Open-Set Domain Adaptation through Self-Supervision

Daniele Rege Cambrin, Kylie Bedwell, Tommaso Natta, Ehsan Ansari Nejad
Politecnico di Torino
Corso Duca degli Abruzzi, 24
10129 Torino, ITALY
s290144@studenti.polito.it, s287581@studenti.polito.it
s282478@studenti.polito.it, s288903@studenti.polito.it

Abstract

- sentence describing the problem - sentence describing our proposed method - sentence summarising the results - sentence about the variations - sentence about the variation results

1. Introduction

In the computer vision research area large amounts of unlabeled data are available, however the cost of labeling this data is high [4, 18]. Domain adaptation is one technique that can be used to exploit the unlabeled data by first training a model on labeled data from a different but similar domain (the *source* domain), and then applying this model to the unlabeled data (the *target* domain). This technique assumes the distribution of both source and target domains are similar and describe the same class labels, also known as the *closed-set* scenario [1]. When applied to real-word scenarios however it is possible that the target domain includes previously unseen classes, known as the *open-set* scenario. These extra class labels in the target domain will cause performance degradation of the classification model and should be identified and isolated. The problem thus consists of two steps: first separating the target domain into known and unknown samples; then conducting domain alignment between the source domain and the known samples of the target domain.

Self-supervised learning can be used to separate the known class samples in the target domain from the unknown samples. Self-supervised learning involves the transformation of data using a known transform (for example by using image rotation), then training a model to predict the transformation [16]. When used in an object classification task and considering the image rotation transformation, the correct orientation of an object is domain-invariant. In this way the model can be trained to predict the correct orientation of the image using data from the source domain, then applied to the images of the target domain. If the orientation of a sample in the target domain is predicted correctly then it is considered to be of a *known* class. Contrarily if the orientation is not prediction correctly it is labelled as *unknown*.

Domain adaptation can then be performed between the samples in the source domain and the samples recognized as known in the target domain. When applying domain adaptation to an open-set scenario (Open-Set Domain Adaptation or OSDA) the samples classified as unknown can be treated as a separate class and incorporated into the Closed-Set Domain Adaptation (CSDA) task [2]. The self-supervised rotation task can also be used to reduce the domain shift during this step, using the Rotation-based Open Set (ROS) method developed by Bucci *et al*. [1].

This study investigates the use of a simplified ROS method for object classification on the *Office-Home* dataset [14]. Alternative self-supervised tasks as well as the inclusion of center loss are also considered and their performance on the object classification task is evaluated.

2. Related Work

Anomaly detection, or outlier/novelty detection, in an open-set scenario can be used to detect samples belonging to the unknown or unseen class. Various different approaches for anomaly detection have been used in the literature as applied to the open-set scenario. Golan and El-Yanic [7] present a method for using geometric transformations to create a self-labeled dataset. In this way the neural classifier learns features that are effective for the detection of anomalies. Sakurada and Yairi [13] on the other hand make use of auto-encoders with dimensionality reduction. This method assumes the data have correlations that can clearly separate normal and anomalous samples when reduced to a lower dimensional subspace. After the test data is projected into the subspace it is then reconstructed and the corresponding reconstruction error is used to identify

the anomalous samples.

Self-supervised learning has been a key concept aimed at reducing the need for human labeling of data. It has also created opportunities for the use of data in problems where supervision is not possible [17]. Self-supervised learning consists of choosing a self-supervised task (or pretext task) to train alongside the main classification task. One possible self-supervised task is image rotation prediction, which is reported to perform best for visual representation learning [6, 16]. However many options are available, including image-patch based methods [8, 10], horizontal flipping [7], or by solving jigsaw puzzles [3, 8].

Domain adaptation techniques have advanced significantly in recent years for the closed-set scenario [2], however for real-world applications the closed-set assumption is often not applicable [11]. It has become increasingly important to develop robust techniques for open-set domain adaptation to address this problem. Recent studies in this field include: the development of a generic approach to learn a linear mapping between the features of the source domain and target domain [2]; the use of self-supervision to improve the generalization of models to different domains [3]; partial domain adaptation by using a discrepancy criterion to partially align features whilst avoiding negative transfer [11]. These techniques have been reported to perform well, and increase the applicability of domain adaptation methods to real-world applications [2, 3, 11].

