Neurocomputing 409 (2020) 60–73

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Deep transfer neural network using hybrid representations of domain discrepancy

Changsheng Lu [a], Chaochen Gu [a,*], Kaijie Wu [a], Siyu Xia [b], Haotian Wang [c], Xinping Guan [a]

[a] Key Laboratory of System Control and Information Processing, Shanghai Jiao Tong University, Shanghai 200240, China
[b] School of Automation, Southeast University, Nanjing 210096, China
[c] Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109, USA

ARTICLE INFO

Article history:
Received 12 June 2019
Revised 4 April 2020
Accepted 16 May 2020
Available online 26 May 2020
Communicated by Lixin Duan

Keywords:
Hybrid representations
Transfer neural networks
Adversarial training
MMD
Domain confusion
Viewpoint estimation

ABSTRACT

Transfer neural networks have been successfully applied in many domain adaptation tasks. The initiative of most of the current transfer networks, essentially, is optimizing a single distance metric between the source domain and target domain, while few studies integrate multiple metrics for training transfer networks. In this paper, we propose an architecture of transfer neural network equipped with hybrid representations of domain discrepancy, which could incorporate the advantages of different types of metrics as well as compensate their imperfections. In our architecture, the Maximum Mean Discrepancy (MMD) and $\mathcal{H}$-divergence based domain adaptations are combined for simultaneous distribution alignment and domain confusion. Through extensive experiments, we find that the proposed method is able to achieve compelling transfer performance across the datasets with domain discrepancy from small scale to large scale. Especially, the proposed method can be promisingly used to predict the viewpoint of 3D-printed workpiece even trained without labels of real images. The visualization of learned features and adapted distributions by our transfer network highlights that the proposed approach could effectively learn the similar features between two domains and deal with a wide range of transfer tasks.

1. Introduction

Excellent transfer ability is a critical feature of intelligent learning model and could help to transit from a known knowledge domain to a new one. In practice, the domain discrepancy would always happen and thus degrade the performance of neural networks because the testing samples may have distribution different from those at training stage due to the influences of environment and diversity of sensors. A mainstream of transfer learning, as called *domain adaptation*, is always embedded in neural networks and used to bridge the gap between source and target domains. For example, person re-identification is a cross-camera retrieval task and may be affected by the image style variations [1]. Thus the style transfer [2,3], which is very hot recently, can be leveraged to alleviate the difference between cameras and generate the images which fit the target camera with the help of cycle-GAN [4,5]. Analogously, the medical images captured from different optical platforms also present domain shift and consequently, the application of domain adaptation approach will be greatly helpful. Transfer learning can bring neural networks many benefits such as increasing generalization ability, reducing the dependence on large amounts of labels and accelerating the training procedure. In many instances, training deep neural networks from scratch is difficult and demands much of time and annotations. Therefore, a simple and effective choice is to transfer the pre-trained model from large-scale datasets, e.g. ImageNet [6], and then fine-tune the whole networks or partial layers, which could work well in plenty of practical applications such as video classification [7] and the analysis of medical image [8,9]. In recent years, transfer neural network appears as an effective method to learn more transferable features and address the lack of substantial labels across tasks of image classification, object detection and image segmentation [15–17]. However, few methods, to our best knowledge, could always hold a remarkable performance in a wide range of transfer problems from small scale to large scale. From this perspective, it points out that different domain adaptation neural networks have their strengths, and yet reserves. Therefore, studying the hybrid method of domain adaptation for neural networks to exert their complementary competence is our main concern. Actually, transferring knowledge from labeled source domain to unlabeled target domain, namely *unsupervised domain adaptation*, is quite challeng-

[*] Corresponding author.
E-mail addresses: ChangshengLuu@gmail.com (C. Lu), jacygu@sjtu.edu.cn (C. Gu), kaijiewu@sjtu.edu.cn (K. Wu), xsy@seu.edu.cn (S. Xia), hautian@umich.edu (H. Wang), xpguan@sjtu.edu.cn (X. Guan).

https://doi.org/10.1016/j.neucom.2020.05.020
0925-2312/© 2020 Elsevier B.V. All rights reserved.

ing and this problem will turn to be more difficult as the domain discrepancy grows. Fig. 1 shows some samples for different transfer tasks, and meanwhile reveals their domain discrepancy. For most of transfer learning algorithms, handwritten digit sets [10–13] are widely used as the basic datasets for testing. However, it would be more challenging if the learning algorithms could transfer the knowledge of synthetic workpiece images rendered in the virtual environment to real ones [18]. Under this assumption, firstly, it may allow us to train the neural networks with the unlabeled real images and the synthetic images that are automatically labeled in the virtual environment, which will free the tediously manual work of labeling. Secondly, it may empower the trained model to predict the viewpoint of real workpiece, which would be greatly valuable to digital twin system in industry. Thus applying transfer neural networks to the problem of learning the domain invariant knowledge of workpiece between virtual and real environments is our main task.

In this paper, a hybrid method of domain adaptation is proposed, which aims at optimizing the two different domain discrepancy metrics simultaneously and establishing the transfer process with the guidance of both Maximum Mean Discrepancy (MMD) [19] and $\mathcal{H}$-divergence [20]. Then, a generic transfer network architecture is constructed and we apply the hybrid method of domain adaptation for the networks. Through the joint loss of main classification task and MMD, and alternate domain adversarial training, the transfer neural network narrows the domain shift gradually and finally becomes a shared predictor for source and target domains. The main contributions of this paper are in the following:

- We discover that the aforementioned two independent metrics, namely MMD and $\mathcal{H}$-divergence, can co-exist and be well-optimized simultaneously. The proposed method integrates the advantages of different metrics and exhibits very excellent overall transfer ability across all tasks from handwritten digit datasets to workpiece datasets.
- The proposed deep model, which is trained with labeled synthetic images and unlabeled real images, can be transferred to the real environment and used to realize viewpoint estimation of workpieces, which proceeds preliminary exploration in applying transfer neural networks into industry.
- We elaborately analyze the transfer performance of neural networks by adopting advanced visualization techniques and the results show that, indeed, the networks can learn the similar features between source and target domains. Meanwhile, we provide a standard approach and interface to build the subset from ImageNet for studying transfer learning beyond the same object classes.

The remainder of this paper is organized as follows. In Section 2, two current main domain discrepancy representations as well as the corresponding derived domain adaptation networks are described. Then, we will detail the proposed network architecture with the hybrid method of domain adaptation in Section 3. In Section 4, extensive experiments are implemented and the learned features by transfer learning are visualized. Finally, we conclude the paper in Section 5.

2. Related work

Thanks to the appropriate domain discrepancy representation, transfer neural networks are able to realize domain adaptation. Therefore, the distance representation between source and target domains is quite important and mainly can be divided into two categories: 1) Maximum Mean Discrepancy (MMD) [19] and 2) domain confusion based distance representation [20,21]. The former representation, namely MMD, utilizes the Euclidian distance to measure the overall mean distance between two sets of observations in Hilbert space, while the latter one uses domain confusion to represent the distance, which means that the domain discrepancy will be smaller if the feature is more indistinguishable for a binary classifier judging whether it comes from source domain or target domain. In the following, we introduce the two domain discrepancy representations and the derived transfer networks. For convenient reference, Table 1 shows the main notations commonly used in this paper.

2.1. MMD and related transfer neural networks

MMD is a kernel-based statistical test and is initially used in answering whether two sample sets X^s and X^t drawn from the same distribution or not. The original form of MMD can be formulated as

$$\mathrm{MMD}(X^s, X^t) = \sup_{f\in\mathcal{F}}\left(\mathbb{E}_{\mathbf{x^s}\in X^s} f(\mathbf{x^s}) - \mathbb{E}_{\mathbf{x^t}\in X^t} f(\mathbf{x^t})\right) \tag{1}$$

where $\mathcal{F}$ is a class of functions $f : \mathcal{X} \mapsto \mathbb{R}$. In order to compute MMD conveniently, f can be expressed as an inner product in Reproducing Kernel Hilbert Space (RKHS), namely $f(\mathbf{x}) = \langle f, \phi(\mathbf{x})\rangle$, where $\phi(\cdot)$ is a mapping function from feature space $\mathcal{X}$ to RKHS. Hereby, Eq. (1) can be simplified as

$$\begin{aligned}\mathrm{MMD}(X^s, X^t) &= \sup_{f\in\mathcal{F}}\left(\mathbb{E}_{\mathbf{x^s}\in X^s}\langle f, \phi(\mathbf{x^s})\rangle - \mathbb{E}_{\mathbf{x^t}\in X^t}\langle f, \phi(\mathbf{x^t})\rangle\right)\\ &= \sup_{f\in\mathcal{F}}\langle f, \mathbb{E}_{\mathbf{x^s}\in X^s}\phi(\mathbf{x^s}) - \mathbb{E}_{\mathbf{x^t}\in X^t}\phi(\mathbf{x^t})\rangle\\ &= \sup_{f\in\mathcal{F}} f\left(\mathbb{E}_{\mathbf{x^s}\in X^s}\phi(\mathbf{x^s}) - \mathbb{E}_{\mathbf{x^t}\in X^t}\phi(\mathbf{x^t})\right)\\ &= \left\|\mathbb{E}_{\mathbf{x^s}\in X^s}\phi(\mathbf{x^s}) - \mathbb{E}_{\mathbf{x^t}\in X^t}\phi(\mathbf{x^t})\right\|.\end{aligned} \tag{2}$$

By squaring the $\mathrm{MMD}(X^s, X^t)$, then we are able to use kernel trick to calculate each expanded term with Gaussian kernel function $\langle\phi(\mathbf{x}), \phi(\mathbf{x}')\rangle = k(\mathbf{x}, \mathbf{x}') = \exp(-\|\mathbf{x} - \mathbf{x}'\|/2\sigma^2)$.

MMD is a very useful domain adaptation tool and firstly adopted in the pioneering work DaNN [22] for object recognition. Since DaNN merely has only two layers and is shallow, DDC [23] uses deeper and more powerful networks, namely AlexNet [24], which improves the feature learning and transfer performance. Instead of utilizing single MMD for one adaptation layer like DaNN and DDC, DAN [25] and JAN [26] adapt multiple fully connected layers and use the variant versions of MMD, which aims at reinforcing the domain adaptation degree. Besides in deep transfer learning, MMD is also widely used in the problem of heterogeneous domain adaptation as an important metric to measure the discrepancy in shared latent space over the domains with different modal data [27,28]. It should note that MMD is very computationally efficient and could well measure the domain discrepancy between two domains. However, the transfer performance would empirically degrade if the dimensionality of the adapted feature increases.

