Cross-Domain Visual Matching via Generalized Similarity Measure and Feature Learning

Liang Lin, Guangrun Wang, Wangmeng Zuo, Xiangchu Feng, and Lei Zhang

Abstract—Cross-domain visual data matching is one of the fundamental problems in many real-world vision tasks, e.g., matching persons across ID photos and surveillance videos. Conventional approaches to this problem usually involves two steps: i) projecting samples from different domains into a common space, and ii) computing (dis-)similarity in this space based on a certain distance. In this paper, we present a novel pairwise similarity measure that advances existing models by i) expanding traditional linear projections into affine transformations and ii) fusing affine Mahalanobis distance and Cosine similarity by a data-driven combination. Moreover, we unify our similarity measure with feature representation learning via deep convolutional neural networks. Specifically, we incorporate the similarity measure matrix into the deep architecture, enabling an end-to-end way of model optimization. We extensively evaluate our generalized similarity model in several challenging cross-domain matching tasks: person re-identification under different views and face verification over different modalities (i.e., faces from still images and videos, older and younger faces, and sketch and photo portraits). The experimental results demonstrate superior performance of our model over other state-of-the-art methods.

Index Terms—Similarity model, cross-domain matching, person verification, deep learning

1 Introduction

Visual similarity matching is arguably considered as one of the most fundamental problems in computer vision and pattern recognition, and this problem becomes more challenging when dealing with cross-domain data. For example, in still-video face retrieval, a newly rising task in visual surveillance, faces from still images captured under a constrained environment are utilized as the queries to find the matches of the same identity in unconstrained videos. Age-invariant and sketch-photo face verification tasks are also examples of cross-domain image matching. Some examples in these applications are shown in Fig. 1.

Conventional approaches (e.g., canonical correlation analysis [1] and partial least square regression [2]) for cross-domain matching usually follow a procedure of two steps:

1) Samples from different modalities are first projected into a common space by learning a transformation. One may simplify the computation by assuming that these cross domain samples share the same projection.
2) A certain distance is then utilized for measuring the similarity/disimilarity in the projection space. Usually Euclidean distance or inner product are used.

- *L. Lin and G. Wang are with School of Data and Computer Science, Sun Yat-sen University, Guangzhou, P.R. China, and also with Collaborative Innovation Center of High Performance Computing, National University of Defense Technology, Changsha 410073, China. E-mail: linliang@ieee.org, wanggrun@mail2.sysu.edu.cn.*
- *W. Zuo is with the School of Computer Science and Technology, Harbin Institute of Technology, Harbin, P.R. China. E-mail: cswmzuo@gmail.com.*
- *X. Feng is with the School of Mathematics and Statistics, Xidian University, Xi'an, P.R. China. E-mail: xcfeng@mail.xidian.edu.cn.*
- *L. Zhang is with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong. E-mail: cslzhang@comp.polyu.edu.hk.*

Manuscript received 19 Aug. 2015; revised 23 Apr. 2016; accepted 4 May 2016. Date of publication 11 May 2016; date of current version 12 May 2017. Recommended for acceptance by E. G. Learned-Miller.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TPAMI.2016.2567386

Suppose that $\mathbf{x}$ and $\mathbf{y}$ are two samples of different modalities, and $\mathbf{U}$ and $\mathbf{V}$ are two projection matrices applied on $\mathbf{x}$ and $\mathbf{y}$, respectively. $\mathbf{Ux}$ and $\mathbf{Vy}$ are usually formulated as linear similarity transformations mainly for the convenience of optimization. A similarity transformation has a good property of preserving the shape of an object that goes through this transformation, but it is limited in capturing complex deformations that usually exist in various real problems, e.g., translation, shearing, and their compositions. On the other hand, Mahalanobis distance, Cosine similarity, and their combination have been widely studied in the research of similarity metric learning, but it remains less investigated on how to unify feature learning and similarity learning, in particular, how to combine Mahalanobis distance with Cosine similarity and integrate the distance metric with deep neural networks for end-to-end learning.

To address the above issues, in this work we present a more general similarity measure and unify it with deep convolutional representation learning. One of the key innovations is that we generalize the existing similarity models from two aspects. First, we extend the similarity transformations $\mathbf{Ux}$ and $\mathbf{Vy}$ to the affine transformations by adding a translation vector into them, i.e., replacing $\mathbf{Ux}$ and $\mathbf{Vy}$ with $\mathbf{L_A x} + \mathbf{a}$ and $\mathbf{L_B y} + \mathbf{b}$, respectively. Affine transformation is a generalization of similarity transformation without the requirement of preserving the original point in a linear space, and it is able to capture more complex deformations. Second, unlike the traditional approaches choosing either Mahalanobis distance or Cosine similarity, we combine these two measures under the affine transformation. This combination is realized in a data-driven fashion, as discussed in the Appendix, resulting in a novel generalized similarity measure, defined as:

$$S(\mathbf{x}, \mathbf{y}) = [\mathbf{x}^T \ \mathbf{y}^T \ 1] \begin{bmatrix} \mathbf{A} & \mathbf{C} & \mathbf{d} \\ \mathbf{C}^T & \mathbf{B} & \mathbf{e} \\ \mathbf{d}^T & \mathbf{e}^T & f \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \\ 1 \end{bmatrix}, \quad (1)$$

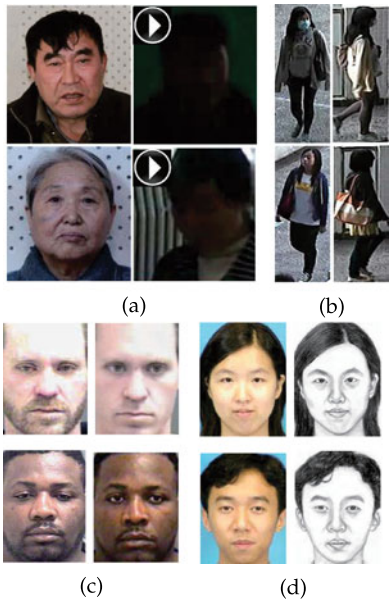


Fig. 1. Typical examples of matching cross-domain visual data. (a) Faces from still images and vidoes. (b) Front- and side-view persons. (c) Older and younger faces. (d) Photo and sketch faces.

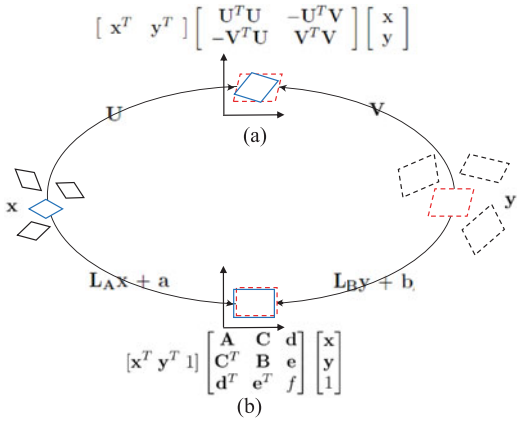


Fig. 2. Illustration of the generalized similarity model. Conventional approaches project data by simply using the linear similarity transformations (i.e., $\mathbf{U}$ and $\mathbf{V}$), as illustrated in (a), where Euclidean distance is applied as the distance metric. As illustrated in (b), we improve existing models by i) expanding the traditional linear similarity transformation into an affine transformation and ii) fusing Mahalanobis distance and Cosine similarity. One can see that the case in (a) is a simplified version of our model. Please refer to Appendix section for the deduction details.

where sub-matrices $\mathbf{A}$ and $\mathbf{B}$ are positive semi-definite, representing the self-correlations of the samples in their own domains, and $\mathbf{C}$ is a correlation matrix crossing the two domains.

Fig. 2 intuitively explains the idea.[1] In this example, it is observed that Euclidean distance under the linear transformation, as (a) illustrates, can be regarded as a special case of our model with $\mathbf{A} = \mathbf{U}^T\mathbf{U}$, $\mathbf{B} = \mathbf{V}^T\mathbf{V}$, $\mathbf{C} = -\mathbf{U}^T\mathbf{V}$, $\mathbf{d} = \mathbf{0}$, $\mathbf{e} = \mathbf{0}$, and $f = 0$. Our similarity model can be viewed as a generalization of several recent metric learning models [3], [4]. Experimental results validate that the introduction of $(\mathbf{d}, \mathbf{e}, f)$ and more flexible setting on $(\mathbf{A}, \mathbf{B}, \mathbf{C})$ do improve the matching performance significantly.

Another innovation of this work is that we unify feature representation learning and similarity measure learning. In literature, most of the existing models are performed in the original data space or in a pre-defined feature space, that is, the feature extraction and the similarity measure are studied separately. These methods may have several drawbacks in practice. For example, the similarity models heavily rely on feature engineering and thus lack of generality when handling problems under different scenarios. Moreover, the interaction between the feature representations and similarity measures is ignored or simplified, thus limiting their performances. Meanwhile, deep learning, especially the Convolutional Neural Network (CNN), has demonstrated its effectiveness on learning discriminative features from raw data and benefited to build end-to-end learning frameworks. Motivated by these works, we build a deep architecture to integrate our similarity measure with the CNN-based feature representation learning. Our architecture takes raw images of different modalities as the inputs and automatically produce their representations by sequentially stacking shared sub-network upon domain-specific subnetworks. Upon these layers, we further incorporate the components of our similarity measure by stimulating them with several appended structured neural network layers. The feature learning and the similarity model learning are thus integrated for end-to-end optimization.

In sum, this paper makes three main contributions to cross-domain similarity measure learning.

- First, it presents a generic similarity measure by generalizing the traditional linear projection and distance metrics into a unified formulation. Our model can be viewed as a generalization of several existing similarity learning models.
- Second, it integrates feature learning and similarity measure learning by building an end-to-end deep architecture of neural networks. Our deep architecture effectively improves the adaptability of learning with data of different modalities.
- Third, we extensively evaluate our framework on four challenging tasks of cross-domain visual matching: person re-identification across views,[2] and face verification under different modalities (i.e., faces from still images and videos, older and younger faces, and sketch and photo portraits). The experimental results show that our similarity model outperforms other state-of-the-arts in three of the four tasks and achieves the second best performance in the other one.

The rest of the paper is organized as follows. Section 2 reviews related work. Section 3 introduces our generalized similarity model and discusses its connections to existing works. Section 4 presents the proposed deep neural network architecture and the learning algorithm in Section 4.2. The experimental results, comparisons and ablation studies are presented in Section 5. Section 6 concludes the paper.

1. Fig. 2 does not imply that our model geometrically aligns two samples to be matched. Using this example we emphasize the superiority of the affine transformation over the traditional linear similarity transformation on capturing pattern variations in the feature space.

2. Person re-identification is arguably a cross-domain matching problem. We introduce it in our experiments since this problem has been receiving increasing attentions recently.

2 Related Work

In literature, to cope with the cross-domain matching of visual data, one can learn a common space for different domains. CCA [1] learns the common space via maximizing cross-view correlation, while PLS [2] is learned via maximizing cross-view covariance. Coupled information-theoretic encoding is proposed to maximize the mutual information [5]. Another conventional strategy is to synthesize samples from the input domain into the other domain. Rather than learning the mapping between two domains in the data space, dictionary learning [6], [7] can be used to alleviate cross-domain heterogeneity, and semi-coupled dictionary learning (SCDL [7]) is proposed to model the relationship on the sparse coding vectors from the two domains. Duan et al. proposed another framework called domain adaptation machine (DAM) [8] for multiple source domain adaption but they need a set of pre-trained base classifiers.

Various discriminative common space approaches have been developed by utilizing the label information. Supervised information can be employed by the Rayleigh quotient [1], treating the label as the common space [9], or employing the max-margin rule [10]. Using the SCDL framework, structured group sparsity was adopted to utilize the label information [6]. Generalization of discriminative common space to multiview was also studied [11]. Kan et al. proposed a multiview discriminant analysis (MvDA [12]) method to obtain a common space for multiple views by optimizing both inter-view and intra-view Rayleigh quotient. In [13], a method to learn shape models using local curve segments with multiple types of distance metrics was proposed.

