

RESEARCH

Open Access



Transfer deep convolutional activation-based features for domain adaptation in sensor networks

Zhong Zhang^{1,2*} and Donghong Li^{1,2}

Abstract

In this paper, we propose a novel method named transfer deep convolutional activation-based features (TDCAF) for domain adaptation in sensor networks. Specifically, we first train a siamese network with weight sharing to map the images from different domains into the same feature space, which can learn domain-invariant information. Since various feature maps in one convolutional layer of the siamese network contain different kinds of information, we propose a novel vertical pooling strategy to aggregate them into one convolutional activation summing map (CASM) which contains the completed information and preserves the spatial information. We stretch the CASM into one feature vector to obtain the TDCAF. Finally, we feed the proposed TDCAF into a Support Vector Machine (SVM) for classification. The proposed TDCAF is validated on three generalized image databases and three cloud databases, and the classification results outperform the other state-of-the-art methods.

Keywords: Sensor networks, Domain adaptation, Deep convolutional activations, Vertical pooling

1 Introduction

With the rapid development of wireless communications and electronics, the sensor networks have attracted much attention due to the great value for practical applications, such as health, weather forecast, military, and so on [1–3].

Imagine that we train a classifier with images captured from one sensor. Can we utilize the classifier to recognize objects captured from other sensors in the network, and hope the classifier still works well? Similarly, there are about 2424 weather stations distributed in China, and some of them are connected by a sensor network. Since the weather stations are distributed in various locations, and installed with different capturing equipment, the ground-based cloud images captured by them are various. Can we train a classifier with images from one weather station, and still hope the classifier works well on other weather stations? The above two problems are about training on one domain while testing on another domain. Here, a domain often refers to a database collected by one

sensor where samples belong to the same feature space and follow the same data distribution.

As for object recognition, many traditional classification methods have shown attractive performance on a specific sensor (domain) [4–6]. However, such methods often fail when presented with a novel domain, and a series of works have shown that when classifiers are evaluated outside of their training domains, the performance degrades significantly [7, 8]. In order to solve this problem, a desirable alternative is known as domain adaptation which has been extensively studied [9–13]. The objective of domain adaptation is to adapt classifiers trained on the source domain to the target domain for keeping acceptable performance. Here, the source domain contains a great amount of labelled samples so that a classifier can be reliably trained. The target domain usually includes a few labelled samples and a lot of unlabelled samples. The existing methods for domain adaptation mainly focus on deriving new domain-invariant representations. Gong et al. [14] proposed a kernel-based method to learn feature representations. They utilized a geodesic flow kernel to model the domain shift by integrating an infinite number of subspaces that characterize changes in geometric and statistical properties from the source domain to the

*Correspondence: zhong.zhang8848@gmail.com

¹Tianjin Key Laboratory of Wireless Mobile Communications and Power Transmission, Tianjin Normal University, Tianjin, China

²College of Electronic and Communication Engineering, Tianjin Normal University, Tianjin, China

target domain. Hoffman et al. [15] formed a linear transformation that mapped features from the target domain to the source domain. Baktashmotlagh et al. [16] introduced a domain invariant projection approach to extract the information that is invariant across the source and target domains. More recently, as convolutional neural networks (CNNs) have shown remarkable performance in image classification [17], researchers have applied CNNs to facilitate domain adaptation and obtained promising performance. Some researchers [18, 19] came up with the domain-adversarial approaches, where high-level representations from CNNs are optimized to minimize the loss on the source domain and maximize the loss on the domain classifier. Long et al. [20] proposed a deep adaptation network (DAN) architecture which generalizes deep convolutional neural network to the domain adaptation scenario.

Most domain adaptation algorithms focus on generalized classification tasks. However, as for ground-based cloud classification, there are a few literature studying domain adaptation when the cloud images are presented with the domain shift. What is more, successful classification of cloud types plays an important role in the research of climate change and meteorological services [21–23]. Therefore, this is an especially important problem to apply domain adaptation to ground-based cloud classification.

In this paper, we propose a novel method named transfer deep convolutional activation-based features (TDCAF) for domain adaptation in sensor networks. The proposed TDCAF can be applied to the generalized image classification and ground-based cloud classification. Specifically, we use unlabelled samples to train a siamese network that predicts similarity scores. The siamese network consists of two convolutional neural network (CNN) models, and we train the models with weight sharing, that is, the parameters of two CNNs are the same. The inputs of the siamese network are sample pairs where a sample is from one sensor (the source domain) and another one from another sensor (the target domain). Then we take deep convolutional activations in feature maps of the siamese network as features to represent images. We utilize deep convolutional activations for two reasons. First, many studies have shown that the convolutional activation-based features perform better than the fully connected layer-based features [24–26]. Second, convolutional activations in feature maps can be intuitively interpreted as local features of images. As for one convolutional layer, there are various feature maps which contain different kinds of information. In order to obtain completed information and preserve the spatial information, we propose a novel vertical pooling strategy named vertical pooling to aggregate these feature maps into one feature map which is defined as the convolutional activation summing map (CASM). Then the CASM is stretched

into one feature vector, and we define the resulting feature vector as the TDCAF. It should be noted that we demonstrate the effectiveness of the proposed TDCAF on the tasks of both the generalized image classification and ground-based cloud classification.