2.2. Domain confusion based distance representation

Another type of domain discrepancy representation is based on the domain confusion. Although there are different formulations, e.g $\mathcal{H}$-divergence [20], $\mathcal{H}\Delta\mathcal{H}$-divergence [21] and $\mathcal{A}$-distance [20], the central ideas are same. Namely, the learned features from source and target domains should be sufficiently mixed and indistinguishable in a common space. Thus, the adapted features with high domain confusion could be classified by shared decision boundary. In order to quantize domain confusion based distance, we suppose the source and target domains are $\mathcal{S}$ and $\mathcal{T}$ and the

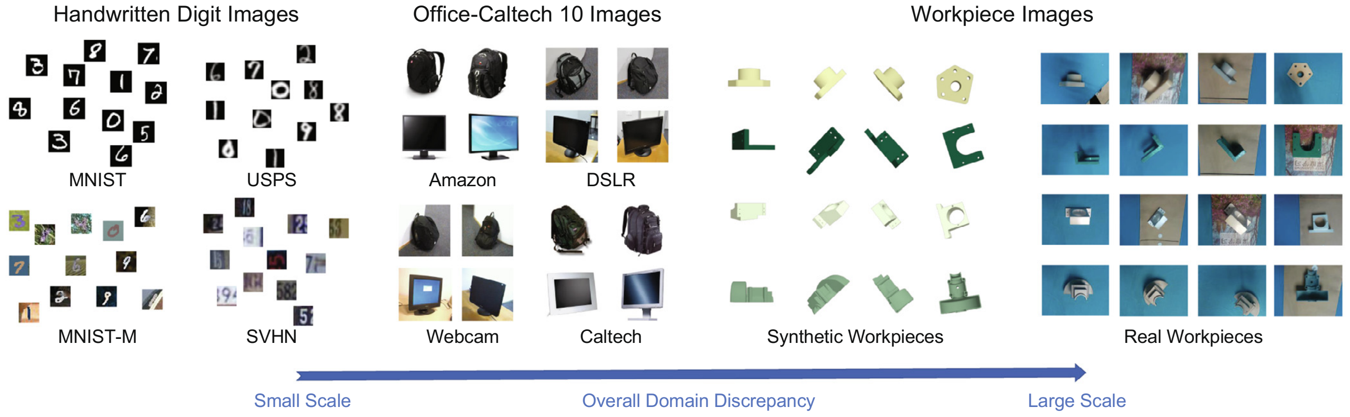


Fig. 1. An overview of image sets for transfer learning with different domain discrepancy. At left, there are four well-known domains used for handwritten digit recognition, namely MNIST [10], USPS [11], MNIST-M [12] and SVHN [13]. In the middle, it is the Office-Caltech 10 image set [14] which also consists of four domains. At right, the workpiece image set collected by us is comprised of 1) synthetic workpiece images which are rendered from CAD (Computer-aided Design) models at different poses in the virtual environment, and 2) the real 3D-printed counterpart images. Unlike most of the tasks transferring knowledge within a real environment, intuitively, the transfer task from textureless synthetic workpiece images to real environment should be with the large-scale domain discrepancy and very challenging due to the disturbances of illumination, texture, shadow, and various backgrounds.

Table 1
Important notations.

Symbol	Definition
$\mathcal{S}, \mathcal{T}$	Source domain and target domain
$\mathbf{x^s}, y^s$	Source sample and its category label
$\mathbf{x^t}, y^t$	Target sample and its category label
z^s, z^t	Domain label
$\mathcal{X}$	Feature space
$\mathcal{F}$	A set of functions mapping extracted feature to $\mathbb{R}$
$\phi(\cdot)$	Mapping function from the feature space to RKHS
k	Gaussian kernel function
$\mathcal{H}$	A class of binary hypotheses
F	Feature extractor
G_c	Task classifier
G_d	Domain classifier identifying which domain the sample $\mathbf{x}$ is from
θ	Model parameters of neural network

samples from $\mathcal{S}$ are all labeled 0 while the target samples are labeled 1. Given $\mathcal{H}$ a class of hypotheses $h : \mathcal{X} \mapsto \{0, 1\}$, the $\mathcal{H}$-divergence between $\mathcal{S}$ and $\mathcal{T}$ can be written as

$$\begin{aligned} d_{\mathcal{H}}(\mathcal{S}, \mathcal{T}) &= 2\sup_{h \in \mathcal{H}} |\mathbb{E}_{\mathbf{x^s} \in \mathcal{S}} \mathbb{I}[\mathbb{h}(\mathbf{x^s}) \neq \mathbb{1}] - \mathbb{E}_{\mathbf{x^t} \in \mathcal{T}} \mathbb{I}[\mathbb{h}(\mathbf{x^t}) \neq \mathbb{1}]| \\ &= \mathbb{2}\sup_{\mathbb{h} \in \mathcal{H}} \left| \mathbb{1} - \underbrace{\left(\mathbb{E}_{\mathbf{x^s} \in \mathcal{S}} \mathbb{I}[\mathbb{h}(\mathbf{x^s}) \neq \mathbb{0}] + \mathbb{E}_{\mathbf{x^t} \in \mathcal{T}} \mathbb{I}\left[\mathbb{h}\left(\mathbf{x^t}\right) \neq \mathbb{1}\right]\right)}_{\text{domain confusion term } \delta} \right| \end{aligned} \tag{3}$$

where $\mathbb{I}[\cdot]$ is the indicator function. And the domain confusion δ always ranges from 0 to 1 if hypothesis h is properly trained. Therefore, the domain discrepancy tends to be smaller when domain confusion δ is larger.

Currently, the minimization of domain confusion based distance is always realized by *domain adversarial training* [12,29–31]. The architecture ADDA proposed by Tzeng et al. [29,30] trains a domain classifier to minimize the δ in Eq. (3) as well as training a feature extractor to generate the features confusing the domain classifier. After such adversarial training, the transfer neural network could simultaneously optimize the domain shift $d_{\mathcal{H}}(\mathcal{S}, \mathcal{T})$ and transfer the knowledge between tasks. DANN [12] uses a reversal layer which contradicts the gradients of domain classification loss and thus simplifies the domain adversarial training. CDAN [32] further jointly takes the domain-specific features as well as class-specific predictions as input to conditional domain discriminator for adaptation. Unlike DANN, ADDA and CDAN optimize the domain confusion based distance through constructing an assistant domain classifier, recently, MCD [31] introduces a novel paradigm of adversarial domain adaptation which alternately maximizes two classifiers' discrepancy and then minimizes it by updating the shared feature extractor. Moreover, the methods like GTA [33] and CyCADA [34] even incorporates AC-GAN [35] and Cycle-GAN [4] respectively in hope of realizing domain confusion along with generating fake high-fidelity images. Though such kind of methods are impressive for excellent performance in simple transfer tasks, it might be vulnerable for slightly tougher problems and hard to train and converge because of the addition of strict constraints. Generally, $\mathcal{H}$-divergence is easily used and performs very well in the transfer tasks especially with small domain shift, but severely depends on the balanced training between two adversarial networks.

3. Proposed approach

In this section, we will first analyze the difference between two aforementioned domain discrepancy representations, and then elicit the motivation of using hybrid representations in domain adaptation which could integrate the advantages of MMD and $\mathcal{H}$-divergence. Next, a transfer network architecture based on both MMD and $\mathcal{H}$-divergence, and the corresponding training procedures are introduced, as shown in Fig. 2. The proposed approach is simple and yet could perform very well in various kinds of transfer problems, where individual MMD or $\mathcal{H}$-divergence based domain adaptation method is hard to achieve.

3.1. Overall idea of using hybrid representations in domain adaptation

Without ambiguity, we denote the source image $\mathbf{x^s}$ as well as its label y^s drawn from source domain $\mathcal{S} = \{X^s, Y^s\}$ and the unlabeled target image $\mathbf{x^t}$ drawn from target domain $\mathcal{T} = X^t$. Then we use a feature extractor F to take $\mathbf{x^s}$ and $\mathbf{x^t}$ as inputs and generate the corresponding features $F(\mathbf{x^s})$ and $F(\mathbf{x^t})$ which are assumed in feature space $\mathcal{X}$. Next, two classifiers, namely the task classifier G_c and domain classifier G_d, are employed for fulfilling the main task classification and domain classification respectively. Provided the labeled source domain $\mathcal{S}$ and unlabeled target domain $\mathcal{T}$, our final goal is to learn the feature extractor F and task classifier G_c which can correctly predict the label y^t for any input target image $\mathbf{x^t}$.

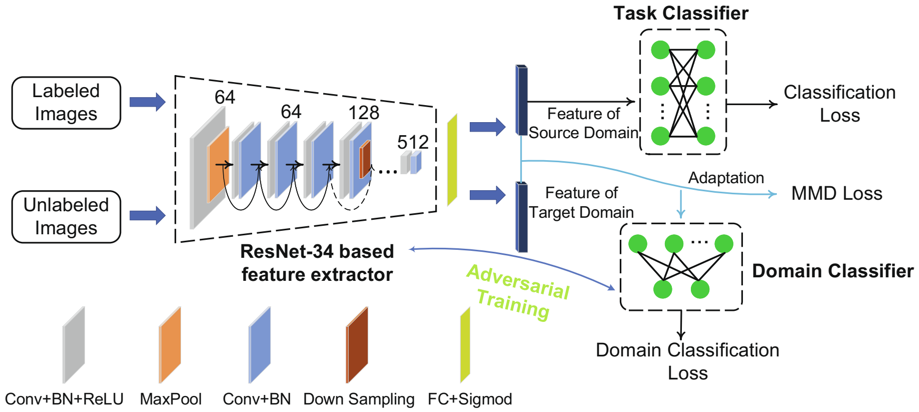


Fig. 2. Proposed transfer neural network architecture with hybrid representations of domain discrepancy. The feature extractor takes both the labeled and unlabeled images as input and outputs the deep features of source and target domains respectively. Then in the first aspect, the deep feature from source domain flows into the task classifier and results in the task classification loss; Secondly, the MMD based domain adaptation is applied over the extracted features, which produces the MMD loss; Thirdly, the domain classifier generates the domain classification loss by discriminating the domains of deep features, which can be used for realizing another type of domain adaptation that is based on $\mathcal{H}$-divergence. Via joint loss optimization and alternate adversarial training between domain classifier and feature extractor, the neural network is able to establish the transfer process as well as strengthen the transfer ability under hybrid metrics.

Currently, most of transfer neural networks are based on either MMD or $\mathcal{H}$-divergence. The essence of MMD is to pull the mean of source and target features to be closer in RKHS and thus reaching the purpose of aligning two marginal distributions $P(F(\mathbf{x^s}))$ and $P(F(\mathbf{x^t}))$. Due to the similarity of images in the same class between $\mathcal{S}$ and $\mathcal{T}$, the conditional distributions $P(y|\mathbf{x^s})$ and $P(y|\mathbf{x^t})$ would attract mutually and also be pulled closer to some extent under the optimization of the overall mean discrepancy of two domains. For $\mathcal{H}$-divergence based transfer networks, as indicated in Eq. (3), it always trains an assistant domain classifier G_d to distinguish which domain the extracted feature $F(\mathbf{x})$ belongs to, which can be regarded as trying to approximate the $\mathcal{H}$-divergence. Meanwhile, it trains F to generate the undistinguishable features to confuse G_d as much as possible, which can be regarded as optimizing $\mathcal{H}$-divergence. After such an adversarial training between G_d and F, the $P(z|\mathbf{x^s})$ and $P(z|\mathbf{x^t})$ (z is the domain prediction of G_d) are aligned and converge towards $P(z|\mathbf{x^s}) = P(z|\mathbf{x^t}) = [0.5, 0.5]^{\mathrm{T}}$ if maximum confusion is acquired.