Rotation-based Open Set (ROS) is a specific technique developed by Bucci *et al*. [1]. ROS is a two-stage method for open-set domain adaptation. The first stage separates samples in the target domain into known and unknown categories by training the model on a multi-rotation recognition task. The rotation recognition task includes the use of the center loss to improve performance by learning a center of the features and minimizing the distances between features and their corresponding centers [15]. The second stage conducts domain alignment, training both semantic and rotation classifiers to classify known target samples. Bucci *et al*. [1] also propose the use of the harmonic mean of the average class accuracies for the known and unknown classes as a more robust and balanced evaluation metric.

3. Method

This study aims to solve the open-set scenario by using self-supervised learning with domain adaptation. Image rotation recognition has been chosen as the self-supervised task due to it's good performance for visual representation learning [6, 16]. The rotation recognition task involves taking the original image sample from the source domain and rotating it clockwise a set amount (for example by 0°, 90°, 180° or 270°). A rotation classifier is then trained to predict the correct orientation of the object. The correct orientation of the object, however, is not an inherent property of an image, for example consider the pens in Figure 1. When analyzing a rotated image of a pen it is not possible to infer the original rotation. The original image is therefore included in the rotation classifier and the relative rotation analysed instead. In training the rotation classifier the features of the original sample are concatenated with the features of the rotated sample. By including the original samples the network is also able to learn features that are more discriminative between class labels, focusing more on the shape of the object and less on the texture [1], as shown in Figure 2.

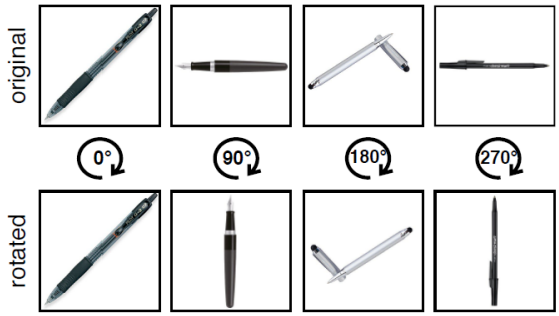


Figure 1. Relative orientations of pens with respect to the original images [1].

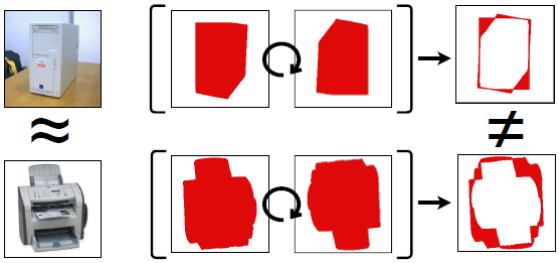


Figure 2. Image rotations help the network learn features that discriminate the shape of objects [1].

To solve the open-set domain adaptation problem a simplified version of the ROS method (Figure 3) is used. This version uses a single-head rotation classifier and does not include the center loss, however the effect of the center loss is evaluated subsequently as a variation to this method. Alternative self-supervised tasks are also considered, including horizontal flipping and through solving jigsaw puzzles. These variations are analyzed in Section 5.

3.1. Stage I - known/unknown separation

Stage 1 of the simplified ROS method involves training both an object classifier and a rotation recognition task on data from the source domain, as shown on the left side of Figure 3, where D_s is the source domain dataset, E is the encoder, C_1 is the object classifier and R_1 is the rotation classifier. The object prediction is based on the features of