In summary, there are three main contributions of the proposed TDCAF.

(1) We fine-tune a siamese network with weight sharing to learn a similarity metric using unlabelled samples. The network forces the features from the source and target domains to the same feature space, which could learn the domain-invariant representations.

(2) We extract deep convolutional activations from feature maps as features, and employ the proposed vertical pooling to aggregate deep convolutional activations across all feature maps, so that we can extract completed features and preserve the spatial information of images.

The rest of this paper is organized as follows. Section 2 introduces the proposed TDCAF for domain adaptation in detail. Section 3 provides comprehensive analysis of the proposed TDCAF on generalized image databases and ground-based cloud image databases. We conclude this paper in Section 4.

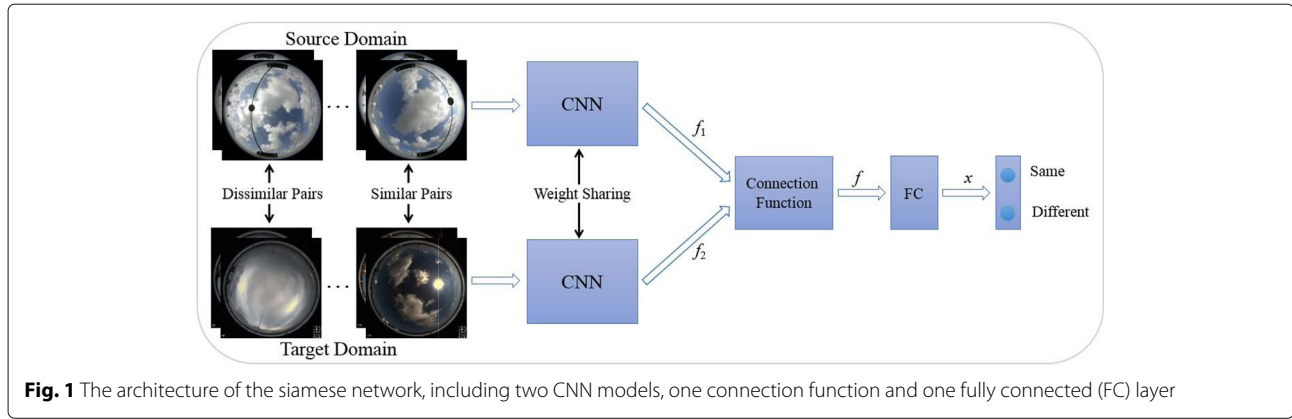
2 Method

In this section, we introduce the procedure of the proposed TDCAF. First, we utilize unlabelled samples to train a siamese network that predicts similarity scores for sample pairs. Then, we extract deep convolutional activation-based features from the trained siamese network. Finally, we apply the proposed vertical pooling strategy for the final feature representation.

2.1 The siamese network

Figure 1 briefly illustrates the architecture of the siamese network. Given a sample pair as the input, the siamese network predicts the similarity score of the two samples. The siamese network usually consists of two CNN models, one connection function and one fully connected (FC) layer. We utilize sample pairs to train the siamese network with weight sharing. Here, sample pairs consist of similar pairs and dissimilar pairs. When two samples from the source domain and the target domain belong to the same class (the different classes), we define them as a similar pair (a dissimilar pair). We utilize such sample pairs to train the siamese network so that the samples from different domains can be forced to the same feature space and could learn some domain-invariant characteristics. The CNN model can be CaffeNet [17], VGG19 [27], or ResNet-50 [28], where we change the number of kernels in the final FC layer according to the number of classes for fine-tuning the networks.

It should be noted that we train the siamese network with weight sharing. That means the trainable parameters



used in the two CNN models are the same. The weight sharing strategy is an important principle in the siamese network because it helps to reduce the total number of trainable parameters. Furthermore, weight sharing leads to more efficient training and more effective model especially when some similar local structures appear in the input feature space.

The connection function is used to connect the output vectors of the two CNN models. In our model, we define the connection function as

$$f = (f_1 - f_2)^2 \quad (1)$$

where f_1 and f_2 are the output vectors of the two CNN models, respectively, and they are both 1024-dim vectors. f is the 1024-dim output vector of the connection function.

