Neurocomputing 316 (2018) 166–177

Contents lists available at ScienceDirect

Neurocomputing

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

Pedestrian recognition in multi-camera networks based on deep transfer learning and feature visualization

Jing-Tao Wang[a], Guo-Li Yan[b], Hui-Yan Wang[b], Jing Hua[b,*]

[a] School of Computer Science and Technology, Harbin Institute Of Technology, China
[b] School of Computer and Information Engineering, Zhejiang Gongshang University, Hangzhou, China

ARTICLE INFO

Article history:
Received 31 December 2017
Revised 25 June 2018
Accepted 16 July 2018
Available online 13 August 2018

Keywords:
Convolutional neural network
Deep transfer learning
Feature visualization
Pedestrian recognition
Feature fusion

ABSTRACT

The extensive deployment of surveillance cameras in public places, such as subway stations and shopping malls, necessitates automated visual-data processing approaches to match pedestrians across non-overlapping multiple cameras. However, due to the insufficient number of labeled training samples in real surveillance scene, it is difficult to train an effective deep neural network for cross-camera pedestrian recognition. Moreover, the cross-camera variation in viewpoint, illumination, and background makes the task even more challenging. To address these issues, in this paper we propose to transfer the parameters of a pre-trained network to our target network and then update the parameters adaptively using training samples from the target domain. More importantly, we develop new network structures that are specially tailored for cross-camera pedestrian recognition task, and implement a simple yet effective multi-level feature fusion method that yield more discriminative and robust features for pedestrian recognition. Specifically, rather than conventionally perform classification on the single-level feature of the last feature layer, we instead utilize multi-level feature by associating feature visualization with multi-level feature fusion. As another contribution, we have published our codes and extracted features to facilitate further research. Extensive experiments are conducted on WARD, PRID and MARS datasets, we show that the proposed method consistently outperforms state-of-the-arts.

1. Introduction

Nowadays, the extensive deployment of surveillance cameras in public places such as airports, subway stations, shopping malls, and banks results in a considerable amount of visual data and necessitates automated data analysis technologies. One of the most important visual data processing tasks is to match individuals over non-overlapping camera views automatically, i.e., to search a target pedestrian out of multiple cameras. To simplify this problem, we assume that the target pedestrian is a known subject that has registered in the recognition system beforehand. Based on this close-set assumption, our potential application scenarios can be searching for a student in the campus surveillance system or finding an employee in the company surveillance system, etc. Automated pedestrian recognition is essential for saving manpower costs and improving video processing efficiency in visual surveillance. However, it is a challenging problem due to the significant cross-camera variations in viewpoint, illumination, occlusion, and background clutter, which can cause large intraclass (each pedestrian is considered as a class) variations as shown in Fig. 1.

To address the problem of serious intraclass variations, researchers have done a lot of work (e.g., [1–3]) with regard to pedestrian images' feature modeling, aiming at designing feature representations that are robust to both illumination and viewpoint change. Several effective approaches have been proposed, such as the ensemble of localized features (ELF) [1], symmetry-driven accumulation of local features (SDALF) [2], kBiCov [4] and local descriptors encoded by fisher vector (LDFV) [3]. These handcrafted feature descriptors have achieved some improvements over traditional feature representations, but how to learn a more robust and discriminative feature that can be adaptive to different surveillance scenarios still remains an open problem.

On the other hand, deep learning has gained much attention in recent years and has lead to breakthroughs in many research fields, including speech recognition, object detection, image classification, and scene understanding. The success of deep learning largely owes to its multi-layer nonlinear structure, which has the capacity for approaching any complicated nonlinear function. Therefore, in this paper, we resort to deep learning methods to solve the problem of pedestrian recognition.

* Corresponding author.
E-mail addresses: ingtaow@yahoo.com (J.-T. Wang), xwang@zjgsu.edu.cn (J. Hua).

https://doi.org/10.1016/j.neucom.2018.07.063
0925-2312/

Fig. 1. Illustration of intraclass variations. A pedestrian is regarded as a class, while multiple pedestrians are regarded as multiple different classes.

However, the publicly available multi-camera pedestrian datasets are relatively small, for example, the WARD dataset contains 4,786 images of 70 pedestrians. It is difficult to train an effective deep neural network with such small number of samples. This motivates us to adopt the idea of transfer learning. More specifically, we first formulate pedestrian recognition into classification problem, then transfer the parameters of a pre-trained classification network, which are trained on a large-scale dataset, to our target classification neural network. Afterwards, we update the parameters adaptively using the available small-scale training dataset. More importantly, we revise the pre-trained network architecture to achieve more effective pedestrian-recognition network structure, and associate feature visualization and multi-level feature fusion to extract more discriminative and robust features in a semi-supervised way. To summarize, the key contributions of this work are:

- We propose a novel transfer learning framework for pedestrian recognition, which incorporates feature visualization to select features and revise network architectures in a supervised way, and propose a multi-level feature fusion method that yield more discriminative and robust features for pedestrian recognition.
- We publish our codes and extracted features on the Internet to facilitate further research on cross-camera pedestrian recognition.
- Comprehensive experiments have been conducted on three real world datasets. Empirical evaluation shows that our proposed method consistently and significantly outperforms the existing methods. The reasons behind the observations are also presented together with other empirical findings.