In fact, MMD and $\mathcal{H}$-divergence are from two different perspectives to represent the domain discrepancy. MMD starts from the mean value aspect while $\mathcal{H}$-divergence is from the domain confusion aspect, resulting in that the domain adaptation based on each of them has its advantages. MMD based approach could force to reduce the mean distance between source and target domains and thus well align the two distributions despite that they have relatively large discrepancy, but not necessary to achieve excellent domain confusion like $\mathcal{H}$-divergence based one. On the contrary, $\mathcal{H}$-divergence based method could make the features of two domains to be mixed adequately and deal very well with the transfer tasks especially with small-scale difficulty, but it also could not guarantee the mean deviation between source and target domains to be diminished and thus the domain-level confusion would be insufficient and even hard to handle the case when domain shift hugely grows. Empirically, neither MMD nor $\mathcal{H}$-divergence based method could keep superior performance than each other across all different transfer tasks. Therefore, it is significant to develop a hybrid method in domain adaptation to universally handle various transfer problems and inherit the advantages of both MMD and $\mathcal{H}$-divergence based approaches. We find that MMD and $\mathcal{H}$-divergence can co-exist and be optimized simultaneously within a transfer network. And surprisingly, the trained transfer model with the hybrid representations of domain discrepancy could combine both strengths, enhance the transfer ability and show extraordinary learning performance in various tasks.

3.2. Transfer network architecture and training steps

In order to concretize the hybrid method in domain adaptation, we provide a neural network exemplar as shown in Fig. 2. Considering that ResNet [36] can avoid vanishing gradient and sufficiently learn the features, we adopt ResNet34 as the default backbone of feature extractor F. And we use fully connected linear layers to construct the main task classifier G_c and domain classifier G_d. Assumed that the network's each two-stream input batches from source domain $\mathcal{S}$ and target domain $\mathcal{T}$ are $\mathbf{B^s}$ and $\mathbf{B^t}$ respectively. Therefore, we could acquire the extracted features $F(\mathbf{B^s})$ and $F(\mathbf{B^t})$ by feeding the $\mathbf{B^s}$ and $\mathbf{B^t}$ to F simultaneously. Then we train G_c merely with the labels of source images since the labels of target images are absent. The loss function of the main classification task can be formulated as

$$\mathcal{L}_{\mathrm{cls}}\left(\theta_f, \theta_c\right) = -\mathbb{E}_{(\mathbf{x^s}, y^s)\sim\left(\mathcal{X}^s, \mathcal{Y}^s\right)} \mathbf{1}[y^s] \log p(y|\mathbf{x^s}) \tag{4}$$

where θ_f and θ_c is the model parameters of F and G_c, and $p(y|\mathbf{x^s})$ is the C-dimensional probability output of G_c using softmax function. $\mathbf{1}[i]$ is the identity vector where i-th entry is 1. In order to transfer the knowledge from source domain to target domain and align the features of two domains, firstly, we apply the MMD based domain adaptation over feature space $\mathcal{X}$, which accords with the empirical risk minimization. Thus we have the MMD loss function

$$\begin{aligned}\mathcal{L}_{\mathrm{MMD}}\left(\theta_f\right) &= \mathrm{MMD}^2 = \left\|\mathbb{E}_{\mathbf{x^s}} \in \mathbf{B^s}\phi(F(\mathbf{x^s})) - \mathbb{E}_{\mathbf{x^t}} \in \mathbf{B^t}\phi(F(\mathbf{x^t}))\right\|^2 \\ &= \left\|\tfrac{1}{|\mathbf{B^s}|}\sum_{\mathbf{x^s}\in\mathbf{B^s}}\phi(F(\mathbf{x^s})) - \tfrac{1}{|\mathbf{B^t}|}\sum_{\mathbf{x^t}\in\mathbf{B^t}}\phi(F(\mathbf{x^t}))\right\|^2 \\ &= \tfrac{1}{|\mathbf{B^s}|^2}\sum_{\mathbf{x^s_i},\mathbf{x}^s_j\in\mathbf{B^s}} k\left(F(\mathbf{x^s_i}), F\left(\mathbf{x^s_j}\right)\right) + \tfrac{1}{|\mathbf{B^t}|^2}\sum_{\mathbf{x^t_i},\mathbf{x}^t_j\in\mathbf{B^t}} k\left(F(\mathbf{x^t_i}), F\left(\mathbf{x^t_j}\right)\right) \\ &\quad - \tfrac{2}{|\mathbf{B^s}||\mathbf{B^t}|}\sum_{\mathbf{x^s_i}\in\mathbf{B^s},\mathbf{x}^t_j\in\mathbf{B^t}} k\left(F(\mathbf{x^s_i}), F\left(\mathbf{x^t_j}\right)\right)\end{aligned} \tag{5}$$

where $|\mathbf{B}^\bullet|$ is the batch size and k is Gaussian kernel function. In order to mitigate the risk of selecting inappropriate kernel function

k, the weighted kernel functions are adopted, namely $\langle\phi(\cdot),\phi(\cdot)\rangle=\sum_u\beta_u k_u(\cdot,\cdot)$ where $\sum_u\beta_u=1$ and $\beta_u\geqslant 0$. In addition, we also feed the extracted feature $F(\mathbf{x})$ to domain classifier G_d which will result in two-dimensional probability $p(z|\mathbf{x})$ for the judgement of whether source domain or target domain it belongs to. The domain classification loss function can be written as

$$\mathcal{L}_{\mathrm{d}}\left(\theta_f,\theta_d\right)=-\mathbb{E}_{\mathbf{x^s}\sim X^s}\mathbf{1}[z^s]\log p(z|\mathbf{x^s})-\mathbb{E}_{\mathbf{x^t}\sim X^t}\mathbf{1}\left[z^t\right]\log p(z|\mathbf{x^t}) \tag{6}$$

where θ_d is model parameters of G_d and z is the domain prediction of G_d given image $\mathbf{x}$. z^s is domain label of source image $\mathbf{x^s}$ and equal to 0 and analogously, z^t is equal to 1.

Based on the above three loss functions $\mathcal{L}_{\mathrm{cls}},\mathcal{L}_{\mathrm{MMD}}$, and $\mathcal{L}_{\mathrm{d}}$, we are able to train the transfer neural network by optimizing the joint loss functions in adversarial fashion. At each iteration, we will first fix the parameters of domain classifier G_d, namely θ_d, and optimize the θ_f and θ_c by jointly minimizing $\mathcal{L}_{\mathrm{cls}}+\lambda(\mathcal{L}_{\mathrm{MMD}}-\xi\mathcal{L}_{\mathrm{d}})$. Then alternately, we freeze the parameters of feature extractor F and main task classifier G_c, namely θ_f and θ_c, and optimize θ_d by minimizing the domain classification loss function $\lambda\xi\mathcal{L}_{\mathrm{d}}$. The whole adversarial training process can be formulated as

$$\begin{aligned}\theta_f,\theta_c &= \underset{\theta_f,\theta_c}{\arg\min}\quad \mathcal{L}_{\mathrm{cls}}\left(\theta_f,\theta_c\right)+\lambda\left(\mathcal{L}_{\mathrm{MMD}}\left(\theta_f\right)-\xi\mathcal{L}_{\mathrm{d}}\left(\theta_f,\hat{\theta}_d\right)\right)\\ \theta_d &= \underset{\theta_d}{\arg\min}\quad \lambda\xi\mathcal{L}_{\mathrm{d}}\left(\hat{\theta}_f,\theta_d\right)\end{aligned} \tag{7}$$

where λ is the tradeoff between the supervised classification loss $\mathcal{L}_{\mathrm{cls}}$ and the summation loss of $\mathcal{L}_{\mathrm{MMD}}$ and $\mathcal{L}_{\mathrm{d}}$ which aims for domain adaptation. We fix $\lambda=1$ throughout our paper. ξ is the weight between MMD and $\mathcal{H}$-divergence and is also set to be 1 as regarding both metrics have equal importance.

The hybrid method in domain adaptation, as demonstrated above, integrates both MMD and $\mathcal{H}$-divergence based domain adaptations to the network simultaneously, which would be much helpful to exert the advantages of two metrics and improve the adaptability to various transfer problems.

4. Experiments

In this section, we test our method in four types of datasets to demonstrate that the transfer neural network with hybrid adaptations could combine and even strengthen the advantages of MMD and $\mathcal{H}$-divergence based domain adaptation networks. Moreover, we visualize the learned features and the distribution alignment between source and target domains, the results of which further ensure the effectiveness of the proposed method.

4.1. Experiment setup

4.1.1. Evaluation metrics

In order to quantitatively evaluate the performance of transfer networks, we use average precision (AP) and classification accuracy as evaluation metrics. The definition of AP and classification accuracy are

$$\begin{aligned}\mathrm{AP}&=\sum_n(R_n-R_{n-1})P_n\\ \mathrm{Accuracy}&=\frac{\sum_{i=1}^{N}\mathbb{I}\left[\mathbb{y}_{\mathbb{i}}=\mathbb{y}_{\mathbb{i}}^{\mathbb{t}}\right]}{N}\end{aligned} \tag{8}$$

where R_n and P_n are the recall and precision at the n-th threshold, respectively. N is total number of test images in target domain and the indicator function $\mathbb{I}\left[\mathbb{y}_{\mathbb{i}}=\mathbb{y}_{\mathbb{i}}^{\mathbb{t}}\right]$ is 1 if the prediction y_i equals to the ground truth y_i^t; otherwise $\mathbb{I}\left[\mathbb{y}_{\mathbb{i}}=\mathbb{y}_{\mathbb{i}}^{\mathbb{t}}\right]=\mathbb{0}$.

4.1.2. Compared methods

For simplicity, we abbreviate the proposed transfer neural network using hybrid representations of domain discrepancy as TNN-Hybrid. Firstly, four baselines: Source Only I, Source Only II, TNN-MMD, and TNN-$\mathcal{H}$ are provided for ablation study. Except Source Only I which is based on AlexNet, the rest three share the same backbone ResNet34 with TNN-Hybrid. Source Only I and II is merely trained with source data and without any adaptation; TNN-MMD uses only MMD for domain adaptation; and TNN-$\mathcal{H}$ uses $\mathcal{H}$-divergence for adaptation. Secondly, the popular MMD based transfer learning networks, e.g. DDC [23], DAN [25] and JAN [26], and the $\mathcal{H}$-divergence based methods such as DANN [12], ADDA [30], MCD [31] and GTA [33] are also adopted for comparison in experiments.