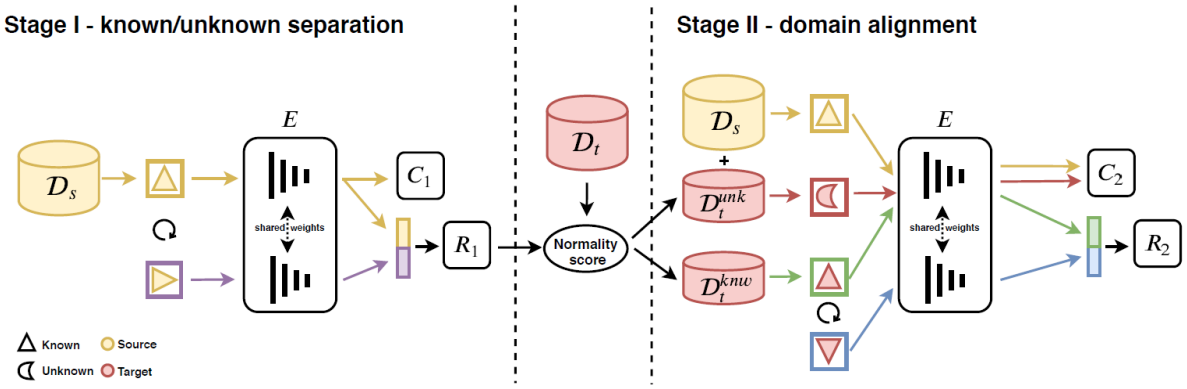


Figure 3. Rotation-based Open Set (ROS) method schematic illustration [1].

the original source samples, whereas the rotation prediction is based on the concatenated features of the original and rotated samples. The object classification and rotation recognition tasks are trained simultaneously to minimize the total loss objective function given by:

$$L_{tot} = L_{cls} + \alpha_1 L_{rot}, \tag{1}$$

where L_{cls} is the loss from the object classifier, L_{rot} is the loss from the rotation classifier, and α_1 is the weight assigned to the rotation recognition task. The value of α_1 is tuned according to the performance of the network, as detailed in Section 4.1.

Target evaluation

Once the rotation classifier has been well trained it can be used to identify the known classes of the target domain, and separate these samples from the unknown classes. The method is illustrated in the center of Figure 3, where D_t is the target domain dataset, D_t^{unk} is the dataset of target samples identified as belonging to an unknown class, and D_t^{knw} is the dataset of target samples identified as belonging to a known class. The rotation classifier is applied to each of the target samples and generates a score for each of the possible rotations. The normality score is then computed as the rotation with the maximum score (highest prediction).

The precision of the normality score is then evaluated using the AUC (area under receiver operating characteristics (ROC) curve) metric. The AUC metric reduces analysis of the ROC curve to a single scalar value which can be used to compare the performance of classifiers [5]. The higher the AUC score the better the classifier overall, with scores > 0.5 showing an improvement over random guessing.

If the AUC is > 0.5 then the normality score can be used to conduct the known/unknown separation subject to a selected threshold, i.e. if the normality score is above the threshold the sample is considered as known and added to D_t^{knw}, otherwise it is considered as unknown and added to D_t^{unk}. The threshold value is also tuned according to the performance of the network, as detailed in Section 4.1.

3.2. Stage II - domain alignment

In stage 2 of the simplified ROS method the unknown dataset created from the target domain D_t^{unk} is combined with the samples from the source domain D_s, with the unknown samples representing an additional 'unknown' class. The object classifier C_2 is thus trained to recognize samples belonging to the unknown class. Concurrently, the known target samples D_t^{knw} are used for the source-target adaptation by training the rotation recognition task R_2. The method is shown on the right side of Figure 3. As in stage I, the object classification and rotation recognition tasks are trained simultaneously to minimize the total loss objective function given by:

$$L_{tot} = L_{cls} + \alpha_2 L_{rot}, \tag{2}$$

where α_2 is the weight assigned to the rotation recognition task of stage 2. The value of α_2 is also selected to provide the best performance, detailed further in Section 4.1.

Final evaluation

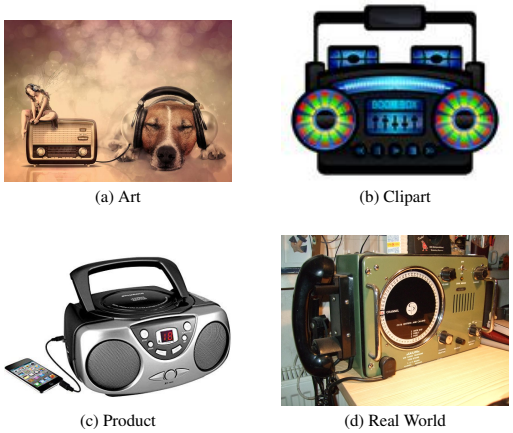
Once the classifiers have been trained the final evaluation can be completed. Three metrics are considered for the evaluation: the accuracy of the object classifier in correctly classifying objects of the known category OS^*; the accuracy of the object classifier in correctly identifying objects of the unknown category UNK; and the harmonic mean HOS between the two accuracies OS^* and UNK, as defined in [1]. The harmonic mean is calculated as:

$$HOS = 2\,\frac{OS^* \times UNK}{OS^* + UNK}, \tag{3}$$

and provides a balanced measure of performance of the classifier at recognizing both known and unknown samples.

