Efficient and Robust Pseudo-Labeling for Unsupervised Domain Adaptation

Hochang Rhee* and Nam Ik Cho†
† Department of ECE, INMC, Seoul National University, Seoul, Korea
E-mail: nicho@snu.ac.kr Tel/Fax: +82-2-880-1810

***Abstract*—Unsupervised domain adaptation is to transfer knowledge from an annotated source domain to a fully-unlabeled target domain. The conventional methods consider the data which exceed a certain threshold of confidence as pseudo-labeled data for the target domain, and thus choosing the appropriate threshold affects the target performance. In this paper, we propose a new confidence-based weighting scheme for obtaining pseudo-labels and an adaptive threshold adjustment strategy to provide sufficient and accurate pseudo-labels throughout the training. To be precise, our confidence-based weighting scheme generates pseudo-labels to have a different contribution based on the confidence, which makes the performance less sensitive to the threshold. Also, the proposed adaptive threshold adjustment method chooses the threshold according to the degree of adaptation of a network to the target domain, and thus obviates the need for an exhaustive search for the appropriate threshold. Experimental results on a digit classification task show that the proposed methods efficiently utilizes the pseudo-labels to preserve sufficiency and accuracy.**

I. Introduction

Recent researches using Deep Neural Network (DNN) have achieved state-of-the-art results in many computer vision tasks through the supervised training of DNN [1]–[5]. However, the successful use of DNN usually requires a large amount of well-annotated training datasets which is cost expensive. Moreover, the training dataset needs to be recollected for a new task, even if it is related to the previously trained one. Hence, acquiring a sufficient number of training data and/or using the related well-annotated datasets have been one of the significant issues in DNN researches. In this respect, the transfer learning [6] is an excellent technique that can alleviate the cost of recollecting large scale labeled data by transferring knowledge from a different but related domain. Also, domain adaptation is a sub-problem of the general transfer learning, which pays attention to the case where the annotated training data (*i.e.*, source domain) and the unannotated training data (*i.e.*, target domain) share the same task but follow different distributions.

Domain adaptation methods are categorized into supervised, semi-supervised, and unsupervised according to the degree of target domain labeling, while the source domain dataset is fully labeled. In supervised [7] and semi-supervised domain adaptation [8], [9], only part of the target data samples are labeled, but not enough to achieve a satisfactory model for the target domain. In unsupervised domain adaptation (UDA) [10]–[12], no target data samples are labeled. Among the three categories, this work focuses on the unsupervised domain adaptation, where all target data samples are unlabeled.

Most of the previous methods focused on minimizing the discrepancy between the distributions of source and target data [13]–[17]. However, mapping the source and target distributions to a common one can mix samples with different labels together. Hence recent studies started to consider the learning of discriminative representations for the target domain. For example, some of them adopted pseudo-labeling schemes to learn target discriminative representations, which encourage separation between the classes in the target domain [12], [18]–[20].

Pseudo-labeling [21] is the process of artificially labeling the unlabeled data to convert the unsupervised learning into the semi-supervised. We follow the premise that samples with high confidence output can be used as accurate labels [22], [23]. Specifically, unlabeled samples with confidence over a certain threshold are deemed as ground-truth labels, which we call pseudo-labels. Performance of pseudo-labeling technique relies heavily on the threshold, where choosing the appropriate threshold is critical to achieving high performance. A high threshold leads to more accurate pseudo-labels resulting in high performance, but also leads to the lack of labels. On the other hand, a low threshold results in a large number of pseudo-labels but also in poor performance due to inaccurate pseudo-labels. Thus, finding the appropriate threshold to achieve high performance is a major issue in pseudo-labeling.

Also, rather than fixing the threshold throughout the training, conventional methods constantly increase the threshold as the training progresses [18]–[20]. As the training progresses, a network adapts to the target domain implying that more pseudo-labels with high confidence will be generated compared to the early stage of training. When the training has been progressed to some extent, a network is capable of providing sufficient pseudo-labels even with a high threshold. Hence, a network focuses on generating accurate pseudo-labels by increasing the threshold throughout the training.