As shown in Fig. 1, we then take f as the input of the FC, and the resulting vector x can be expressed as

$$x = \theta \circ f \quad (2)$$

where \circ denotes the convolutional operation. θ is the parameters of the FC, and its dimension is 1024.

Since this is a binary classification problem, we utilize the final layer to convert x to a 2-dim vector (z_1, z_2) which is then fed into the softmax function to obtain the predicted probability of the input sample pair belonging to the same class. The formulation of the softmax function is

$$\hat{p}_i = \frac{e^{z_i}}{\sum_{k=1}^2 e^{z_k}} \quad (3)$$

where \hat{p}_i is the predicted probability, and $\hat{p}_1 + \hat{p}_2 = 1$.

Finally, we use the cross-entropy loss for this binary classification

$$Loss = \sum_{i=1}^2 -p_i \log(\hat{p}_i) \quad (4)$$

where p_i is the true probability. As for a similar pair, $p_1 = 1$, and $p_2 = 0$. While for a dissimilar pair, $p_1 = 0$, and $p_2 = 1$.

In the forward propagation, according to Eq. (3), Eq. (4) can be reformulated as

$$Loss = -p_1 \log \frac{e^{z_1}}{\sum_{k=1}^2 e^{z_k}} - p_2 \log \frac{e^{z_2}}{\sum_{k=1}^2 e^{z_k}} \quad (5)$$

As for a similar pair, i.e., $p_1 = 1$, and $p_2 = 0$. Equation (5) can be rewritten as

$$Loss = -\log \frac{e^{z_1}}{\sum_{k=1}^2 e^{z_k}} \quad (6)$$

and as for a dissimilar pair, i.e., $p_1 = 0$, and $p_2 = 1$. Equation (5) is reformulated as

$$Loss = -\log \frac{e^{z_2}}{\sum_{k=1}^2 e^{z_k}} \quad (7)$$

We adopt the mini-batch stochastic gradient descent (SGD) [29] and error backpropagation algorithm (BP) to train the siamese network. In the backpropagation, we take the derivative of Eqs. (6) and (7) with respect to z_1 and z_2 , respectively, and obtain

$$Loss' = \frac{e^{z_1}}{\sum_{k=1}^2 e^{z_k}} - 1 \quad (8)$$

$$Loss' = \frac{e^{z_2}}{\sum_{k=1}^2 e^{z_k}} - 1 \quad (9)$$

Generally, since a large number of trainable parameters should be learned for CNN, an effective model requires lots of training samples. If we train a CNN model with insufficient training samples, it would lead to overfitting. To address the problem, we train the siamese network by fine-tuning the pre-trained CNN model.

2.2 Transfer deep convolutional activation-based features

The convolutional layers are main components of the CNN model and can capture more local image characteristics [30, 31]. Hence, we extract deep convolutional activation-based features from a certain convolutional layer to represent images. Suppose that there are N

feature maps from a certain convolutional layer of the CNN model. As shown in Fig. 2a, different feature maps tend to have various activations for the same image, meaning that these feature maps describe different patterns. Hence, in order to obtain completed features, all feature maps for a convolutional layer should be considered for the image representation. Traditional methods aggregate all convolutional activations from one feature map into one activation value as shown in Fig. 2a. The resulting activation value is chosen from the maximum value or average value of one feature map. Then, the activation value of each feature map is concatenated into a N -dim feature vector. However, the feature vector is insensitive to spatial distribution variation.

To address this problem, we propose the vertical pooling strategy, which contains the sum operation or the max operation. As for the sum operation, the deep convolutional activations at the same position of all feature maps are added, resulting in the CASM with the size of $H \times W$. Then the CASM is straightened into a $(H \times W)$ -dim TDCAF which thus obtains completed information and preserves the spatial information of images. For describing the sum operation more clearly, let $f^n(a, b)$ be the convolutional activations at position (a, b) from the n -th feature map f^n , and the sum-operation feature $F_s(a, b)$ at this position is defined as

$$F_s(a, b) = \sum_{n=1}^N f^n(a, b) \quad (a \in H, b \in W) \quad (10)$$

Then an image can be represented as $F_s = \{F_s(1, 1), F_s(1, 2), \dots, F_s(a, b)\}$. The process is shown in Fig. 2b.

Similarly, as for the max operation, we preserve the maximum convolutional activation at the same position for all feature maps where the resulting convolutional activation is salient and more robust to local transformations. The max-operation feature $F_m(a, b)$ is formulated as

$$F_m(a, b) = \max_{1 \leq n \leq N} f^n(a, b) \quad (11)$$

and an image can be represented as $F_m = \{F_m(1, 1), F_m(1, 2), \dots, F_m(a, b)\}$.