Moreover, for most existing multiview analysis methods, the target is defined based on the standard inner product or distance between the samples in the feature space. In the field of metric learning, several generalized similarity/distance measures have been studied to improve recognition performance. In [4], [14], the generalized distance/similarity measures are formulated as the difference between the distance component and the similarity component to take into account both cross inner product term and two norm terms. Li et al. [3] adopted the second-order decision function as distance measure without considering the positive semi-definite (PSD) constraint. Chang and Yeung [15] suggested an approach to learn locally smooth metrics using local affine transformations while preserving the topological structure of the original data. These distance/similarity measures, however, were developed for matching samples from the same domain, and they cannot be directly applied to cross domain data matching.

To extend traditional single-domain metric learning, Mignon and Jurie [16] suggested a cross-modal metric learning (CMML) model, which learns domain-specific transformations based on a generalized logistic loss. Zhai et al. [17] incorporated the joint graph regularization with the heterogeneous metric learning model to improve the cross-media retrieval accuracy. In [16], [17], Euclidean distance is adopted to measure the dissimilarity in the latent space. Instead of explicitly learning domain-specific transformations, Kang et al. [18] learned a low rank matrix to parameterize the cross-modal similarity measure by the accelerated proximal gradient (APG) algorithm. However, these methods are mainly based on the common similarity or distance measures and none of them addresses the feature learning problem under the cross-domain scenarios.

Instead of using hand-crafted features, learning feature representations and contextual relations with deep neural networks, especially the convolutional neural network (CNN) [19], has shown great potential in various pattern recognition tasks such as object recognition [20] and semantic segmentation [21]. Significant performance gains have also been achieved in face recognition [22] and person re-identification [23], [24], [25], [26], mainly attributed to the progress in deep learning. Recently, several deep CNN-based models have been explored for similarity matching and learning. For example, Andrew et al. [27] proposed a multi-layer CCA model consisting of several stacked nonlinear transformations. Li et al. [28] learned filter pairs via deep networks to handle misalignment, photometric and geometric transforms, and achieved promising results for the person re-identification task. Wang et al. [29] learned fine-grained image similarity with deep ranking model. Yi et al. [30] presented a deep metric learning approach by generalizing the Siamese CNN. Ahmed et al. [25] proposed a deep convolutional architecture to measure the similarity between a pair of pedestrian images. Besides the shared convolutional layers, their network also includes a neighborhood difference layer and a patch summary layer to compute cross-input neighborhood differences. Wang et al. [26] proposed a deep ranking framework to learn the joint representation of an image pair and return the similarity score directly, in which the similarity model is replaced by full connection layers.

Our deep model is partially motivated by the above works, and we target on a more powerful solution of cross-domain visual matching by incorporating a generalized similarity function into deep neural networks. Moreover, our network architecture is different from existing works, leading to new state-of-the-art results on several challenging person verification and recognition tasks.

3 Generalized Similarity Model

In this section, we first introduce the formulation of our deep generalized similarity model and then discuss the connections between our model and existing similarity learning methods.

3.1 Model Formulation

According to the discussion in Section 1, our generalized similarity measure extends the traditional linear projection and integrates Mahalanobis distance and Cosine similarity into a generic form, as shown in Eqn. (1). As we derive in the Appendix, $\mathbf{A}$ and $\mathbf{B}$ in our similarity measure are positive semi-definite but $\mathbf{C}$ does not obey this constraint. Hence, we can further factorize $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$, as:

$$\begin{aligned} \mathbf{A} &= {\mathbf{L}_\mathbf{A}}^T \mathbf{L}_\mathbf{A}, \\ \mathbf{B} &= {\mathbf{L}_\mathbf{B}}^T \mathbf{L}_\mathbf{B}, \\ \mathbf{C} &= -{\mathbf{L}_\mathbf{C}^\mathbf{x}}^T \mathbf{L}_\mathbf{C}^\mathbf{y}. \end{aligned} \tag{2}$$

Moreover, our model extracts feature representation (i.e., $\mathbf{f}_1(\mathbf{x})$ and $\mathbf{f}_2(\mathbf{y})$) from the raw input data by utilizing the

CNNs. Incorporating the feature representation and the above matrix factorization into Eqn. (1), we can thus have the following similarity model:

$$\begin{aligned}\tilde{S}(\mathbf{x},\mathbf{y}) &= S(\mathbf{f}_1(\mathbf{x}),\mathbf{f}_2(\mathbf{y})),\\ &= [\mathbf{f}_1(\mathbf{x})^T\ \mathbf{f}_2(\mathbf{y})^T\ 1]\begin{bmatrix}\mathbf{A} & \mathbf{C} & \mathbf{d}\\ \mathbf{C}^T & \mathbf{B} & \mathbf{e}\\ \mathbf{d}^T & \mathbf{e}^T & f\end{bmatrix}\begin{bmatrix}\mathbf{f}_1(\mathbf{x})\\ \mathbf{f}_2(\mathbf{y})\\ 1\end{bmatrix},\\ &= \|\mathbf{L_A}\mathbf{f}_1(\mathbf{x})\|^2 + \|\mathbf{L_B}\mathbf{f}_2(\mathbf{y})\|^2 + 2\mathbf{d}^T\mathbf{f}_1(\mathbf{x})\\ &\quad - 2(\mathbf{L}_\mathbf{C}^\mathbf{x}\mathbf{f}_1(\mathbf{x}))^T(\mathbf{L}_\mathbf{C}^\mathbf{y}\mathbf{f}_2(\mathbf{y})) + 2\mathbf{e}^T\mathbf{f}_2(\mathbf{y}) + f.\end{aligned} \tag{3}$$

Specifically, $\mathbf{L_A}\mathbf{f}_1(\mathbf{x})$, $\mathbf{L}_\mathbf{C}^\mathbf{x}\mathbf{f}_1(\mathbf{x})$, $\mathbf{d}^T\mathbf{f}_1(\mathbf{x})$ can be regarded as the similarity components for $\mathbf{x}$, while $\mathbf{L_B}\mathbf{f}_2(\mathbf{y})$, $\mathbf{L}_\mathbf{C}^\mathbf{y}\mathbf{f}_2(\mathbf{y})$, $\mathbf{d}^T\mathbf{f}_2(\mathbf{y})$ accordingly for $\mathbf{y}$. These similarity components are modeled as the weights that connect neurons of the last two layers. For example, a portion of output activations represents $\mathbf{L_A}\mathbf{f}_1(\mathbf{x})$ by taking $\mathbf{f}_1(\mathbf{x})$ as the input and multiplying the corresponding weights $\mathbf{L_A}$. In the following, we discuss the formulation of our similarity learning.

The objective of our similarity learning is to seek a function $\tilde{S}(\mathbf{x},\mathbf{y})$ that satisfies a set of similarity/disimilarity constraints. Instead of learning similarity function on hand-crafted feature space, we take the raw data as input, and introduce a deep similarity learning framework to integrate nonlinear feature learning and generalized similarity learning. Recall that our deep generalized similarity model is in Eqn. (1). $(\mathbf{f}_1(\mathbf{x}),\mathbf{f}_2(\mathbf{y}))$ are the feature representations for samples of different modalities, and we use $\mathbf{W}$ to indicate their parameters. We denote $\mathbf{\Phi} = (\mathbf{L_A},\mathbf{L_B},\mathbf{L}_\mathbf{C}^\mathbf{x},\mathbf{L}_\mathbf{C}^\mathbf{y},\mathbf{d},\mathbf{e},f)$ as the similarity components for sample matching. Note that $\tilde{S}(\mathbf{x},\mathbf{y})$ is asymmetric, i.e., $\tilde{S}(\mathbf{x},\mathbf{y}) \neq \tilde{S}(\mathbf{y},\mathbf{x})$. This is reasonable for cross-domain matching, because the similarity components are domain-specific.

Assume that $\mathcal{D} = \{(\{\mathbf{x}_i,\mathbf{y}_i\},\ell_i)\}_{i=1}^N$ is a training set of cross-domain sample pairs, where $\{\mathbf{x}_i,\mathbf{y}_i\}$ denotes the ith pair, and ℓ_i denotes the corresponding label of $\{\mathbf{x}_i,\mathbf{y}_i\}$ indicating whether $\mathbf{x}_i$ and $\mathbf{y}_i$ are from the same class:

$$\ell_i = \ell(\mathbf{x}_i,\mathbf{y}_i) = \begin{cases}-1, & c(\mathbf{x}) = c(\mathbf{y})\\ 1, & \text{otherwise},\end{cases} \tag{4}$$

where $c(\mathbf{x})$ denotes the class label of the sample $\mathbf{x}$. An ideal deep similarity model is expected to satisfy the following constraints:

$$\tilde{S}(\mathbf{x}_i,\mathbf{y}_i)\begin{cases}< -1, & \text{if } \ell_i = -1\\ \geq 1, & \text{otherwise}\end{cases} \tag{5}$$

for any $\{\mathbf{x}_i,\mathbf{y}_i\}$.

Note that the feasible solution that satisfies the above constraints may not exist. To avoid this scenario, we relax the hard constraints in Eqn. (5) by introducing a hinge-like loss:

$$G(\mathbf{W},\mathbf{\Phi}) = \sum_{i=1}^N (1 - \ell_i\tilde{S}(\mathbf{x}_i,\mathbf{y}_i))_+. \tag{6}$$

To improve the stability of the solution, some regularizers are further introduced, resulting in our deep similarity learning model:

$$(\hat{\mathbf{W}},\hat{\mathbf{\Phi}}) = \arg\min_{\mathbf{W},\mathbf{\Phi}} \sum_{i=1}^N (1 - \ell_i\tilde{S}(\mathbf{x}_i,\mathbf{y}_i))_+ + \Psi(\mathbf{W},\mathbf{\Phi}), \tag{7}$$

where $\Psi(\mathbf{W},\mathbf{\Phi}) = \lambda\|\mathbf{W}\|^2 + \mu\|\mathbf{\Phi}\|^2$ denotes the regularizer on the parameters of the feature representation and generalized similarity models.

3.2 Connection with Existing Models

Our generalized similarity learning model is a generalization of many existing metric learning models, while they can be treated as special cases of our model by imposing some extra constraints on $(\mathbf{A},\mathbf{B},\mathbf{C},\mathbf{d},\mathbf{e},f)$.

Conventional similarity model usually is defined as $S_\mathbf{M}(\mathbf{x},\mathbf{y}) = \mathbf{x}^T\mathbf{M}\mathbf{y}$, and this form is equivalent to our model, when $\mathbf{A} = \mathbf{B} = 0$, $\mathbf{C} = \frac{1}{2}\mathbf{M}$, $\mathbf{d} = \mathbf{e} = \mathbf{0}$, and $f = 0$. Similarly, the Mahalanobis distance $D_\mathbf{M}(\mathbf{x},\mathbf{y}) = (\mathbf{x}-\mathbf{y})^T\mathbf{M}(\mathbf{x}-\mathbf{y})$ is also regarded as a special case of our model, when $\mathbf{A} = \mathbf{B} = \mathbf{M}$, $\mathbf{C} = -\mathbf{M}$, $\mathbf{d} = \mathbf{e} = \mathbf{0}$, and $f = 0$.

In the following, we connect our similarity model to two state-of-the-art similarity learning methods, i.e., LADF [3] and Joint Bayesian [4].

In [3], Li et al. proposed to learn a decision function that jointly models a distance metric and a locally adaptive thresholding rule, and the so-called LADF (i.e., Locally-Adaptive Decision Function) is formulated as a second-order large-margin regularization problem. Specifically, LADF is defined as:

$$F(\mathbf{x},\mathbf{y}) = \mathbf{x}^T\mathbf{A}\mathbf{x} + \mathbf{y}^T\mathbf{A}\mathbf{y} + 2\mathbf{x}^T\mathbf{C}\mathbf{y} + \mathbf{d}^T(\mathbf{x}+\mathbf{y}) + f. \tag{8}$$

One can observe that $F(\mathbf{x},\mathbf{y}) = S(\mathbf{x},\mathbf{y})$ when we set $\mathbf{B} = \mathbf{A}$ and $\mathbf{e} = \mathbf{d}$ in our model.

