# Cross-Domain Visual Representations via Unsupervised Graph Alignment

**Baoyao Yang, Pong C. Yuen**
Department of Computer Science, Hong Kong Baptist University, Hong Kong
byyang@comp.hkbu.edu.hk, pcyuen@comp.hkbu.edu.hk

## Abstract

In unsupervised domain adaptation, distributions of visual representations are mismatched across domains, which leads to the performance drop of a source model in the target domain. Therefore, distribution alignment methods have been proposed to explore cross-domain visual representations. However, most alignment methods have not considered the difference in distribution structures across domains, and the adaptation would subject to the insufficient aligned cross-domain representations. To avoid the misclassification/misidentification due to the difference in distribution structures, this paper proposes a novel unsupervised graph alignment method that aligns both data representations and distribution structures across the source and target domains. An adversarial network is developed for unsupervised graph alignment, which maps both source and target data to a feature space where data are distributed with unified structure criteria. Experimental results show that the graph-aligned visual representations achieve good performance on both cross-dataset recognition and cross-modal re-identification.

## 1 Introduction

In machine learning and pattern recognition, the generalization ability of a model decreases in the test dataset with the deviation of distributions between the training and test datasets. To overcome the problem of dataset bias when labels are unavailable in the test dataset, unsupervised domain adaptation (UDA) (Gopalan, Li, and Chellappa 2011) has been proposed. Recent researches (e.g., (Sun, Feng, and Saenko 2016; Tzeng et al. 2017)) have shown that aligning data distributions across the training dataset (source domain) and the test dataset (target domain) is a promising approach to obtain cross-domain representations for unsupervised domain adaptation. After distribution alignment, the cross-domain representations have similar distributions in the source and target domains, and therefore the target-domain generalization error is reduced.

To achieve the cross-domain representations, many unsupervised domain adaptation methods (Long et al. 2014b; 2015; Sun, Feng, and Saenko 2016; Sun and Saenko 2016) proposed to align the statistical measures (e.g., mean, variances) across the source and target domains. These methods perform well when source and target data are distributed with structures (Tenenbaum, De Silva, and Langford 2000; Long et al. 2014a; Hou et al. 2016) that can be nicely reproduced by the mean and covariance. But this assumption is difficult to guarantee in real-world applications. When the assumption is invalid, data distributions in the source and target domains are failed to be aligned by merely shifting the means and/or covariances of the source and target data. Other methods learned the cross-domain representations by subspace alignment (Fernando et al. 2013; Sun and Saenko 2015). Represented using the similar bases, source and target representations in the subspaces are regarded as aligned. However, the source and target data in each subspace may be variously distributed with different distribution structures, resulting in a mismatch between distributions of source and target data in the subspaces.

Instead of directly modeling the distribution alignment across domains, (Ganin et al. 2016; Tzeng et al. 2017) approximated the distribution alignment by learning the domain-invariant features that are indistinguishable to domains. Although target data are confused with source data in the feature space, this approximation is insufficient. Because without the guide of target labels, the mapping for each target data is arbitrary. Therefore, the structural information in the target distribution that needs to be preserved is destroyed in the feature space.

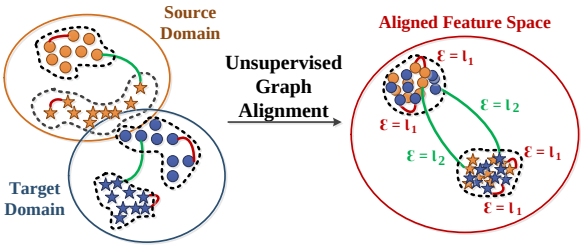


Figure 1: **Basic idea of unsupervised graph alignment**: Data are distributed with different structures in the source and target domains. Complete graphs are built in the source and target domains (edges are partially drawn) for alignment. In the aligned feature space, nodes of source and target data are closely located. The distribution structures are aligned across domains by constraining the values of source and target edges using the same criteria.

In this paper, we argue that data are distributed with different distribution structures (Tenenbaum, De Silva, and Langford 2000) in the source and target domains. That is, source/target data may be tightly or discretely located in the representation space without any prior geographic assumptions. Thus, the distribution alignment is insufficient in the existing methods that ignored the difference in distribution structures. Instead, this paper proposes an unsupervised graph alignment method to explore cross-domain representations, where source and target data have both similar representations and similar distribution structures. As shown in Figure 1, source data are linked to build a complete graph to represent the structural information of data distribution in the source domain. Similarly, a complete graph is built among target data. The values of edges in the source and target graphs are the geometric distance (Wang and Mahadevan 2013) between each pairs of nodes (data representations) in each graph, and therefore, structural information of source and target distributions is recorded in the edges of source and target graphs, respectively. The distribution alignment across domains can then be modeled as the alignment between the source and target graphs. Domain-indiscrimination loss is adopted for node alignment. To achieve the edge alignment without target labels, unified criteria are designed for source and target edges in the feature space. The unified criteria constrain that values of source and target edges should be minimized or approach to a fixed distance. We also propose a consistency constraint to preserve the target structural information among target features, so that arbitrary mapping of target data can be avoided with the guidance of the target structural information.

The contributions of this paper are listed as follows.

1. We propose an unsupervised graph alignment method to address the problem of structure difference in distribution for unsupervised domain adaptation. The proposed method aligns representations of source and target data, while matching the distribution structures in the source and target domains.
2. We design unified criteria for edges in both source and target domains. Constrained by the unified edge criteria, edges that represent the structures of source and target distributions are aligned without target labels.
3. We develop an adversarial network to learn the graph-aligned representations with similar distribution structures in the source and target domains. The graph-aligned representations not only are invariant to domains, but also preserve structural information in each domain.

## 2 Related Work

The problem of comparing distributions was addressed in (Gretton et al. 2007), and a statistic of Maximum Mean Discrepancy (MMD) was proposed to measure the probability of two samples from different distributions. As unsupervised domain adaptation that aims to align distributions across the source and target domains has become popular in recent years, the technique of MMD has been applied in many unsupervised domain adaptation methods to explore source and target features with similar distributions. For example, source and target data are mapped to a domain-invariant feature space with the criterion of MMD in (Gong, Grauman, and Sha 2013; Long et al. 2014b); TSC (Long et al. 2013) and DsGsDL (Yang, Ma, and Yuen 2018) adopted MMD in dictionary learning models to obtain the aligned sparse representations; and (Long et al. 2015) formulated the criterion of MMD as a loss function in deep-learning models. However, (Sun, Feng, and Saenko 2016) found that merely matching the statistical measure of mean was not enough for distribution alignment, because the source and target data could also diverge in the covariance. Therefore, they proposed to align the second-order of statistics for unsupervised domain adaptation. Incorporating the convolutional network, (Sun and Saenko 2016) then extended this idea to a deep-learning version.

