Shuang Li, Shi-ji Song, Cheng Wu, 2018. Layer-wise domain correction for unsupervised domain adaptation. *Frontiers of Information Technology & Electronic Engineering*, 19(1): 91-103. https://doi.org/10.1631/FITEE.1700774

Layer-wise domain correction for unsupervised domain adaptation

Key words: Unsupervised domain adaptation; Maximum mean discrepancy; Residual network; Deep learning

Corresponding author: Shi-ji SONG
E-mail: shijis@mail.tsinghua.edu.cn
ORCID: http://orcid.org/0000-0001-7361-9283

Motivation

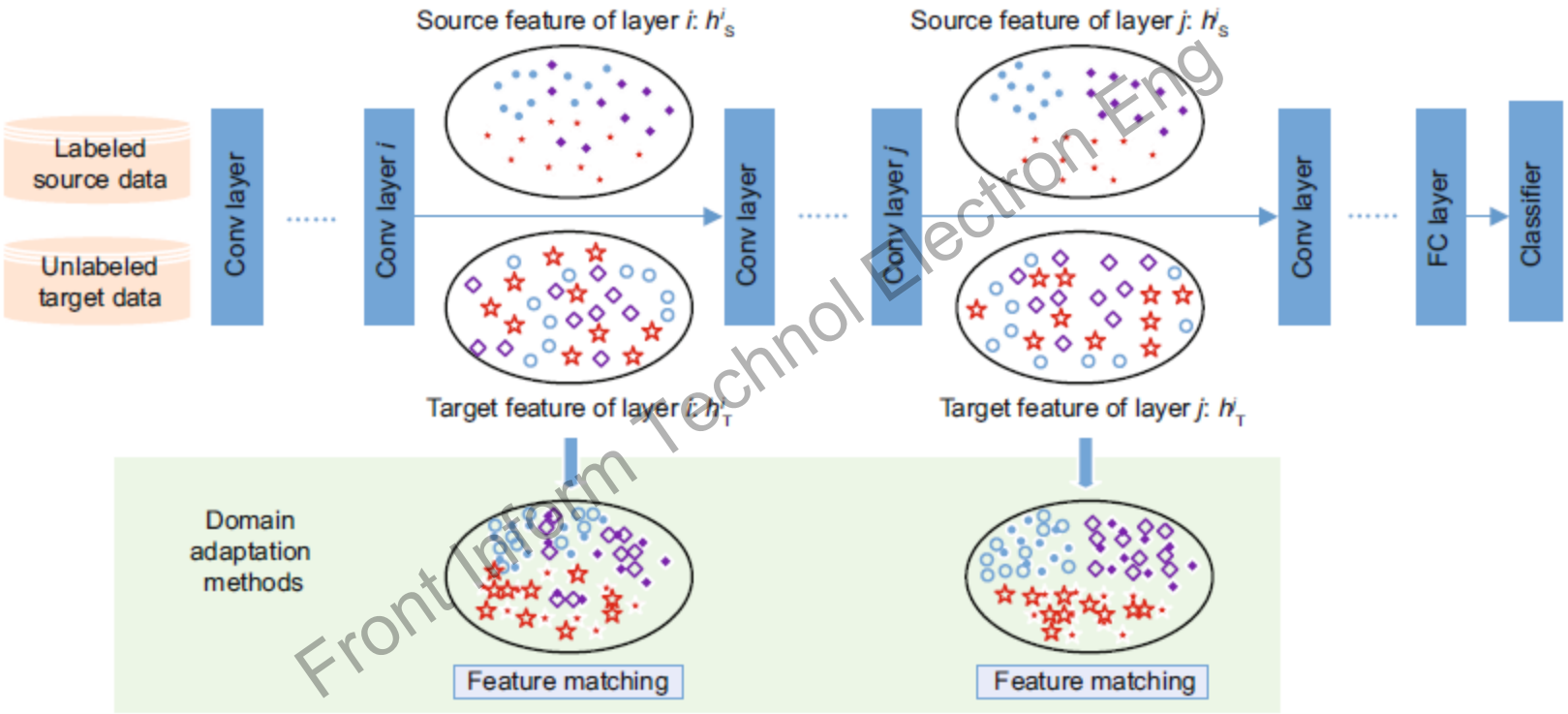


Fig. 1 Illustrating the aims of most deep domain adaptation methods

For the traditional methods, we can use labeled source data to train a perfect deep neural network. However, since the source and target data are under different distributions, for each layer, the representations of both domains, i.e., h^i_S and h^i_T, h^j_S and h^j_T, are divergent, which result in the dramatic degradation of classification performance on target data. Most deep domain adaptation methods (including LDC) aim to mitigate the domain discrepancy by different kinds of feature matching approaches, which can bridge both domains, and benefit the final target classification