It should be noted that LADF treats $\mathbf{x}$ and $\mathbf{y}$ using the same metrics, i.e., $\mathbf{A}$ for both $\mathbf{x}^T\mathbf{A}\mathbf{x}$ and $\mathbf{y}^T\mathbf{A}\mathbf{y}$, and $\mathbf{d}$ for $\mathbf{d}^T\mathbf{x}$ and $\mathbf{d}^T\mathbf{y}$. Such a model is reasonable for matching samples with the same modality, but may be unsuitable for cross-domain matching where $\mathbf{x}$ and $\mathbf{y}$ are with different modalities. Compared with LADF, our model uses $\mathbf{A}$ and $\mathbf{d}$ to calculate $\mathbf{x}^T\mathbf{A}\mathbf{x}$ and $\mathbf{d}^T\mathbf{x}$, and uses $\mathbf{B}$ and $\mathbf{e}$ to calculate $\mathbf{y}^T\mathbf{B}\mathbf{y}$ and $\mathbf{e}^T\mathbf{y}$, making our model more effective for cross-domain matching.

In [4], Chen et al. extended the classical Bayesian face model by learning a joint distributions (i.e., intra-person and extra-person variations) of sample pairs. Their decision function is posed as the following form:

$$J(\mathbf{x},\mathbf{y}) = \mathbf{x}^T\mathbf{A}\mathbf{x} + \mathbf{y}^T\mathbf{A}\mathbf{y} - 2\mathbf{x}^T\mathbf{G}\mathbf{y}. \tag{9}$$

Note that the similarity metric model proposed in [14] also adopted such a form. Interestingly, this decision function is also a special variant of our model by setting $\mathbf{B} = \mathbf{A}$, $\mathbf{C} = -\mathbf{G}$, $\mathbf{d} = \mathbf{0}$, $\mathbf{e} = \mathbf{0}$, and $\mathbf{f} = 0$.

In summary, our similarity model can be regarded as the generalization of many existing cross-domain matching and metric learning models, and it is more flexible and suitable for cross-domain visual data matching.

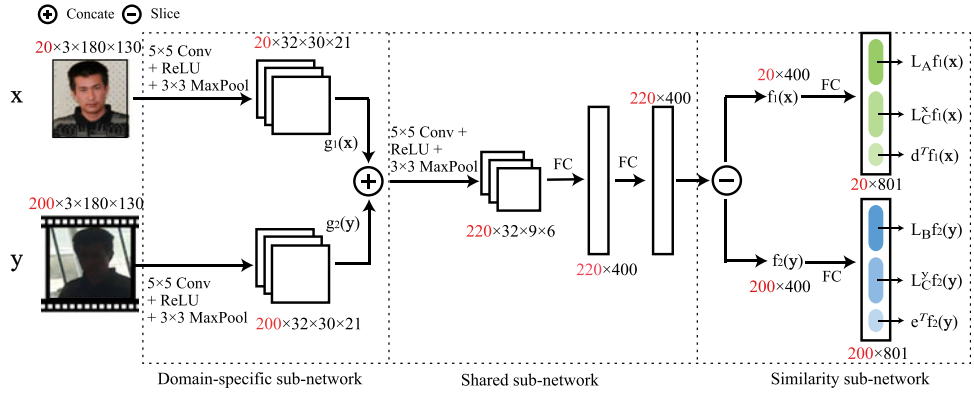


Fig. 3. Deep architecture of our similarity model. This architecture is comprised of three parts: domain-specific sub-network, shared sub-network and similarity sub-network. The first two parts extract feature representations from samples of different domains, which are built upon a number of convolutional layers, max-pooling operations and fully-connected layers. The similarity sub-network includes two structured fully-connected layers that incorporate the similarity components in Eqn. (3).

4 Joint Similarity and Feature Learning

In this section, we introduce our deep architecture that integrates the generalized similarity measure with convolutional feature representation learning.

4.1 Deep Architecture

As discussed above, our model defined in Eqn. (7) jointly handles similarity function learning and feature learning. This integration is achieved by building a deep architecture of convolutional neural networks, which is illustrated in Fig. 3. It is worth mentioning that our architecture is able to handle the input samples of different modalities with unequal numbers, e.g., 20 samples of $\mathbf{x}$ and 200 samples of $\mathbf{y}$ are fed into the network in a way of batch processing.

From left to right in Fig. 3, two domain-specific sub-networks $\mathbf{g}_1(\mathbf{x})$ and $\mathbf{g}_2(\mathbf{y})$ are applied to the samples of two different modalities, respectively. Then, the outputs of $\mathbf{g}_1(\mathbf{x})$ and $\mathbf{g}_2(\mathbf{y})$ are concatenated into a shared sub-network $\mathbf{f}(\cdot)$. We make a superposition of $\mathbf{g}_1(\mathbf{x})$ and $\mathbf{g}_2(\mathbf{y})$ to feed $\mathbf{f}(\cdot)$. At the output of $\mathbf{f}(\cdot)$, the feature representations of the two samples are extracted separately as $\mathbf{f}_1(\mathbf{x})$ and $\mathbf{f}_2(\mathbf{y})$, which is indicated by the slice operator in Fig. 3. Finally, these learned feature representations are utilized in the structured fully-connected layers that incorporate the similarity components defined in Eqn. (3). In the following, we introduce the detailed setting of the three sub-networks.

Domain-specific sub-network. We separate two branches of neural networks to handle the samples from different domains. Each network branch includes one convolutional layer with 3 filters of size 5×5 and the stride step of 2 pixels. The rectified nonlinear activation is utilized. Then, we follow by a one max-pooling operation with size of 3×3 and its stride step is set as 3 pixels.

Shared sub-network. For this component, we stack one convolutional layer and two fully-connected layers. The convolutional layer contains 32 filters of size 5×5 and the filter stride step is set as 1 pixel. The kernel size of the max-pooling operation is 3×3 and its stride step is 3 pixels. The output vectors of the two fully-connected layers are of 400 dimensions. We further normalize the output of the second fully-connected layer before it is fed to the next sub-network.

Similarity sub-network. A slice operator is first applied in this sub-network, which partitions the vectors into two groups corresponding to the two domains. For the example in Fig. 3, 220 vectors are grouped into two sets, i.e., $\mathbf{f}_1(\mathbf{x})$ and $\mathbf{f}_2(\mathbf{y})$, with size of 20 and 200, respectively. $\mathbf{f}_1(\mathbf{x})$ and $\mathbf{f}_2(\mathbf{y})$ are both of 400 dimensions. Then, $\mathbf{f}_1(\mathbf{x})$ and $\mathbf{f}_2(\mathbf{y})$ are fed to two branches of neural network, and each branch includes a fully-connected layer. We divide the activations of these two layers into six parts according to the six similarity components. As is shown in Fig. 3, in the top branch the neural layer connects to $\mathbf{f}_1(\mathbf{x})$ and outputs $\mathbf{L_A f_1}(\mathbf{x})$, $\mathbf{L_C^x f_1}(\mathbf{x})$, and $\mathbf{d}^T\mathbf{f_1}(\mathbf{x})$, respectively. In the bottom branch, the layer outputs $\mathbf{L_B f_2}(\mathbf{y})$, $\mathbf{L_C^y f_2}(\mathbf{y})$, and $\mathbf{e}^T\mathbf{f_2}(\mathbf{y})$, respectively, by connecting to $\mathbf{f}_2(\mathbf{y})$. In this way, the similarity measure is tightly integrated with the feature representations, and they can be jointly optimized during the model training. Note that f is a parameter of the generalized similarity measure in Eqn. (1). Experiments show that the value of f only affects the learning convergence rather than the matching performance. Thus we empirically set $f = -1.9$ in our experiments.

In the deep architecture, we can observe that the similarity components of $\mathbf{x}$ and those of $\mathbf{y}$ do not interact to each other by the factorization until the final aggregation calculation, that is, computing the components of $\mathbf{x}$ is independent of $\mathbf{y}$. This leads to a good property of efficient matching. In particular, for each sample stored in a database, we can precomputed its feature representation and the corresponding similarity components, and the similarity matching in the testing stage will be very fast.

4.2 Model Training

In this section, we discuss the learning method for our similarity model training. To avoid loading all images into memory, we use the mini-batch learning approach, that is, in each training iteration, a subset of the image pairs are fed into the neural network for model optimization.

For notation simplicity in discussing the learning algorithm, we start by introducing the following definitions:

$$\begin{aligned} \tilde{\mathbf{x}} &\triangleq [\,\mathbf{L_A f_1}(\mathbf{x}) \quad \mathbf{L_C^x f_1}(\mathbf{x}) \quad \mathbf{d}^T\mathbf{f_1}(\mathbf{x})\,]^T \\ \tilde{\mathbf{y}} &\triangleq [\,\mathbf{L_B f_2}(\mathbf{y}) \quad \mathbf{L_C^y f_2}(\mathbf{y}) \quad \mathbf{e}^T\mathbf{f_2}(\mathbf{y})\,]^T, \end{aligned} \tag{10}$$

where $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$ denote the output layer's activations of the samples $\mathbf{x}$ and $\mathbf{y}$. Prior to incorporating Eqn. (10) into the similarity model in Eqn. (3), we introduce three transformation matrices (using Matlab representation):

$$\begin{aligned}\mathbf{P}_1 &= \left[\mathbf{I}^{r\times r} \quad \mathbf{0}^{r\times(r+1)}\right],\\ \mathbf{P}_2 &= \left[\mathbf{0}^{r\times r} \quad \mathbf{I}^{r\times r} \quad \mathbf{0}^{r\times 1}\right],\\ \mathbf{p}_3 &= \left[\mathbf{0}^{1\times 2r} \quad 1^{1\times 1}\right]^T,\end{aligned} \tag{11}$$

where r equals to the dimension of the output of shared neural network (i.e., the dimension of $f(\mathbf{x})$ and $f(\mathbf{y})$), an $\mathbf{I}$ indicates the identity matrix. Then, our similarity model can be re-written as:

$$\begin{aligned}\tilde{S}(\mathbf{x},\mathbf{y}) = (\mathbf{P}_1\tilde{\mathbf{x}})^T\mathbf{P}_1\tilde{\mathbf{x}} + (\mathbf{P}_1\tilde{\mathbf{y}})^T\mathbf{P}_1\tilde{\mathbf{y}} - 2(\mathbf{P}_2\tilde{\mathbf{x}})^T\mathbf{P}_2\tilde{\mathbf{y}}\\ + 2\mathbf{p}_3^T\tilde{\mathbf{x}} + 2\mathbf{p}_3^T\tilde{\mathbf{y}} + f.\end{aligned} \tag{12}$$

Incorporating Eqn. (12) into the loss function Eqn. (6), we have the following objective:

$$\begin{aligned}&G(\mathbf{W},\Phi;\mathcal{D})\\ &= \sum_{i=1}^{N}\{1-\ell_i[(\mathbf{P}_1\widetilde{\mathbf{x}}_i)^T\mathbf{P}_1\widetilde{\mathbf{x}}_i + (\mathbf{P}_1\widetilde{\mathbf{y}}_i)^T\mathbf{P}_1\widetilde{\mathbf{y}}_i - ,\\ &\quad 2(\mathbf{P}_2\widetilde{\mathbf{x}}_i)^T\mathbf{P}_2\widetilde{\mathbf{y}}_i + 2\mathbf{p}_3^T\widetilde{\mathbf{x}}_i + 2\mathbf{p}_3^T\widetilde{\mathbf{y}}_i + f]\}_+\end{aligned} \tag{13}$$

where the summation term denotes the hinge-like loss for the cross domain sample pair $\{\tilde{\mathbf{x}}_i, \tilde{\mathbf{y}}_i\}$, N is the total number of pairs, $\mathbf{W}$ represents the feature representation of different domains and Φ represents the similarity model. $\mathbf{W}$ and Φ are both embedded as weights connecting neurons of layers in our deep neural network model, as Fig. 3 illustrates.