4.1.3. Implementation details

There are four different types of datasets employed throughout our experiments: 1) Workpiece dataset; 2) Handwritten digit datasets [10–13]; 3) Office-Caltech 10 dataset [14] and 4) ImageNet based dataset [6]. Due to the resolution of the handwritten image is limited and thus could not use very deep feature extractor, we adopt the relative shallow CNN architectures which are previously used in [12,31] and modify the network by adding batch normalization and dropout after each layer. For the remaining datasets, we use the proposed architecture (see Fig. 2), in which the parameters of employed ResNet34 are pre-trained on ImageNet, and the last pooling layer is replaced by adaptive average pooling in order to get the feature map with fixed size 8 × 8. A fully-connected layer with Sigmoid activation function is inserted behind the feature extractor as adaptation layer which is initialized with the normal distribution of mean $u=0$ and standard deviation $\sigma=0.5$. In addition, the domain classifier of proposed architecture has one hidden layer with 512 ReLU activation units [37]. For the MMD computation, four Gaussian kernel functions with standard deviation of $\sigma=2,4,8,16$ are used. All the networks leverage SGD algorithm with a learning rate of 0.001 and momentum of 0.9 for optimization except the handwritten digit transfer tasks which use Adam algorithm [38].

4.2. Experiments on workpiece dataset

In order to investigate the feasibility of applying transfer neural networks to industry for object viewpoint estimation, we build a workpiece dataset upon eight different workpieces and each type of workpieces contains 8610 synthetic images and 840 real images. It should note that we convert and degrade the classic pose estimation problem into the image classification problem in this experiment. As shown in Fig. 3, firstly, the eight different workpiece models are designed in the CAD software 3Ds-Max. Then, we define the main axis of each workpiece and make it point to each node of the viewpoint sphere which is discretized by the latitudes at the step of 45° and longitudes at the step of 90°. For simplicity, only seven blue nodes of the viewpoint sphere in Fig. 3 are selected. Thus the synthetic workpiece images with seven different viewpoints can be generated by rendering the CAD workpiece model with a fixed camera in the virtual environment. In order to augment the number of workpiece images, the main axis of workpiece randomly points the adjacency of each node and rotates the workpiece around itself. Given one node, all the rendered images are categorized as a viewpoint class, for example, as revealed by the synthetic images of workpiece one (WP1) at each row in Fig. 3. Finally, we fabricate the real workpieces using 3D printing technology and capture the real workpiece images analogously. In order to challenge the capability of transfer networks, the real images are collected under more complex backgrounds and illuminations, which lays the hurdle to transfer the knowledge

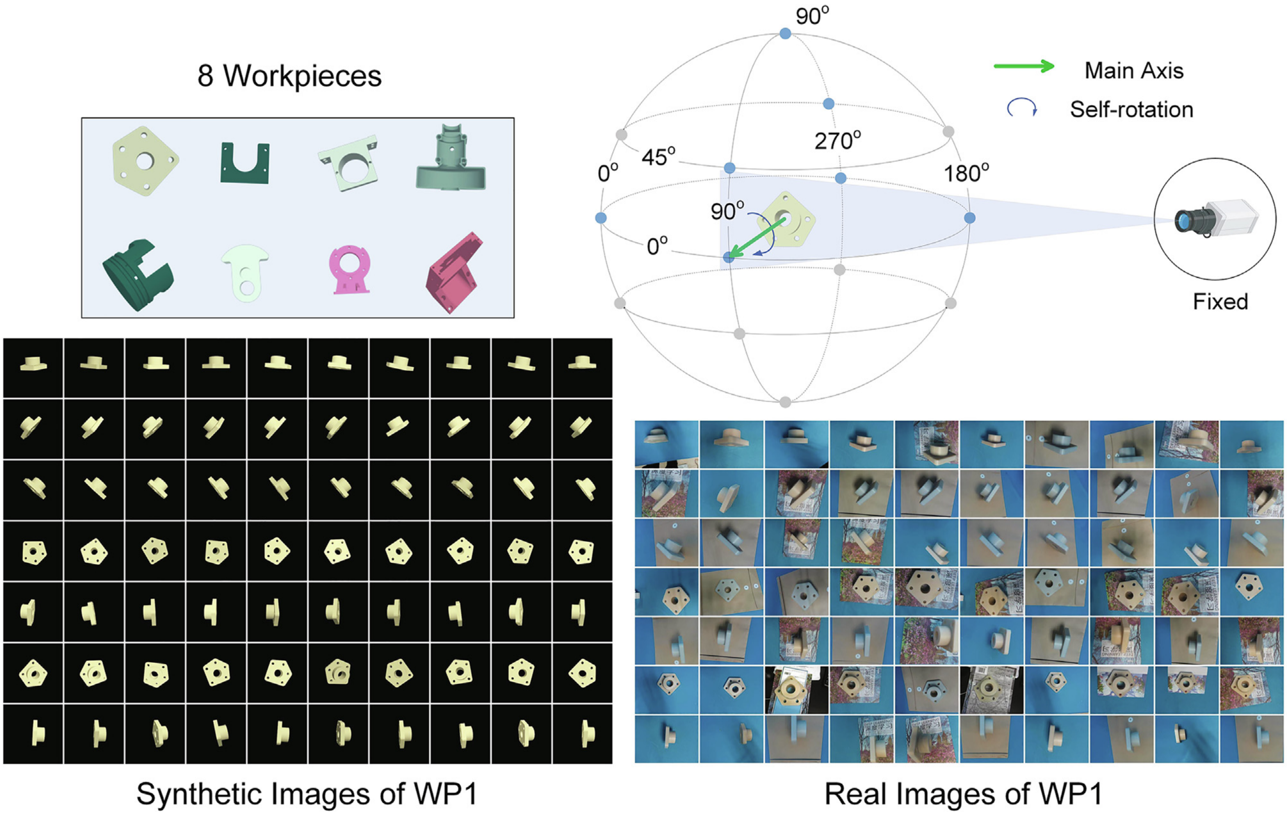


Fig. 3. Synthetic-real workpiece dataset generation and image exemplars of workpiece one (WP1). Eight different CAD workpiece models shown on the top left are leveraged to build the workpiece images. Each workpiece model is placed in the center of a viewpoint sphere (top right), pointing its main axis to each node of sphere and thus being rendered under simple pure color background by a fixed camera in virtual environment, where the WP1's synthetic images with seven different pose categories are shown on the bottom left (and the images with slight pose variations at each row belonging to a common viewpoint class). The real workpiece image generation follows the same configurations with the synthetic images except for establishing the whole process in a real-world environment and with three backgrounds: blue tablecloth, brown envelope, and book. The part real images of WP1 are shown on the bottom right.

from synthetic workpiece images to real images. Though the huge domain gap in this experiment, we use the entire labeled synthetic images and unlabeled real images to train all the compared models, and evaluate on the training set of target domain. The detailed classification results across seven different viewpoints on WP1 compared to baselines and popular transfer network architectures are shown in Table 2. The top *three* scores regarding average precision (AP) and accuracy across all compared methods are stressed in boldface. Among MMD distance based methods, JAN [26] is based on ResNet50 while DDC [23] and DAN [25] are on top of AlexNet. From Table 2, it can observe that they can achieve higher scores to some extent than the backbone-sharing network Source Only II (based on ResNet34) and Source Only I (based on AlexNet) respectively, which demonstrates that transfer learning can effectively alleviate the domain discrepancy. Among JAN, DAN and DDC, the reason that JAN could perform better comprehensively than DDC and DAN may be due to the different adoptions of feature extractor and MMD adaptation strategy. In $\mathcal{H}$-divergence based methods, MCD [31] uses ResNet34 as backbone and DANN employs AlexNet. Apparently in Table 2, MCD [31] performs much better than DANN [12] and TNN-$\mathcal{H}$ and gains much in average accuracy compared to Source Only II, which leads to 76.7%. The reason behind this may be due to that MCD aligns the distributions of source and target by utilizing two task-specific classifiers, which is effective for this task. However, when comparing to TNN-MMD and JAN, MCD shows little advantage and exhibits large margin no matter in accuracy or AP to TNN-MMD. Actually, in front of the large difference of appearance between synthetic and real workpiece images, $\mathcal{H}$-divergence based method would be hard to handle such domain shift and align the class-specific distribution between source and target domains.

Comparing the transfer performance among TNN-$\mathcal{H}$, TNN-MMD and TNN-Hybrid, we can see that TNN-Hybrid could well inherit the high scores of TNN-MMD and avoid the influence of unsatisfactory transfer results of TNN-$\mathcal{H}$. Moreover, TNN-Hybrid is able to boost the performance in many viewpoint classes and achieves the second best overall accuracy of 88.7% in WP1, as shown in Table 2 (The best score is also held by our TNN-Hybrid-ResNet50). It should note that, although $\mathcal{H}$-divergence based adaptation does not effect as leading role in this transfer task, it still could increase the domain confusion and thus help to improve network's transfer ability. As aforementioned, both MMD and $\mathcal{H}$-divergence representations have their own strengths and yet reserves, the proposed TNN-Hybrid aims to adopt both advantages and exerts its best competence for transfer learning, which will be also reflected in subsequent extensive experiments.

For comprehensively testing the performance of compared transfer networks for viewpoint estimation, we implement all transfer tasks on the synthetic-real workpiece dataset from WP1 to WP8 and the quantitative results are shown in Table 3. The accuracy scores achieved by Source Only I and II from WP2 to WP7 are much smaller than those in WP1 and WP8, which indicates that the transfer tasks in WP2 to WP7 are very difficult. Nevertheless, the proposed TNN-Hybrid could still perform effectively and gain the mean average precision (mAP) and mean accuracy (mAcc) across all viewpoint categories vastly in eight workpieces compared to the Source Only II baseline. Again, TNN-Hybrid follows the advances of TNN-MMD in most of the workpieces except some

Table 2
Results of average precision (AP) and accuracy (Acc) of workpiece one (WP1) in seven pose classes. The last column is the mean of AP and Acc. Top three scores are in boldface.

Method	Metrics	$(90^\circ,0^\circ)$	$(45^\circ,90^\circ)$	$(45^\circ,270^\circ)$	$(0^\circ,0^\circ)$	$(0^\circ,90^\circ)$	$(0^\circ,180^\circ)$	$(0^\circ,270^\circ)$	Avg.
Source Only I	AP	84.1	73.6	79.0	67.8	50.3	62.5	45.6	60.7
	Acc	65.8	57.5	55.8	32.5	64.2	90.0	60.0	60.8
Source Only II	AP	65.6	55.8	73.4	94.5	60.4	90.2	60.0	70.2
	Acc	45.0	25.0	61.7	95.8	60.0	77.5	60.0	60.7
DDC [23]	AP	83.3	80.2	81.1	82.7	57.2	79.7	39.2	67.3
	Acc	70.0	59.2	61.7	53.3	**73.3**	86.6	54.2	65.5
DAN [25]	AP	98.9	46.1	30.5	98.5	42.2	90.9	43.4	64.0
	Acc	88.3	53.3	22.5	76.7	45.8	95.8	30.0	58.9
JAN [26]	AP	**100.0**	61.2	56.3	**99.9**	66.8	**99.9**	44.7	76.4
	Acc	**100.0**	62.5	36.7	**100.0**	55.0	**100.0**	45.8	71.4
DANN [12]	AP	75.2	63.8	65.9	51.5	48.9	76.2	44.1	57.6
	Acc	59.2	47.5	56.7	43.3	52.5	93.3	46.7	57.0
MCD [31]	AP	**100.0**	**100.0**	**99.8**	28.6	60.3	55.5	60.5	70.1
	Acc	**100.0**	**100.0**	**99.2**	21.7	60.0	85.8	70.0	76.7
TNN-MMD	AP	**99.9**	93.8	95.1	**99.9**	**78.3**	99.6	**79.8**	**94.8**
	Acc	97.5	81.7	92.5	**100.0**	**67.5**	97.5	**71.7**	**86.9**
TNN-$\mathcal{H}$	AP	48.6	25.7	47.8	94.5	20.7	84.8	46.4	55.5
	Acc	58.3	14.2	41.7	90.0	14.2	80.8	51.7	50.1
TNN-Hybrid	AP	99.7	**97.7**	**99.8**	**99.8**	**73.3**	**99.7**	**76.1**	**94.4**
	Acc	**99.7**	**86.7**	**97.5**	**99.2**	**67.5**	**99.2**	**74.2**	**88.7**
TNN-Hybrid-AlexNet	AP	97.3	78.6	78.6	59.9	62.7	53.6	55.2	67.7
	Acc	93.9	70.8	77.5	61.2	64.2	61.7	60.8	70.0
TNN-Hybrid-ResNet50	AP	99.8	**96.9**	**99.9**	**99.8**	**81.3**	**99.8**	**81.7**	**97.1**
	Acc	97.5	**90.0**	**99.2**	**99.2**	**68.3**	**99.1**	**80.0**	**90.5**

Table 3
Results of mean average precision (mAP) and mean accuracy (mAcc) across all pose classes from workpiece one (WP1) to workpiece eight (WP8). Top three scores are in boldface.

