Supplementary Material: Visual Domain Adaptation with Manifold Embedded Distribution Adaptation*

Jindong Wang, Wenjie Feng, Yiqiang Chen, Han Yu, Meiyu Huang, Philip S. Yu

Institute of Computing Technology, Chinese Academy of Sciences

Update

We notice that there are a huge number of deep transfer learning methods appear with new state-of-the-art results. Hence, we add this section to keep track of the latest results. We add the results on *Office-31, Office-Home, and Image-CLEF DA datasets.*

From the results, we can see that MEDA **significantly** outperforms all existing deep adversarial methods on Office-Home dataset (by 4.2%). On Office-31 and Image-CLEF DA datasets, the improvement is marginal. However, note that MEDA only needs ResNet to perform finetune on the source domaind data. Then, the finetuned model can easily extract features for the source and target domain. Compared to other methods that need to train the whole model with several extra hyperparameters to tune, MEDA is significantly improving the baseline results..

1. Office-31 dataset

Table 1: Classification accuracy (%) on Office-31 dataset by following protocol in [22] using ResNet50 features (deep methods are with ResNet50 as backbone)

	Method	$A \rightarrow D$	$A \rightarrow W$	$D \rightarrow A$	$D \rightarrow W$	$W \rightarrow A$	$W \rightarrow D$	Average
cvpr16	ResNet-50	68.4	96.7	99.3	68.9	62.5	60.7	76.1
icml15	DAN [9]	80.5	97.1	99.6	78.6	63.6	62.8	80.4
nips16	RTN [11]	84.5	96.8	99.4	77.5	66.2	64.8	81.6
icml15	DANN [5]	82.0	96.9	99.1	79.7	68.2	67.4	82.2
cvpr17	ADDA [19]	86.2	96.2	98.4	77.8	69.5	68.9	82.9
icml17	JAN [12]	85.4	97.4	99.8	84.7	68.6	70.0	84.3
cvpr17	GTA [16]	89.5	97.9	99.8	87.7	72.8	71.4	86.5
aaai18	MADA [14]	90.1	97.4	99.6	87.8	70.3	66.4	85.2
cvpr18	CAN [21]	81.5	63.4	85.5	65.9	99.7	98.2	82.4
aaai19	JDDA [3]	82.6	95.2	99.7	79.8	57.4	66.7	80.2
ACMMM18	Our MEDA	86.2	85.9	72.3	97.4	73.4	99.4	85.8

2. Office-Home dataset:

3. Image-CLEF DA dataset:

Now the original supplementary:

1 Classification Accuracy on Office-31 Dataset

We check the performance of MEDA in another widely-used dataset: Office-31 [15]. Office-31 is a standard benchmark for domain adaptation tasks in multimedia analysis. It consists of 4,652 images and 31 categories from three domains: Amazon (A), Webcam (w), and DSLR (D). Each two domain can construct a domain adaptation task, leading to 6 tasks: $A \rightarrow D$, $A \rightarrow W$, $\cdots$, $W \rightarrow D$. We follow the protocol in [22] and evaluate all the methods in these tasks.

Table 4 shows the classification of MEDA and other state-of-the-art traditional and deep domain adaptation methods. Note that MEDA and other traditional methods used the DeCaf7 features, while deep methods used the original

*The oroginal paper can be found at `https://arxiv.org/abs/1807.07258`. Contact: wangjindong@ict.ac.cn

Table 2: Classification accuracy (%) on Office-Home dataset using ResNet50 features (deep methods are with ResNet50 as backbone)