4. Experiments

The proposed method was tested on the *Office-Home* dataset [14]. This dataset consists of four separate domains for image classification: Art, Clipart, Product and Real World. An example of the difference between images in these domains is shown in Figure 4. For this experiment each domain included two sets of images, one when behaving as the source domain and another as the target. The dataset contained 65 classes in total. The classes were sorted alphabetically and the first 45 classes were treated as known and formed the list of images for the source domain, the remaining 20 were considered unknown and included in the list of images for the domain when behaving as the target domain.



(a) Art (b) Clipart

(c) Product (d) Real World

Figure 4. Domains in the Office-Home dataset [14].

The experiments were conducted using a ResNet18 convolutional neural network architecture, employing a learning rate of 0.001, a batch size of 128. Image data augmentation techniques were also employed to improve performance of the model, these included random resized crop, color jitter and random grayscale.

4.1. Ablation study

The simplified ROS method includes three hyper-parameters that need to be tuned according to the network: the weight assigned to the rotation recognition tasks in stage 1 (α_1); the threshold of normality score for considering a target sample as belong to a known category; and the weight assigned to the rotation recognition tasks in stage 2 (α_2). The α_1 parameter of stage 1 was tuned first. Figure 5 shows the class accuracy of stage 1 when trained on the *Art* source domain for a number of epochs and different α_1 values, and Figure 6 shows the corresponding rotation recognition accuracy. As expected the accuracy of both increases with increasing number of epochs, however lower weights take much longer to converge for the rotation recognition tasks. Since the two classifiers are trained simultaneously a balance needs to be sought to reduce over-fitting of either task.

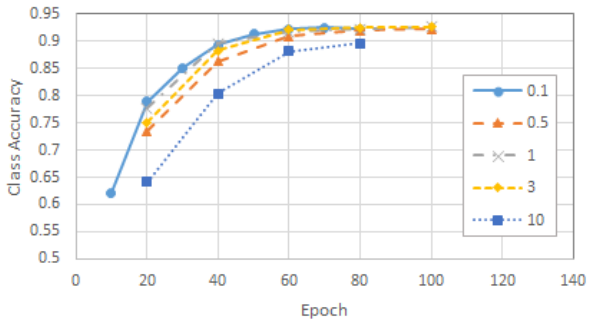


Figure 5. Class accuracy during stage 1 training on the Art domain for various weights.

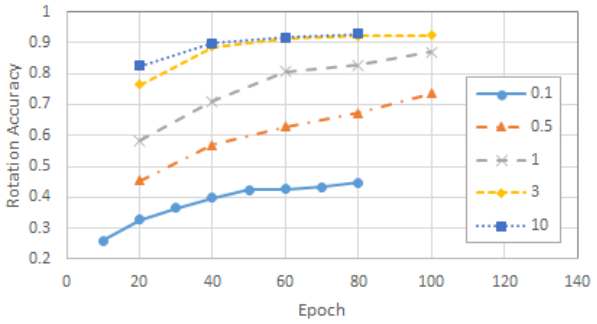


Figure 6. Rotation recognition accuracy during stage 1 training on the Art domain for various weights.

The AUC metric was then used to compare the performance of the various weights. The AUC value for the network with *Art* as the source domain and *Clipart* as the target domain is shown in Table 1. The highest AUC measured was 0.5578 for a weight of 10 at 40 epochs, therefore 10 was chosen at the values for α_1 in stage 1.

	Weight (α_1)				
Epoch	0.1	0.5	1	3	10
40	0.5293	0.4944	0.5245	0.5492	0.5578
80	0.5203	0.4873	0.5312	0.5397	0.5548

Table 1. AUC for Art – Clipart

The next step involved tuning the threshold. The threshold should be chosen such that the number of samples recognized as known is similar to the number of samples that are expected to be recognized as known. Considering again the *Art* as source and *Clipart* as target domain combination as an example, the total number of samples in the *Clipart* target domain is 4365, and 3064 of those samples share the same category as those in the *Art* source domain. The

model was thus evaluated with different thresholds and the number of samples identified as known analysed. Table 2 shows the results of the analysis on the *Art–Clipart* domain combination. A threshold of 0.75 results in the number of known samples being most similar to the expected number of known samples of 3064, and so 0.75 was chosen as the threshold for that combination.