However, above-mentioned methods concerning the selection of threshold have limitations. First, the performance is extremely sensitive to the threshold, where a slight difference in threshold can have a huge negative effect. Exhaustive search is done to find the right threshold which is time-consuming and needs to be done individually for different datasets. Second, prior works adjust the threshold depending on the progress of training or the accuracy of the network with respect to

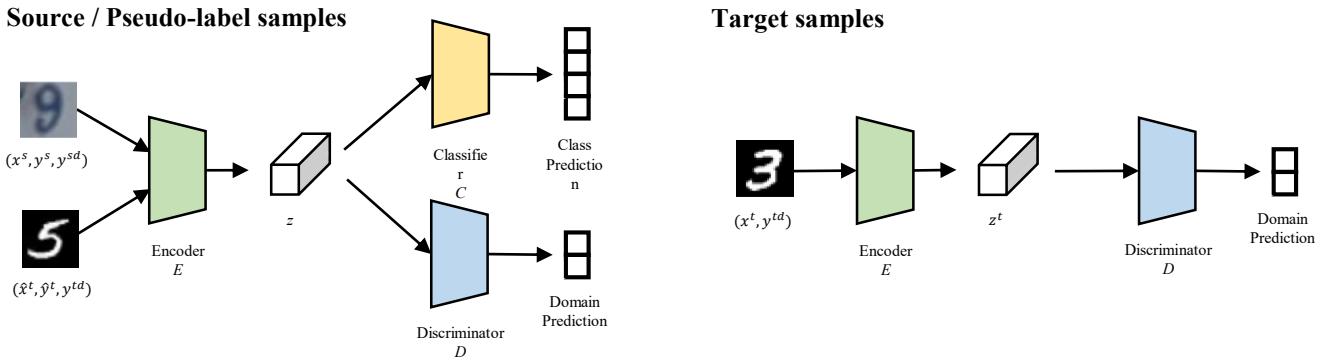


Fig. 1: Overall structure of baseline model. Baseline model is composed of an encoder E, a discriminator D, and a classifier C. The encoder embeds the source domain sample x^s, target domain sample x^t, and pseudo-labeled sample $\hat{x}^t$ into latent representation z. The discriminator aims at distinguishing the domain origin of latent representation z. For final classification, classifier C is built based on the latent representation z, expecting z to preserve both the common feature of source and target domains and category information. Source and pseudo-label samples contribute to both category loss and adversarial loss whereas target samples with no pseudo-labels contribute only to category loss.

the source domain. Despite the fact that the pseudo-labels are generated from the target domain samples, the threshold has no dependency on the target domain.

To overcome the above-stated limitations of conventional methods, we propose a confidence-based weighting scheme on constructing pseudo-labels and an adaptive threshold adjustment method. Our confidence-based weighting scheme constrains both disadvantages of high and low threshold. We grant high weights for pseudo-labels with high confidence and low weights for the low confidence. As a result, the performance becomes less sensitive to the threshold and thus obviates the need for an exhaustive search for the appropriate threshold. Also, the proposed adaptive threshold adjustment method chooses the threshold according to the degree of adaptation of a network to the target domain. The threshold becomes target domain dependent rewarding the target domain to generate sufficient and accurate pseudo-labels throughout the training. Through the proposed method, a network is capable of adjusting the threshold in a adaptive way with no need for exhaustive searching.

In summary, the main contributions of this paper are as follows:

- Introducing a confidence-based weighting scheme for generating pseudo-labels when training a network
- Proposing an adaptive threshold adjustment method for modifying the threshold in an adaptive process

The rest of this paper is organized as follows. Section 2 is dedicated to related works. The model and the training procedure are presented in Section 3. In Section 4, the dataset used for the analysis and the method's validation are described. Experiments and associated results are also detailed in Section 4. Finally, conclusive remarks are drawn in Section 5.

II. Related Works

We summarize the works focusing on unsupervised domain adaptation employing pseudo-labels to learn the target discriminative representations. Methods for re-labeling the unlabeled target samples [22], [23] have been applied to many unsupervised domain adaptation tasks where back propagation of the category loss for target samples is based on pseudo-labeled samples. Most researches [18]–[20] give pseudo-labels equal contribution to the category loss, despite the fact that confidence of pseudo-labels differs. Hu *et al.* [13] used a fixed threshold throughout the training. Due to the performance's sensitivity to the threshold different thresholds were set for various datasets.