2. Related work

2.1. Pedestrian recognition

Typically, methods for pedestrian recognition include 2 components. One is feature modeling, i.e., designing feature representations that are robust to illumination and viewpoint changes. The other is discriminative model learning, which can be further divided into metric learning based methods and classifier based methods. The basic idea behind metric learning approaches is to learn distance or similarity measurement models that make the feature vectors from same image pairs have higher similarity than feature vectors from different image pairs. Meanwhile, the classifier based methods take pedestrian recognition as a problem of multi-classification. Therefore, they have a different configuration in application compared with metric learning based methods. Specifically, instead of querying a probe image captured by one camera among a gallery captured by another camera based on similarity, the classifier based methods are designed for recognizing the identity of a pedestrian in a video captured from the multi-camera surveillance network. Apart from the direct application, another application is searching for an interested pedestrian throughout the multi-camera surveillance network by matching the identity labels based on the classification results.

In order to address the problem of serious intraclass variations caused by different camera views, some researchers have committed to finding a robust feature representation [5–8]. Park et al. [5] proposed a visual search engine using color feature to identify pedestrians under multi-camera networks. However, pedestrians that dressed in similar color clothes could not be distinguished well with only color features. Therefore, to improve matching accuracy, texture and spatial features were added into feature representations in some works. Bak et al. [9] proposed solving pedestrian recognition based on the Haar feature and DCD feature. They used the Adaboost algorithm to select the most distinguishing features from the Haar feature pool. Then, the selected features were processed by a predefined distance function to calculate feature similarity. Conversely, Farenzena et al. [2] proposed a method based on symmetric local features. They segmented the silhouette of the human body into an upper part and lower part, and extracted salient parts of the body according to the axes of symmetry and asymmetry of the human body. Then, color and texture features were extracted from each part and fused. Integrating color, texture, and spatial information, this method further improved recognition results. Recently, researchers have proposed some other new feature representations, such as saliency features [10,11]. Zhao et al. proposed to extract saliency information of human images in an unsupervised manner. Each human image was densely segmented into a grid of local patches. Then, dense color histogram features and dense SIFT features were extracted from each path. The salience score of each patch was learned based on the distance of the k-nearest neighbor. A salience feature map of each human image was then obtained for pedestrian identification. Different from the above methods, we proposed using convolutional neural networks (CNNs) for extracting multi-level deep features of human images. Using feature visualization, we chose discriminative network layers' outputs to construct pedestrian feature representation. Then, the chosen layers' outputs were extracted and fused effectively by the proposed novel feature fusion method. The work that is most related to our study is [12] by Razavian et al. that use features extracted from the OverFeat network [13] as a generic image representation for recognition tasks. However, they only use features extracted from a single layer to feed a linear SVM classier, while we choose multiple layers to model fused feature and use visualization to assist this process.

As for discriminative model learning, some works have focused on developing similarity metrics that make feature vectors from same image pairs have higher similarity scores than feature vectors from different image pairs. Prosser et al. [14] proposed a method based on support vector ranking. They reformulated the person re-identification problem as a ranking problem, and learned a subspace where the potential true match was given the highest ranking rather than any direct distance measure. Based on this work, Zheng et al. [15] proposed a relative distance

comparison learning model. Their approach aimed to minimize intraclass variation while maximizing interclass variation. Besides, Liao et al. [16] proposed a metric learning method called Cross-view Quadratic Discriminant Analysis (XQDA). They learned a discriminant low-dimensional subspace by cross-view quadratic discriminant analysis and applied the distance function on the derived subspace. More recently, Yang et al. [17] proposed a logistic discriminant metric learning method, which learn an optimal distance function by constructing a locally adaptive decision rule with the help of privileged information. Wang et al. [18] proposed a locality constraint distance metric learning to reduce the influence among different traffic scenes. On the other hand, there are also some works that aimed to identify pedestrians by classification methods, which is fairly close to object recognition works [19–22], except that classification based pedestrian recognition methods focus on distinguishing between intra-classes instead of inter-classes. For example, Teixeira et al. [23] proposed to quantify SIFT features based on the vocabulary tree. The extracted features were then sent to a support vector machine (SVM) classifier for pedestrian recognition. Based on this work, Wang et al. [24] proposed to match pedestrians across multiple non-overlapping camera views based on multi-feature fusion and incremental learning. They performed matching based on a classification method, and achieved good object recognition performance. Recently, Wang et al. [25] proposed a multilevel important salient feature and a novel support-vector-machine (SVM) based incremental learning method for pedestrian recognition in multi-camera networks. Their method further improved the accuracy compared to existing classification-based recognition methods. In this paper, we took pedestrian recognition as a problem of classification, and used multi-level deep features to construct multiple binary SVM classifiers. We then formulate a linear weighted sum approach to transfer the decision value outputs of these binary classifiers into the final classification result.