Method	Ar → Cl	Ar → Pr	Ar → Rw	Cl → Ar	Cl → Pr	Cl → Rw	Pr → Ar	Pr → Cl	Pr → Rw	Rw → Ar	Rw → Cl	Rw → Pr	Avg
AlexNet	26.4	32.6	41.3	22.1	41.7	42.1	20.5	20.3	51.1	31.0	27.9	54.9	34.3
DAN [9]	31.7	43.2	55.1	33.8	48.6	50.8	30.1	35.1	57.7	44.6	39.3	63.7	44.5
DANN [5]	36.4	45.2	54.7	35.2	51.8	55.1	31.6	39.7	59.3	45.7	46.4	65.9	47.3
JAN [12]	35.5	46.1	57.7	36.4	53.3	54.5	33.4	40.3	60.1	45.9	47.4	67.9	48.2
CDAN-RM [10]	36.2	47.3	58.6	37.3	54.4	58.3	33.2	43.9	62.1	48.2	48.1	70.7	49.9
CDAN-M [10]	38.1	50.3	60.3	39.7	56.4	57.8	35.5	43.1	63.2	48.4	48.5	71.1	51.0
ResNet-50	34.9	50.0	58.0	37.4	41.9	46.2	38.5	31.2	60.4	53.9	41.2	59.9	46.1
DAN [9]	43.6	57.0	67.9	45.8	56.5	60.4	44.0	43.6	67.7	63.1	51.5	74.3	56.3
DANN [5]	45.6	59.3	70.1	47.0	58.5	60.9	46.1	43.7	68.5	63.2	51.8	76.8	57.6
JAN [12]	45.9	61.2	68.9	50.4	59.7	61.0	45.8	43.4	70.3	63.9	52.4	76.8	58.3
CDAN-RM [10]	49.2	64.8	72.9	53.8	62.4	62.9	49.8	48.8	71.5	65.8	56.4	79.2	61.5
CDAN-M [10]	50.6	65.9	73.4	55.7	62.7	64.2	51.8	49.1	74.5	68.2	56.9	80.7	62.8
MEDA	**54.6**	**75.2**	**77.0**	**56.5**	**72.8**	**72.3**	**59.0**	**51.9**	**78.2**	67.7	**57.2**	**81.8**	**67.0**

Table 3: Classification accuracy (%) on Image-CLEF DA dataset using ResNet-50 as features(deep methods are with ResNet50 as backbone)

Method	I → P	P → I	I → C	C → I	C → P	P → C	Avg
AlexNet	66.2	70.0	84.3	71.3	59.3	84.5	73.9
DAN [9]	67.3	80.5	87.7	76.0	61.6	88.4	76.9
DANN [5]	66.5	81.8	89.0	79.8	63.5	88.7	78.2
JAN [12]	67.2	82.8	91.3	80.0	63.5	91.0	79.3
CDAN-RM [10]	67.0	84.8	92.4	81.3	64.7	91.6	80.3
CDAN-M [10]	67.7	83.3	91.8	81.5	63.0	91.5	79.8
ResNet-50	74.8	83.9	91.5	78.0	65.5	91.2	80.7
DAN [9]	74.5	82.2	92.8	86.3	69.2	89.8	82.5
DANN [5]	75.0	86.0	96.2	87.0	74.3	91.5	85.0
RTN [11]	75.6	86.8	95.3	86.9	72.7	92.2	84.9
JAN [12]	76.8	88.0	94.7	89.5	74.2	91.7	85.8
MADA [14]	75.0	87.9	96.0	88.8	75.2	92.2	85.8
CDAN-RM [10]	77.2	88.3	**98.3**	90.7	76.7	94.0	87.5
CDAN-M [10]	78.3	91.2	96.7	91.2	77.2	93.7	88.1
CAN [21]	78.2	87.5	94.2	89.5	75.8	89.2	85.7
iCAN [21]	**79.5**	89.7	94.7	89.9	78.5	92.0	87.4
MEDA	**79.7**	**92.5**	95.7	**92.2**	**78.5**	**95.5**	**89.0**

images. From these results, we can observe that MEDA still achieves the best classification accuracy. This indicates the effectiveness of MEDA over other methods.