3 Experimental results

In this section, we first introduce the databases and the experimental setup. Then, we validate the effectiveness of the proposed TDCAF on generalized image databases and ground-based cloud image databases.

3.1 Databases and experimental setup

As for the generalized image classification, we utilize three generalized image databases, i.e., Amazon (images downloaded from online merchants), Webcam (low-resolution

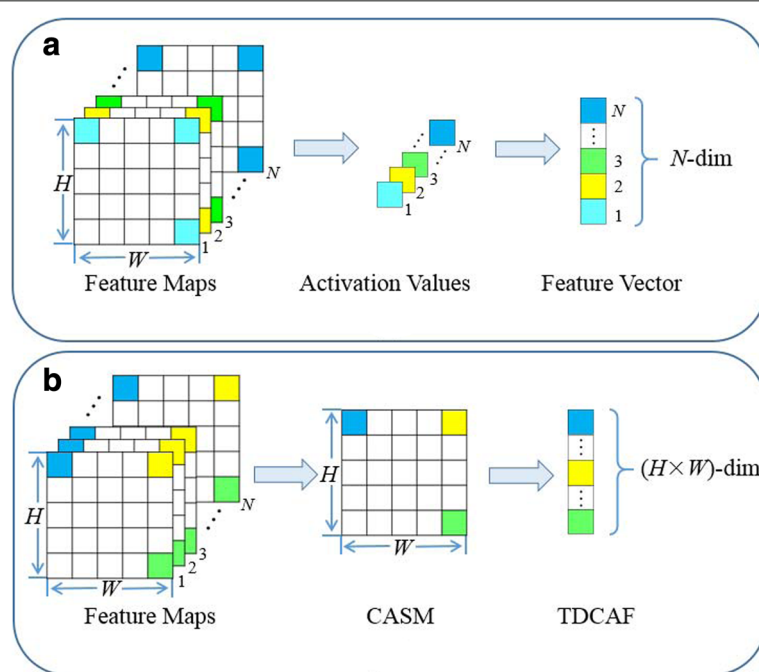


Fig. 2 The flowcharts of feature extraction of **a** traditional methods and **b** the TDCAF method



Fig. 3 Example images from the “laptop_computer” class in **a** Amazon, **b** Webcam, and **c** DSLR

images by a web camera), and DSLR (high-resolution images by a digital SLR camera). Each of them contains 31 classes, and the total number of the three databases is 2818, 498, and 795, respectively. The sizes of the images are 300×300 , 445×445 , and 1000×1000 , respectively. Figure 3 shows example images of the “laptop_computer” class from the three domains. We can see that these images possess significant differences in many aspects, such as captured conditions, resolutions, and viewpoints. Hence, the three databases belong to different domains.

As for the ground-based cloud classification, there are three cloud databases collected by different weather stations. According to the international cloud classification system published in World Meteorological Organization (WMO), the cloud images are separated into seven classes. Figure 4 shows the differences among these domains with example cloud images of each class. The

first cloud database is the IAP_e database, which is provided by the Institute of Atmospheric Physics, Chinese Academy of Sciences. The cloud images in this database are captured in Yangjiang, Guangdong Province, China, and have 2272×1704 pixels with strong illuminations and some occlusions. The second cloud database is the CAMS_e database, which is provided by Chinese Academy of Meteorological Sciences. The cloud images in this database are captured in the same location as the IAP_e database, but the acquisition device is different from that of the IAP_e database. The size of cloud image in this database is 1392×1040 pixels with weak illuminations and no occlusion. The third cloud database is the MOC_e database, which is provided by Meteorological Observation Centre, China Meteorological Administration. Different from the first two cloud databases, the cloud images in this database are taken in Wuxi, Jiangsu Province, China.

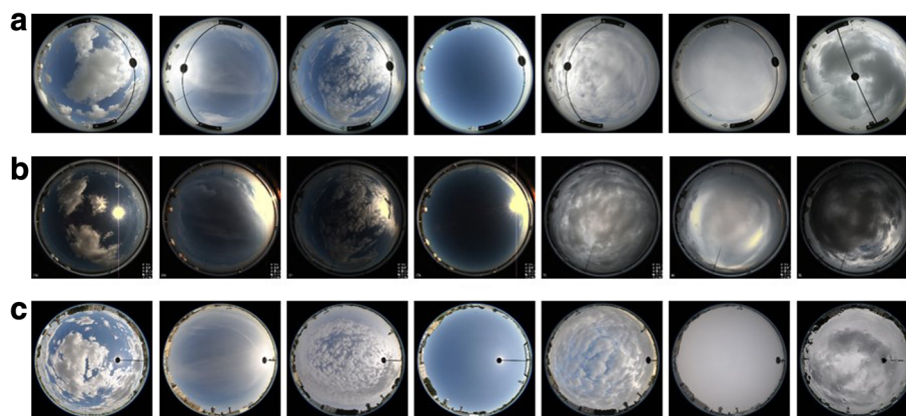


Fig. 4 Cloud samples from **a** the CAMS_e database, **b** the IAP_e database, and **c** the MOC_e database