On the other hand, (Gopalan, Li, and Chellappa 2011) proposed a concept of domain shift that modeled the distribution alignment across the source and target domains as the shift of subspaces in the manifolds. The subspaces were also formulated as the bases in dictionary learning models in (Ni, Qiu, and Chellappa 2013), which aligned the source and target domains by interpolating subspaces between the source and target domains. (Fernando et al. 2013) then summarized these methods as subspace alignment and represented the subspaces using eigenvectors extracted from PCA.

Besides, domain-indistinguishable features were presented in (Ganin et al. 2016) to align distributions for unsupervised domain adaptation. (Ganin et al. 2016) believed that the source and target distributions were aligned in the domain-indistinguishable features. With the popularity of generative adversarial networks, adversarial adaptation networks (Tzeng et al. 2017; Chen et al. 2018; Hu et al. 2018) were developed. These networks introduced a domain classifier to adversary the mapping of domain-invariant features, so that the mapping networks were optimized while refining the domain classifier.

## 3 Proposed Method

In this section, we introduce the proposed unsupervised graph alignment method that obtains cross-domain representations in unsupervised domain adaptation. As shown in Figure 2, the network for unsupervised graph alignment is composed of a source CNN, a target CNN, a domain discriminator, and a classifier. In the training phase, given source images with labels and unlabeled target images, the unsupervised graph alignment network is trained with four losses to achieve both node and edge alignments. Domain loss is adopted to align the features of source and target samples for node alignment. Edge alignment that aligns the distribution structures across domains is achieved by classification, discrepancy, and consistency losses. In the testing phase, the cross-domain representations of the source and target images, named as source and target features, are obtained by the trained source and target CNNs, respectively. The target features are then classified by the trained classifier to predict the class label, or matched to the source features for re-identification.

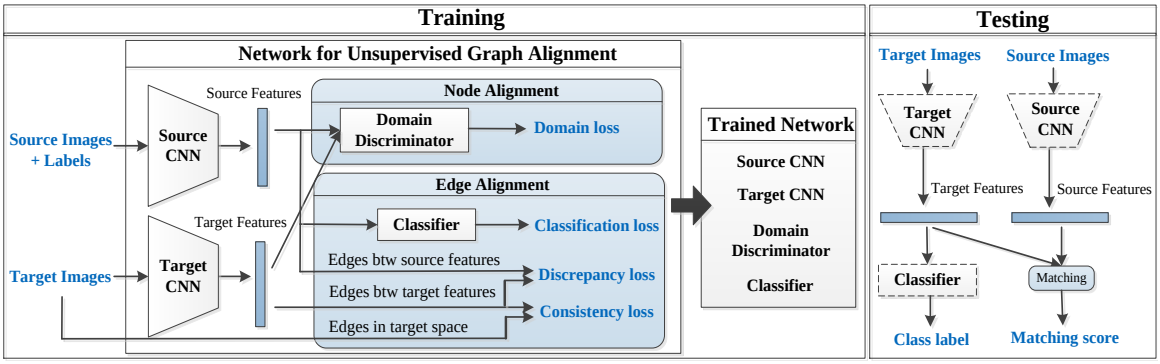


Figure 2: An overview of the proposed network

### 3.1 Graphs for Distribution Alignment

Denote source and target samples as $X_s$ and $X_t$, respectively. We obtain the source and target features (represented as $Z_s$ and $Z_t$, respectively) by mapping source and target samples using source and target CNNs, respectively. In each domain $d \in \{s, t\}$, the features $Z_d$ are linked to build a complete graph $G_{f_d} =< Z_d, \mathcal{E}_{f_d} >$, where nodes $Z_d$ are the features of samples from domain $d$, and the edges $\mathcal{E}_{f_d}$ represent the distance between each pair of features. Therefore, the structural information of feature distribution in each domain $d$ is contained in the edge of $\mathcal{E}_{f_d}$ in graph $G_{f_d}$. The edge values are calculated based on cosine similarity. For example, $\mathcal{E}_{f_d}^{ij} = \exp(-\frac{\boldsymbol{z}_d^i(\boldsymbol{z}_d^j)'}{\|\boldsymbol{z}_d^i\|\|\boldsymbol{z}_d^j\|})$ is the value of the edge between nodes (features) $\boldsymbol{z}_d^i$ and $\boldsymbol{z}_d^j$. Similarly, we build a graph for target data $X_t$, to represent the structural information of distribution for target data. The graph for target data is formed as $G_t =< X_t, \mathcal{E}_t >$, where $\mathcal{E}_t^{ij} = \exp(-\frac{\boldsymbol{x}_t^i(\boldsymbol{x}_t^j)'}{\|\boldsymbol{x}_t^i\|\|\boldsymbol{x}_t^j\|})$ represents the edge between target samples $\boldsymbol{x}_t^i$ and $\boldsymbol{x}_t^j$.

### 3.2 Domain-indiscrimination Loss for Node Alignment

To align the source and target distributions, we first consider the node alignment that aligns feature representations of source and target samples. However, directly learning the transformation between source and target samples is infeasible, as the pair-wise information of the source and target samples is unknown without target labels. Motivated by the idea of domain indiscrimination (Ganin et al. 2016), we introduce a domain discriminator $D$ to align source and target features. Source features $Z_s$ and target features $Z_t$ are regarded as aligned if $D$ cannot correctly predict the domain labels for source and target features $Z = \{Z_s, Z_t\}$. We write the mapping of the source and target CNNs as $M_s$ and $M_t$, i.e., $Z_s = M_s(X_s)$, $Z_t = M_t(X_t)$. The domain-indiscrimination loss for node alignment is then formulated as a cross-entropy loss function,

$$\begin{aligned}\mathcal{L}_{n_M}(X_s, X_t, M_s, M_t, D) = &-\mathbb{E}_{X_t}[\log D(M_t(X_t))] \\ &- \mathbb{E}_{X_s}[\log(1 - D(M_s(X_s)))]\end{aligned} \quad (1)$$

where domain labels of source and target samples are 1 and 0, respectively.