Threshold	Known Samples
0.3	3778
0.5	3486
0.7	3123
0.75	3019
0.8	2872

Table 2. Samples recognised as known for various thresholds for the Art – Clipart domain combination

The final step of the ablation study was to tune the stage 2 weight (α_2). The evaluation metrics for stage 2 involve the harmonic mean between the accuracy of the known classes and the ability of the classifier to recognize the unknown category, as discussed in 3.2. The results of stage 2 for the *Art–Clipart* domain combination are shown in Table 3. The highest harmonic mean value is 0.4399, for a weight of 3 at 20 epochs.

Weight (α_2)	Epoch	OS*	UNK	HOS
0.01	20	0.3026	0.5778	0.3972
	40	0.3008	0.5743	0.3948
0.1	20	0.3091	0.5896	0.4056
	40	0.3257	0.5601	0.4119
0.5	20	0.3280	0.5731	0.4172
	40	0.3404	0.5495	0.4204
3	20	0.3542	0.5802	0.4399
	40	0.3690	0.5000	0.4246
10	20	0.3114	0.6156	0.4136
	40	0.3367	0.5743	0.4246

Table 3. Stage 2 evaluation metrics for the Art – Clipart domain combination

Similar analyses were also conducted for the 11 other possible domain combinations, the results of which can be found in the supplementary material[1].

4.2. Results

The results for the simplified ROS model for each of the domain combinations is shown in Table 4. The harmonic mean for each of the different domain combinations lies approximately between 40% and 55%. Different combinations performed better than others, however the results were sensitive to the input parameters. Depending on the application, prioritizing the accuracy of the categorization of the known class may prove beneficial and further training would be required. Results for the complete ROS method were published in [1] and reported a harmonic mean between approximately 55% and 75%. As expected the complete method performs better than the simplified method adopted in this study, however the simplified method could be employed in situations where a less complex approach is desired.

Domain				
Source	Target	OS*	UNK	HOS
Art	Clipart	35.42	58.02	43.99
	Product	42.27	46.21	44.15
	Real World	42.50	77.01	54.77
Product	Clipart	38.05	55.94	45.29
	Art	33.53	46.24	38.87
	Real World	56.50	37.72	45.24
Real World	Clipart	43.87	50.76	47.07
	Art	46.86	44.54	45.67
	Product	69.44	33.95	45.60
Clipart	Real World	45.46	69.06	54.83
	Art	33.71	58.14	42.67
	Product	47.75	60.02	53.19

Table 4. Accuracy (%) for the simplified ROS model on the Office-Home dataset

5. Variations

Two different variations to the simplified ROS method were also analysed. The first considered the use of alternative self-supervised tasks. The second considered the addition of the center loss to the rotation recognition task.

5.1. Alternative self-supervised tasks

The simplified ROS method reported used rotation recognition as the self-supervised task, however many different self-supervised tasks are possible to be employed for use in open-set domain adaptation. This sudy investigated two alternative self-supervised tasks: horizontal flipping; and solving jigsaw puzzles.

Horizontal Flipping

Horizontal flipping is similar to image rotation however instead rotating the image it is flipped along its center y-axis, creating a mirror-image. Horizontal flipping is shown in Figure 7.

[1]Additional results and the complete code is available at https://github.com/DarthReca/AML-Project.

(a) Raw image (b) Horizontally flipped

Figure 7. Horizontal flipping of an image, adapted from [14].

Jigsaw Puzzle

The self-supervised tasks of solving jigsaw puzzles involves being able to recognize an original image given its shuffled parts. The method involves first taking the original image, splitting the image into a number of sections, then creating a set of permutations with a different ordering of those sections. The method can be seen in Figure 8 as used by [3]. The shuffled images (permutations) are then fed into the convolutional neural network to train the jigsaw classifier alongside the object classifier. This study considered the images broken up into 9 sections, and then recombined based on a random permutation among 30 available permutations. For each image a shuffled one is generated and allocated an index. The convolutional neural network is then trained to predict its index. The Jigsaw puzzle task is formalized as a classification problem over recomposed images with the same dimension of the original one. In this way object recognition and patch reordering can share the same network backbone.