Table 1 Configurations of the ResNet-50

Config.	ResNet-50	Padding
conv_1	$\begin{bmatrix} 7 \times 7, 64 \end{bmatrix} \times 1, \text{stride } 2$	3
	$3 \times 3, \text{max pooling, stride } 2$	0
conv_2	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3, \text{stride } 2$	$\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$
conv_3	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4, \text{stride } 2$	$\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$
conv_4	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6, \text{stride } 2$	$\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$
conv_5	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3, \text{stride } 2$	$\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$
fc	Average pooling, 1000-d, softmax	

The left part in "[]" indicates the size of receipt fields and the right part indicates the number of filter banks. Max pooling is implemented by a 3×3 pixel window. Both the convolution stride and max pooling stride are set to two pixels. The fully connected (FC) layer has 1000 channels

Moreover, the cloud images have 2828×4288 pixels with strong illuminations and some occlusion. It is obvious that cloud images from the three cloud databases vary in locations, illuminations, occlusions, and resolutions. Hence, they belong to different domains. The total number of the three databases is 3533, 2491, and 2107, respectively.

We adopt ResNet-50 [28] as the CNN model in the following experiments, and the configurations of the ResNet-50 are outlined in Table 1. We resize all images to 224×224 . When we train the siamese network, the training images consist of two parts, namely, all source domain images and half of images in each class from the target domain. The remaining images of the target domain are used for test. We implement experiments independently for 10 times and the final results represent the average accuracy over these 10 times. As the inputs of the siamese

network, the ratio between similar pairs and dissimilar pairs is 1:1. The number of training epochs is set to 75. The learning rate is initialized as 0.001 and then set to 0.0001 for the final five epochs. We adopt the SGD [29] to update the parameters of the siamese network. We use NVIDIA TITAN XP GPU to implement the algorithm.

After obtaining the trained siamese network, we feed forward a 224×224 test image to any ResNet-50 in our network. Since the shallower convolutional layer contains more texture information, we extract the activations of conv_3 (see Table 1) which results in the feature map with the size of 32×32 . Therefore, the dimensionality of the proposed TDCAF is 1024. In other words, each test image is represented as an 1024-dim vector. It should be noted that we test the sum operation or the max operation for feature representation. Finally, the feature vectors of all test images from the target domain are fed into a SVM classifier with the radial basis function (RBF) kernel for classification.

3.2 Generalized image classification

We compare the proposed TDCAF with two excellent methods [11, 14]. As for the two methods, we follow the abovementioned experimental settings. The recognition results are listed in Table 2. The proposed TDCAF significantly improves the performances over the two competing methods. It is because we simultaneously learn the domain-invariant information from the source and target domains by using a sharing weight strategy. Furthermore, the deep learning model could learn highly nonlinear features which are beneficial to the domain adaptation problem. The proposed TDCAF achieves the highest accuracy of 73.4% in the situation of DSLR \rightarrow Webcam. This is reasonable as these two domains are similar. As for the proposed TDCAF, in most of cases, the max operation outperforms the sum operation, possibly because there are more discriminative information contained by using the max operation.

3.3 Ground-based cloud classification

We compare the proposed TDCAF with two state-of-the-art methods, i.e., the bag-of-words model (BoW) [32]

Table 2 Classification accuracies (%) for the generalized image classification

Source	Target	Kate et al. [11]	Gong et al. [14]	TDCAF	
				Sum operation	Max operation
Amazon	Webcam	43.5	35.7	69.6	70.1
Amazon	DSLR	29.4	35.8	67.6	66.5
DSLR	Amazon	28.2	36.1	68.7	69.3
DSLR	Webcam	31.8	49.6	73.4	73.2
Webcam	Amazon	42.9	35.5	71.8	72.4
Webcam	DSLR	27.6	49.7	70.3	71.5

Table 3 Classification accuracies (%) for the ground-based cloud classification

Source	Target	BoW	CLBP	TDCAF	
				Sum operation	Max operation
IAP_e	CAMS_e	39.2	41.4	70.5	70.9
IAP_e	MOC_e	41.6	43.5	73.6	74.8
CAMS_e	IAP_e	36.1	37.6	68.5	79.4
CAMS_e	MOC_e	35.4	38.5	68.3	67.6
MOC_e	IAP_e	37.5	40.9	69.2	70.5
MOC_e	CAMS_e	38.6	41.3	68.4	69.7