On the other hand, the domain discriminator $D$ is designed as an adversary network, to ensure its discrimination during the alignment. The loss function for domain discriminator $D$ is written as

$$\begin{aligned}\mathcal{L}_{n_D}(X_s, X_t, M_s, M_t, D) = &-\mathbb{E}_{X_s}[\log D(M_s(X_s))] \\ &- \mathbb{E}_{X_t}[\log(1 - D(M_t(X_t)))]\end{aligned} \quad (2)$$

The parameters of source CNN, target CNN, and domain discriminator are updated simultaneously. A discriminative discriminator is learned with the loss of Equation (2). Constrained by Equation (1), even the discriminative discriminator cannot correctly predict the domain labels for source and target features. In other words, source features $Z_s$ and target features $Z_t$ are aligned.

### 3.3 Edge Alignment with Unified Criteria Across Domains

As discussed in Section 1, distribution structures in the source and target domains should also be aligned to match data distributions across domains. As the distribution structural information in the source and target domains is recorded in the edges of graphs $G_{f_s}$ and $G_{f_t}$, respectively, the alignment of distribution structures can be modeled as the alignment between edges of $\mathcal{E}_{f_s}$ and $\mathcal{E}_{f_t}$. A straightforward method to achieve edge alignment involves regarding edge information as auxiliary features and directly aligning $\mathcal{E}_{f_s}$ and $\mathcal{E}_{f_t}$. However, expressed based on data similarity, edge information is indistinguishable and less representative, especially in the target domain where labels are unknown. Therefore, direct alignment of $\mathcal{E}_{f_s}$ and $\mathcal{E}_{f_t}$ has limited help for distribution alignment across domains.

Instead, we propose unified criteria for edges in both graphs $G_{f_s}$ and $G_{f_t}$ to minimize the discrepancy between source and target edges in the feature space. As shown in Figure 1, edges of $\mathcal{E}_{f_s}$ and $\mathcal{E}_{f_t}$ are divided into two categories: 1) internal edges (red edges) that link samples from the same class, and 2) interacted edges (green edges) that connect samples from different classes. We make the restriction that all of the edges in graphs $G_{f_s}$ and $G_{f_t}$ should meet the following criteria: 1) values of internal edges are minimized, and 2) values of interacted edges are approached to a fixed distance $\ell$. Expressed using the unified representations, the edges of $\mathcal{E}_{f_s}$ and $\mathcal{E}_{f_t}$ are aligned in the feature space.

To achieve these unified criteria, we first consider the formulation in the source domain. With label information, a classifier is introduced to preserve the discrimination in the feature space. Representing the classifier as $C$, the classification loss is formulated as

$$\mathcal{L}_{e_C}(X_s, M_s, C) = -\mathbb{E}_{(X_s, \boldsymbol{y}_s)}[\sum_k \mathbb{1}_{[\boldsymbol{y}_s=k]} \log C(M_s(X_s))] \quad (3)$$

where $\boldsymbol{y}_s$ denotes labels of source data $X_s$ and $k$ is the index of class. $C(M_s(X_s)) \in \mathbb{R}^{K\times n}$ is a probability matrix, where $K$ is the number of classes and $n$ is the number of source samples. $\mathbb{1}_{[\boldsymbol{y}_s=k]}$ represents a unit vector with nonzero value in the $k$-th element.

In the source domain, the edge $\mathcal{E}_{f_s}^{ij}$ between source features $\boldsymbol{z}_s^i$ and $\boldsymbol{z}_s^j$ can be allocated as internal edges or interacted edges based on the labels of $\boldsymbol{z}_s^i$ and $\boldsymbol{z}_s^j$. Thus, the values of edges between source features from the same class needs to be minimized, while the values of edges between samples from different classes should be approached to the value of $\ell$. That is to minimize the following loss function.

$$\mathcal{L}_{e_{D_s}}(\mathcal{E}_{f_s}) = \sum_{\boldsymbol{y}_s^i=\boldsymbol{y}_s^j} \parallel \mathcal{E}_{f_s}^{ij} \parallel_2^2 + \sum_{\boldsymbol{y}_s^i\neq\boldsymbol{y}_s^j} \parallel \ell - \mathcal{E}_{f_s}^{ij} \parallel_2^2 \quad (4)$$

With Equations (3) and (4), source features $Z_s$ are clustered into $K$ groups, where $K$ is the number of classes. Features within the same group have the same label and are closely distributed. Conversely, features from different groups have different labels, and the distances between samples from different groups are unified as $\ell$. In specific, we set $\ell = \mu_{f_s} + 2\sigma_{f_s}$, where $\mu_{f_s}$ and $\sigma_{f_s}$ are the mean and standard deviation of $\mathcal{E}_{f_s}$, respectively.

However, unlike the source domain, labels are unavailable in the target domain. Thus, target feature edges $\mathcal{E}_{f_t}$ cannot be categorized according to labels of feature pairs. In this paper, we propose obtaining the unified criteria for $\mathcal{E}_{f_t}$ by restricting target samples to be mapped to one group of source features. Each target feature is thus of high probability for one class. Introducing the above-mentioned classifier $C$, the probability of the target feature $Z_t^i$ for different classes can be represented as $P^i = C(M_t(X_t^i)) \in \mathbb{R}^{K\times 1}$. $P^{ik}$ denotes the $k$-th element in $P^i$, which records the probability of $Z_t^i$ for class $k$. We minimize $Z_t^i$ of high probability for two different classes, i.e., $P^{ik_1} * P^{ik_2}$, where $k_1 \neq k_2$. Summing $P^{ik_1} * P^{ik_2}$ for all samples and classes, the loss function can then be formulated as

$$\begin{aligned}\mathcal{L}_{e_{D_t}}(X_t, M_t, C) = &(C(M_t(X_t)))' C(M_t(X_t)) \\ &- tr((C(M_t(X_t)))' C(M_t(X_t)))\end{aligned} \quad (5)$$

With Equation (4) and (5), each target feature is distributed close to source samples from a certain class $k$, and away from samples from other classes with a fixed distance of $\ell$. In other words, the values of edges in $\mathcal{E}_{f_t}$ are minimized or optimized to $\ell$. Hence, edges $\mathcal{E}_{f_s}$ and $\mathcal{E}_{f_t}$ are unified in the feature space.