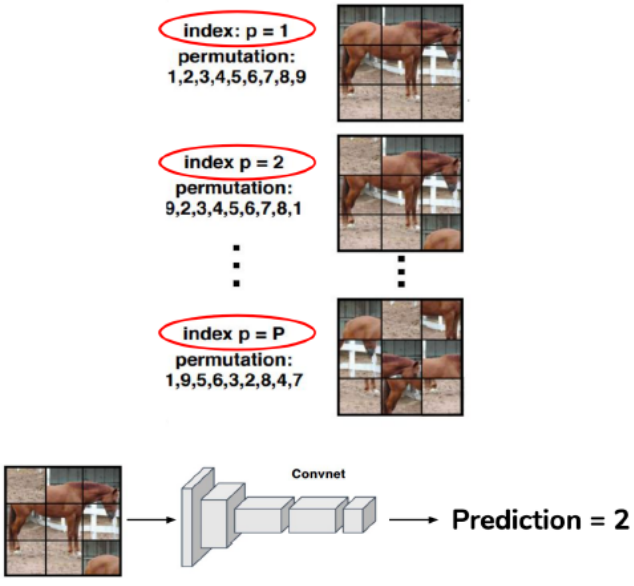


Figure 8. Jigsaw puzzle self-supervised task method, adapted from [3].

Results

The harmonic mean results of the simplified ROS method employing different self-supervised tasks for a subset of domain configurations are shown in Table 5. The first two configurations show the best performance using rotation recognition as the self-supervised task, however the third configuration performs best with horizontal flipping. It should be noted that in calculating these results the threshold parameter needed to be tuned independently for each self-supervised task, as this would have a significant effect on the results. The rotation recognition task seems to be the most robust method for conducting open-set domain adaption, in agreeance with the findings from Xu *et al*. [16] and Gidaris *et al*. [6]. As shown by this short study alternative self-supervised tasks may however perform better for some domain configurations.

Domain				
Source	**Target**	**Rotation**	**Horizontal Flipping**	**Jigsaw Puzzle**
Art	Clipart	43.99	35.11	18.64
Product	Art	38.87	25.01	36.17
Real World	Product	45.60	53.58	49.46

Table 5. Harmonic mean (%) for the simplified ROS model with different self-supervised tasks on the Office-Home dataset

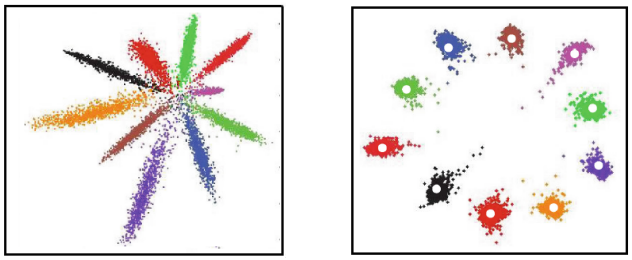
5.2. Center Loss

The center loss was introduced by Wen *et al*. [15] as an additional objective function for enhancing the discriminative power of learned deep features. It involves learning the center of the deep features for each class, prioritizing features that are close to the center and adding a penalty to the distance between the deep features and their corresponding centers [15]. The center loss function is optimized alongside the soft max loss, with the two losses jointly supervised. A hyper-parameter is introduced representing the weight of the center loss contribution, and is tuned to balance the two loss functions. The difference between the features learned by a classifier training with the soft max loss function on the MNIST dataset [9], compared with the inclusion of the center loss during training is shown in Figure 9. The features learned with the inclusion of center loss can clearly be seen to provide greater discriminative power.

The center loss has been implemented during the first step of the model, and used to understand the effect of this loss on the separation phase.

Results

An ablation study was performed to select the most appropriate weight (α_w) of the center loss contribution. The initial weights considered were based on those reported by Bucci *et al*. [1] in their research on the Office-Home [14]



(a) Soft Max Loss (b) Center Loss

Figure 9. Comparison of features learned when including the center loss on the MNIST dataset, adapted from [15].