Considering the network's adaptation to the target domain, some researches altered the threshold as the training progressed. Chen *et al*. [19] gradually increased the threshold according to the current epoch of training. Finding the right increase rate for the threshold acquires exhaustive search, which is time-consuming. Zou *et al*. [20] adjusted the threshold based on the model's accuracy with respect to the source domain. The concept behind the "source accuracy dependent threshold" is that the source accuracy reflects the network's training progress. However, most networks are pre-trained on the source domain, which outputs high source accuracy even in the early stage of training. Thus, a model's accuracy on the source domain is not enough to show the progress of training. This paper attempts to improve the method of generating pseudo-labels based on the confidence levels and to set the threshold in an adaptive, target-dependent manner.

III. Method

In this section, we provide the details of the proposed methods. We first give a brief overview of the task formulation. Then we introduce our confidence-based weighting scheme

on pseudo-labels. Finally, we provide the formulation of the adaptive threshold adjustment.

A. Task Formulation

In UDA, we are given a source domain dataset $D_s = \{x_i^s, y_i^s\}_{i=1}^{n_s}$ ($x_i^s \in X_s, y_i^s \in Y_s$) and a target domain dataset $D_t = \{x_j^t\}_{j=1}^{n_t}$ ($x_j^t \in X_t$). n_s and n_t denotes the number of samples from the source and target domain respectively, X_s and Y_s denotes the set of source data and labels, and X_t denotes the set of target data. The source and target domain data have the joint probability distributions $P(X_s, Y_s)$ and $Q(X_t, Y_t)$ respectively, with $P \neq Q$ in general. We assume that the source and target domains contain the same object classes. Unless otherwise specified, the symbols s and t used in the superscript or subscript denote the source domain and the target domain respectively.

B. Baseline model

We describe our proposed method through a baseline model illustrated in Figure 1, composed of an encoder, a discriminator and a classifier. We build our baseline model upon DANN [24], where we adopt pseudo-labels to contribute to the category loss. The encoder, denoted as E, attempts to map any image from either source or target domain into a latent representation $z = E(x)$, which is expected to be domain invariant and category informative. The discriminator, denoted as D, judges whether the latent representation is derived from the source or target domain, forcing the latent representation to become domain invariant. The classifier, denoted as C, tries to classify the categories of the input latent representations.

To achieve high classification performance in respect to target domain samples, the latent representation z needs to be domain invariant and category informative. The domain invariance can be achieved by training the encoder to map both source and target samples into a common domain representation so that the discriminator is incapable of judging the domain origination of the latent representation. The latent representation can become category informative by training the classifier through correctly labeled source samples and pseudo-labeled target samples.

The objective function of the discriminator to achieve domain invariance is formulated as

$$L_D = \min_{W^D}\Big(\sum_{x^s \in X_s} H(D(z^s), y^{sd}) + \sum_{x^t \in X_t} H(D(z^t), y^{td})\Big) \tag{1}$$

where $H(\cdot,\cdot)$ is the cross entropy loss used in softmax layer and W^D denotes the parameters of the discriminator D. y^{sd} and y^{td} are 2-class domain labels each representing the source domain and target domain label. The discriminator's objective is to correctly predict the domain labels of the given input z.

The objective function of the encoder to achieve domain invariance is formulated as

$$L_G = \min_{W^E}\left(\sum_{x^t \in X_t} H(D(z^t), y^{sd})\right) \tag{2}$$

where W^E represents the parameters of the encoder E. The encoder focuses on forcing the discriminator to falsely predict the target samples as source samples.

The objective function of the encoder E and the classifier C to make the latent representation contain category information is formulated as

$$L_C = \min_{W^E, W^C}\Big(\sum_{x^s \in X^s} H(C(z^s), y^s) + \sum_{\hat{x}^t \in \hat{X}_t} H(C(\hat{z}^t), \hat{y}^t)\Big) \tag{3}$$

where $\hat{x}^t$ represents pseudo-labeled samples, $\hat{y}^t$ represents pseudo-labels for $\hat{x}^t$ and W^C represents the parameters of the classifier C. The classifier is trained based on all the source samples and pseudo-labeled samples from target samples.

Target samples are pseudo-labeled when the confidence exceeds a certain threshold. Pseudo-labels are considered reliable and used as ground truth labels to contribute to the category loss of the classifier.

C. Confidence-based weighting scheme

When training the classifier by category loss, prior works [18]–[20] give equal contribution to category loss for each pseudo-labeled samples even though the confidence of the samples vary. With the premise that highly confident pseudo labels are mostly correct [22], [23], pseudo-labels with different confidence should have different contribution to the category loss. Pseudo-labels with high confidence imply a high probability of matching the ground-truth labels, which should give high contribution to the category loss. In contrast, pseudo-labels with low confidence have the risk of being falsely labeled meaning that low contribution should be given to the category loss to suppress the risk. Therefore, we propose to give pseudo-labels weight that is proportional to their confidence.