But without constraints from target labels, the mapping of each target sample is independent, and therefore target data are arbitrarily mapped close to source features from a random class. Consequently, the target structural information in the graph of target data ($G_t$) is likely to be broken among target features. To avoid arbitrary alignment and preserve target structural information in the feature space, we propose a consistency loss for target features. Given the edges of target data ($\mathcal{E}_t$) and the edges of target features ($\mathcal{E}_{f_t}$), the consistency loss is designed as

$$\mathcal{L}_{e_T}(\mathcal{E}_{f_t}, \mathcal{E}_t) = \sum_{i,j} \parallel \mathcal{H}(\mathcal{E}_{f_t}^{ij}, \mu_{f_t}) - \mathcal{H}(\mathcal{E}_t^{ij}, \mu_t) \parallel_2^2 \quad (6)$$

where $\mathcal{H}(\mathcal{E}, \mu) = 1/(1+e^{-(\mathcal{E}-\mu)})$ is a logistic function with sigmoid's midpoint $\mu$, and $\mu_t$ and $\mu_{f_t}$ are the mean of $\mathcal{E}_t$ and $\mathcal{E}_{f_t}$, respectively. Constrained by Equation (6), similar (dissimilar) target samples remain closely (distantly) located in the feature space. Thus, the structural information in the target domain is preserved among the aligned target features.

Combining node and edge alignments, the graph-aligned representations are obtained by solving the following optimizations,

$$\begin{aligned}&\min_{M_s} \mathcal{L}_{n_M}(X_s, M_s, D) + \mathcal{L}_{e_C}(X_s, M_s, C) + \mathcal{L}_{e_{D_s}}(\mathcal{E}_{f_s}) \\ &= -\mathbb{E}_{X_s}[\log(1 - D(M_s(X_s)))] \\ &\quad - \mathbb{E}_{(X_s, \boldsymbol{y}_s)}[\sum_k \mathbb{1}_{[\boldsymbol{y}_s=k]} \log C(M_s(X_s))] \\ &\quad + \sum_{\boldsymbol{y}_s^i=\boldsymbol{y}_s^j} \parallel \mathcal{E}_{f_s}^{ij} \parallel_2^2 + \sum_{\boldsymbol{y}_s^i\neq\boldsymbol{y}_s^j} \parallel \ell - \mathcal{E}_{f_s}^{ij} \parallel_2^2\end{aligned} \quad (7)$$

$$\begin{aligned}&\min_{M_t} \mathcal{L}_{n_M}(X_t, M_t, D) + \mathcal{L}_{e_{D_t}}(X_t, M_t, C) + \mathcal{L}_{e_T}(\mathcal{E}_{f_t}, \mathcal{E}_t) \\ &= -\mathbb{E}_{X_t}[\log(D(M_t(X_t)))] \\ &\quad + \sum_{i,j} \parallel \mathcal{H}(\mathcal{E}_{f_t}^{ij}, \mu_{f_t}) - \mathcal{H}(\mathcal{E}_t^{ij}, \mu_t) \parallel_2^2 \\ &\quad + (C(M_t(X_t)))' C(M_t(X_t)) \\ &\quad - tr((C(M_t(X_t)))' C(M_t(X_t)))\end{aligned} \quad (8)$$

$$\begin{aligned}&\min_D \mathcal{L}_{n_D}(X_s, X_t, M_s, M_t, D) \\ &= -\mathbb{E}_{X_s}[\log(D(M_s(X_s)))] - \mathbb{E}_{X_t}[\log(1 - D(M_t(X_t)))]\end{aligned} \quad (9)$$

$$\begin{aligned}&\min_C \mathcal{L}_{e_C}(X_s, M_s, C) + \mathcal{L}_{e_{D_t}}(X_t, M_t, C) \\ &= -\mathbb{E}_{(X_s, \boldsymbol{y}_s)}[\sum_k \mathbb{1}_{[\boldsymbol{y}_s=k]} \log C(M_s(X_s))] \\ &\quad + (C(M_t(X_t)))' C(M_t(X_t)) \\ &\quad - tr((C(M_t(X_t)))' C(M_t(X_t)))\end{aligned} \quad (10)$$

### 3.4 Optimization

We denote the parameters of source CNN, target CNN, domain discriminator, and classifier as $\theta_s$, $\theta_t$, $\theta_D$, and $\theta_C$, respectively. To solve the optimization problem of unsupervised graph alignment, we iteratively optimize $\theta_s$, $\theta_t$, $\theta_D$, and $\theta_C$ by fixing the other components unchanged. In each iteration, the gradients for parameters are calculated, and the parameters are then updated via backpropagation with mini-batch stochastic gradient descent (SGD) (Long et al. 2015). In the following, we present the gradients for the parameters of each component.

Table 1: Performance (%) of Digit Recognition Across Datasets

| Method | **M → U** | **M → S** | **U → M** | **S → M** | Avg. |
|---|---|---|---|---|---|
| GFK | 10.3±0.0 | 9.2±0.7 | 9.9±0.0 | 11.2±0.0 | 10.2 |
| SA | 8.8±0.6 | 11.6±2.3 | 11.2±0.0 | 10.2±0.7 | 10.5 |
| Coral | 9.7±2.2 | 12.2±2.2 | 10.2±0.0 | 9.9±0.5 | 10.5 |
| LeNet | 61.5±0.4 | 17.2±0.3 | 46.5±0.6 | 56.8±0.5 | 45.5 |
| DAN | 69.1±0.5 | 19.3±0.4 | 60.5±0.7 | 65.2±0.3 | 53.5 |
| WDAN | 72.6±0.3 | 23.4±0.2 | 65.4±0.4 | 67.4±0.4 | 57.2 |
| ADDA | 90.4±0.7 | **34.4**±4.1 | 96.1±0.4 | 63.2±4.5 | 71.0 |
| Ours | **94.1**±1.6 | 32.0±0.9 | **98.3**±0.1 | **85.0**±1.0 | **77.4** |