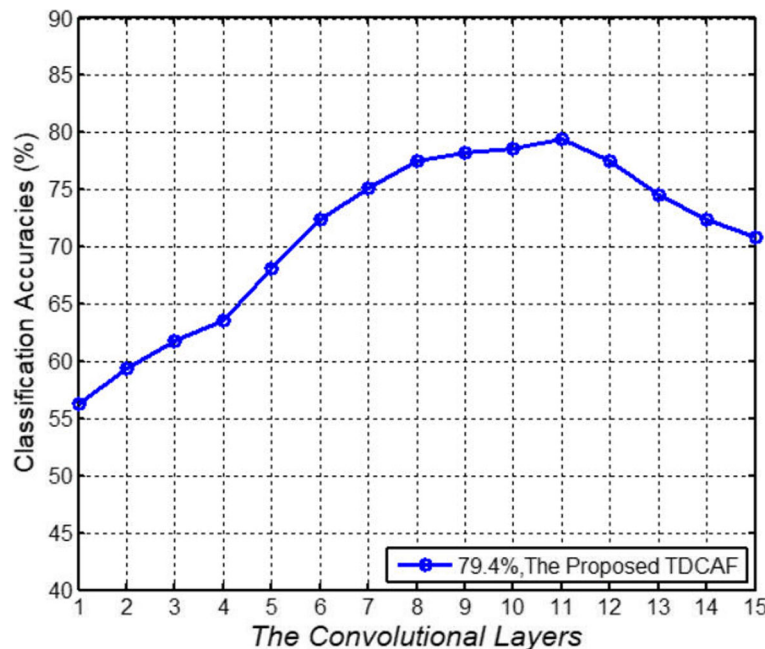
and the completed local binary patterns (CLBP) [33]. For the BoW, we first extract patch features for each cloud image. Each patch feature is an 81-dim vector, which is formed by stretching a 9×9 neighborhood around each pixel. Then, we learn a dictionary with K -means clustering [34] over patch vectors. The size of dictionary for each class is set to be 200, which results in an 1400-dim vector for each cloud image. For the CLBP, there are three operators, i.e., CLBP_C, CLBP_S, and CLBP_M, and we combine them hybridly. Specifically, we calculate a joint 2D histogram of CLBP_S and CLBP_C, and then the histogram is converted to a 1D histogram, which is then concatenated with CLBP_M to generate a joint histogram. The dimensionality of the CLBP for each cloud image is $(10 \times 2 + 10) + (18 \times 2 + 18) + (26 \times 2 + 26) = 162$. Finally, feature vectors of all cloud

images are fed into a SVM classifier with RBF kernel for classification.

The experimental results are listed in Table 3. It is obvious that the BoW and the CLBP do not deal with such the domain adaptation task well, while the proposed TDCAF can obtain better performances. The reason is that the siamese network is trained by sample pairs and weight sharing, and the proposed TDCAF could learn domain-invariant information. Moreover, the vertical pooling could obtain the completed and spatial information by aggregating features across all the feature maps. The proposed TDCAF contains two operations, i.e., the sum operation and the max operation. From Table 3, we can see that the max operation achieves better performance than the sum operation in most situations. It is because there are generally some interferences and noises in the cloud images, and the max operation could select the salient and discriminative features. It should be noted that when we take the CAMS_e database as the source domain, and the MOC_e database as the target domain, we obtain the poorest performance of 67.6%. The reason is that the CAMS_e database is greatly different from the MOC_e database in illuminations, capturing locations, occlusions, and image resolutions.

3.4 Influence of parameter variances

We take the CAMS_e to IAP_e shift as an example to analyze the proposed TDCAF in the aspect of the selection of the convolutional layers for extracting features. In

**Fig. 5** Classification accuracies (%) of the proposed TDCAF in different convolutional layers on the CAMS_e to IAP_e shift

a CNN, the shallow convolutional layers usually contain structural and textural local features while the deep convolutional layers usually contain high-level semantic information. Since the appearance of clouds can be treated as a kind of natural texture, we select the shallow convolutional layers for feature representation. We evaluate the performance of the proposed TDCAF when using different convolutional layers. The index of convolutional layers varies from 1 to 15, as shown in Fig. 5. From Fig. 5, the experimental results indicate that the highest result of the proposed method is obtained when we utilize the 11th convolutional layer in conv_3 for feature representation.