Domains	Metric	Standard	Weight α_w 0.001	0.01
Art → Clipart	AUC	**0.5590**	0.5486	0.5302
	Accuracy	0.8988	0.9150	0.9150
Product → Art	AUC	0.4997	**0.5077**	0.5023
	Accuracy	0.9513	0.9612	0.9631
Art → RealWorld	AUC	0.4956	**0.5000**	0.4967
	Accuracy	0.8988	0.9128	0.9139

Table 6. Comparison of center loss model with standard model

and Office31 datasets [12]. High values of α_w were found to give too much importance to the center loss at the expense of the soft max loss, and the model converged extremely slowly. For this reason the results analysed here consider only weight values lower or equal to 0.01.

Table 6 shows a comparison of the AUC values with and without the introduction of center loss, with $\alpha_1 = 10$ trained for 40 epochs. The inclusion of center loss was able to improve the accuracy for all domain configurations considered. It also improved the AUC metric for the second two domain configurations, however was detrimental in the first configuration. Due to the similarity of the results presented in 6 the findings are inconclusive, and further investigations would be necessary to understand if the center loss could be used effectively in this scenario.

6. Conclusions

- summarise the results - add future recommendations

6.1. Acknowledgements

The authors would like to thank Silvia Bucci for her assistance and guidance in completing this study.

References

[1] Silvia Bucci, Mohammad Reza Loghmani, and Tatiana Tommasi. On the effectiveness of image rotation for open set domain adaptation. In *CVPR*, 2020. 1, 2, 3, 5, 6

[2] Pau Panareda Busto, Ahsan Iqbal, and Juergen Gall. Open set domain adaptation for image and action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(2):413–429, 2020. 1, 2

[3] Fabio Maria Carlucci, Antonio D'Innocenteand Silvia Bucci, Barbara Caputo, and Tatiana Tommasi. Domain generalization by solving jigsaw puzzles. In *CVPR*, 2019. 2, 6

[4] Gabriela Csurka. Domain adaptation for visual applications: A comprehensive survey. In *CVPR*, 2017. 1

[5] Tom Fawcett. An introduction to roc analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006. 3

[6] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *CVPR*, 2018. 2, 6

[7] Izhak Golan and Ran El-Yaniv. Deep anomaly detection using geometric transformations. In *NeurIPS*, 2018. 1, 2

[8] Dahun Kim, Donghyeon Cho, Donggeun Yoo, and In So Kweon. Learning image representations by completing damaged jigsaw puzzles. In *WACV*, 2018. 2

[9] Yann LeCun, Corinna Cortes, and Christopher J.C. Burges. The mnist database of handwritten digits. http://yann.lecun.com/exdb/mnist/, 1998. 6

[10] T. Nathan Mundhenk, Daniel Ho, and Barry Y. Chen. Improvements to context based self-supervised learning. In *CVPR*, 2018. 2

[11] Chuan-Xian Ren, Pengfei Ge, Peiyi Yang, and Shuicheng Yan. Learning target-domain-specific classifier for partial domain adaptation. *IEEE Transactions on Neural Networks and Learning Systems*, 32(5):1989–2001, 2021. 2

[12] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *ECCV*, 2010. 7

[13] Mayu Sakurada and Takehisa Yairi. Anomaly detection using autoencoders with nonlinear dimensionality reduction. In *Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis*, pages 4–11, 2014. 1

[14] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *CVPR*, 2017. 1, 4, 6

[15] Yandong Wen, Kaipeng Zhang, Zhifeng Li1, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *ECCV*, 2016. 2, 6, 7

[16] Jiaolong Xu, Liang Xiao, and Antonio M. Lopez. Self-supervised domain adaptation for computer vision tasks. *IEEE*, 7:156694–156706, 2019. 1, 2, 6

[17] Burhaneddin Yaman, Seyed Amir Hossein Hosseini, Steen Moeller, Jutta Ellermann, Kâmil Uğurbil, and Mehmet Akçakaya. Self-supervised learning of physics-guided reconstruction neural networks without fully sampled reference data. *Magnetic Resonance in Medicine*, 84(6):3172–3191, 2020. 2

[18] Lei Zhang and David Zhang. Robust visual knowledge transfer via extreme learning machine-based domain adaptation. *IEEE Transactions on Image Processing*, 25(10):4959–4973, 2016. 1