Table 2: Performance (%) of Object Recognition Across Datasets on *Office-10 + Caltech-10* Dataset

| Method | **D → C** | **W → C** | **A → C** | **C → D** | **C → W** | **C → A** | Avg. |
|---|---|---|---|---|---|---|---|
| GFK | 36.4±0.0 | 26.4±0.0 | 41.4±0.0 | 42.0±0.0 | 43.7±0.0 | 56.2±0.0 | 41.0 |
| SA | 34.4±0.0 | 32.3±0.0 | 40.6±0.0 | 43.7±0.0 | 40.6±0.0 | 45.4±0.0 | 39.5 |
| Coral | 33.8±0.0 | 33.8±0.0 | 45.1±0.0 | 45.9±0.0 | 46.4±0.0 | 52.1±0.0 | 42.8 |
| LeNet | 61.2±3.6 | 60.5±2.2 | 74.6±2.1 | 77.7±2.2 | 69.9±5.1 | 86.6±1.4 | 71.8 |
| ADDA (LeNet) | 74.7±3.9 | 75.9±2.9 | 78.4±1.5 | 28.0±5.1 | 47.1±5.3 | 77.7±3.5 | 66.0 |
| Ours (LeNet) | **80.6**±0.0 | **81.7**±0.0 | **82.7**±0.1 | **81.4**±0.7 | **80.0**±0.2 | **91.2**±0.1 | **82.9** |
| AlexNet | 80.8±0.4 | 76.1±0.5 | 83.8±0.3 | 89.0±0.3 | 83.1±0.3 | 91.1±0.2 | 84.0 |
| DDC (AlexNet) | 80.5±0.2 | 76.9±0.4 | 84.3±0.5 | 89.1±0.3 | 85.5 ±0.3 | 91.3±0.3 | 84.6 |
| DAN (AlexNet) | 82.0±0.4 | 81.5±0.3 | 86.0±0.5 | 90.5±0.1 | **92.0**±0.4 | 92.0±0.3 | 87.3 |
| Ours (AlexNet) | **85.8**±0.1 | **85.6**±0.2 | **86.8**±0.4 | **91.6**±0.4 | 89.1±0.5 | **92.7**±0.2 | **88.6** |

**Gradients for $\theta_D$ and $\theta_C$** Equations (9) and (10) are derivable, because $\mathcal{L}_{n_D}$ and $\mathcal{L}_{e_C}$ are typical cross-entropy losses, and $\mathcal{L}_{e_{D_t}}$ is a quadratic loss. Thus, the gradients for $\theta_D$ and $\theta_C$ can be obtained by simply computing the derivatives of $\partial\mathcal{L}_{n_D}/\partial\theta_D$ and $\partial(\mathcal{L}_{e_C}+\mathcal{L}_{e_{D_t}})/\partial\theta_C$, respectively. The formulations are not detailed presented in this paper due to the limited space.

**Gradient for $\theta_s$** The gradients of $\mathcal{L}_{n_M}$ and $\mathcal{L}_{e_C}$ for $\theta_s$ in Equation (7) can also be computed by derivation. On the other hand, $\mathcal{L}_{e_{D_s}}$ in Equation (7) is a formulation based on the pair-wise similarity $\mathcal{E}^{ij}_{f_s}$. This formulation causes a complexity of $O(n^2)$, where $n$ is the number of source data. However, the complexity of $O(n^2)$ is undesirable in deep-learning networks learned from large-scale datasets. Inspired by the unbiased estimation (Gretton et al. 2012), we adopt the strategy of sub-sampling without replacement to reduce the complexity of $\mathcal{L}_{e_{D_s}}$ to $O(n)$. In particular, the unbiased estimator for $\mathcal{L}_{e_{D_s}}$ is written as

$$\frac{2}{n}\sum_{i}^{n/2}(\mathbb{1}_{[\boldsymbol{y}_s^a=\boldsymbol{y}_s^b]} \parallel \boldsymbol{u}^i_{f_s} \parallel_2^2 \;+\; \mathbb{1}_{[\boldsymbol{y}_s^a\neq\boldsymbol{y}_s^b]} \parallel \ell - \boldsymbol{u}^i_{f_s} \parallel_2^2) \quad (11)$$

where $\boldsymbol{u}^i_{f_s} = \mathcal{E}^{ab}_{f_s} = \exp(-\frac{\boldsymbol{z}_s^a(\boldsymbol{z}_s^b)'}{\|\boldsymbol{z}_s^a\|\|\boldsymbol{z}_s^b\|})$, $a = 2i-1$ and $b = 2i$. In deep-learning networks, normalization losses are used for source features $\boldsymbol{z}_s^a$ and $\boldsymbol{z}_s^b$, and therefore, $\boldsymbol{u}^i_{f_s}$ can be approximated as $\boldsymbol{u}^i_{f_s} = \exp(-M_s(\boldsymbol{x}_s^a)(M_s(\boldsymbol{x}_s^b))')$.

In the mini-batch SGD, we only need to consider the gradient with respect to each $\boldsymbol{u}^i_{f_s}$, so that we only introduce the computation of gradient for $\mathcal{L}_{e_{D_s}}(\boldsymbol{u}^i_{f_s})$. The gradient of $\mathcal{L}_{e_{D_s}}$ can then be calculated by the combination of the gradients from each $\boldsymbol{u}^i_{f_s}$. We have

$$\begin{aligned}\frac{\partial\mathcal{L}_{e_{D_s}}(\boldsymbol{u}^i_{f_s})}{\partial\theta_s} &= \frac{\partial\mathcal{L}_{e_{D_s}}(\boldsymbol{u}^i_{f_s})}{\boldsymbol{u}^i_{f_s}}\frac{\partial\boldsymbol{u}^i_{f_s}}{\partial\theta_s} \\ &= 2(\boldsymbol{u}^i_{f_s}-\mathbb{1}_{[\boldsymbol{y}_s^a\neq\boldsymbol{y}_s^b]}\ell)\frac{\partial\boldsymbol{u}^i_{f_s}}{\partial\theta_s}\end{aligned} \quad (12)$$

In summary, the gradient for $\theta_s$ in Equation (7) is the summation of the gradients of $\mathcal{L}_{n_M}$, $\mathcal{L}_{e_C}$ and $\mathcal{L}_{e_{D_s}}$.