The objective function in Eqn. (13) is defined in sample-pair-based form. To optimize it using SGD, one should apply a certain scheme to generate mini-batches of the sample pairs, which usually costs much computation and memory. Note that the sample pairs in training set $\mathcal{D}$ are constructed from the original set of samples from different modalities $\mathcal{Z} = \{\{\mathcal{X}\},\{\mathcal{Y}\}\}$, where $\mathcal{X} = \{\mathbf{x}^1,\ldots,\mathbf{x}^j,\ldots,\mathbf{x}^{M_\mathbf{x}}\}$ and $\mathcal{Y} = \{\mathbf{y}^1,\ldots,\mathbf{y}^j,\ldots,\mathbf{y}^{M_\mathbf{y}}\}$. The superscript denotes the sample index in the original training set, e.g., $\mathbf{x}^j \in \mathcal{X} = \{\mathbf{x}^1,\ldots,\mathbf{x}^j,\ldots,\mathbf{x}^{M_\mathbf{x}}\}$ and $\mathbf{y}^j \in \mathcal{Y} = \{\mathbf{y}^1,\ldots,\mathbf{y}^j,\ldots,\mathbf{y}^{M_\mathbf{y}}\}$, while the subscript denotes the index of sample pairs, e.g., $\mathbf{x}_i \in \{\mathbf{x}_i,\mathbf{y}_i\} \in \mathcal{D}$. $M_\mathbf{x}$ and $M_\mathbf{y}$ denote the total number of samples from different domains. Without loss of generality, we define $\mathbf{z}^j = \mathbf{x}^j$ and $\mathbf{z}^{M_\mathbf{x}+j} = \mathbf{y}^j$. For each pair $\{\mathbf{x}_i,\mathbf{y}_i\}$ in $\mathcal{D}$, we have $\mathbf{z}^{j_{i,1}} = \mathbf{x}_i$ and $\mathbf{z}^{j_{i,2}} = \mathbf{y}_i$ with $1 \le j_{i,1} \le M_\mathbf{x}$ and $M_\mathbf{x}+1 \le j_{i,2} \le M_\mathbf{z}(= M_\mathbf{x}+M_\mathbf{y})$. And we also have $\widetilde{\mathbf{z}}^{j_{i,1}} = \widetilde{\mathbf{x}}_i$ and $\widetilde{\mathbf{z}}^{j_{i,2}} = \widetilde{\mathbf{y}}_i$.

Therefore, we rewrite Eqn. (13) in a sample-based form:

$$\begin{aligned}&L(\mathbf{W},\Phi;\mathcal{Z})\\ &= \sum_{i=1}^{N}\{1-\ell_i[(\mathbf{P}_1\widetilde{\mathbf{z}}^{j_{i,1}})^T\mathbf{P}_1\widetilde{\mathbf{z}}^{j_{i,1}} + (\mathbf{P}_1\widetilde{\mathbf{z}}^{j_{i,2}})^T\mathbf{P}_1\widetilde{\mathbf{z}}^{j_{i,2}} - ,\\ &\quad 2(\mathbf{P}_2\widetilde{\mathbf{z}}^{j_{i,1}})^T\mathbf{P}_2\widetilde{\mathbf{z}}^{j_{i,2}} + 2\mathbf{p}_3^T\widetilde{\mathbf{z}}^{j_{i,1}} + 2\mathbf{p}_3^T\widetilde{\mathbf{z}}^{j_{i,2}} + f]\}_+\end{aligned} \tag{14}$$

Given $\Omega = (\mathbf{W},\Phi)$, the loss function in Eqn. (7) can also be rewritten in the sample-based form:

$$H(\Omega) = L(\Omega;\mathcal{Z}) + \Psi(\Omega). \tag{15}$$

The objective in Eqn. (15) can be optimized by the mini-batch back propagation algorithm. Specifically, we update the parameters by gradient descent:

$$\Omega = \Omega - \alpha\frac{\partial}{\partial\Omega}H(\Omega), \tag{16}$$

where α denotes the learning rate. The key problem of solving the above equation is calculating $\frac{\partial}{\partial\Omega}L(\Omega)$. As is discussed in [31], there are two ways to this end, i.e., pair-based gradient descent and sample-based gradient descent. Here we adopt the latter to reduce the requirements on computation and memory cost.

Suppose a mini-batch of training samples $\{\mathbf{z}^{j_{1,\mathbf{x}}},\ldots,\mathbf{z}^{j_{n_\mathbf{x},\mathbf{x}}},\mathbf{z}^{j_{1,\mathbf{y}}},\ldots,\mathbf{z}^{j_{n_\mathbf{y},\mathbf{y}}}\}$ from the original set $\mathcal{Z}$, where $1 \le j_{i,\mathbf{x}} \le M_\mathbf{x}$ and $M_\mathbf{x}+1 \le j_{i,\mathbf{y}} \le M_\mathbf{z}$. Following the chain rule, calculating the gradient for all pairs of samples is equivalent to summing up the gradient for each sample,

$$\frac{\partial}{\partial\Omega}L(\Omega) = \sum_j \frac{\partial L}{\partial\tilde{\mathbf{z}}^j}\frac{\partial\tilde{\mathbf{z}}^j}{\partial\Omega}, \tag{17}$$

where j can be either $j_{i,\mathbf{x}}$ or $j_{i,\mathbf{y}}$.

Using $\mathbf{z}^{j_{i,\mathbf{x}}}$ as an example, we first introduce an indicator function $\mathbf{1}_{\mathbf{z}^{j_{i,\mathbf{x}}}}(\mathbf{z}^{j_{i,\mathbf{y}}})$ before calculating the partial derivative of output layer activation for each sample $\frac{\partial L}{\partial\tilde{\mathbf{z}}^{j_{i,\mathbf{x}}}}$. Specifically, we define $\mathbf{1}_{\mathbf{z}^{j_{i,\mathbf{x}}}}(\mathbf{z}^{j_{i,\mathbf{y}}}) = 1$ when $\{\mathbf{z}^{j_{i,\mathbf{x}}},\mathbf{z}^{j_{i,\mathbf{y}}}\}$ is a sample pair and $\ell_{j_{i,\mathbf{x}},j_{i,\mathbf{y}}}\tilde{S}(\mathbf{z}^{j_{i,\mathbf{x}}},\mathbf{z}^{j_{i,\mathbf{y}}}) < 1$. Otherwise we let $\mathbf{1}_{\mathbf{z}^{j_{i,\mathbf{x}}}}(\mathbf{z}^{j_{i,\mathbf{y}}}) = 0$. $\ell_{j_{i,\mathbf{x}},j_{i,\mathbf{y}}}$, indicating where $\mathbf{z}^{j_{i,\mathbf{x}}}$ and $\mathbf{z}^{j_{i,\mathbf{y}}}$ are from the same class. With $\mathbf{1}_{\mathbf{z}^{j_{i,\mathbf{x}}}}(\mathbf{z}^{j_{i,\mathbf{y}}})$, the gradient of $\mathbf{z}^{\tilde{j}_{i,\mathbf{x}}}$ can be written as

$$\frac{\partial L}{\partial\tilde{\mathbf{z}}^{j_{i,\mathbf{x}}}} = -\sum_{j_{i,\mathbf{y}}} 2\mathbf{1}_{\mathbf{z}^{j_{i,\mathbf{x}}}}(\mathbf{z}^{j_{i,\mathbf{y}}})\ell_{j_{i,\mathbf{x}},j_{i,\mathbf{y}}}(\mathbf{P}_1^T\mathbf{P}_1\tilde{\mathbf{z}}^{j_{i,\mathbf{x}}} - \mathbf{P}_2^T\mathbf{P}_2\tilde{\mathbf{z}}^{j_{i,\mathbf{y}}} + \mathbf{p}_3). \tag{18}$$

The calculation of $\frac{\partial L}{\partial\tilde{\mathbf{z}}^{j_{i,\mathbf{y}}}}$ can be conducted in a similar way. The algorithm of calculating the partial derivative of output layer activation for each sample is shown in Algorithm 1.

Note that all the three sub-networks in our deep architecture are differentiable. We can easily use the back-propagation procedure [19] to compute the partial derivatives with respect to the hidden layers and model parameters Ω. We summarize the overall procedure of deep generalized similarity measure learning in Algorithm 2.

If all the possible pairs are used in training, the sample-based form allows us to generate $n_\mathbf{x} \times n_\mathbf{y}$ sample pairs from a mini-batch of $n_\mathbf{x}+n_\mathbf{y}$. On the other hand, the sample-pair-based form may require $2n_\mathbf{x}n_\mathbf{y}$ samples or less to generate $n_\mathbf{x} \times n_\mathbf{y}$ sample pairs. In gradient computation, from Eqn. (18), for each sample we only require calculating $\mathbf{P}_1^T\mathbf{P}_1\tilde{\mathbf{z}}^{j_{i,\mathbf{x}}}$ once and $\mathbf{P}_2^T\mathbf{P}_2\tilde{\mathbf{z}}^{j_{i,\mathbf{y}}}$ $n_\mathbf{y}$ times in the sample-based form. While in the sample-pair-based form, $\mathbf{P}_1^T\mathbf{P}_1\tilde{\mathbf{z}}^{j_{i,\mathbf{x}}}$ and $\mathbf{P}_2^T\mathbf{P}_2\tilde{\mathbf{z}}^{j_{i,\mathbf{y}}}$ should be computed $n_\mathbf{x}$ and $n_\mathbf{y}$ times, respectively. In sum, the sample-based form generally results in less computation and memory cost.

Algorithm 1. Calculate the Derivative of the Output Layer's Activation for Each Sample

Input:
The output layer's activation for all samples
Output:
The partial derivatives of output layer's activation for all the samples

for each sample $\mathbf{z}^j$ **do**
  Initialize the partner set $\mathcal{M}^j$ containing the sample $\mathbf{z}^j$ with $\mathcal{M}^j = \emptyset$;
  for each pair $\{\mathbf{x}_i, \mathbf{y}_i\}$ **do**
    if pair $\{\mathbf{x}_i, \mathbf{y}_i\}$ contains the sample $\mathbf{z}^j$ **then**
      if pair $\{\mathbf{x}_i, \mathbf{y}_i\}$ satisfies $\ell_i \tilde{S}(\mathbf{x}_i, \mathbf{y}_i) < 1$ **then**
        $\mathcal{M}^i \leftarrow \{\mathcal{M}^i$, the corresponding partner of $\mathbf{z}^j$ in $\{\mathbf{x}_i, \mathbf{y}_i\}\}$;
      end if
    end if
  end for
  Compute the derivatives for the sample $\mathbf{z}^j$ with all the partners in $\mathcal{M}^j$, and sum these derivatives to be the desired partial derivative for sample $\mathbf{z}^j$'s output layer's activation using Eqn. (18);
end for

Algorithm 2. Generalized Similarity Learning

Input:
Training set, initialized parameters $\mathbf{W}$ and $\mathbf{\Phi}$, learning rate α, $t \leftarrow 0$
Output:
Network parameters $\mathbf{W}$ and $\mathbf{\Phi}$

while $t <= T$ **do**
  Sample training pairs $\mathcal{D}$;
  Feed the sampled images into the network;
  Perform a feed-forward pass for all the samples and compute the net activations for each sample $\mathbf{z}^i$;
  Compute the partial derivative of the output layer's activation for each sample by Algorithm 1.
  Compute the partial derivatives of the hidden layers' activations for each sample following the chain rule;
  Compute the desired gradients $\frac{\partial}{\partial \Omega} H(\Omega)$ using the back-propagation procedure;
  Update the parameters using Eqn. (16);
end while

Batch process implementation. Suppose that the training image set is divided into K categories, each of which contains O_1 images from the first domain and O_2 images from the second domain. Thus we can obtain a maximum number $(K \times O_1) \times (K \times O_2)$ of pairwise samples, which is quadratically more than the number of source images $K \times (O_1 + O_2)$. In real application, since the number of stored images may reach millions, it is impossible to load all the data for network training. To overcome this problem, we implement our learning algorithm in a batch-process manner. Specifically, in each iteration, only a small subset of cross domain image pairs are generated and fed to the network for training. According to our massive experiments, randomly generating image pairs is infeasible, which may cause the image distribution over the special batch becoming scattered, making valid training samples for a certain category very few and degenerating the model. Besides, images in any pair are almost impossible to come from the same class, making the positive samples very few. In order to overcome this problem, an effective cross domain image pair generation scheme is adopted to train our generalized similarity model. For each round, we first randomly choose $\widehat{K}$ instance categories. For each category, a number of $\widehat{O_1}$ images first domain and a number of $\widehat{O_2}$ from second domain are randomly selected. For each selected images in first domain, we randomly take samples from the second domain and the proportions of positive and negative samples are equal. In this way, images distributed over the generated samples are relatively centralized and the model will effectively converge.