Method	Metrics	WP1	WP2	WP3	WP4	WP5	WP6	WP7	WP8
Source Only I	mAP	60.7	48.1	47.3	44.0	56.9	53.8	46.0	70.8
	mAcc	60.8	48.9	47.8	49.2	56.4	53.0	44.9	67.7
Source Only II	mAP	70.2	22.2	32.2	31.7	36.8	44.6	27.7	61.4
	mAcc	60.7	24.3	33.9	30.7	36.9	42.1	29.5	50.8
DDC [23]	mAP	67.3	51.0	45.0	49.0	58.5	58.5	45.4	73.0
	mAcc	65.5	52.7	50.0	47.3	57.2	60.6	54.3	75.3
DAN [25]	mAP	64.0	54.1	48.3	40.1	70.3	**66.0**	**65.7**	51.0
	mAcc	58.9	48.9	41.8	37.4	61.4	54.4	**59.9**	33.4
JAN [26]	mAP	76.4	53.5	**57.7**	58.6	**79.6**	65.1	**77.8**	62.7
	mAcc	71.4	43.2	45.1	52.5	**69.2**	61.6	56.3	43.7
DANN [12]	mAP	57.6	34.5	40.6	41.2	53.3	48.0	44.5	69.0
	mAcc	57.0	38.4	43.0	41.4	53.5	47.1	46.3	64.6
MCD [31]	mAP	70.0	32.4	**61.2**	37.5	53.8	61.5	31.3	97.7
	mAcc	76.7	46.3	**63.0**	44.4	**63.9**	61.4	34.9	91.3
TNN-MMD	mAP	**94.8**	**59.3**	53.8	**82.5**	53.5	54.1	49.2	**99.3**
	mAcc	**86.9**	**57.4**	58.9	**72.8**	52.1	60.5	49.7	**97.4**
TNN-$\mathcal{H}$	mAP	55.5	26.2	33.3	31.9	30.5	30.7	36.1	50.5
	mAcc	50.1	30.7	31.1	30.8	29.5	30.9	32.6	52.0
TNN-Hybrid	mAP	**94.4**	**69.4**	57.2	**70.4**	**65.9**	**65.2**	62.9	**98.4**
	mAcc	**88.7**	**68.3**	**63.5**	**63.9**	60.5	**65.9**	**59.5**	**95.6**
TNN-Hybrid-AlexNet	mAP	67.7	43.6	42.9	48.3	49.6	58.2	48.1	78.3
	mAcc	70.0	52.1	47.4	51.7	59.2	**64.3**	50.1	73.1
TNN-Hybrid-ResNet50	mAP	**97.1**	**88.2**	**79.0**	**61.6**	**65.9**	**84.9**	**75.7**	**98.9**
	mAcc	**90.5**	**78.6**	**81.5**	**57.9**	**71.1**	**83.2**	**76.9**	**95.6**

drops in WP8 and WP4. From the Table 3, it is straightforward that the proposed TNN-Hybrid is very suitable to deal with the transfer tasks in workpiece dataset and could harmoniously integrate MMD and $\mathcal{H}$-divergence within a common architecture.

Besides the ablation experiments for studying the effect of whether or not using hybrid representations, we also investigate the influence of different backbone settings to our method. As indicated in Table 2 and Talbe 3, we can see that our ResNet50 version of TNN-Hybrid holds most of best accuracy scores as well as AP and outperforms TNN-Hybrid which is on the basis of ResNet34. However, if we use AlexNet as backbone, though it could conduct better than Source Only I, the performance is quite restricted. Thus the appropriate network settings are also quite important. Therefore, we provide two types of Source Only baselines for conveniently comparing different transfer methods throughout the whole experiments.

4.3. Experiments on handwritten digit dataset

In this experiment, we evaluate the transfer neural network with hybrid representations on the entire and standard public handwritten digit datasets: MNIST [10], USPS [11], MNIST-M [12] and Street View House Numbers (SVHN) [13]. The image samples of four handwritten digit datasets are displayed in Fig. 1 and there clearly exists discrepancy among four domains. We adopt four frequently employed transfer tasks in this experiment, namely MNIST to MNIST-M, USPS to MNIST, SVHN to MNIST and MNIST to USPS, where the former dataset is as source domain and the latter is as target domain. The results of handwritten digit recognition with various transfer networks are illustrated in Table 4. Recall that DANN [12], ADDA [30], TNN-$\mathcal{H}$, GTA [33] and MCD [31] are all based on $\mathcal{H}$-divergence representation. As we can see, $\mathcal{H}$-divergence based methods perform quite well for the transfer tasks in handwritten digit datasets and reveal stronger transfer ability than MMD based method. For instance, TNN-$\mathcal{H}$ outperforms TNN-MMD in four handwritten digit recognition transfer tasks, as shown in Table 4. It also could observe that TNN-Hybrid holds the better results than TNN-$\mathcal{H}$ and TNN-MMD across all transfer tasks, which indicates that hybrid method in domain adaptation can not only successfully incorporate the advantages of $\mathcal{H}$-divergence based adaptation but also get promoted in the manner of combining both domain discrepancy representations. Even surprisingly, TNN-Hybrid achieves one best accuracy scores 93.03% in the task of MNIST to MNIST-M compared to the state-of-the-art transfer networks MCD and GTA. In fact, it would be difficult for GTA to align distribution between MNIST and MNIST-M as the target image owns higher complexity of background and generating fake high-fidelity images under such situation is with stricter constraints and not easy. On the contrary to the experiments over synthetic-real workpiece dataset in Section 4.2, $\mathcal{H}$-divergence based adaptation takes over the leading role from MMD based adaptation, and fortunately it does not affect the proposed method and TNN-Hybrid could still perform well on the basis of the advantages of two domain adaptation methods.

4.4. Experiments on Office-Caltech 10 and ImageNet based dataset

We further implement extensive experiments on the public dataset Office-Caltech 10 [14] and the self-built ImageNet based dataset. Office-Caltech 10 is composed of 10 common categories from Office-31 dataset [39] and Caltech-256 dataset [40]. There are totally 2533 images in Office-Caltech 10 dataset and four domains which can be called as Amazon (**A**), Webcam (**W**), DSLR (**D**) and Caltech (**C**). The images in Amazon domain are downloaded from the website of amazon.com while the images in Webcam and DSLR are captured by a low-resolution web camera and high-resolution digital SLR camera in office environment respectively. The corresponding image samples of four domains in Office-Caltech 10 are shown in Fig. 1. Given that Office-Caltech 10 is commonly adopted by many transfer learning methods for comparison, therefore, we build eight transfer tasks in experiment, which are $\mathbf{A} \rightarrow \mathbf{W}, \mathbf{A} \rightarrow \mathbf{D}, \mathbf{A} \rightarrow \mathbf{C}, \mathbf{D} \rightarrow \mathbf{W}, \mathbf{W} \rightarrow \mathbf{D}, \mathbf{C} \rightarrow \mathbf{A}, \mathbf{C} \rightarrow \mathbf{W}$ and $\mathbf{C} \rightarrow \mathbf{D}$. The experimental results on Office-Caltech 10 are as shown in Table 5. Firstly, it is straightforward that the transfer neural networks TNN-MMD and TNN-$\mathcal{H}$ achieve similar accuracy scores across eight transfer tasks in Office-Caltech 10, which reveals the approximately equal transfer capability in this dataset between MMD and $\mathcal{H}$-divergence based adaptation methods. Secondly, we can see that the proposed TNN-Hybrid could also hold excellent performance in Office-Caltech 10 and keep pace with TNN-MMD and TNN-$\mathcal{H}$. The obtained accuracy scores by TNN-Hybrid are quite balanced and even the worst performance can reach 87.4% (see the transfer task of $\mathbf{A} \rightarrow \mathbf{C}$). Thirdly, compared to other popular transfer methods listed in Table 5, TNN-Hybrid is very competitive and can achieve 3/8 best accuracy scores in Office-Caltech 10, which evidently demonstrates its effectiveness. Moreover, compared to TNN-Hybrid, our TNN-Hybrid-ResNet50 reveals higher transfer potential, which accords to the results achieved in Section 4.2.

In order to demonstrate that two domain discrepancy representations, namely MMD distance and $\mathcal{H}$-divergence, could be optimized together within our architecture, we use the first transfer task $\mathbf{A} \rightarrow \mathbf{W}$ on Office-Caltech 10 as an example and draw the curves of training losses and task classification accuracy, as shown in Fig. 4. It can observe that the task classification loss, MMD loss and domain classification loss could be simultaneously optimized and at the meantime, the accuracy gets promoted, which verifies the effectiveness of proposed method. Notably, the domain classification loss curve converges to 1.386 which is the theoretical value of adversarial balance between domain classifier G_d and feature extractor F when maximum confusion achieves.

In fact, Office-Caltech 10 is a quite small-scale dataset whose number of images per category averages 63.3 and is with a maximum of 151 (like “backpack” category in Caltech domain) and a minimum of 8 (like “mug” in DSLR domain), which will restrain the transfer performance of deep networks that demand large amounts of data. In addition, in order to investigate the performance of transfer networks beyond same object types, for instance, the transfer tasks of cat → tiger and dog → wolf, we build a standard ImageNet based dataset[1] for future research. The ImageNet based dataset can be divided into six object types, which are cat, dog, car, tiger, wolf, and truck respectively, as shown in Fig. 5. Moreover, it consists of 7114 images in total and has an average of 1185 with a minimum of 815 and a maximum of 1552 images per category, which are abundant enough for training a deep network. With ImageNet based dataset, we construct two transfer tasks in our experiment, namely **Task A**: $\{\text{cat}, \text{dog}, \text{car}\} \rightarrow \{\text{tiger}, \text{wolf}, \text{truck}\}$ and **Task B**: $\{\text{tiger}, \text{wolf}, \text{truck}\} \rightarrow \{\text{cat}, \text{dog}, \text{car}\}$, and quantitative results are also shown in Table 5. Although the transfer objects between the source domain and target domain are different and the backgrounds are the diverse nature scenes (see Fig. 5), the knowledge still could be transferred as long as there existing common characters between two domains. From Table 5, we can learn that the two tasks in ImageNet based dataset are not difficult as the high accuracy scores achieved by Source Only I and II. Naturally, the proposed TNN-Hybrid could yet obtain very competitive results, especially in **Task A** which reaches 97.2%. Similar to the experimental results on Office-Caltech 10, again, TNN-Hybrid could perfectly integrate both MMD and $\mathcal{H}$-divergence based adaptations and fully exert their competence.