4 Conclusion

In this paper, we have introduced an effective domain adaptation method TDCAF for the generalized classification and the ground-based cloud classification in sensor networks. We utilize sample pairs to train a siamese network with weight sharing, and therefore, the siamese network could learn the domain-invariant information from the source and target domains. We employ the vertical pooling to obtain the TDCAF from all feature maps of one convolutional layer, which includes completed and spatial information. We have conducted experiments to verify the proposed TDCAF on three generalized image databases, i.e., Amazon, Webcam, and DSLR, and three cloud databases, i.e., IAP_e, CAMS_e, and MOC_e. Comparing with the state-of-the-art methods, the classification accuracies demonstrate the effectiveness of the proposed TDCAF.

Abbreviations

BoW: Bag-of-words; BP: Backpropagation algorithm; CASM: Convolutional activations summing map; CLBP: Completed local binary patterns; CNN: Convolutional neural network; DAN: Deep adaptation network; FC: Fully-connected layer; RBF: Radial basis function; SGD: Stochastic gradient descent; SLR: Single lens reflex; SVM: Support vector machine; TDCAF: Transfer deep convolutional activations-based features; WMO: World meteorological organization

Acknowledgements

The authors would like to thank the editor and anonymous reviewers for their helpful comments and suggestions in improving the quality of this paper. This work was supported by National Natural Science Foundation of China under Grant No. 61711530240, Natural Science Foundation of Tianjin under Grant No. 17JCZDJC30600, the Fund of Tianjin Normal University under Grant No. 135202RC1703, the Open Projects Program of National Laboratory of Pattern Recognition under Grant No. 201700001, and the China Scholarship Council No. 201708120040.

Funding

This work was supported by National Natural Science Foundation of China under Grant No. 61711530240, Natural Science Foundation of Tianjin under Grant No. 17JCZDJC30600, the Fund of Tianjin Normal University under Grant No. 135202RC1703, the Open Projects Program of National Laboratory of Pattern Recognition under Grant No. 201700001, and the China Scholarship Council No. 201708120040.

Availability of data and materials

The generalized image databases are available online.

Authors' contributions

ZZ contributed to the conception and algorithm design of the study. DL implement the experiments and ZZ verified all the results. ZZ and DL drafted and revised the manuscript. Both authors read and approved the final manuscript.

Authors' information

Zhong Zhang received the Ph.D. degree from Institute of Automation, Chinese Academy of Sciences. He is currently an Associate Professor at Tianjin Normal University. Donghong Li is currently pursuing the M.S. degree at Tianjin Normal University.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 29 January 2018 Accepted: 14 February 2018

Published online: 01 March 2018

References

1. MZ Hasan, H Al-Rizzo, M Günay, Lifetime maximization by partitioning approach in wireless sensor networks. *EURASIP J. Wirel. Commun. Netw.* **2017**, 15 (2017)
2. F Zhao, X Sun, H Chen, R Bie, Outage performance of relay-assisted primary and secondary transmissions in cognitive relay networks. *EURASIP J. Wirel. Commun. Netw.* **2014**(1), 60 (2014)
3. F Zhao, W Wang, H Chen, Q Zhang, Interference alignment and game-theoretic power allocation in mimo heterogeneous sensor networks communications. *Signal Process.* **126**, 173–9 (2016)
4. O Boiman, E Shechtman, M Irani, in *IEEE Conference on Computer Vision and Pattern Recognition*. In defense of nearest-neighbor based image classification (IEEE, Anchorage, 2008), pp. 1–8
5. A Bosch, A Zisserman, X Munoz, in *ACM International Conference on Image and Video Retrieval*. Representing shape with a spatial pyramid kernel (ACM, Amsterdam, 2007), pp. 401–8
6. A Cheddad, H Kusetoğlu, H Grahn, in *International Symposium on Image and Signal Processing and Analysis*. Object recognition using shape growth pattern (IEEE, Ljubljana, 2017), pp. 47–52
7. F Perronnin, J Sánchez, YL Xerox, in *IEEE Conference on Computer Vision and Pattern Recognition*. Large-scale image categorization with explicit data embedding (IEEE, San Francisco, 2010), pp. 2297–2304
8. A Torralba, AA Efros, in *IEEE Conference on Computer Vision and Pattern Recognition*. Unbiased look at dataset bias (IEEE, Providence, 2011), pp. 1521–1528
9. F Zhao, L Wei, H Chen, Optimal time allocation for wireless information and power transfer in wireless powered communication systems. *IEEE Trans. Veh. Technol.* **65**(3), 1830–5 (2016)
10. A Bergamo, L Torresani, in *Advances in Neural Information Processing Systems*. Exploiting weakly-labeled web images to improve object classification: a domain adaptation approach (NIPS, Vancouver, 2010), pp. 181–189
11. K Saenko, B Kulis, M Fritz, T Darrell, in *European Conference on Computer Vision*. Adapting visual category models to new domains (Springer, Crete, 2010), pp. 213–26
12. X Li, M Fang, J-J Zhang, J Wu, Learning coupled classifiers with rgb images for rgb-d object recognition. *Pattern Recognit.* **61**, 433–46 (2017)
13. L Zhang, Z He, Y Liu, Deep object recognition across domains based on adaptive extreme learning machine. *Neurocomputing.* **239**, 194–203 (2017)
14. B Gong, Y Shi, F Sha, K Grauman, in *IEEE Conference on Computer Vision and Pattern Recognition*. Geodesic flow kernel for unsupervised domain adaptation (IEEE, Providence, 2012), pp. 2066–73
15. J Hoffman, E Rodner, J Donahue, T Darrell, K Saenko, Efficient learning of domain-invariant image representations. *International Conference on Learning Representations*, arXiv:1301.3224 (2013). <https://arxiv.org/abs/1301.3224>
16. M Baktashmotlagh, MT Harandi, BC Lovell, M Salzmann, in *IEEE International Conference on Computer Vision*. Unsupervised domain adaptation by domain invariant projection (IEEE, Sydney, 2013), pp. 769–76