Motivation

- We focus on the setting where both f_S and f_T are deep neural networks. The lack of labeled target data prevents us from training f_T directly. Instead, we adapt f_S to the target domain through additive 'correction' residual layers for target data.
- Our adaptation approach leverages the fact that deep neural networks learn their own internal data representation as well as a classifier. We 'correct' this internal representation for target data by adding small correction terms to the hidden representations to make them mimic source data.

Main idea

1. Let $h_S(\boldsymbol{x})$ be the representation of input $\boldsymbol{x}$ at a given layer in f_S. To adapt f_S, we must learn a hidden representation $h_T(\boldsymbol{x})$ for target data such that

$$P_S(h_S(\boldsymbol{x}), y) \approx P_T(h_T(\boldsymbol{x}), y)$$

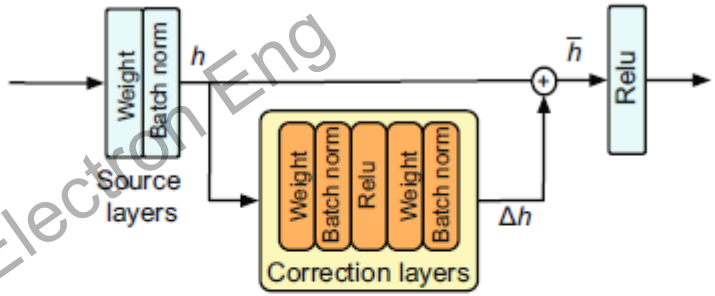


Fig. 2 Architecture of corrections added after a source network weight layer

Original source layers are blue, and correction layers are in the yellow box. References to color refer to the online version of this figure

2. We adapt the representation $h_S(\boldsymbol{x})$ to $h_T(\boldsymbol{x})$ with the help of an additive corrective term $\Delta h(\boldsymbol{x})$, i.e.