4.5. Analysis of learned features of transfer neural network

In order to intuitively judge and validate the effectiveness of the proposed transfer neural network with hybrid method in domain adaptation, we implement experiments to visualize the distribution as well as the discriminative area of deep features learned by the trained transfer model from source and target domains.

4.5.1. Visualizing distribution of learned features

In this experiment, we employ the compared deep transfer models trained in the dataset of workpiece one (WP1) for visualizing the distribution of extracted features which are used for domain adaptation. Since the deep features are high-dimensional, e.g. 512D in our architecture and 256D in DDC, it is hard to directly interpret and understand. With the help of useful dimensionality reduction algorithm t-SNE [41], we are capable of turning the extracted features into two dimensions and drawing the distribution of source and target features within graphs, as shown in Fig. 6. The more distinguishable of each visualized viewpoint cate-

[1] https://github.com/AlanLuSun/ImageNet-Based-Dataset-for-Transfer-Learning.

Table 4
The classification accuracy of different transfer tasks on four handwritten digit datasets MNIST [10], USPS [11], SVHN [13] and MNIST-M [12]. The experiments are conducted with the entire standard datasets and all methods are with same network backbone protocol as indicated in [12,31]. Notation (†) means the results of the corresponding methods are cited from published papers.

Method	MNIST to USPS	USPS to MNIST	SVHN to MNIST	MNIST to MNIST-M
Source Only	68.9	62.1	65.7	52.3
DANN† [12]	77.1	73.0	73.9	76.7
ADDA† [30]	89.4	90.1	76.0	-
GTA [33]	**93.6**	89.2	89.5	75.3
MCD [31]	93.0	**96.4**	**94.4**	74.1
TNN-MMD	79.9	75.1	72.7	52.5
TNN-$\mathcal{H}$	81.9	82.0	77.0	90.0
TNN-Hybrid	89.2	89.3	84.7	**93.03**

Table 5
The accuracy of various transfer tasks on Office-Caltech 10 dataset and ImageNet based dataset. Top three scores are in boldface.

Method	Office-Caltech 10 dataset								ImageNet based Dataset	
	A→W	A→D	A→C	D→W	W→D	C→A	C→W	C→D	Task A	Task B
Source Only I	73.9	80.3	77.6	86.1	90.4	89.4	72.9	82.2	89.5	79.4
Source Only II	78.6	84.1	79.9	89.8	96.8	88.8	89.2	83.4	92.7	89.1
DDC [23]	81.0	80.9	84.5	91.2	97.5	91.0	79.0	85.4	93.1	82.7
DAN [25]	75.9	85.4	78.8	96.6	98.1	89.9	73.6	77.1	92.3	83.4
JAN [26]	**95.9**	**87.9**	**87.4**	**96.9**	95.5	90.5	88.1	88.5	**95.7**	**95.0**
DANN [12]	67.8	77.1	78.7	91.5	98.1	90.4	75.9	82.8	88.2	78.7
GTA [33]	41.7	51.6	50.3	67.1	86.0	67.2	48.1	65.6	73.2	69.3
MCD [31]	**88.5**	**89.2**	87.3	**98.6**	99.3	87.2	87.8	89.1	92.3	83.1
TNN-MMD	**88.5**	86.6	86.9	89.8	**100.0**	**94.1**	**90.5**	**90.4**	94.5	89.9
TNN-$\mathcal{H}$	80.7	84.7	86.5	88.5	**100.0**	**93.2**	**90.2**	89.2	94.3	89.0
TNN-Hybrid	87.8	**89.8**	**87.4**	89.2	**100.0**	**93.4**	89.5	**92.9**	**97.2**	**90.8**
TNN-Hybrid-AlexNet	82.0	85.4	84.1	92.2	98.7	91.3	82.7	86.0	92.9	82.6
TNN-Hybrid-ResNet50	**94.6**	**87.9**	**90.1**	**98.3**	97.5	93.0	**90.2**	**90.4**	**96.8**	**93.4**

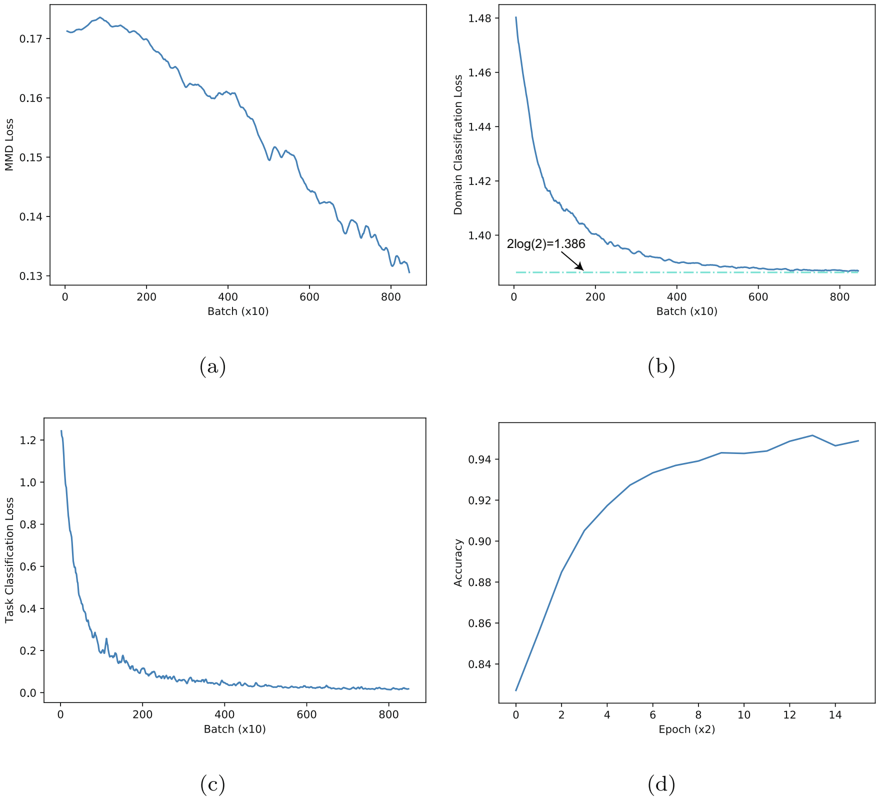


Fig. 4. Example of training loss curves and classification accuracy using TNN-Hybrid-ResNet50 in transfer task A→W on Office-Caltech 10 dataset. (a) MMD loss curve; (b) domain classification loss curve; (c) task classification loss curve; (d) accuracy curve.

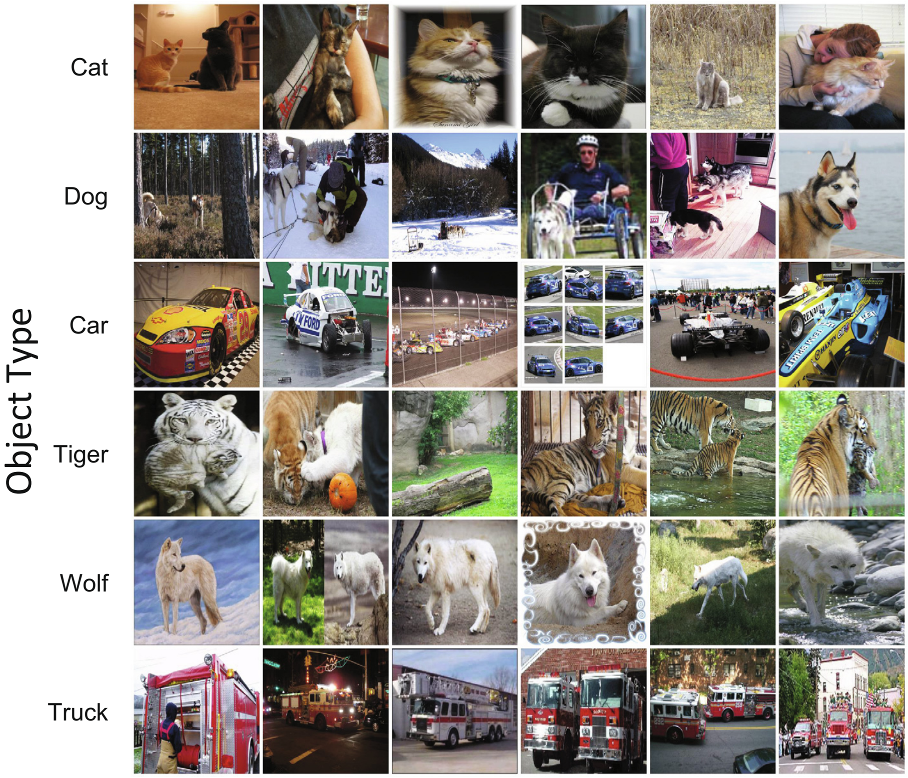
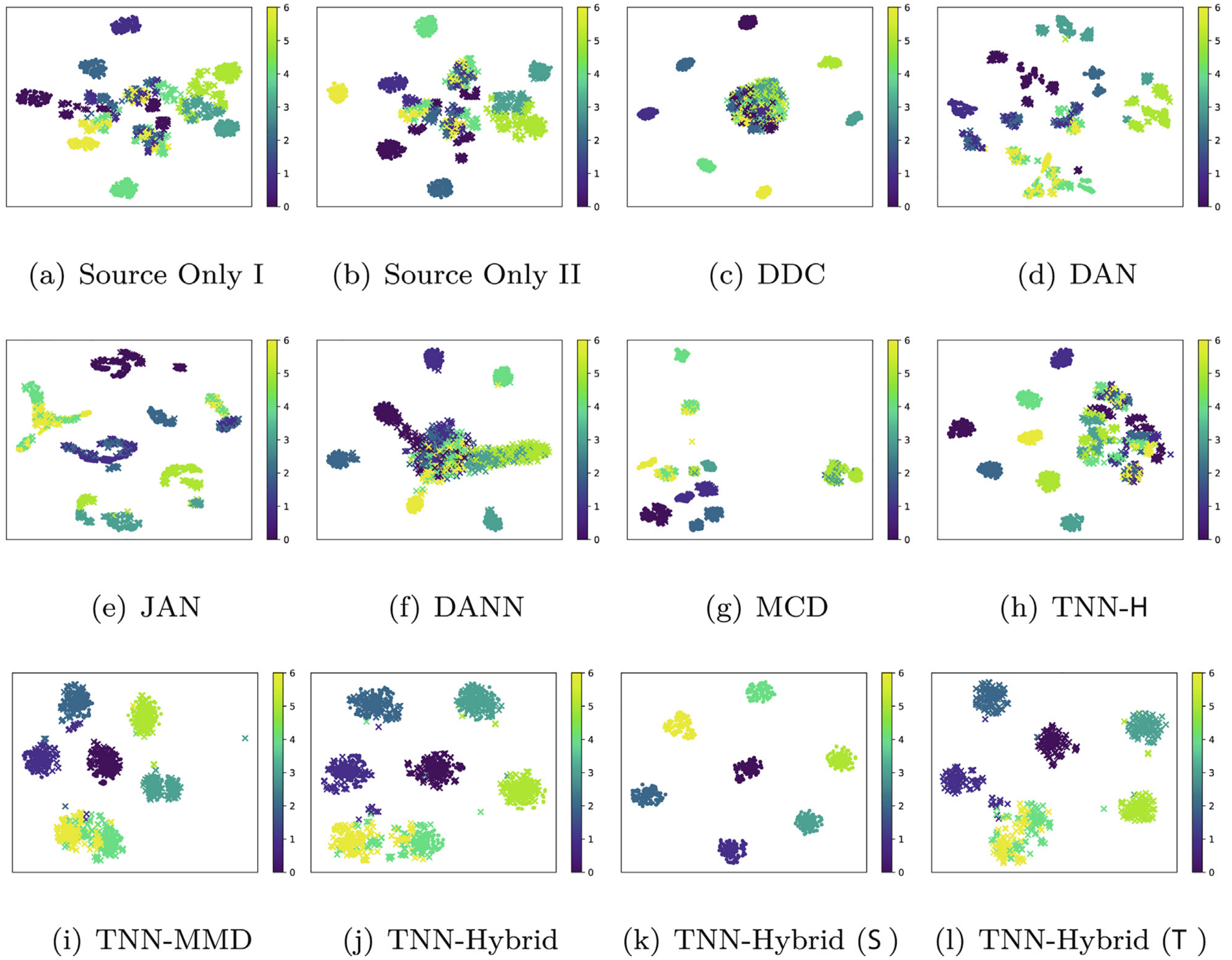