2 Parameter Sensitivity

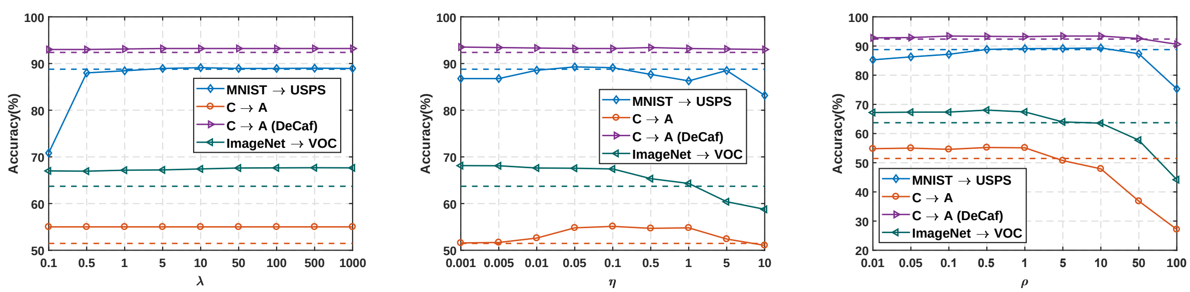
In order to check the regularization parameter sensitivity of MEDA, we run MEDA with a wide range of parameter values for λ, η, and ρ on several selected tasks. Then we compare its performance with the best baseline method. The results are in Figure 1. Those results indicate that MEDA can achieve a **robust** performance with regard to a wide range of parameter values.

Specifically, the best choices of those parameters are: $\lambda \in [0.5, 1000], \eta \in [0.01, 1]$, and $\rho \in [0.01, 5]$. To sum up, the performance of MEDA stays robust with a large range of regularization parameter choice. Therefore, the parameters do not need to be fine-tuned in practical applications.

Remark: Although MEDA is easy to use, and its parameters do not have to be fine-tuned, in addition, for research purpose, we also interpret how to further tune those parameters. We choose parameters according to following rules: 1st, SRM on source domain is very important, so we tend to choose a small η to make sure MEDA does not degenerate. 2nd, distribution adaptation is inevitable in SRM, so we choose a slightly larger λ to make it count. 3rd, we choose ρ by following [1]. 4th, p is set following [2].

Table 4: Classification accuracy (%) on Office-31 dataset by following protocol in [22] using DeCaf6 features (deep methods are with AlexNet as backbone)

Method	A → D	A → W	D → A	D → W	W → A	W → D	Average
SVM	55.7	50.6	46.5	93.1	43.0	97.4	64.4
TCA [13]	45.4	40.5	36.5	78.2	34.1	84.0	53.1
GFK [7]	52.0	48.2	41.8	86.5	38.6	87.5	59.1
SA [4]	46.2	42.5	39.3	78.9	36.3	80.6	54.0
DANN [6]	34.0	34.1	20.1	62.0	21.2	64.4	39.3
CORAL [17]	57.1	53.1	51.1	94.6	47.3	98.2	66.9
AlexNet [8]	63.8	61.6	51.1	95.4	49.8	99.0	70.1
DDC [20]	64.4	61.8	52.1	95.0	52.2	98.5	70.6
DAN [9]	67.0	68.5	54.0	96.0	53.1	99.0	72.9
RTN [11]	71.0	**73.3**	50.5	**96.8**	51.0	99.6	73.7
DCORAL [18]	66.4	66.8	52.8	95.7	51.5	99.2	72.1
DUCDA [22]	68.3	68.3	53.6	96.2	51.6	**99.7**	73.0
MEDA	**69.5**	69.9	**58.0**	94.0	**56.0**	96.8	**74.0**



(a) λ (b) η (c) ρ

Figure 1: Parameter sensitivity on selected tasks: (a) λ, (b) η, (c) ρ.

3 Detailed Results of the Estimation of Adaptive Factor μ

The evaluation of the adaptive factor μ is important for the dynamic distribution adaptation. Other than the results in the main paper (Section 4.5.2), we list all the results of our estimation of μ in both Table 5 and Table 6. The results clearly demonstrated the effectiveness of our solution in quantitatively estimating the adaptive factor.

Although most of the $\hat{\mu}$ were close to μ_{opt}, we also noticed that there was relatively larger gap between $\hat{\mu}$ and μ_{opt} on two tasks (i.e. C → D and U → M). It is interesting to observe that the estimations on their opposite (i.e. D → C and M → U if we change the source and target) were satisfactory. We think this is probably because of the high variance of the target domain. We will continue to explore more properties of this phenomenon and improve the estimation.