We set fully confident pseudo-labels, labels with confidence of 1.0, to have weight of 1.0 whereas we empirically set pseudo-labels with confidence of the threshold value to have half the contribution compared to fully confident pseudo-labels. That is, we assign pseudo-labels with confidence of the threshold value with a weight of 0.5. We set weight of pseudo-labels to be linearly proportional to the confidence of the pseudo-labels as

$$w(x^t) = \frac{0.5}{1-th} conf(x^t) + \frac{0.5-th}{1-th} \tag{4}$$

where $conf(x^t)$ represents the confidence of target sample x^t derived from the classifier and th denotes the threshold value.

We can modify Equation (3) by applying Equation (4) to reformulate a new category loss considering the confidence of pseudo-labels as

$$L_C = \min_{W^E, W^C} (\sum_{x^s \in X^s} H(C(z^s), y^s) + \sum_{\hat{x}^t \in \hat{X}_t} w(\hat{x}^t) H(C(\hat{z}^t), \hat{y}^t)) \quad (5)$$

where the weight term is applied to the cross entropy term of pseudo-labels.

Through the confidence-based weighting scheme of pseudo-labels, we are able to reduce the disadvantages of both the low and high threshold. Inaccurate pseudo-labels generated by the low threshold are given small weight and thus the classifier suffers less from the inaccuracy. High threshold undergoes the insufficiency of the pseudo-labels, which is compensated by the pseudo-labels provided from the low threshold. In result, the performance of a network does not drop significantly from the inaccuracy and insufficiency of pseudo-labels, meaning that the performance becomes less sensitive to the threshold.

D. Adaptive threshold adjustment

To focus on the lack of pseudo-labels in the early stage of training and the accuracy of pseudo-labels in the later stage, many studies increase the threshold according to the progress of training [19], [20]. The progress of training is determined by the current epoch or the accuracy of the network on the source domain samples. However, these terms are not target domain dependent and thus do not fully reflect the progress of training. Therefore, we propose an adaptive threshold adjustment strategy, setting the threshold according to the model's degree of adaptation to the target domain. A model with low adaptation to the target domain is not capable of classifying target samples, which results in low confidence output, whereas well adapted model gives high confidence output. Therefore, we consider a model's degree of adaptation to the target domain as the average confidence output of the target samples. The modified threshold based on a model's confidence in respect to the target domain is formulated as

$$th = \max(\frac{\sum_{x^t \in X_t} conf(x^t)}{n_t}, \alpha) \quad (6)$$

where n_t represents the number of the target samples and α represents the minimum threshold. A minimum threshold is determined to prevent too low threshold in the early stage of training, when the model has no adaptation to the target domain. Too low threshold will generate many inaccurate pseudo-labels which can prevent the network from learning the target discriminate representation. Through the proposed method, the network can adaptively choose the appropriate threshold to maintain sufficiency and accuracy of pseudo-labels throughout the training. In result, a model can be robust to various datasets and model capacity.

Algorithm 1 Optimization procedure of network

Input: The source domain x^s and its category label y^s, target domain sample x^t without category label

Output : The parameter of whole network, $W = \{W^E, W^D, W^C\}$

1: Pre-train E and C with $x^s \in X_s$
2: **while** not converge **do**
3: Determine th by Equation (6) via C
4: Obtain pseudo-label from $x^t \in X_t$ with samples having confidence over th
5: Update W^D by minimizing L_D in Equation (1)
6: Update W^C, W^E by minimizing $L_E + \alpha L_C$ in Equation (2) and (5)
7: **end while**

E. Overall Objective Function

The overall objective function can be formulated as

$$L = \min_{W^E, W^D, W^C} (L_E + L_D + \beta L_C) \quad (7)$$

where β is a balance parameter. The encoder and the discriminator are optimized in an adversarial manner to map the source and target samples into a common domain representation. The encoder and the classifier are optimized using the source and pseudo-labeled samples to achieve high classification performance. The detailed optimization process is shown in Algorithm 1.