Fig. 5. An overview of ImageNet based dataset. There are six different object types in total, which can be divided into two domains for studying transfer learning, namely $\{\text{cat}, \text{dog}, \text{car}\}$ and $\{\text{tiger}, \text{wolf}, \text{truck}\}$.

gory and the better distribution alignment between source and target features are meaning the higher classification accuracy and better transfer ability. Firstly, we can see that the target features of Source Only I and II in Fig. 6a) and Fig. 6(b) are mixed and badly align with the source features since there is no domain adaptation applied. Compared with the displayed features of various transfer methods from Fig. 6(c)–(i), the class boundaries of the proposed transfer neural network TNN-Hybrid are very clear and the clusters of each category are much compact, which highlights the excellent transfer performance achieved by TNN-Hybrid. Secondly, the feature alignment of TNN-MMD is much better than that of TNN-$\mathcal{H}$ as Fig. 6i) and Fig. 6(h) show, which implies MMD based domain adaptation is more suitable than $\mathcal{H}$-divergence based one regarding the transfer task in workpiece dataset. In addition, we also can find that the feature distribution plot of TNN-Hybrid is similar to that of TNN-MMD but performs better in overall compactness and alignment, which illustrates that the hybrid adaptations could exert better on the basis of MMD based domain adaptation. Admittedly, the above analysis fully accords with the aforementioned numerical results on WP1. Thirdly, from Fig. 6(k) and (l), the features from source domain are much more distinguishable than the features from target domain although after domain adaptation, which is extremely sensible because the deep transfer model is supervised under labeled source images and unlabeled target images. Overall, the distribution plots of source and target features in Fig. 6 are quite straightforward, vivid and helpful, which evidently demonstrates the effectiveness of the proposed method.

4.5.2. Interpreting learned deep features

Although the deep neural network has always been considered as a black box, many valuable works are trying to unveil what features the deep model has learned. For example, Grad-CAM [42] and CAM [43] are two well-known as well as convenient methods for producing the coarse discriminative area which implicitly reflects the attention of the trained convolutional neural networks. In this experiment, we use the Grad-CAM [42] to visualize the discriminative area of the feature map generated by the last pooling layer of ResNet-34 based feature extractor in our architecture. The examples of the discriminative area learned by the proposed transfer model TNN-Hybrid across eight workpieces are shown in Fig. 7. As we can see, the highlighted area in synthetic images is within or near the workpiece objects despite the TNN-Hybrid are trained under image level labels. Actually, the deep model could learn the common spatial characteristics in the images that belong to the same class and then focus on the critical area for prediction. Notably, the backgrounds of real workpiece images are much more complicated than synthetic ones and TNN-Hybrid are trained without labels of real images. Even so, the visualized discriminative area of the proposed TNN-Hybrid could successfully highlight the workpiece objects in real images, which indicates the transfer model has learned the common features between the source domain and target domain and is equipped with the transfer ability to some extent. In addition, we also visualize the discriminative area of the feature maps on ImageNet based dataset, as shown in Fig. 8. The visualization results of Fig. 8(a) and (b) are produced by using the deep model TNN-Hybrid trained in transfer **Task B** and **Task A** respectively. Thus, the image groups of Fig. 8(a) and (b) are both belonging to target domains. Again, we can see that the TNN-Hybrid trained in two opposite transfer tasks could learn the similar features, e.g. the textures between cat and tiger, and focus on the class-related area, which demonstrates the proposed transfer neural network with hybrid method in domain adaptation could effectively transfer the information from the source domain to target domain.



(a) Source Only I (b) Source Only II (c) DDC (d) DAN

(e) JAN (f) DANN (g) MCD (h) TNN-H

(i) TNN-MMD (j) TNN-Hybrid (k) TNN-Hybrid (S) (l) TNN-Hybrid (T)

Fig. 6. Visualization of distribution for extracted deep features from synthetic images (source domain) and real images (target domain) of workpiece one by using t-SNE [41]. The source and target features are marked by color dots (·) and crosses (×) respectively, where each color represents a viewpoint class of workpiece as indicated in the color bar. (a) to (j) are the plots of feature distribution alignment with different deep transfer networks and drawing the source and target features simultaneously. In order to further observe the distribution difference of learned features between the source domain and target domain, the source and target features extracted by TNN-Hybrid are also separately shown. in (k) and (l).

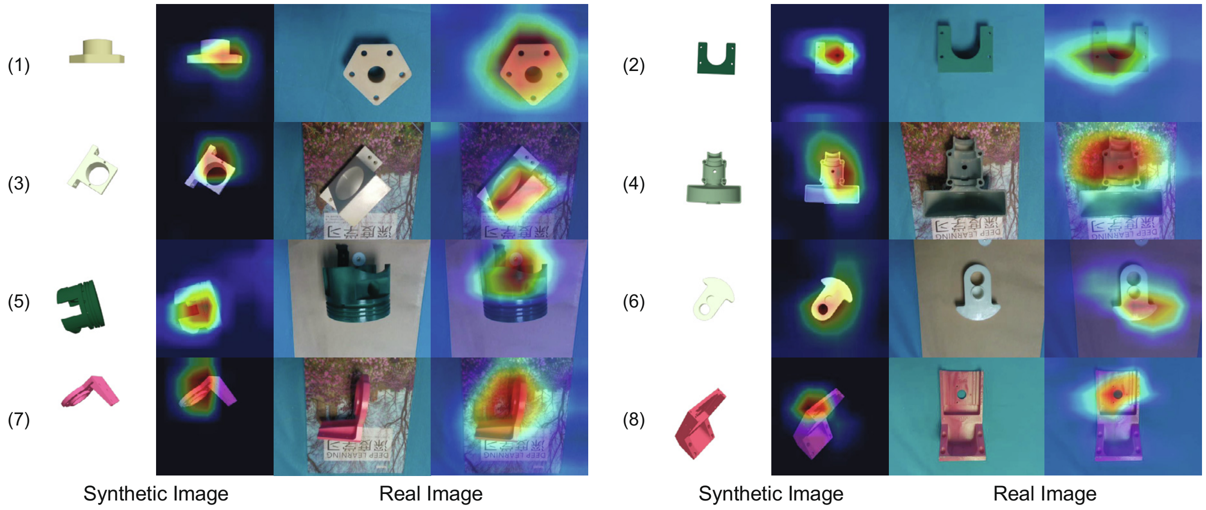


Fig. 7. Visualization of the discriminative area where the proposed transfer neural network model TNN-Hybrid focuses. The eight synthetic and real workpiece images along with the heat map which indicates the focused area of deep transfer model are displayed at each indexed row.

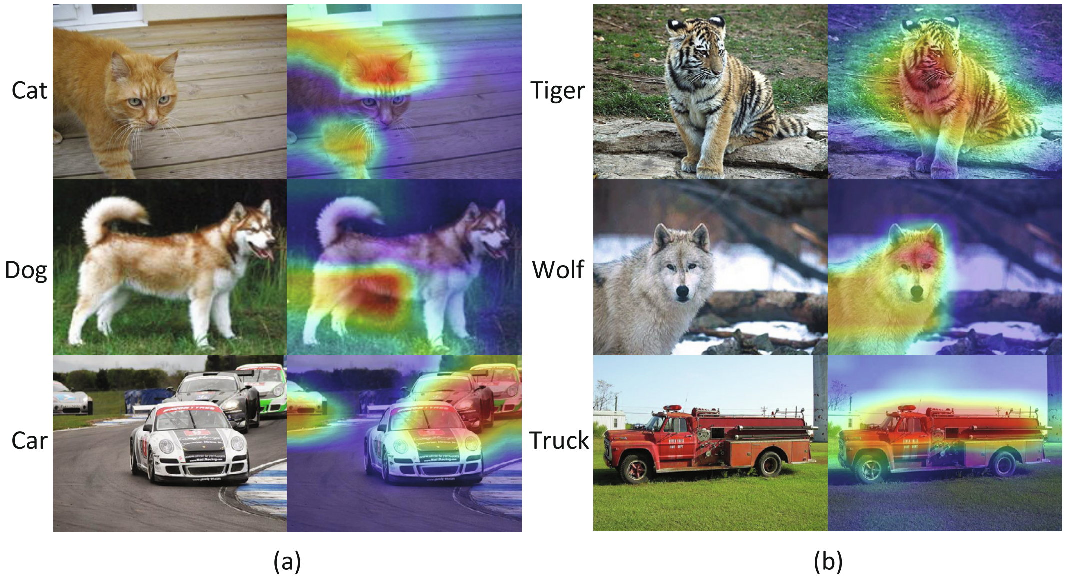


Fig. 8. Visualizing the discriminative area of learned features on ImageNet based dataset by TNN-Hybrid. The heat maps of (a) are generated by using the transfer model trained in Task B, namely transferring from $\{tiger, wolf, truck\}$ to $\{cat, dog, car\}$, while the heat maps of (b) are produced on the contrary.