5 Experiments

In this section, we apply our similarity model in four representative tasks of matching cross-domain visual data and adopt several benchmark datasets for evaluation: i) person re-identification under different views on CUHK03 [28] and CUHK01 [32] datasets; ii) age-invariant face recognition on MORPH [33], CACD [34] and CACD-VS [35] datasets; iii) sketch-to-photo face matching on CUFS dataset [36]; iv) face verification over still-video domains on COX face dataset [37]. On all these tasks, state-of-the-art methods are employed to compare with our model.

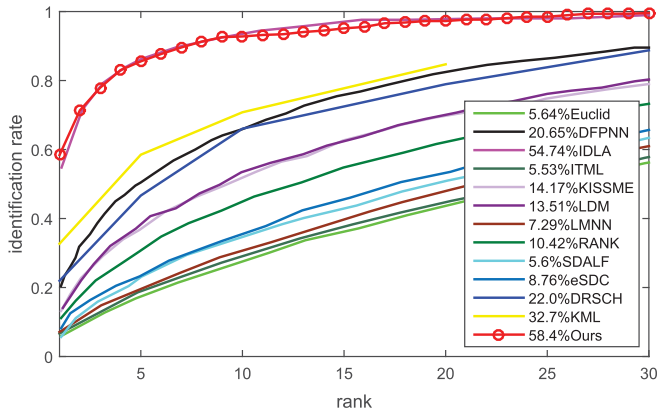
Experimental setting. Mini-batch learning is adopted in our experiments to save memory cost. In each task, we randomly select a batch of sample from the original training set to generate a number of pairs (e.g., $4{,}800$). The initial parameters of the convolutional and the full connection layers are set by two zero-mean Gaussian Distributions, whose standard deviations are 0.01 and 0.001 respectively. Other specific settings to different tasks are included in the following sections.

In addition, ablation studies are presented to reveal the benefit of each main component of our method, e.g., the generalized similarity measure and the joint optimization of CNN feature representation and metric model. We also implement several variants of our method by simplifying the similarity measures for comparison.

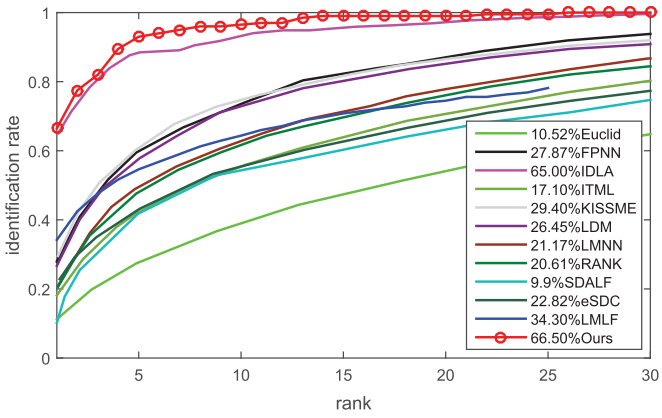
5.1 Person Re-Identification

Person re-identification, aiming at matching pedestrian images across multiple non-overlapped cameras, has attracted increasing attentions in surveillance. Despite that considerable efforts have been made, it is still an open problem due to the dramatic variations caused by viewpoint and pose changes. To evaluate this task, CUHK03 [28] dataset and CUHK01 [32] dataset are adopted in our experiments.

CUHK03 dataset [28] is one of the largest databases for person re-identification. It contains 14,096 images of 1,467 pedestrians collected from five different pairs of camera views. Each person is observed by two disjoint camera views and has an average of 4.8 images in each view. We follow the standard setting of using CUHK03 to randomly partition this dataset for 10 times, and a training set (including 1,367 persons) and a testing set (including 100 persons) are obtained without overlap.



(a) CUHK03



(b) CUHK01

Fig. 4. CMC curves on (a) CUHK03 [28] dataset and (b) CUHK01 [32] for evaluating person re-identification. Our method has superior performances over existing state-of-the-arts overall.

CUHK01 dataset [32] contains 971 individuals, each having two samples from disjoint cameras. Following the setting in [25], [28], we partition this dataset into a training set and a testing set: 100 individuals for testing and the others for training.

For evaluation on these two benchmarks, the testing set is further randomly divided into a gallery set of 100 images (i.e., one image per person) and a probe set (including images of individuals from different camera views in contrast to the gallery set) without overlap for 10 times. We use Cumulative Matching Characteristic (CMC) [38] as the evaluation metric in this task.

In our model training, all of the images are resized to 250×100, and cropped to the size of 230×80 at the center with a small random perturbation. During every round of learning, 4,800 pairs of samples are constructed by selecting 60 persons (or classes) and constructing 80 pairs for each person (class). For CUHK01, due to each individual only have two samples, the 80 pairs per individual will contain some duplicated pairs.

Results on CUHK03. We compare our approach with several state-of-the-art methods, which can be grouped into three categories. First, we adopt five distance metric learning methods based on fixed feature representation, i.e., the Information Theoretic Metric Learning (ITML) [5], the Local Distance Metric Learning (LDM) [39], the Large Margin Nearest Neighbors (LMNN) [40], the learning-to-rank method (RANK) [41], and the Kernel-based Metric Learning method (KML) [23]. Following their implementation, the handcrafted features of dense color histograms and dense SIFT uniformly sampled from patches are adopted. Second, three methods specially designed for person re-identification are employed in the experiments: SDALF [42], KISSME [43], and eSDC [44]. Moreover, several recently proposed deep learning methods, including DRSCH [45], DFPNN [28] and IDLA [25], are also compared with our approach. DRSCH [45] is a supervised hashing framework for integrating CNN feature and hash code learning, while DFPNN and IDLA have been introduced in Section 2.

The results are reported in Fig. 4a. It is encouraging to see that our approach significantly outperforms the competing methods (e.g., improving the state-of-the-art rank-1 accuracy from 54.74 percent (IDLA [25]) to 58.39 percent). Among the competing methods, ITML [5], LDM [39], LMNN [40], RANK [41], KML [23], SDALF [42], KISSME [43], and eSDC [44] are all based on hand-crafted features. And the superiority of our approach against them should be attributed to the deployment of both deep CNN features and generalized similarity model. DRSCH [45], DFPNN [28] and IDLA [25] adopted CNN for feature representation, but their matching metrics are defined based on traditional linear transformations.

Results on CUHK01. Fig. 4b shows the results of our method and the other competing approaches on CUHK01. In addition to those used on CUHK03, one more method, i.e., LMLF [24], is used in the comparison experiment. LMLF [24] learns mid-level filters from automatically discovered patch clusters. According to the quantitative results, our method achieves a new state-of-the-art with a rank-1 accuracy of 66.50 percent.

5.2 Age-Invariant Face Recognition

Age invariant face recognition is to decide whether two images with different ages belong to the same identity. The key challenge is to handle the large intra-subject variations caused by aging process while distinguishing different identities. Other factors, such as illumination, pose, and expression, make age invariant face recognition more difficult. We conduct the experiments using three datasets, i.e., MORPH [33], CACD [34], and CACD-VS [35].

MORPH [33] contains more than 55,000 face images of 13,000 individuals, whose ages range from 16 to 77. The average number of images per individual is 4. The training set consists of 20,000 face images from 10,000 subjects, with each subject having two images with the largest age gap. The test set is composed of a gallery set and a probe set from the remaining 3,000 subjects. The gallery set is composed of the youngest face images of each subject. The probe set is composed of the oldest face images of each subject. This experimental setting is the same with those adopted in [46] and [34].

CACD [34] is a large scale dataset released in 2014, which contains more than 160,000 images of 2,000 celebrities. We adopt a subset of 580 individuals from the whole database in our experiment, in which we manually remove the noisy images. Among these 580 individuals, the labels of images from 200 individuals have been originally provided, and we annotate the rest of the data. CACD includes large variations

TABLE 1
Experimental Results for Age-Invariant Face Recognition

(a) Recognition rates on the MORPH dataset.	
Method	Recognition rate
TDBN [48]	60%
3D Aging Model [50]	79.8%
MFDA [49]	83.9%
HFA [46]	91.1%
CARC [34]	92.8%
Ours	**94.4%**

(b) Verification accuracy on the CACD-VS dataset.	
Method	verification accuracy
HD-LBP [51]	81.6%
HFA [46]	84.4%
CARC [34]	87.6%
Deepface [52]	85.4%
Ours	**89.8%**

not only in pose, illumination, expression but also in ages. Based on CACD, a verification subset called CACD-VS [35] is further developed, which contains 2,000 positive pairs and 2,000 negative pairs. The setting and testing protocol of CACD-VS are similar to the well-known LFW benchmark [47], except that CACD-VS contains much more samples for each person.

All of the images are resized to 200×150. For data augmentation, images are cropped to the size of 180×130 at the center with a small random perturbation when feeding to the neural network. Sample-based mini-batch setting is adopted, and 4,800 pairs are constructed for each iteration.

Results on MORPH. We compare our method with several state-of-the-art methods, including topological dynamic Bayesian network (TDBN) [48], cross-age reference coding (CARC) [34], probabilistic hidden factor analysis (HFA) [46], multi-feature discriminant analysis (MFDA) [49] and 3D aging model [50]. The results are reported in Table 1(a). Thanks to the use of CNN representation and generalized similarity measure, our method achieves the recognition rate of 94.35 percent, and significantly outperforms the competing methods.

Results on CACD. On this dataset, the protocol is to retrieve face images of the same individual from gallery sets by using a probe set, where the age gap between probe face images and gallery face images is large. Following the experimental setting in [34], we set up 4 gallery sets according to the years when the photos were taken: $[2004-2006]$, $[2007-2009]$, $[2010-2012]$, and $[2013]$. And we use the set of $[2013]$ as the probe set to search for matches in the rest of three sets. We introduce several state-of-the-art methods for comparison, including CARC [34], HFA [46] and one deep learning based method, Deepface [52]. The results of CARC [34] and HFA [46] are borrowed from their papers. The results of Deepface [52] and our approach (i.e., Ours-1) are implemented based on the 200 originally annotated individuals, where 160 samples are used for model training. From the quantitative results reported in Fig. 5, our model achieves superior performances over the competing methods. Furthermore, we also report the result of our method (i.e., Ours-2) by using images of 500 individuals as training

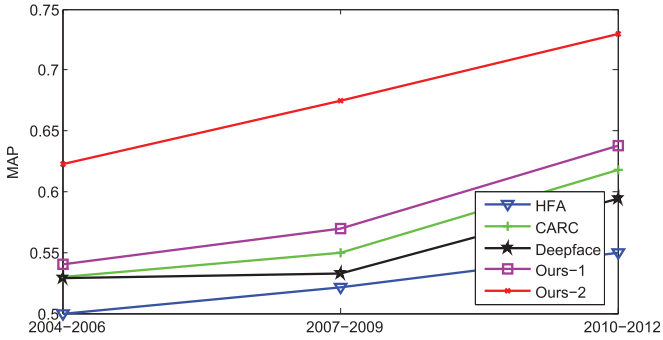


Fig. 5. The retrieval performances on CACD dataset for age-invariant face recognition. Ours-1 and Ours-2 are our method, while the latter uses more training samples.

samples. One can see that, the performance of our model can be further improved by increasing training data.

Results on CACD-VS. Following the setting in [35], we further evaluate our approach by conducting the general face verification experiment. Specifically, for all of the competing methods, we train the models on CACD and test on CACD-VS, and the optimal threshold value for matching is obtained by exhaustive search. The results produced by our methods and the others (i.e., CARC [34], HFA [46], HD-LBP [51] and Deepface [52]) are reported in Table 1(b). It is worth mentioning that our method improves the state-of-the-art recognition rate from 87.6 percent (by CARC [34], [52]) to 89.8 percent. Thanks to the introduction of generalized similarity measure our approach achieves higher verification accuracy than Deepface. Note that an explicit face alignment was adopted in [52] before the CNN feature extraction, which is not in our framework.