$$h_T(\boldsymbol{x}) = h_S(\boldsymbol{x}) + \Delta h(\boldsymbol{x})$$

3. We model $\Delta h(\boldsymbol{x})$ as a small multi-layer neural network as Fig. 2

Major results

- Datasets

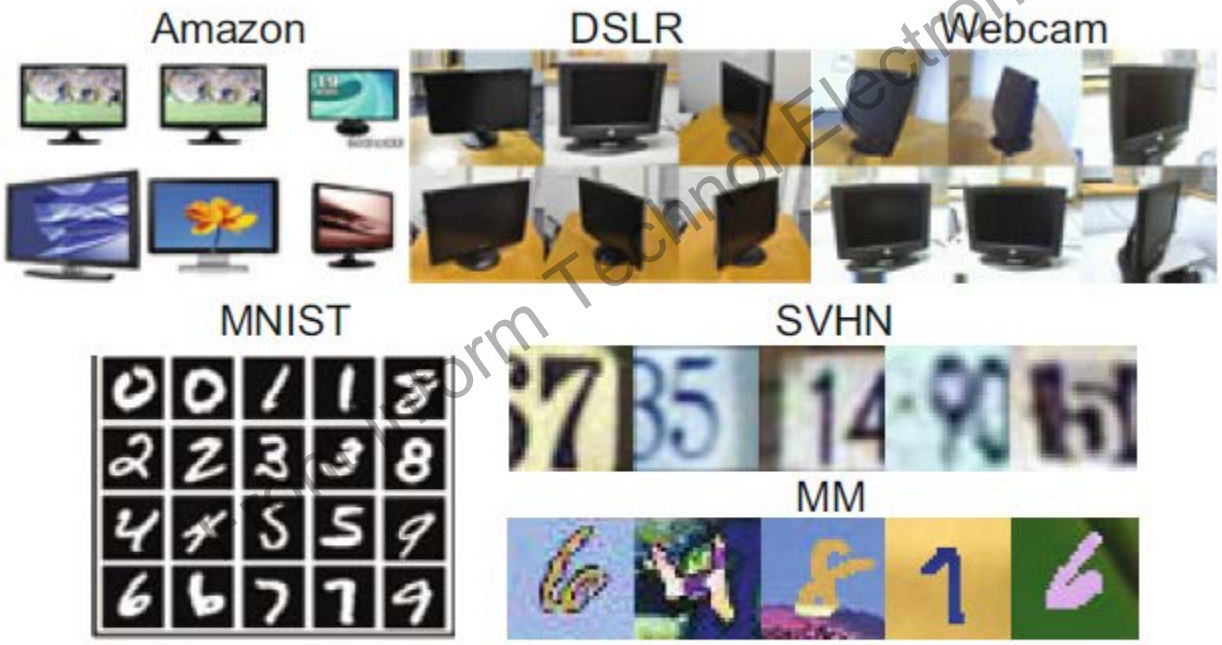


Fig. 3 Example images from different datasets (Amazon (A), DSLR (D), Webcam (W), MNIST, MM, SVHN)