IV. Experiments

Evaluation of domain adaptation in this paper is category classification on the test set of target domain with a model trained with labeled source domain samples and unlabeled target domain samples. We report our performance in terms of rank-1 accuracy of the classifier C. Specifically, we experiment our category classification performance on digit classification task. MNIST [25], USPS [26] and SVHN [27] are the datasets used in the experiment. For more comprehensive comparison, we further evaluate the network trained with only labeled sourced domain images, denoted as "Source Only" and the network trained with only labeled target domain images as "Target Only." Both "Source Only" and "Target Only" are constructed on the baseline model in Section 3.2 and indicate the lower and upper bound performance of domain adaptation.

A. Implementation Details

The pixel values in the input images are re-scaled to [0.0, 1.0] in all experiments. We resize source domain samples to the size of target domain samples to maximize the performance on the target domain. Minimum threshold α is empirically set as 0.7 to prevent too much inaccurate pseudo-labels from interrupting the training. Balance parameter β is set as 0.2 in our experiments.

TABLE I: Result on digit classification

Table 1: The results of unsupervised domain adaptation on digit classification. CBWS denotes confidence-based weighting scheme and ADA denotes adaptive threshold adjustment.

Source Target	MNIST USPS	USPS MNIST	SVHN MNIST
Source Only	82.74	69.50	59.97
Baseline	88.49	86.41	76.01
CBWS	90.74	88.10	77.90
ADA	93.04	93.90	81.01
CBWS+ADA	**94.52**	**96.14**	**82.41**
Target Only	94.96	98.88	98.88

B. Experiment Results

We experiment on four models: baseline model, a model with confidence-based weighting scheme (CBWS), adaptive threshold adjustment (ADA) model, and finally a model where both CBWS and ADA are applied. The baseline model uses a fixed threshold of 0.7 throughout the whole training. Performance of each models are shown in Table 1.

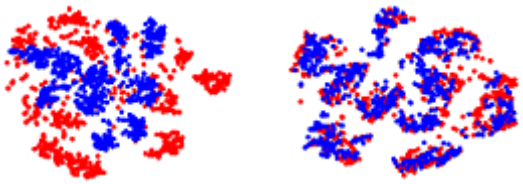
The proposed CBWS and ADA both outperforms the baseline model showing that both methods better utilizes pseudo-labels when learning target discriminative representation. From the fact the ADA model outperforms CBWS model, we can imply that the ADA model is capable of generating sufficient and accurate pseudo-labels whereas the CBWS model only forces the false pseudo-labels to have less influence. Our final model employing both CBWS and ADA outperforms all other models implying that the final model retains the appropriate threshold throughout the training.

Our final model benefits most in the setting of adapting SVHN to MNIST compared to other settings. Discrepancy between SVHN and MNIST is large compared to MNIST and USPS where false pseudo-labels are more likely to be produced. When the risk of false pseudo-labeling is high, CBWS and ADA's ability to lessen the negative influence of false pseudo-labels is maximized.

We further visualize the distribution of the learnt latent representation to show the effect of domain adaptation. Latent representation before adaptation is illustrated in Figure 2 (a) where the discrepancy between the source and target domain is large. In contrast, Figure 2 (b) demonstrates the result of domain adaptation where the discrepancy is significantly reduced.

V. Conclusion

We have proposed a confidence-based weighting scheme for generating pseudo-labels and an adaptive threshold adjustment strategy to handle an unsupervised domain adaptation task efficiently. Confidence-based weighting scheme grants pseudo-labels with weight proportional to the confidence of target samples. As a result, the negative influence of insufficient and inaccurate pseudo-labels is reduced, making the performance less sensitive to the threshold. Adaptive threshold adjustment is proposed to modify the threshold according to the model's adaptation to the target domain. Accordingly, a model is capable of maintaining the appropriate threshold throughout the training without any exhaustive search. Evaluation of the proposed methods for the digit classification shows that the proposed methods contribute to performance gain in unsupervised domain adaptation task. In the future, we will calibrate the confidence and accuracy of the classifier to generate more reliable pseudo-labels.



(a) Before Adaptation (b) After Adaptation

Fig. 2: The distribution of source and target domain samples before and after the adaptation. Source domain samples are shown in red and target domain samples are shown in blue. (a) shows the distribution of source and target domain samples before adaptation. (b) shows the distribution of source and target domain samples after adaption

Acknowledgment

This research was supported by Samsung Electronics Co., Ltd.

References

[1] A. Krizhenvshky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional networks," in *Proceeding of the IEEE Conference on Advances in Neural Information Processing Systems (NIPS)*, pages 1097-1105, 2012

[2] K. Simoyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *arXiv preprint arXiv:1409.1556*, 2014.