5.3 Sketch-Photo Face Verification

Sketch-photo face verification is an interesting yet challenging task, which aims to verify whether a face photo and a drawing face sketch belong to the same individual. This task has an important application of assisting law enforcement, i.e., using face sketch to find candidate face photos. It is however difficult to match photos and sketches in two different modalities. For example, hand-drawing may bring unpredictable face distortion and variation compared to the real photo, and face sketches often lack of details that can be important cues for preserving identity.

We evaluate our model on this task using the CUFS dataset [36]. There are 188 face photos in this dataset, in which 88 are selected for training and 100 for testing. Each face has a corresponding sketch that is drawn by the artist. All of these face photos are taken at frontal view with a normal lighting condition and neutral expression.

All of the photos/sketches are resized to 250×200, and cropped to the size of 230×180 at the center with a small random perturbation. 1,200 pairs of photos and sketches (i.e., including 30 individuals with each having 40 pairs) are constructed for each iteration during the model training. In the testing stage, we use face photos to form the gallery set and treat sketches as the probes.

We employ several existing approaches for comparison: the eigenface transformation based method (ET) [53], the multi-scale Markov random field based method (MRF) [36], and MRF+ [54] (i.e., the lighting and pose robust version of

TABLE 2
Recognition Rates on the CUFS Dataset for Sketch-Photo Face Verification

Method	Recognition rate
ET [53]	71.0%
MRF [36]	96.0%
MRF+ [54]	99.0%
Ours	**100.0%**

[36]). It is worth mentioning that all of these competing methods need to first synthesize face sketches by photo-sketch transformation, and then measure the similarity between the synthesized sketches and the candidate sketches, while our approach works in an end-to-end way. The quantitative results are reported in Table 2. Our method achieves 100 percent recognition rate on this dataset.

5.4 Still-Video Face Recognition

Matching person faces across still images and videos is a newly rising task in intelligent visual surveillance. In these applications, the still images (e.g., ID photos) are usually captured under a controlled environment while the faces in surveillance videos are acquired under complex scenarios (e.g., various lighting conditions, occlusions and low resolutions).

For this task, a large-scale still-video face recognition dataset, namely COX face dataset, has been released recently,[3] which is an extension of the COX-S2V dataset [60]. This COX face dataset includes 1,000 subjects and each has one high quality still image and 3 video cliques respectively captured from three cameras. Since these cameras are deployed under similar environments (e.g., similar results are generated for the three cameras in [37], we use the data captured by the first camera in our experiments.

Following the setting of COX face dataset, we divide the data into a training set (300 subjects) and a testing set (700 subjects), and conduct the experiments with 10 random splits. There are two sub-tasks in the testing: i) matching video frames to still images (V2S) and ii) matching still images to video frames (S2V). For V2S task we use the video frames as probes and form the gallery set by the still images, and inversely for S2V task. The split of gallery/probe sets is also consistent with the protocol required by the creator. All of the image are resized to 200×150, and cropped to the size of 180×130 with a small random perturbation. 1,200 pairs of still images and video frames (i.e., including 20 individuals with each having 60 pairs) are constructed for each iteration during the model training.

Unlike the traditional image-based verification problems, both V2S and S2V are defined as the point-to-set matching problem, i.e., one still image to several video frames (i.e., 10 sampled frames). In the evaluation, we calculate the distance between the still image and each video frame by our model and output the average value over all of the distances. For comparison, we employ several existing point-to-set distance metrics: dual-space linear discriminant analysis (PSD) [55], manifold-manifold distance (PMD) [56], hyperplane-based distance (PAHD) [57], kernelized convex geometric distance (PCHD) [58], and covariance kernel based distance (PSDML) [59]. We also compare with the point-to-set correlation learning (PSCL-EA) method [37], which specially developed for the COX face dataset. The recognition rates of all competing methods are reported in Table 3, and our method achieves excellent performances, i.e., the best in S2V and the second best in V2S. The experiments show that our approach can generally improve performances in the applications to image-to-image, image-to-video, and video-to-image matching problems.

3. The COX face DB is collected by Institute of Computing Technology Chinese Academy of Sciences, OMRON Social Solutions Co. Ltd, and Xinjiang University.

TABLE 3
Recognition Rates on the COX Face Dataset

Method	V2S	S2V
PSD [55]	9.90%	11.64%
PMD [56]	6.40%	6.10%
PAHD [57]	4.70%	6.34%
PCHD [58]	7.93%	8.89%
PSDML [59]	12.14%	7.04%
PSCL-EA [37]	**30.33%**	28.39%
Ours	28.45%	**29.02%**

5.5 Ablation Studies

In order to provide more insights on the performance of our approach, we conduct a number of ablation studies by isolating each main component (e.g., the generalized similarity measure and feature learning). Besides, we also study the effect of using sample-pair-based and sample-based batch settings in term of convergence efficiency.

Generalized similarity model. We design two experiments by using handcrafted features and deep features, respectively, to justify the effectiveness of our generalized similarity measure.

(i) We test our similarity measure using the fixed handcrafted features for person re-identification. The experimental results on CUHK01 and CUHK03 clearly demonstrate the effectiveness of our model against the other similarity models without counting on deep feature learning. Following [44], we extract the feature representation by using patch-based color histograms and dense SIFT descriptors. This feature representation is fed into a full connection layer for dimensionality reduction to obtain a 400-dimensional vector. We then invoke the similarity sub-network (described in Section 4) to output the measure. On both CUHK01 and CUHK03, we adopt several representative similarity metrics for comparison, i.e., ITML [5], LDM [39], LMNN [40], and RANK [41], using the same feature representation.

The quantitative CMC curves and the recognition rates of all these competing models are shown in Figs. 6a and 6b for CUHK03 and CUHK01, respectively, where "Generalized" represents our similarity measure. It is observed that our model outperforms the others by large margins, e.g., achieving the rank-1 accuracy of 31.85 percent against 13.51 percent by LDM on CUHK03. Most of these competing methods learn Mahalanobis distance metrics. In contrast, our metric model combines Mahalanobis distance with Cosine similarity in a generic form, leading to a more general and effective solution in matching cross-domain data.

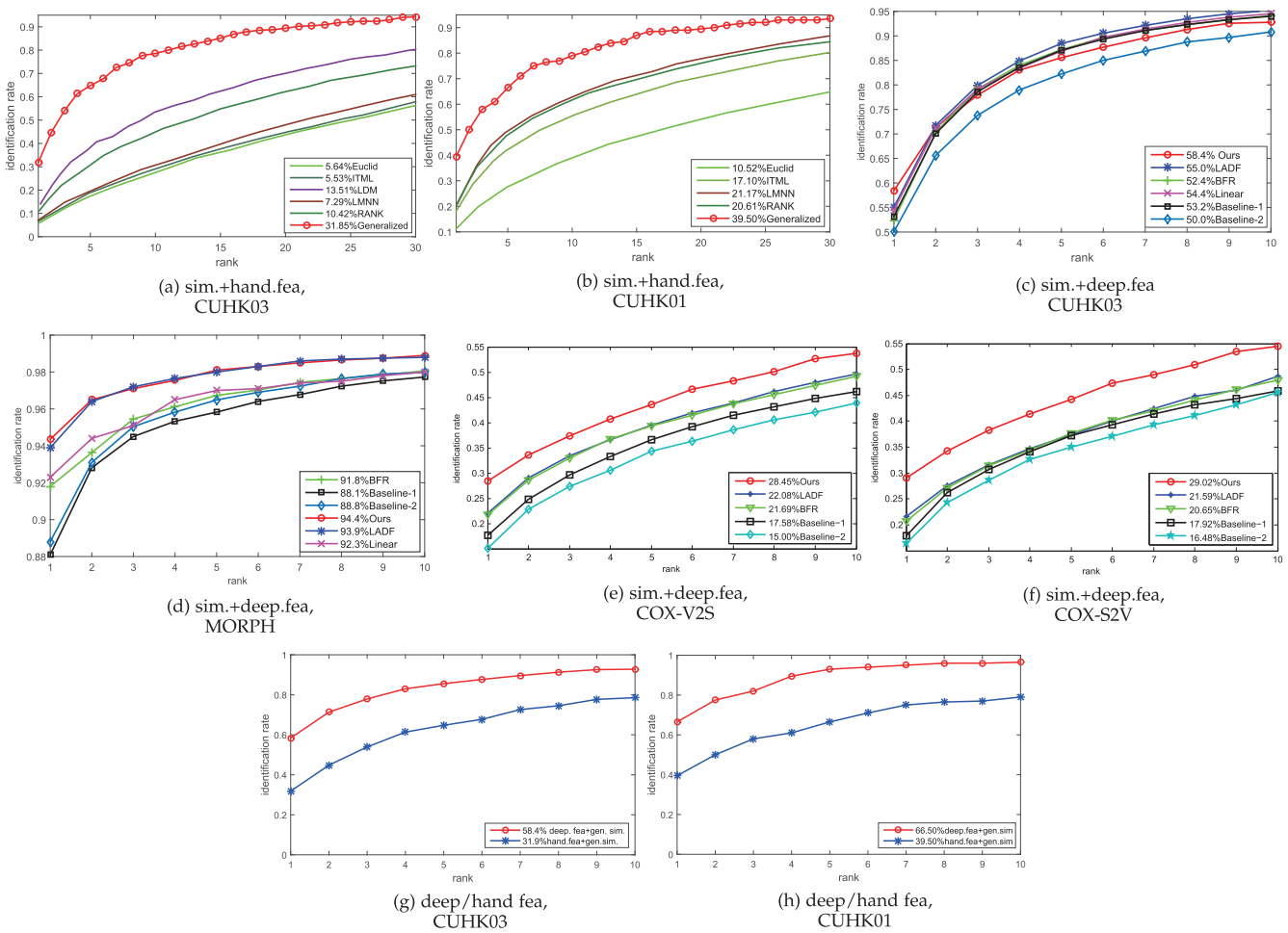


Fig. 6. Results of the ablation studies demonstrating the effectiveness of each main component of our framework. The CMC curve and recognition rate are used for evaluation. The results of different similarity models are shown using the handcrafted features (in (a) and (b)) and using the deep features (in (c)-(f)), respectively. (g) and (h) show the performances with/without the deep feature learning while keeping the same similarity model.

(ii) On the other hand, we incorporate several representative similarity measures into our deep architecture and jointly optimize these measures with the CNN feature learning. Specifically, we simplify our network architecture by removing the top layer (i.e., the similarity model), and measure the similarity in either the Euclidean embedding space (as Baseline-1) or in the inner-product space (as Baseline-2). These two variants can be viewed as two degenerations of our similarity measure (i.e., affine Euclidean distance and affine Cosine similarity). To support our discussions in Section 3.2, we adopt the two distance metric models LADF [3] and BFR (i.e., Joint Bayesian) [4] into our deep neural networks. Specifically, we replace our similarity model by the LADF model defined in Eqn. (8) and the BFR model defined in Eqn. (9), respectively. Moreover, we implement one more variant (denoted as "Linear" in this experiment), which applies similarity transformation parameters with separate linear transformations for each data modality. That is, we remove affine transformation while keeping separate linear transformation by setting $\mathbf{d} = \mathbf{0}$, $\mathbf{e} = \mathbf{0}$ and $f = 0$ in Eqn. (1). Note that the way of incorporating these metric models into the deep architecture is analogously to our metric model. The experiment is conducted on four benchmarks: CUHK03, MORPH, COX-V2S and COX-S2V, and the results are shown in Figs. 6c, 6d, 6e, 6f, respectively. Our method outperforms the competing methods by large margins on MORPH and COX face dataset. On CUHK03 (i.e., Fig. 6c), our method achieves the best rank-1 identification rate (i.e., $58.39\,\text{percent}$) among all the methods. In particular, the performance drops by $4\,\text{percent}$ when removing the affine transformation on CUHK03.