Major results

- Experiments and results

Table 1 Classification accuracy on Office-31 dataset*

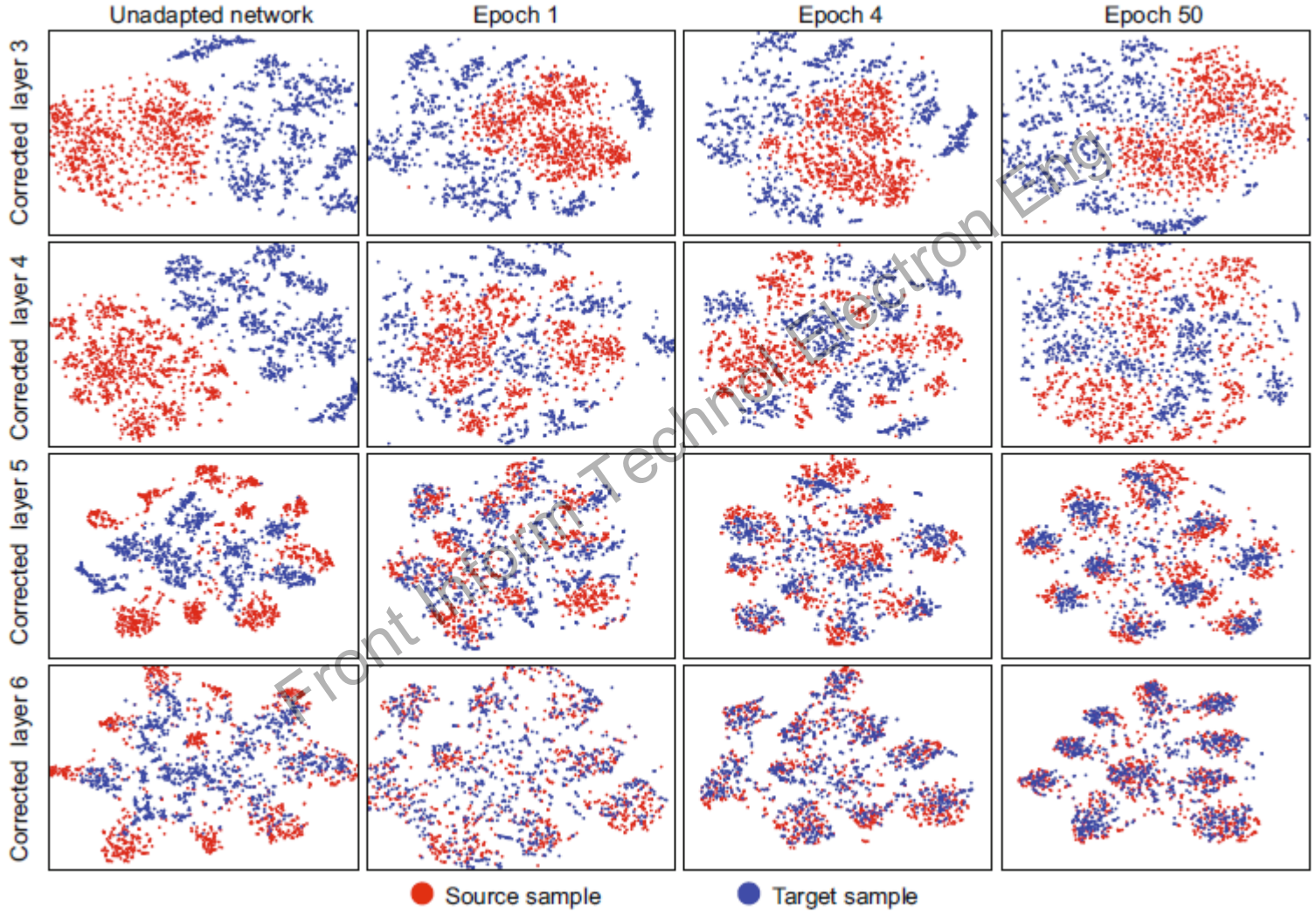
Method	Classification accuracy						
	A→W	D→W	W→D	A→D	D→A	W→A	Average
TCA (Pan et al., 2011)	59.0±0.0	90.2±0.0	88.2±0.0	57.8±0.0	51.6±0.0	47.9±0.0	65.8
GFK (Gong et al., 2012)	58.4±0.0	93.6±0.0	91.0±0.0	58.6±0.0	52.4±0.0	46.1±0.0	66.7
AlexNet (Krizhevsky et al., 2017)	60.6±0.4	95.4±0.2	99.0±0.1	64.2±0.3	45.5±0.5	48.3±0.5	68.8
ResNet18 (He et al., 2016)	65.7±0.2	95.9±0.1	98.3±0.2	69.8±0.2	51.7±0.1	49.0±0.3	71.7
DDC (Tzeng et al., 2014)	61.0±0.5	95.0±0.3	98.5±0.3	64.9±0.4	47.2±0.5	49.4±0.6	69.3
RevGrad (Ganin and Lempitsky, 2015)	73.0±0.6	96.4±0.4	99.2±0.3	72.8±0.4	54.4±0.3	53.6±0.2	74.9
DAN (Long et al., 2015)	68.5±0.3	96.0±0.1	99.0±0.1	66.8±0.2	50.0±0.4	49.8±0.3	71.7
RTN (Long et al., 2016b)	73.3±0.3	96.8±0.2	99.6±0.1	71.0±0.2	50.5±0.3	51.0±0.1	73.7
LDC-AlexNet	73.7±0.4	95.6±0.6	96.9±0.3	65.3±0.2	55.1±0.6	53.3±0.2	73.3
LDC-ResNet18	**78.6±0.5**	**98.7±0.1**	**100.0±0.0**	**79.1±0.3**	**61.9±0.3**	**59.6±0.5**	**79.7**

*: AlexNet is an AlexNet model, and ResNet18 is a ResNet18 model

Table 2 Classification accuracy on MNIST, MM, and SVHN datasets

Method	Classification accuracy			
	MNIST→MM	SVHN→MNIST	SVHN→MM	Average
Source-network	49.0±3.4	65.6±2.2	49.7±0.2	54.8
RevGrad (Ganin and Lempitsky, 2015)	81.0±1.0	73.5±0.5	54.9±1.1	69.8
LDC	**84.8±1.2**	**89.5±2.1**	**71.4±0.3**	**81.9**

Major results



t-SNE visualizations of different adapted layers at different stages of training from SVHN to MNIST

Conclusions

- We have proposed layer-wise domain correction, a new method for unsupervised domain adaptation. LDC leverages the power of residual networks to learn small layer-wise transformations which reduce the propagation of domain differences.

- In contrast to prior work, it does not retrain the weights of the original classifier from scratch and is therefore substantially faster during the adaptation time. In addition to the speedup, LDC requires substantially less storage for each target domain: only the weights of the corrected layers have to be stored in addition to the original network.

- The additional layers increase the capacity of the neural network, which may be a reason for its excellent generalization performance.