**Gradient for $\theta_t$** Similarly, unbiased estimation (Gretton et al. 2012) is adopted to reduce the complexity of $\mathcal{L}_{e_T}$ in Equation (8). We write unbiased estimator for $\mathcal{L}_{e_T}$ as

$$\frac{2}{n}\sum_{i}^{n/2}(\parallel \mathcal{H}(\boldsymbol{u}^i_{f_t},\mu_{f_t}) - \mathcal{H}(\boldsymbol{u}^i_t,\mu_t) \parallel_2^2) \quad (13)$$

where $\boldsymbol{u}^i_{f_t} = \mathcal{E}^{ab}_{f_t} \approx \exp(-M_t(\boldsymbol{x}_t^a)(M_t(\boldsymbol{x}_t^b))')$, $\boldsymbol{u}^i_t = \mathcal{E}^{ab}_t$, $a = 2i-1$ and $b = 2i$. For each $\boldsymbol{u}^i_{f_s}$, we have

$$\begin{aligned}\frac{\partial\mathcal{L}_{e_T}(\boldsymbol{u}^i_{f_t},\boldsymbol{u}^i_t)}{\partial\theta_t} &= \frac{\partial\mathcal{L}_{e_T}(\boldsymbol{u}^i_{f_t})}{\partial B}\frac{\partial B}{\partial\boldsymbol{u}^i_{f_t}}\frac{\partial\boldsymbol{u}^i_{f_t}}{\partial\theta_t} \\ &= 2(B-\mathcal{H}(\boldsymbol{u}^i_t,\mu_t))B(1-B)\frac{\partial\boldsymbol{u}^i_{f_t}}{\partial\theta_t}\end{aligned} \quad (14)$$

where $B = \mathcal{H}(\boldsymbol{u}^i_{f_t},\mu_{f_t})$.

Then, we achieve the gradient of Equation (8) as the combination of the gradients of $\mathcal{L}_{n_M}$, $\mathcal{L}_{e_{D_t}}$ and $\mathcal{L}_{e_T}$.

Table 3: Performance (%) of Cross-modal Person Re-identification on RegDB Dataset

| Dataset | Method | Target-domain Training Dataset | | | | | Target-domain Testing Dataset | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | r = 1 | r = 5 | r = 10 | r = 20 | mAP | r = 1 | r = 5 | r = 10 | r = 20 | mAP |
| Visible → Thermal | mLBP | 2.5 | 5.8 | 8.6 | 12.4 | 3.2 | 2.0 | 5.1 | 7.3 | 10.9 | 3.3 |
| | HOG | 8.7 | 16.2 | 21.8 | 29.7 | 7.8 | 7.3 | 14.3 | 19.8 | 28.2 | 7.2 |
| | AlexNet | 18.0 | 27.3 | **35.3** | **43.4** | 16.6 | 7.4 | 15.1 | 20.5 | 28.2 | 7.9 |
| | Ours | **23.4** | **29.4** | 34.3 | 43.2 | **24.4** | **10.6** | **20.3** | **26.4** | **35.3** | **10.7** |
| Thermal → Visible | mLBP | 2.3 | 6.5 | 10.9 | 16.7 | 3.1 | 2.5 | 6.4 | 9.9 | 14.5 | 3.0 |
| | HOG | 6.8 | 13.3 | 18.3 | 25.0 | 8.7 | 6.6 | 12.9 | 17.5 | 23.8 | 8.2 |
| | AlexNet | 17.5 | 29.6 | 37.1 | 45.8 | 15.9 | 7.2 | 16.7 | 22.3 | 30.9 | 7.8 |
| | Ours | **28.4** | **33.9** | **39.5** | **49.6** | **28.9** | **13.1** | **25.1** | **33.2** | **43.4** | **13.0** |

# 4 Experimental Results

In this section, we evaluate the proposed method on cross-dataset digit and object recognition. Evaluations are also performed on cross-modal re-identification. For each pair of datasets, experiments are conducted 10 times, and the averaged results in the target domains are reported. In Section 4.4, we visualize the aligned source and target features to further analyze the performance of the proposed unsupervised graph alignment method.

## 4.1 Cross-dataset Digit Recognition

Experiments of cross-dataset digit recognition are done across the full training set of three benchmarks (MNIST (LeCun et al. 1998), USPS and SVHN (Netzer et al. 2011) datasets). Ten classes of digits are contained in each dataset. In short, characters **M**, **U** and **S** are used to represent MNIST, USPS, and SVHN datasets, respectively. Four adaptation directions (i.e., **M** → **U**, **M** → **S**, **U** → **M**, and **S** → **M**) are used for evaluation. Following the network settings in (Tzeng et al. 2017), the source and target CNNs are implemented with the LeNet (LeCun et al. 1998), and the domain discriminator is implemented with three fully connected layers: two layers with 500 hidden units followed by the final discriminator output. Results are compared with statistical measure alignment methods (DAN (Long et al. 2015), WDAN (Yan et al. 2017), Coral (Sun, Feng, and Saenko 2016)), subspace alignment methods (GFK (Gong et al. 2012), SA (Fernando et al. 2013)), and ADDA (Tzeng et al. 2017) that learned the domain-indistinguishable features.

As shown in Table 1, the proposed method performs well among these four digit recognition experiments. The averaged accuracy of the proposed method is **77.4%**, which is the highest among all unsupervised domain adaptation methods. We achieve more than **6%** improvement over the second-best result. The proposed method also achieves the best results in the datasets of **M** → **U**, **U** → **M**, and **S** → **M**. However, our result in the **M** → **S** dataset is not as good as that of other datasets. This may be because the SVHN dataset is not well segmented, resulting in extensive noises contained in the structural information. These noises can spread to the aligned features via the consistency loss in the proposed adversarial network. Consequently, the recognition accuracy in the SVHN dataset is relatively low, but we still obtain the second best result in **M** → **S** dataset.