It is interesting to discover that most of these competing methods can be treated as special cases of our model. And our generalized similarity model can fully take advantage of convolutional feature learning by developing the specific deep architecture, and can consistently achieve superior performance over other variational models.

Deep feature learning. To show the benefit of deep feature learning, we adopt the handcrafted features (i.e., color histograms and SIFT descriptors) on CUHK01 and CHUK03 benchmark. Specifically, we extract this feature representation based on the patches of pedestrian images and then build the similarity measure for person re-identification. The results on CUHK03 and CHUK01 are reported in Figs. 6g and 6h, respectively. We denote the result by using the handcrafted features as "hand.fea + gen.sim" and the result by end-to-end deep feature learning as "deep.fea + gen.sim". It is obvious that without deep feature representation the performance drops significantly, e.g., from 58.4 to 31.85 percent on CUHK03 and from 66.5 to 39.5 percent on

CUHK01. These above results clearly demonstrate the effectiveness of utilizing deep CNNs for discriminative feature representation learning.

Sample-pair-based versus sample-based batch setting. In addition, we conduct an experiment to compare the sample-pair-based and sample-based in term of convergence efficiency, using the CUHK03 dataset. Specifically, for the sample-based batch setting, we select 600 images from 60 people and construct 60,000 pairs in each training iteration. For the sample-pair-based batch setting, 300 pairs are randomly constructed. Note that each person on CUHK03 has 10 images. Thus, 600 images are included in each iteration and the training time per iteration is almost the same for the both settings. Our experiment shows that in the sample-based batch setting, the model achieves rank-1 accuracy of $58.14\,\mathrm{percent}$ after about 175,000 iterations, while in the other setting the rank-1 accuracy is $46.96\,\mathrm{percent}$ after 300,000 iterations. These results validate the effectiveness of the sample-based form in saving the training cost.

6 Conclusion

In this work, we have presented a novel generalized similarity model for cross-domain matching of visual data, which generalizes the traditional two-step methods (i.e., projection and distance-based measure). Furthermore, we integrated our model with the feature representation learning by building a deep convolutional architecture. Experiments were performed on several very challenging benchmark dataset of cross-domain matching. The results show that our method outperforms other state-of-the-art approaches.

There are several directions along which we intend to extend this work. The first is to extend our approach for larger scale heterogeneous data (e.g., web and user behavior data), thereby exploring new applications (e.g., rich information retrieval). Second, we plan to generalize the pairwise similarity metric into triplet-based learning for more effective model training.

Appendix

Derivation of Equation (1)

As discussed in Section 1, we extend the two linear projections $\mathbf{U}$ and $\mathbf{V}$ into affine transformations and apply them on samples of different domains, $\mathbf{x}$ and $\mathbf{y}$, respectively. That is, we replace $\mathbf{Ux}$ and $\mathbf{Vy}$ with $\mathbf{L_A x} + \mathbf{a}$ and $\mathbf{L_B y} + \mathbf{b}$, respectively. Then, the affine Mahalanobis distance is defined as:

$$\begin{aligned} D_{\mathbf{M}} &= \|(\mathbf{L_A x} + \mathbf{a}) - (\mathbf{L_B y} + \mathbf{b})\|_2^2 \\ &= \begin{bmatrix} \mathbf{x}^T & \mathbf{y}^T & 1 \end{bmatrix} \mathbb{S}_{\mathbf{M}} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \\ 1 \end{bmatrix}, \end{aligned} \tag{19}$$

where the matrix $S_{\mathbf{M}}$ can be further unfolded as:

$$\mathbb{S}_{\mathbf{M}} = \begin{bmatrix} \mathbf{L_A^T L_A} & -\mathbf{L_A^T L_B} & \mathbf{L_A^T}(\mathbf{a}-\mathbf{b}) \\ -\mathbf{L_B^T L_A} & \mathbf{L_B^T L_B} & \mathbf{L_B^T}(\mathbf{b}-\mathbf{a}) \\ (\mathbf{a}^T - \mathbf{b}^T)\mathbf{L_A} & (\mathbf{b}^T - \mathbf{a}^T)\mathbf{L_B} & \|\mathbf{a}-\mathbf{b}\|_2^2 \end{bmatrix}. \tag{20}$$

Furthermore, the affine Cosine similarity is defined as the inner product in the space of affine transformations:

$$\begin{aligned} S_{\mathbf{I}} &= (\widehat{\mathbf{L}}_{\mathbf{A}}\mathbf{x} + \widehat{\mathbf{a}})^T(\widehat{\mathbf{L}}_{\mathbf{B}}\mathbf{y} + \widehat{\mathbf{b}}), \\ &= \begin{bmatrix} \mathbf{x}^T & \mathbf{y}^T & 1 \end{bmatrix} \mathbb{S}_{\mathbf{I}} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \\ 1 \end{bmatrix}. \end{aligned} \tag{21}$$

The corresponding matrix $\mathbb{S}_{\mathbf{I}}$ is,

$$\mathbb{S}_{\mathbf{I}} = \begin{bmatrix} \mathbf{0} & \frac{\widehat{\mathbf{L}}_{\mathbf{A}}^T\widehat{\mathbf{L}}_{\mathbf{B}}}{2} & \frac{\widehat{\mathbf{L}}_{\mathbf{A}}^T\widehat{\mathbf{b}}}{2} \\ \frac{\widehat{\mathbf{L}}_{\mathbf{B}}^T\widehat{\mathbf{L}}_{\mathbf{A}}}{2} & \mathbf{0} & \frac{\widehat{\mathbf{L}}_{\mathbf{B}}^T\widehat{\mathbf{a}}}{2} \\ \frac{\widehat{\mathbf{b}}^T\widehat{\mathbf{L}}_{\mathbf{A}}}{2} & \frac{\widehat{\mathbf{a}}^T\widehat{\mathbf{L}}_{\mathbf{B}}}{2} & \widehat{\mathbf{a}}^T\widehat{\mathbf{b}} \end{bmatrix}, \tag{22}$$

We propose to fuse $D_{\mathbf{M}}$ and $S_{\mathbf{I}}$ by a weighted aggregation as follows:

$$\begin{aligned} S &= \mu D_{\mathbf{M}} - \lambda S_{\mathbf{I}} \\ &= \begin{bmatrix} \mathbf{x}^T & \mathbf{y}^T & 1 \end{bmatrix} \mathbb{S} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \\ 1 \end{bmatrix}. \end{aligned} \tag{23}$$

Note that $D_{\mathbf{M}}$ is an affine distance (i.e., nonsimilarity) measure while $S_{\mathbf{I}}$ is an affine similarity measure. Analogous to [14], we adopt $\mu D_{\mathbf{M}} - \lambda S_{\mathbf{I}}$ $(\mu, \lambda \geq 0)$ to combine $D_{\mathbf{M}}$ and $S_{\mathbf{I}}$. The parameters μ , λ, $D_{\mathbf{M}}$ and $S_{\mathbf{I}}$ are automatically learned through our learning algorithm. Then, the matrix $\mathbb{S}$ can be obtained by fusing $\mathbb{S}_{\mathbf{M}}$ and $\mathbb{S}_{\mathbf{I}}$:

$$\mathbb{S} = \begin{bmatrix} \mathbf{A} & \mathbf{C} & \mathbf{d} \\ \mathbf{C}^T & \mathbf{B} & \mathbf{e} \\ \mathbf{d}^T & \mathbf{e}^T & f \end{bmatrix}, \tag{24}$$

where

$$\begin{aligned} \mathbf{A} &= \mu\mathbf{L}_{\mathbf{A}}^T\mathbf{L}_{\mathbf{A}} \\ \mathbf{B} &= \mu\mathbf{L}_{\mathbf{B}}^T\mathbf{L}_{\mathbf{B}} \\ \mathbf{C} &= -\mu\mathbf{L}_{\mathbf{A}}^T\mathbf{L}_{\mathbf{B}} - \lambda\frac{\widehat{\mathbf{L}}_{\mathbf{A}}^T\widehat{\mathbf{L}}_{\mathbf{B}}}{2} \\ \mathbf{d} &= \mu\mathbf{L}_{\mathbf{A}}^T(\mathbf{a}-\mathbf{b}) - \lambda\frac{\widehat{\mathbf{L}}_{\mathbf{A}}^T\widehat{\mathbf{b}}}{2} \\ \mathbf{e} &= \mu\mathbf{L}_{\mathbf{B}}^T(\mathbf{b}-\mathbf{a}) - \lambda\frac{\widehat{\mathbf{L}}_{\mathbf{B}}^T\widehat{\mathbf{a}}}{2} \\ f &= \mu\|\mathbf{a}-\mathbf{b}\|_2^2 - \lambda\widehat{\mathbf{a}}^T\widehat{\mathbf{b}}. \end{aligned} \tag{25}$$

In the above equations, we use 6 matrix (vector) variables, i.e., $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{d}$, $\mathbf{e}$ and f, to represent the parameters of the generalized similarity model in a generic form. On one hand, given μ, λ, $\mathbb{S}_{\mathbf{M}}$ and $\mathbb{S}_{\mathbf{I}}$, these matrix variables can be directly determined using Eqn. (25). On the other hand, if we impose the positive semi-definite constraint on $\mathbf{A}$ and $\mathbf{B}$, it can be proved that once $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{d}$, $\mathbf{e}$ and f are determined there exist at least one solution of μ, λ, $\mathbb{S}_{\mathbf{M}}$ and $\mathbb{S}_{\mathbf{I}}$, respectively, that is, $\mathbb{S}$ is guaranteed to be decomposed into the weighted Mahalanobis distance and Cosine similarity. Therefore, the generalized similarity measure can be learned by optimizing $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{d}$, $\mathbf{e}$ and f under the positive

semi-definite constraint on **A** and **B**. In addition, **C** is not required to satisfy the positive semidefinite condition and it may not be a square matrix when the dimensions of **x** and **y** are unequal.

ACKNOWLEDGMENTS

This work was supported in part by the Hong Kong Scholar Program, in part by Guangdong Natural Science Foundation under Grant S2013050014548 and 2014A030313201, in part by Program of Guangzhou Zhujiang Star of Science and Technology under Grant 2013J2200067, and in part by the Fundamental Research Funds for the Central Universities. This work was also supported by Special Program for Applied Research on Super Computation of the NSFC-Guangdong Joint Fund (the second phase).

REFERENCES

[1] D. Hardoon, S. Szedmak, and J. Shawe-Taylor, "Canonical correlation analysis: An overview with application to learning methods," *Neural Comput.*, vol. 16, no. 12, pp. 2639–2664, 2004.

[2] A. Sharma and D. W. Jacobs, "Bypassing synthesis: Pls for face recognition with pose, low-resolution and sketch," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognition*, 2011, pp. 593–600.

[3] Z. Li, S. Chang, F. Liang, T. S. Huang, L. Cao, and J. R. Smith, "Learning locally-adaptive decision functions for person verification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2013, pp. 3610–3617.

[4] D. Chen, X. Cao, L. Wang, F. Wen, and J. Sun, "Bayesian face revisited: A joint formulation," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 566–579.

[5] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon, "Information-theoretic metric learning," in *Proc. Int. Conf. Mach. Learn.*, 2007, pp. 209–216.

[6] Y. T. Zhuang, Y. F. Wang, F. Wu, Y. Zhang, and W. M. Lu, "Supervised coupled dictionary learning with group structures for multi-modal retrieval," in *Proc. 27th AAAI Conf. Artif. Intell.*, 2013, pp. 1070–1076.

[7] S. Wang, D. Zhang, Y. Liang, and Q. Pan, "Semi-coupled dictionary learning with applications to image super-resolution and photo-sketch synthesis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2012, pp. 2216–2223.

[8] L. Duan, D. Xu, and I. W. Tsang, "Domain adaptation from multiple sources: A domain-dependent regularization approach," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 3, pp. 504–518, Mar. 2012.

[9] D. Ramage, D. Hall, R. Nallapati, and C. D. Manning, "Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora," in *Proc. Conf. Empirical Methods Natural Language Process.*, 2009, pp. 248–256.