5. Conclusion

The two well-known domain discrepancy representations, as well as the derived deep transfer networks, are detailedly discussed in this paper. Based on that, we propose hybrid method in domain adaptation and corresponding transfer neural network architecture, which can fully excavate the potentials of different methods and exert their best competence within a common network. Moreover, the proposed method shows strong universality and is very convenient to be employed. The achieved competitive results across all different transfer tasks either simple or difficult demonstrate that the method using hybrid representations of domain discrepancy could well incorporate the advantages of both MMD and $\mathcal{H}$-divergence based adaptations and even strengthen the network's transfer ability. In the future, we will study the hybrid approaches to further improving the transfer performance with semi-supervised training by adding a few labels from the target domain and integrating more other novel domain adaptation methods.

CRediT authorship contribution statement

Changsheng Lu: Conceptualization, Methodology, Writing - review & editing, Software. **Chaochen Gu:** Supervision, Writing - original draft. **Kaijie Wu:** Supervision, Visualization, Investigation. **Siyu Xia:** Writing - review & editing. **Haotian Wang:** Data curation, Software, Validation. **Xinping Guan:** Writing - review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

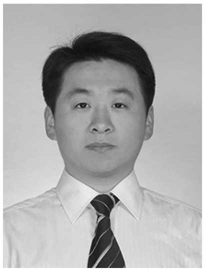
The authors would like to thank anonymous reviewers for insightful comments. This work was supported by National Key R&D Program of China, Project No. 2018YFB1703201, and Shanghai Action plan for Science and Technology Innovation, Project No. 19511105900.

References

[1] Z. Zhong, L. Zheng, Z. Zheng, S. Li, Y. Yang, Camera style adaptation for person re-identification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 5157–5166.
[2] Y. Jing, Y. Yang, Z. Feng, J. Ye, Y. Yu, M. Song, Neural style transfer: A review, arXiv preprint arXiv:1705.04058..
[3] L.A. Gatys, A.S. Ecker, M. Bethge, Image style transfer using convolutional neural networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2414–2423.
[4] J.-Y. Zhu, T. Park, P. Isola, A.A. Efros, Unpaired image-to-image translation using cycle-consistent adversarial networks, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 2223–2232.
[5] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in: Advances in neural information processing systems, 2014, pp. 2672–2680..
[6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: a large-scale hierarchical image database, in: 2009 IEEE conference on computer vision and pattern recognition, IEEE, 2009, pp. 248–255..
[7] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, L. Fei-Fei, Large-scale video classification with convolutional neural networks, in: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, 2014, pp. 1725–1732.
[8] H. Ravishankar, P. Sudhakar, R. Venkataramani, S. Thiruvenkadam, P. Annangi, N. Babu, V. Vaidya, Understanding the mechanisms of deep transfer learning for medical images, in: Deep Learning and Data Labeling for Medical Applications, Springer, 2016, pp. 188–196..
[9] N. Tajbakhsh, J.Y. Shin, S.R. Gurudu, R.T. Hurst, C.B. Kendall, M.B. Gotway, J. Liang, Convolutional neural networks for medical image analysis: Full training or fine tuning?, IEEE Trans Med. Imag. 35 (5) (2016) 1299–1312.
[10] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, et al., Gradient-based learning applied to document recognition, Proc. IEEE 86 (11) (1998) 2278–2324.
[11] J.J. Hull, A database for handwritten text recognition research, IEEE Trans. Pattern Anal. Mach. Intell. 16 (5) (1994) 550–554.
[12] 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 (1) (2016) 2096, 2030.
[13] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, A. Y. Ng, Reading digits in natural images with unsupervised feature learning, in: NIPS workshop on deep learning and unsupervised feature learning, 2011..
[14] B. Gong, Y. Shi, F. Sha, K. Grauman, Geodesic flow kernel for unsupervised domain adaptation, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2012, pp. 2066–2073.
[15] M. Wang, W. Deng, Deep visual domain adaptation: a survey, Neurocomputing 312 (2018) 135–153.
[16] M. Oquab, L. Bottou, I. Laptev, J. Sivic, Learning and transferring mid-level image representations using convolutional neural networks, in: Proceedings of

the IEEE conference on computer vision and pattern recognition, 2014, pp. 1717–1724.
[17] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 3431–3440.
[18] C. Lu, H. Wang, C. Gu, K. Wu, X. Guan, Viewpoint estimation for workpieces with deep transfer learning from cold to hot, in: International Conference on Neural Information Processing, Springer, 2018, pp. 21–32.
[19] K.M. Borgwardt, A. Gretton, M.J. Rasch, H.-P. Kriegel, B. Schölkopf, A.J. Smola, Integrating structured biological data by kernel maximum mean discrepancy, Bioinformatics 22 (14) (2006) e49–e57.
[20] S. Ben-David, J. Blitzer, K. Crammer, F. Pereira, Analysis of representations for domain adaptation, Adv. Neural Inform. Process. Syst. (2007) 137–144.
[21] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, J.W. Vaughan, A theory of learning from different domains, Mach. Learn. 79 (1–2) (2010) 151–175.
[22] M. Ghifary, W.B. Kleijn, M. Zhang, Domain adaptive neural networks for object recognition, in: Pacific Rim International Conference on Artificial Intelligence, Springer, 2014, pp. 898–904.
[23] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, T. Darrell, Deep domain confusion: Maximizing for domain invariance, Comput. Sci..
[24] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems, 2012, pp. 1097–1105..
[25] M. Long, Y. Cao, J. Wang, M. Jordan, Learning transferable features with deep adaptation networks, in: International Conference on Machine Learning, 2015, pp. 97–105.
[26] M. Long, H. Zhu, J. Wang, M.I. Jordan, Deep transfer learning with joint adaptation networks, in: International Conference on Machine Learning, 2017, pp. 2208–2217.
[27] J. Li, K. Lu, Z. Huang, L. Zhu, H.T. Shen, Transfer independently together: a generalized framework for domain adaptation, IEEE Trans. Cybern. 49 (6) (2018) 2144–2155.
[28] J. Li, K. Lu, Z. Huang, L. Zhu, H.T. Shen, Heterogeneous domain adaptation through progressive alignment, IEEE Trans. Neural Networks Learn. Syst. 30 (5) (2018) 1381–1391.
[29] E. Tzeng, J. Hoffman, T. Darrell, K. Saenko, Simultaneous deep transfer across domains and tasks, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 4068–4076.
[30] E. Tzeng, J. Hoffman, K. Saenko, T. Darrell, Adversarial discriminative domain adaptation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 7167–7176.
[31] K. Saito, K. Watanabe, Y. Ushiku, T. Harada, Maximum classifier discrepancy for unsupervised domain adaptation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 3723–3732.
[32] M. Long, Z. Cao, J. Wang, M. I. Jordan, Conditional adversarial domain adaptation, in: Advances in Neural Information Processing Systems, 2018, pp. 1640–1650..
[33] S. Sankaranarayanan, Y. Balaji, C.D. Castillo, R. Chellappa, Generate to adapt: aligning domains using generative adversarial networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 8503–8512.
[34] J. Hoffman, E. Tzeng, T. Park, J.-Y. Zhu, P. Isola, K. Saenko, A. Efros, T. Darrell, Cycada: Cycle-consistent adversarial domain adaptation, in: Proceedings of the 35th International Conference on Machine Learning, 2018..
[35] A. Odena, C. Olah, J. Shlens, Conditional image synthesis with auxiliary classifier gans, in: Proceedings of the 34th International Conference on Machine Learning-Volume 70, JMLR. org, 2017, pp. 2642–2651.
[36] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
[37] V. Nair, G.E. Hinton, Rectified linear units improve restricted boltzmann machines, in: Proceedings of the 27th international conference on machine learning (ICML-10), 2010, pp. 807–814.
[38] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980..
[39] K. Saenko, B. Kulis, M. Fritz, T. Darrell, Adapting visual category models to new domains, in: European conference on computer vision, Springer, 2010, pp. 213–226.
[40] G. Griffin, A. Holub, P. Perona, Caltech-256 object category dataset..
[41] L.V.D. Maaten, G. Hinton, Visualizing data using t-sne, J. Mach. Learn. Res. 9 (Nov) (2008) 2579–2605.
[42] R.R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad-cam: visual explanations from deep networks via gradient-based localization, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 618–626.
[43] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, A. Torralba, Learning deep features for discriminative localization, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2921–2929.

Changsheng Lu received the B.S. degree in Automation from Southeast University, Nanjing, China, in 2017. Currently, he is an academic M.S. student with the Key Laboratory of System Control and Information Processing, Shanghai Jiao Tong University. He has wide research interests mainly including computer vision, transfer learning, image processing, pattern recognition, and robotics. Particularly, he is interested in the theories and algorithms that empower robot to see, think and conduct more like a human. Previously, he was awarded the national scholarship for graduate student, and listed in the first term of Huawei F(X) future scientist program member and the outstanding undergraduate of Southeast University. He has served as the reviewers of IEEE Computational Intelligence Magazine, Pattern Recognition, and Journal of Visual Communication and Image Representation.



Chaochen Gu is currently an Associate Professor at School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University. He received his bachelor degree from Shandong University, Jinan, China, in 2007, and the Ph.D. degree in Mechanical Engineering from Shanghai Jiao Tong University, Shanghai, China, in 2013. His current research interests include industry robotics, machine vision, and man-machine interfaces.

Kaijie Wu is an Associate Professor at School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University. He received his Ph.D. degree in Biomedical Engineering from Tianjin University, Tianjin, China, in 2006. His current research explores biomedical optical imaging, medical information processing, and pattern recognition.

Siyu Xia received his BE and MS degrees in automation engineering from Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2000 and 2003, respectively, and the PhD degree in pattern recognition and intelligence system from Southeast University, Nanjing, China, in 2006. He is currently working as an associate professor in the School of Automation at Southeast University, Nanjing, China. His research interests include object detection, applied machine learning, social media analysis, and intelligent vision systems. He was the recipient of the Science Research Famous Achievement Award in Higher Institution of China in 2015. He has served as the reviewer of many journals including TIP, T-SMC-B, T-IFS, T-MM, IJPRAI, and Neurocomputing. He received Outstanding Reviewer Award for Journal of Neurocomputing in 2016. He has also served on the PC/SPC for the conferences including AAAI, ACM-MM, ICME, and ICMLA. He is a member of IEEE and ACM.

Haotian Wang received his master degree from Department of Electrical Engineering and Computer Science, University of Michigan, USA and bachelor degree from School of Electronic Information and Electrical Engineering, Shanghai Jiaotong University, China. Currently his research interests include computer vision and machine learning.

Xinping Guan received the B.S. degree in mathematics from Harbin Normal University, Harbin, China, and the M.S. degree in applied mathematics and the Ph.D. degree in electrical engineering, both from Harbin Institute of Technology, in 1986, 1991, and 1999, respectively. He is currently a Professor of Department of Automation, Shanghai Jiao Tong University, Shanghai 200240, China. He is the (co) author of more than 200 papers in mathematical, technical journals, and conferences. He is the Special appointment professor of Cheung Kong Scholars Programme. His current research interests include functional differential and difference equations, robust control and intelligent control for time-delay systems, chaos control and synchronization, and congestion control of networks.