[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)*, 2016.

[4] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE Conference on Computer vision and pattern recognition (CVPR)*, pages 1440-1448, 2015.

[5] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A.L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," in *IEEE Transactions on Pattern analysis and Machine intelligence*, 2016.

[6] S. Pan and Q. Yang, "A survey on transfer learning," in *IEEE Transactions on Knowledge and Data Engineering (KDE)*, pages 1345-1359, 2009.

[7] H.D. III, "Frustratingly easy domain adaptation," in *arXiv preprint arXiv:0907.1815*, 2009.

[8] Y. Yao and G. Doretto, "Boosting for transfer learning with multiple sources," in *Proceedings of the IEEE Conference on Computer vision and pattern recognition (CVPR)*, pages 1855-1862, 2010

[9] A. Kumar, A. Saha, and H. Daume, "Co-regularization based semi-supervised domain adaptation," in *Proceeding of the IEEE Conference on Advances in Neural Information Processing Systems (NIPS)*, pages 478-486, 2010

[10] M. Baktashmotlagh, M.T. Harandi, B.C, and M.Salzmann, "Unsupervised domain adaptation by domain invariant projection," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 769-776, 2013.

[11] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *arXiv preprint arXiv:1409.7495*, 2014.

[12] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F.Laviolette, and V.Lempitsky, "Domain-adversarial training of neural networks," in *The Journal of Machine Learning Research*, 2016.

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

[14] M. Long, Y. Cao, J. Wang, and M.I. Jordan, "Learning transferable features with deep adaptation networks," in *arXiv preprint arXiv:1502.02791*, 2015.

[15] M. Gong, K. Zhang, T. Liu, D. Tao, C. Glymour, and B. Schölkopf, "Domain adaptation with conditional transferable components," in *International Conference on Machine Learning"*, pages 2839-2848, 2016.

[16] K. Bousmalis, G. Trigeorgis, N. Silberman, D.Krishnan, and D. Erhan, "Domain separation networks," in *Proceeding of the IEEE Conference on Advances in Neural Information Processing Systems (NIPS)*, 2016.

[17] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proceedings of the IEEE Conference on Computer vision and pattern recognition (CVPR)*, pages 7167-7176, 2017.

[18] L. Hu, M.Kan , S. Shan, and X. Chen, "Duplex generative adversarial network for unsupervised domain adaptation," in *Proceedings of the IEEE Conference on Computer vision and pattern recognition (CVPR)*, pages 1498-1507, 2018.

[19] C. Chen and W. Xie, "Progressive feature alignment for unsupervised domain adaptation," in *arXiv preprint arXiv:1811.08585*, 2019.

[20] Y. Zou, Z. Yu, K. Vijaya, B.V.K, and J. Wang, "Unsupervised domain adaptation for semantic segmentation via class-balanced self-training" in *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 289-305, 2018.

[21] H.D. Lee, "Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks," in *Workshop on Challenges in Representation Learning*, 2013.

[22] O. Sener, H.O. Song, A. Saxena, and S.Savarese, "Learning transferrable representations for unsupervised domain adaptation," in *Proceeding of the IEEE Conference on Advances in Neural Information Processing Systems (NIPS)*, pages 2110-2118, 2016.

[23] K. Saito, Y. Ushiku, and T. Harada, "Asymmetric tri-training for unsupervised domain adaptation," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70 (ICML)*, pages 2988-2997, 2017.

[24] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F.Laviolette, and V.Lempitsky, "Domain-adversarial training of neural networks," in *The Journal of Machine Learning Research*, 2016.

[25] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, "Gradient-based learning applied to document recognition," in *Proceeding of the IEEE Conference on Advances in Neural Information Processing Systems (NIPS)*, pages 2278-2324, 1998.

[26] J.S. Denker, W.R. Gardner, H.P. Graf, D. Henderson, R.E. Howard, W. Hubbard, and I.Guyon, "Neural network recognizer for hand-written zip code digits," in *Proceeding of the IEEE Conference on Advances in Neural Information Processing Systems (NIPS)*, pages 323-331, 1989.

[27] Y. Netzer, T. Wang, A. Coated, A. Bissacco, B. Wu, and A.Y. Ng, "Reading digits in natural images with unsupervised feature learning," in *Proceedings of the IEEE Conference on Advances in Neural Information Processing Systems Workshop (NIPSW)*, volume 2011, page 5, 2011.