17. A Krizhevsky, I Sutskever, GE Hinton, in *Advances in Neural Information Processing Systems*. Imagenet classification with deep convolutional neural networks (NIPS, Lake Tahoe, 2012), pp. 1097–105. <https://arxiv.org/abs/1409.1556>
18. Y Ganin, V Lempitsky, in *International Conference on Machine Learning*. Unsupervised domain adaptation by backpropagation (IMLS, Lille, 2015), pp. 1180–9
19. Y Ganin, E Ustinova, H Ajakan, P Germain, H Larochelle, F Laviolette, M Marchand, V Lempitsky, Domain-adversarial training of neural networks. *J. Mach. Learn. Res.* **17**(59), 1–35 (2016)
20. M Long, Y Cao, J Wang, M Jordan, in *International Conference on Machine Learning*. Learning transferable features with deep adaptation networks (IMLS, Lille, 2015), pp. 97–105
21. F Cui, R Ju, Y Ding, H Ding, X Cheng, Prediction of regional global horizontal irradiance combining ground-based cloud observation and numerical weather prediction. *Adv. Mater. Res.* **1073**, 388–94 (2014)
22. F Zhao, B Li, H Chen, X Lv, Joint beamforming and power allocation for cognitive mimo systems under imperfect csi based on game theory. *Wirel. Pers. Commun.* **73**(3), 679–94 (2013)
23. T Várnai, A Marshak, Effect of cloud fraction on near-cloud aerosol behavior in the modis atmospheric correction ocean color product. *Remote Sens.* **7**(5), 5283–99 (2015)
24. K He, X Zhang, S Ren, J Sun, in *European Conference on Computer Vision*. Spatial pyramid pooling in deep convolutional networks for visual recognition (Springer, Zurich, 2014), pp. 346–61
25. X-S Wei, B-B Gao, J Wu, in *IEEE International Conference on Computer Vision*. Deep spatial pyramid ensemble for cultural event recognition (IEEE, Santiago, 2015), pp. 38–44
26. M Cimpoi, S Maji, A Vedaldi, in *IEEE Conference on Computer Vision and Pattern Recognition*. Deep filter banks for texture recognition and segmentation (IEEE, Boston, 2015), pp. 3828–36
27. K Simonyan, A Zisserman, Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
28. K He, X Zhang, S Ren, J Sun, in *IEEE Conference On Computer Vision And Pattern Recognition*. Deep residual learning for image recognition (IEEE, Las Vegas, 2016), pp. 770–8
29. L Bottou, in *International Conference on Computational Statistics*. Large-scale machine learning with stochastic gradient descent (Springer, Paris, 2010), pp. 177–86
30. MD Zeiler, R Fergus, in *European Conference On Computer Vision*. Visualizing and understanding convolutional networks (Springer, Zurich, 2014), pp. 818–33
31. A Mahendran, A Vedaldi, Visualizing deep convolutional neural networks using natural pre-images. *Int. J. Comput. Vis.* **120**(3), 233–55 (2016)
32. T Leung, J Malik, Representing and recognizing the visual appearance of materials using three-dimensional textons. *Int. J. Comput. Vis.* **43**(1), 29–44 (2001)
33. Z Guo, L Zhang, D Zhang, A completed modeling of local binary pattern operator for texture classification. *IEEE Trans. Image Process.* **19**(6), 1657–63 (2010)
34. S Lloyd, Least squares quantization in pcm. *IEEE Trans. Inf. Theory.* **28**(2), 129–37 (1982)

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)