Table 5: Comparison between μ_{opt} and $\hat{\mu}$ on Office+Caltech10 datasets using both SURF and DeCaf features

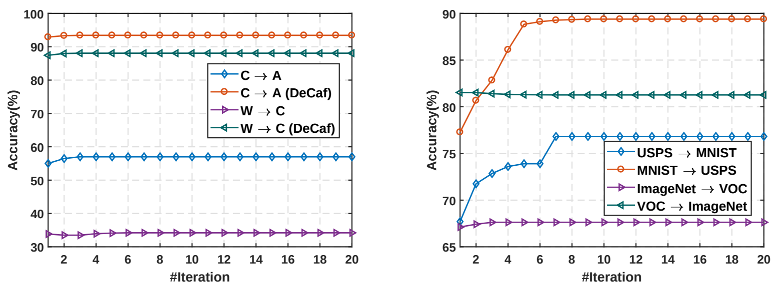
Feature	Task	C → A	C → W	C → D	A → C	A → W	A → D	W → C	W → A	W → D	D → C	D → A	D → W	Avg
SURF	μ_{opt}	56.9	55.3	58.6	45.0	53.2	47.8	34.2	43.0	89.2	35.8	41.5	87.8	54.0
	$\hat{\mu}$	56.5	53.9	50.3	43.9	53.2	45.9	34.0	42.7	88.5	34.9	41.2	87.5	52.7
	Performance Variation	-0.7%	-2.5%	-14.2%	-2.4%	0.0%	-4.0%	-0.6%	-0.7%	-0.8%	-2.5%	-0.7%	-0.3%	-2.4%
DeCaf	μ_{opt}	93.4	95.6	91.7	87.4	88.1	91.7	88.0	93.2	99.4	87.6	93.2	98.0	91.2
	$\hat{\mu}$	93.4	95.6	91.1	87.4	88.1	88.1	93.2	99.4	99.4	87.5	93.2	97.6	92.8
	Performance Variation	0.0%	0.0%	-0.7%	0.0%	0.0%	-3.9%	**+5.9%**	**+6.7%**	0.0%	-0.1%	0.0%	-0.4%	**+1.8%**

Table 6: Comparison between μ_{opt} and $\hat{\mu}$ on USPS+MNIST and ImageNet+VOC2007 datasets

Task	U → M	M → U	V → I	I → V	AVG
μ_{opt}	76.8	89.4	81.2	67.6	78.8
$\hat{\mu}$	72.1	89.5	74.4	67.3	75.8
Performance Variation	-7.20%	**+0.14%**	-0.58%	-1.33%	-3.8%

4 Detailed Results of Empirical Convergence Analysis

We also validate the convergence of MEDA through empirical analysis. We run randomly selected tasks of Office+Caltech datasets and all tasks of USPS+MNIST / ImageNet+VOC datasets. The total iteration number is set to be $T = 20$. Figure 2 presents the results. From those results, we can clearly see that MEDA could reach a steady performance in only a few $(T < 10)$ iterations.



(a) Office+Caltech (b) USPS+MNIST and ImageNet+VOC

Figure 2: Empirical convergence analysis on all datasets.

5 Analysis of Pseudo-labeling

We understand the importance of pseudo-labeling. If the base classifier is very weak, or the pseudo labels are not confident, will that affect the results?

We validate the the robustness of the pseudo-labeling through experiments. We use different base classifiers: 1-NN, 3-NN, SVM-c10, SVM-c100, and Random guessing. Since different classifiers has different learning abilities and will lead to different pseudo-labeling results, we see how that will affect the final results of the MEDA method. The experimental results are in Figure 5 and Table 7.

According to these results, although different classifiers are used to generate different confidence of the pseudo-labeling, our MEDA is **significantly robust** to the classifiers. Therefore, MEDA is very easy to use. (Even random guessing could lead to comparable performance. However, since random guessing is not a technical method, we prefer to use 1-NN)

Table 7: Results of different base classifier used in MEDA.