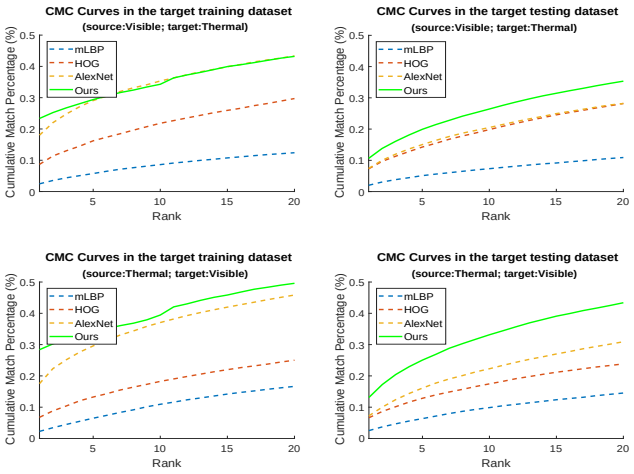


Figure 3: Cumulated matching characteristics (CMC) curves

## 4.2 Cross-dataset Object Recognition

Cross-dataset object recognition experiments are conducted across the *Office-10* and *Caltech-10* (Gong et al. 2012) datasets. In the *Office-10* dataset, 10 classes (e.g., bike, bag, keyboard) of object images are captured from three different conditions: the *Amazon* dataset consists of images downloaded from websites, and the *Dslr* and *Webcam* datasets include images taken by SLR and web cameras, respectively. Ten common categories of images that are shared with the *Office-10* dataset are extracted from the *Caltech-256* dataset to form the *Caltech-10* dataset. In Table 2, character **A**, **D**, **W** and **C** are used as the abbreviations to represent *Amazon*, *Dslr*, *Webcam*, and *Caltech-10* datasets, respectively. Six transfer tasks (**D** → **C**, **W** → **C**, **A** → **C**, **C** → **D**, **C** → **W**, and **C** → **A**) are formed across the *Office-10* and *Caltech-10* datasets for evaluation. The source and target CNNs in the proposed method are implemented using both LeNet (LeCun et al. 1998) and AlexNet (Krizhevsky, Sutskever, and Hinton 2012), so that our results can be fairly compared with the results of LeNet-based (ADDA (Tzeng et al. 2017)) and AlexNet-based (DDC (Tzeng et al. 2014), DAN (Long et al.

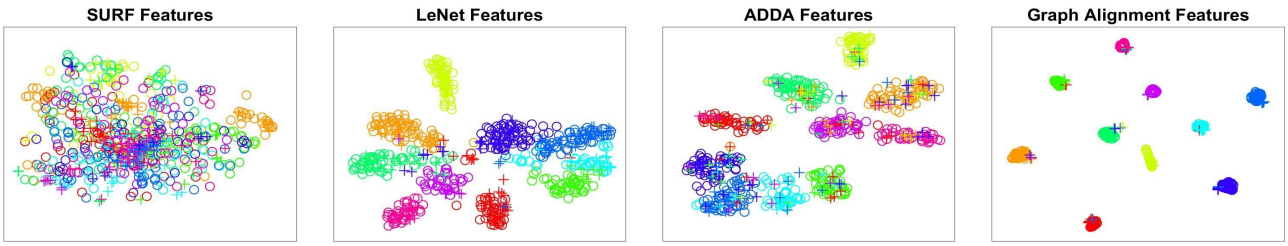


Figure 4: Spacial distributions of source (**C**) and target (**D**) features ($\circ$ : source-domain samples; $+$ : target-domain samples)

2015))) domain adaptation methods. As discussed in (Long et al. 2015), features extracted by the lower convolutional layers are general. Thus, we initialize the AlexNet using the pre-trained ImageNet and only update the last three layers for adaptation. The implementation of domain discriminator is the same as that introduced in Section 4.1.

Table 2 presents the results of object recognition across the *Office-10+Caltech-10* datasets. It is shown that the proposed method improves the recognition performance in the target domain by **11.1**% and **4.6**% with LeNet and AlexNet, respectively. Our method also outperforms other LeNet-based and AlexNet-based unsupervised domain adaptation methods in almost all (expect one) datasets. We obtain an average accuracy of **82.9**% and **88.6**% with LeNet and AlexNet, respectively, which are the best results in Table 2.

### 4.3 Cross-modal Re-identification

The graph-aligned representations are validated across modality on RegDB (Nguyen et al. 2017) dataset that contains images of 412 persons captured by dual camera systems. Two sub datasets are included in RegDB dataset: 1) Visible dataset with 10 visible light images of each person, and 2) Thermal dataset with 10 different thermal images of each person. Following the experimental protocol in (Ye et al. 2018), we randomly split the Visible and Thermal datasets into two halves for training and testing. In the training phase, training images from one modality with labels (source domain) and the training images from the other modality without labels (target domain) are used. For testing, images in the target domain are used as the probe set while images in the source domain with the other modality as the gallery set. The verifications are performed on both target-domain training and testing datasets. To indicate the performance, the standard cumulated matching characteristics (CMC) curves are plotted in Figure 3, and mean average precision (mAP) is listed in Table 3.

As shown in Table 3, the deep-learning (AlexNet) features obtain better performance than the hand-craft features (HOG (Dalal and Triggs 2005) and mLBP (Moore and Bowden 2011)) in the target-domain training dataset. But the performance of AlexNet features in the target-domain testing dataset is as poor as the HOG and mLBP features. In contrast, the proposed method achieves the highest average precision in both training and testing datasets in the target domain, which shows the better generalization ability of the proposed graph-aligned representations. Moreover, it is displayed in Figure 3 that the proposed method gets the highest matching scores at almost all ranks compared to other methods in each experiment.

### 4.4 Visualization

In this section, we visualize graph-aligned representations in the source and target domains to qualitatively show the performance of the proposed method. T-SNE (Maaten and Hinton 2008) is employed, and the dataset of $\mathbf{C} \rightarrow \mathbf{D}$ is selected for visualization. SURF (Bay, Tuytelaars, and Van Gool 2006) features, features extracted from LeNet (LeCun et al. 1998) and ADDA (Tzeng et al. 2017) are also shown in Figure 4 for comparison. Symbols $\circ$ and $+$ are used to mark the features from source and target domains, respectively. Features of different classes are presented in different colors.

As shown in Figure 4, samples from *Caltech-10* (source) and *Dslr* (target) datasets are disorderly distributed in the SURF feature space. This indicates that the hand-crafted features of SURF are not strongly discriminative toward class labels in both the *Caltech-10* and *Dslr* datasets. On the other hand, LeNet features are learned from source images and their labels, and therefore the source-domain LeNet features are more distinguishable for the class labels. But the discrimination is not extended to the Dslr dataset (target domain), as the distributions of LeNet features from *Caltech-10* and *Dslr* datasets are not nicely matched. Compared to LeNet features, the marginal distributions of ADDA features from the *Caltech-10* and *Dslr* datasets are better aligned. However, it can be found that each target sample is arbitrarily aligned to one cluster of source samples. In addition, some samples from different categories are closely distributed in the ADDA feature space, which results in a difficulty of classification for samples from these classes.