[10] J. Zhu, A. Ahmed, and E. P. Xing, "Medlda: Maximum margin supervised topic models for regression and classification," in *Proc. Int. Conf. Mach. Learn.*, 2009, pp. 1257–1264.

[11] A. Sharma, A. Kumar, H. Daume III, and D. W. Jacobs, "Generalized multiview analysis: A discriminative latent space," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2012, pp. 2160–2167.

[12] M. Kan, S. Shan, H. Zhang, S. Lao, and X. Chen, "Multi-view discriminant analysis," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 808–821.

[13] P. Luo, L. Lin, and X. Liu, "Learning compositional shape models of multiple distance metrics by information projection," *IEEE Trans. Neural Netw. Learn. Syst.*, 2015, Doi: 10.1109/TNNLS.2015.244043.

[14] Q. Cao, Y. Ying, and P. Li, "Similarity metric learning for face recognition," in *Proc. Int. Conf. Comput. Vis.*, 2013, pp. 2408–2415.

[15] H. Chang and D.-Y. Yeung, "Locally smooth metric learning with application to image retrieval," in *Proc. 11th Int. Conf. Comput. Vis*, 2007, pp. 1–7.

[16] A. Mignon and F. Jurie, "CMML: A new metric learning approach for cross modal matching," in *Proc. Asian Conf. Comput. Vis*, 2012, pp. 1–14.

[17] X. Zhai, Y. Peng, and J. Xiao, "Heterogeneous metric learning with joint graph regularization for crossmedia retrieval," in *Proc. 27th AAAI Conf. Artif. Intell.*, Jun. 2013, pp. 1198–1204.

[18] C. Kang, S. Liao, Y. He, J. Wang, S. Xiang, and C. Pan, "Cross-modal similarity learning: A low rank bilinear formulation," (2014) [Online]. Available: http://arxiv.org/abs/1411.4738

[19] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural Comput.*, vol. 1, no. 4, pp. 541–551, 1989.

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Adv. Neural Inform. Process. Syst.*, 2012, pp. 1097–1105.

[21] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2015, pp. 3431–3440.

[22] Y. Sun, Y. Chen, X. Wang, and X. Tang, "Deep learning face representation by joint identification-verification," in *Proc. Int. Conf. Adv. Neural Inform. Process. Syst.*, 2014, pp. 1988–1996.

[23] F. Xiong, M. Gou, O. Camps, and M. Sznaier, "Person re-identification using kernel-based metric learning methods," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 1–16.

[24] R. Zhao, W. Ouyang, and X. Wang, "Learning mid-level filters for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2014, pp. 144–151.

[25] E. Ahmed, M. Jones, and T. K. Marks, "An improved deep learning architecture for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2015, pp. 3908–3916.

[26] G. Wang, L. Lin, S. Ding, Y. Li, and Q. Wang, "DARI: Distance metric and representation integration for person verification," in *Proc. AAAI Conf. Artif. Intell.*, 2016, pp. 3611–3617.

[27] G. Andrew, R. Arora, J. Bilmes, and K. Livescu, "Deep canonical correlation analysis," in *Proc. IEEE 30th Int. Conf. Mach. Learn.*, 2013, pp. 1247–1255.

[28] W. Li, R. Zhao, T. Xiao, and X. Wang, "Deepreid: Deep filter pairing neural network for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2014, pp. 152–159.

[29] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu, "Learning fine-grained image similarity with deep ranking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2014, pp. 1386–1393.

[30] D. Yi, Z. Lei, and S. Z. Li, "Deep metric learning for practical person re-identification," *Proc. Int. Conf. Pattern Recog.*, pp. 34–39, 2014.

[31] S. Ding, L. Lin, G. Wang, and H. Chao, "Deep feature learning with relative distance comparison for person re-identification," *Pattern Recog.*, vol. 48, no. 10, pp. 2993–3003, 2015.

[32] W. Li, R. Zhao, and X. Wang, "Human reidentification with transferred metric learning." in *Proc. Asian Conf. Comput. Vis*, 2012, pp. 31–44.

[33] K. Ricanek and T. Tesafaye, "Morph: A longitudinal image database of normal adult age-progression," in *Proc. IEEE Int. Conf. Autom. Face Gesture Recog.*, 2006, pp. 341–345.

[34] B.-C. Chen, C.-S. Chen, and W. H. Hsu, "Cross-age reference coding for age-invariant face recognition and retrieval," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 768–783.

[35] B.-C. Chen, C.-S. Chen, and W. Hsu, "Face recognition and retrieval using cross-age reference coding with cross-age celebrity dataset," *IEEE Trans. Multimedia*, vol. 17, no. 6, pp. 804–815, Jun. 2015.

[36] X. Wang and X. Tang, "Face photo-sketch synthesis and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 11, pp. 1955–1967, Nov. 2009.

[37] Z. Huang, S. Shan, R. Wang, H. Zhang, S. Lao, A. Kuerban, and X. Chen, "A benchmark and comparative study of video-based face recognition on COX face database," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5967–5981, Dec. 2015.

[38] D. Gray, S. Brennan, and H. Tao, "Evaluating appearance models for recognition, reacquisition, and tracking," in *Proc. IEEE 10th Int. Conf. Workshop Perform. Eval. Tracking Surveillance*, 2007.

[39] M. Guillaumin, J. Verbeek, and C. Schmid, "Is that you? metric learning approaches for face identification," in *Proc. Int. Conf. Comput. Vis.*, 2009, pp. 498–505.

[40] K. Q. Weinberger, J. Blitzer, and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," in *Adv. Neural Inform. Process. Syst.*, 2005, pp. 1473–1480.

[41] B. McFee and G. R. Lanckriet, "Metric learning to rank," in *Proc. Int. Conf. Mach. Learn.*, 2010, pp. 775–782.

[42] M. Farenzena, L. Bazzani, A. Perina, V. Murino, and M. Cristani, "Person re-identification by symmetry-driven accumulation of local features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2010, pp. 2360–2367.

[43] M. Kostinger, M. Hirzer, P. Wohlhart, P. M. Roth, and H. Bischof, "Large scale metric learning from equivalence constraints," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2012, pp. 2288–2295.
[44] R. Zhao, W. Ouyang, and X. Wang, "Unsupervised salience learning for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2013, pp. 3586–3593.
[45] R. Zhang, L. Lin, R. Zhang, W. Zuo, and L. Zhang, "Bit-scalable deep hashing with regularized similarity learning for image retrieval," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 4766–4779, Dec. 2015.
[46] D. Gong, Z. Li, D. Lin, J. Liu, and X. Tang, "Hidden factor analysis for age invariant face recognition," in *Proc. Int. Conf. Comput. Vis.*, 2013, pp. 2872–2879.
[47] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," Univ. Massachusetts, Amherst, MA, Tech. Rep., 07-49, 2007.
[48] D. Bouchaffra, "Mapping dynamic Bayesian networks to-shapes: Application to human faces identification across ages," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 8, pp. 1229–1241, Aug. 2012.
[49] Z. Li, U. Park, and A. K. Jain, "A discriminative model for age invariant face recognition," *IEEE Trans. Inform. Forensics Security*, vol. 6, no. 3, pp. 1028–1037, Sep. 2011.
[50] U. Park, Y. Tong, and A. K. Jain, "Age-invariant face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 5, pp. 947–954, May 2010.
[51] D. Chen, X. Cao, F. Wen, and J. Sun, "Blessing of dimensionality: High-dimensional feature and its efficient compression for face verification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2013, pp. 3025–3032.
[52] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2014, pp. 1701–1708.
[53] X. Tang and X. Wang, "Face sketch recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 1, pp. 50–57, Jan. 2004.
[54] W. Zhang, X. Wang, and X. Tang, "Lighting and pose robust face sketch synthesis," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 420–433.
[55] X. Wang and X. Tang, "Dual-space linear discriminant analysis for face recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2004, pp. II–564.
[56] R. Wang, S. Shan, X. Chen, Q. Dai, and W. Gao, "Manifold–manifold distance and its application to face recognition with image sets," *IEEE Trans. Image Process.*, vol. 21, no. 10, pp. 4466–4479, Oct. 2012.
[57] P. Vincent and Y. Bengio, "K-local hyperplane and convex distance nearest neighbor algorithms," in *Proc. Int. Conf. Adv. Neural Inform. Process. Syst.*, 2001, pp. 985–992.
[58] H. Cevikalp and B. Triggs, "Face recognition based on image sets," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2010, pp. 2567–2573.
[59] P. Zhu, L. Zhang, W. Zuo, and D. Zhang, "From point to set: Extend the learning of distance metrics," in *Proc. Int. Conf. Comput. Vis.*, 2013, pp. 2664–2671.
[60] Z. Huang, S. Shan, H. Zhang, S. Lao, A. Kuerban, and X. Chen, "Benchmarking still-to-video face recognition via partial and local linear discriminant analysis on cox-s2v dataset," in *Proc. Asian Conf. Comput. Vis.*, 2013, pp. 589–600.

Liang Lin received the BS and PhD degrees from the Beijing Institute of Technology, Beijing, China, in 1999 and 2008, respectively. He is a professor with the School of Computer Science, Sun Yat-Sen University, China. From 2008 to 2010, he was a postdoctoral research fellow with the Department of Statistics, University of California, Los Angeles. He worked as a visiting scholar with the Department of Computing, Hong Kong Polytechnic University, Hong Kong and with the Department of Electronic Engineering, Chinese University of Hong Kong. His research focuses on new models, algorithms and systems for intelligent processing and understanding of visual data such as images and videos. He has published more than 100 papers in top tier academic journals and conferences. He currently serves as an associate editor of IEEE Transactions on Human-Machine Systems. He received the Best Paper Runners-Up Award in ACM NPAR 2010, Google Faculty Award in 2012, Best Student Paper Award in IEEE ICME 2014, and Hong Kong Scholars Award in 2014.

Guangrun Wang received the BE degree from the School of Information Science and Technology, Sun Yat-sen University, Guangzhou, China, in 2013. He is currently working toward the ME degree in the School of Data and Computer Science, Sun Yat-sen University. His research interests include computer vision and machine learning.

Wangmeng Zuo (M'09, SM'14) received the PhD degree in computer application technology from the Harbin Institute of Technology, Harbin, China, in 2007. From 2005 to 2006, and from 2007 to 2008, he was a research assistant with the Department of Computing, Hong Kong Polytechnic University, Hong Kong. From 2009 to 2010, he was a visiting professor at Microsoft Research Asia. He is currently a professor with the School of Computer Science and Technology, Harbin Institute of Technology. His current research interests include image modeling and low-level vision, discriminative learning, and biometrics. He has authored about 50 papers in those areas. He is an associate editor of the IET Biometrics.

Xiangchu Feng received the BE degree in computational mathematics from Xi'an Jiaotong University and the MS and PhD degrees in applied mathematics from Xidian University, Xi'an, China in 1984, 1989 and 1999, respectively. He is currently a professor in the Department of Information and Computational Science, School of Mathematics and Statistics, Xidian University, Xi'an, China. His current research interests include advanced numerical analysis, image restoration and enhancement based on PDEs and sparse approximation.

Lei Zhang (M'04, SM'14) received the BSc degree in 1995 from the Shenyang Institute of Aeronautical Engineering, Shenyang, P.R. China, and the MSc and PhD degrees in control theory and engineering from Northwestern Polytechnical University, Xian, P.R. China, respectively in 1998 and 2001. From 2001 to 2002, he was a research associate in the Department of Computing, The Hong Kong Polytechnic University. From January 2003 to January 2006 he worked as a postdoctoral fellow in the Department of Electrical and Computer Engineering, McMaster University, Canada. In 2006, he joined the Department of Computing, The Hong Kong Polytechnic University, as an assistant professor. Since July 2015, he has been a full professor in the same department. His research interests include computer vision, pattern recognition, image and video processing, and biometrics, etc. He has published more than 200 papers in those areas. By 2015, his publications have been cited more than 14,000 times in literature. He is currently an associate editor of IEEE Transactions on Image Processing, IEEE Transactions on CSVT and Image and Vision Computing. He recieved the 2012-13 Faculty Award in Research and Scholarly Activities.