Base classifier	**Accuracy of base classifier**	**Accuracy of MEDA**
1-NN	45.20	56.99
3-NN	45.41	55.64
SVM-c10	31.62	54.90
SVM-c100	52.30	55.84
Random	9.60	55.53

References

[1] Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of machine learning research*, 7(Nov):2399–2434, 2006.

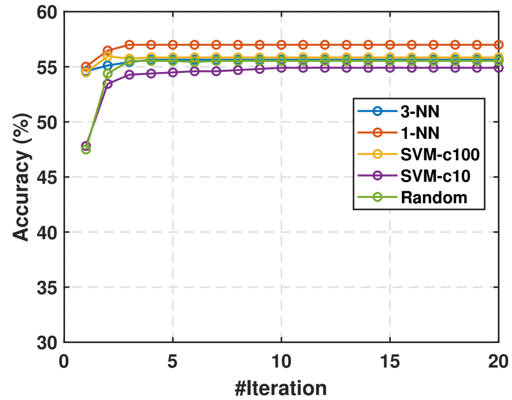


Figure 3: Results of different base classifiers used in MEDA on C $\rightarrow$ A task.

[2] Deng Cai, Xiaofei He, Jiawei Han, and Thomas S Huang. Graph regularized nonnegative matrix factorization for data representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1548–1560, 2011.

[3] Chao Chen, Zhihong Chen, Boyuan Jiang, and Xinyu Jin. Joint domain alignment and discriminative feature learning for unsupervised deep domain adaptation. In *AAAI*, 2019.

[4] Basura Fernando, Amaury Habrard, Marc Sebban, and Tinne Tuytelaars. Unsupervised visual domain adaptation using subspace alignment. In *Proceedings of the IEEE international conference on computer vision*, pages 2960–2967, 2013.

[5] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *ICML*, pages 1180–1189, 2015.

[6] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(59):1–35, 2016.

[7] Boqing Gong, Yuan Shi, Fei Sha, and Kristen Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2066–2073. IEEE, 2012.

[8] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[9] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. Learning transferable features with deep adaptation networks. In *ICML*, pages 97–105, 2015.

[10] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. In *Advances in Neural Information Processing Systems*, pages 1645–1655, 2018.

[11] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I. Jordan. Unsupervised domain adaptation with residual transfer networks. In *NIPS*, 2016.

[12] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Deep transfer learning with joint adaptation networks. In *International Conference on Machine Learning*, pages 2208–2217, 2017.

[13] Sinno Jialin Pan, Ivor W Tsang, James T Kwok, and Qiang Yang. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, 22(2):199–210, 2011.

[14] Zhongyi Pei, Zhangjie Cao, Mingsheng Long, and Jianmin Wang. Multi-adversarial domain adaptation. In *AAAI Conference on Artificial Intelligence*, 2018.

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

[16] Swami Sankaranarayanan, Yogesh Balaji, Carlos D Castillo, and Rama Chellappa. Generate to adapt: Aligning domains using generative adversarial networks. In *CVPR*, 2018.

[17] Baochen Sun, Jiashi Feng, and Kate Saenko. Return of frustratingly easy domain adaptation. In *AAAI*, volume 6, page 8, 2016.

[18] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *European Conference on Computer Vision*, pages 443–450. Springer, 2016.

[19] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Computer Vision and Pattern Recognition (CVPR)*, volume 1, page 4, 2017.

[20] Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell. Deep domain confusion: Maximizing for domain invariance. *arXiv preprint arXiv:1412.3474*, 2014.

[21] Weichen Zhang, Wanli Ouyang, Wen Li, and Dong Xu. Collaborative and adversarial network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3801–3809, 2018.

[22] Junbao Zhuo, Shuhui Wang, Weigang Zhang, and Qingming Huang. Deep unsupervised convolutional domain adaptation. In *Proceedings of the 2017 ACM on Multimedia Conference*, pages 261–269. ACM, 2017.