In contrast, the proposed method aligns both the representations and the distribution structures of the source and target samples. Thus, as shown in the last sub-figure in Figure 4, the graph-aligned features of *Caltech-10* and *Dslr* datasets have similar distributions. Moreover, constrained by the same structural criteria, samples from the same class are clustered in groups with small variances, and samples from different classes are separately distributed. Figure 4 also shows that samples from *Dslr* datasets are less likely to be mapped to the wrong class in the graph alignment feature space. Because the mapping of target samples is guided by the constraint of consistency, and structural information in the target domain are preserved in the graph alignment fea-

ture space. Therefore, the graph-aligned features from the *Dslr* dataset are more discriminative toward class labels.

## 5 Conclusion

This paper proposes an unsupervised graph alignment method to address the problem of structural difference between the source and target distributions in unsupervised domain adaptation. An adversarial network is developed to explore the cross-domain visual representations with losses of the node and edge alignments. Unified criteria are designed for edge alignment and used as loss functions for propagation in the adversarial network. Experimental results show that the proposed method achieves promising results in the tasks of cross-domain recognition and re-identification.

## Acknowledgments

This work was partially supported by the Science Faculty Research Grant of Hong Kong Baptist University, Hong Kong Research Grants Council General Research Fund: *RGC/HKBU12200518*.

## References

Bay, H.; Tuytelaars, T.; and Van Gool, L. 2006. Surf: Speeded up robust features. In *ECCV*.

Chen, Q.; Liu, Y.; Wang, Z.; Wassell, I.; and Chetty, K. 2018. Re-weighted adversarial adaptation network for unsupervised domain adaptation. In *CVPR*.

Dalal, N., and Triggs, B. 2005. Histograms of oriented gradients for human detection. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*.

Fernando, B.; Habrard, A.; Sebban, M.; and Tuytelaars, T. 2013. Unsupervised visual domain adaptation using subspace alignment. In *ICCV*.

Ganin, Y.; Ustinova, E.; Ajakan, H.; Germain, P.; Larochelle, H.; Laviolette, F.; Marchand, M.; and Lempitsky, V. 2016. Domain-adversarial training of neural networks. *JMLR*.

Gong, B.; Shi, Y.; Sha, F.; and Grauman, K. 2012. Geodesic flow kernel for unsupervised domain adaptation. In *CVPR*.

Gong, B.; Grauman, K.; and Sha, F. 2013. Connecting the dots with landmarks: Discriminatively learning domain-invariant features for unsupervised domain adaptation. In *ICML*.

Gopalan, R.; Li, R.; and Chellappa, R. 2011. Domain adaptation for object recognition: An unsupervised approach. In *ICCV*.

Gretton, A.; Borgwardt, K. M.; Rasch, M.; Schölkopf, B.; and Smola, A. J. 2007. A kernel method for the two-sample-problem. In *NIPS*.

Gretton, A.; Borgwardt, K. M.; Rasch, M. J.; Schölkopf, B.; and Smola, A. 2012. A kernel two-sample test. *JMLR*.

Hou, C.-A.; Tsai, Y.-H. H.; Yeh, Y.-R.; and Wang, Y.-C. F. 2016. Unsupervised domain adaptation with label and structural consistency. *IEEE Transactions on Image Processing*.

Hu, L.; Kan, M.; Shan, S.; and Chen, X. 2018. Duplex generative adversarial network for unsupervised domain adaptation. In *CVPR*.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. 2012. Imagenet classification with deep convolutional neural networks. In *NIPS*.

LeCun, Y.; Bottou, L.; Bengio, Y.; and Haffner, P. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*.

Long, M.; Ding, G.; Wang, J.; Sun, J.; Guo, Y.; and Philip, S. Y. 2013. Transfer sparse coding for robust image representation. In *CVPR*.

Long, M.; Wang, J.; Ding, G.; Shen, D.; and Yang, Q. 2014a. Transfer learning with graph co-regularization. *TKDE*.

Long, M.; Wang, J.; Ding, G.; Sun, J.; and Yu, P. S. 2014b. Transfer joint matching for unsupervised domain adaptation. In *CVPR*.

Long, M.; Cao, Y.; Wang, J.; and Jordan, M. 2015. Learning transferable features with deep adaptation networks. In *ICML*.

Maaten, L. v. d., and Hinton, G. 2008. Visualizing data using t-sne. *JMLR*.

Moore, S., and Bowden, R. 2011. Local binary patterns for multi-view facial expression recognition. *CVIU*.

Netzer, Y.; Wang, T.; Coates, A.; Bissacco, A.; Wu, B.; and Ng, A. Y. 2011. Reading digits in natural images with unsupervised feature learning. In *NIPS workshop*.

Nguyen, D.; Hong, H. G.; Kim, K. W.; and Park, K. R. 2017. Person recognition system based on a combination of body images from visible light and thermal cameras. *Sensors*.

Ni, J.; Qiu, Q.; and Chellappa, R. 2013. Subspace interpolation via dictionary learning for unsupervised domain adaptation. In *CVPR*.

Sun, B., and Saenko, K. 2015. Subspace distribution alignment for unsupervised domain adaptation. In *BMVC*.

Sun, B., and Saenko, K. 2016. Deep coral: Correlation alignment for deep domain adaptation. In *ECCV*.

Sun, B.; Feng, J.; and Saenko, K. 2016. Return of frustratingly easy domain adaptation. In *AAAI*.

Tenenbaum, J. B.; De Silva, V.; and Langford, J. C. 2000. A global geometric framework for nonlinear dimensionality reduction. *science*.

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

Tzeng, E.; Hoffman, J.; Saenko, K.; and Darrell, T. 2017. Adversarial discriminative domain adaptation. In *CVPR*.

Wang, C., and Mahadevan, S. 2013. Manifold alignment preserving global geometry. In *IJCAI*.

Yan, H.; Ding, Y.; Li, P.; Wang, Q.; Xu, Y.; and Zuo, W. 2017. Mind the class weight bias: Weighted maximum mean discrepancy for unsupervised domain adaptation. In *CVPR*.

Yang, B.; Ma, J.; and Yuen, P. 2018. Domain-shared group-sparse dictionary learning for unsupervised domain adaptation. In *AAAI*.

Ye, M.; Lan, X.; Li, J.; and Yuen, P. 2018. Hierarchical discriminative learning for visible thermal person re-identification. In *AAAI*.