Recently, many end-to-end deep learning methods have been proposed in the literature of person re-identification [26–35]. Yi et al. [27] used a siamese deep neural network to jointly learn the color feature, texture feature and metric in a unified framework. The siamese CNN outputs a similarity score given a pair of images as input. Similar to this work, Ahmed et al. [26] also used a network with two sub-networks that outputs a similarity value of two input images. But their architecture further includes a layer that computes cross-input neighborhood differences and a subsequent layer that summarizes these differences to capture relationships between two input views. More recently, Chung et al. [28] proposed a two stream CNN architecture where each stream is a siamese network. This architecture learns spatial and temporal information separately and combines the siamese cost of the spatial and temporal streams with a weighted cost function. Zheng et al. [29] proposed a siamese network that simultaneously computes the identification loss and verification loss to learn a discriminative embedding and a similarity measurement at the same time. This method can be easily applied on different pre-trained networks. Instead of training on pair-wise pedestrian images, there are also some deep networks taking image triplets as input and employing a triplet loss function [30,31,35]. However, generating image pairs or triplets for training is often inefficient and unstable. Moreover, the embedding similarity metrics learned in these networks are difficult to generalize for other camera views. Different from these approaches, we focus on recognizing the identities of different pedestrians and resort to a classification network for extracting features followed by training an off-the-shelf classifier with fused features. Our framework is flexible to incorporate many existing classification deep networks and applicable to multi-camera surveillance scenarios.

2.2. Deep features visualization

Feature visualization techniques can be used to help understand deep neural networks and assist in tasks such as object detection [36], tracking [37], and image classification [38]. Zeiler et al. [38] proposed to use a multi-layered deconvolutional network (deconvnet) for feature visualization. The feature activations of each layer were input into the deconvnet and projected back to the pixel space through unpooling, rectification, and filtering, successively. This visualization approach gave insight into the function of intermediate feature layers and helped them find model architectures that outperformed the method proposed by Krizhevsky et al. [19] on the ImageNet classification benchmark. Turcsany et al. [39] visualized the Restricted Boltzmann Machine (RBM) features by displaying the hidden nodes' weight vectors in the shape of the input image data. And they visualized higher layer features in a Deep Belief Network (DBN) by using the connection weights to calculate a linear combination of certain previous layer features. The visualization methods enabled them to compare the distinctiveness of DBN features for face completion. Ma et al. [37] visualized the upsampled outputs of the third, fourth, and fifth layers of the VGG-Net [20] to help interpret different merits of the features that using the output of different convolutional layers. And accordingly, they utilized multiple CNN layers to improve the performance in visual tracking. Inspired by these works, we proposed to visualize feature outputs of the pre-trained network layers, so as to interpret the extracted features of each layer qualitatively, aid the adjustment of the target network, and direct the fusion of multi-layer features. Based on that, we used multi-level deep features to construct multiple binary SVM classifiers and implement feature fusion in the decision-making process. Pedestrians were recognized according to the classification results. Finally, our experimental results verified the effectiveness of the proposed method.

3. Method

3.1. Overall architecture

Given a pedestrian surveillance video captured in a camera network, our goal is to classify the pedestrians in the video frames to achieve recognition. The proposed framework mainly include the following: network pre-training, deep feature visualization, target-network transfer learning, feature fusion, and classification. The pipeline of our proposed approach is depicted in Fig. 2, and the specific steps are as follows:

Step 1. Perform pre-training based on the large-scale source domain (ImageNet dataset in our case) to get an initial network, hereinafter, referred to as the *pre-trained network*.

Step 2. Input pedestrian samples to the *pre-trained network* for hierarchical feature visualization, so as to interpret the extracted features of each layer qualitatively, validate the applicability of the pre-trained network, and aid the construction of new networks on the target domain.

Step 3. Transfer the parameters of the pre-trained network to the new network (i.e., target network) and use the small-scale target dataset to update the target network.

Step 4. Extract multi-level deep features from the pedestrian images in the target domain based on the trained target network, and identify discriminative hierarchical features for constructing pedestrian feature representation using feature visualization.

Step 5. Implement multi-level feature fusion and feed the fused features into the decision-making process of classifiers.

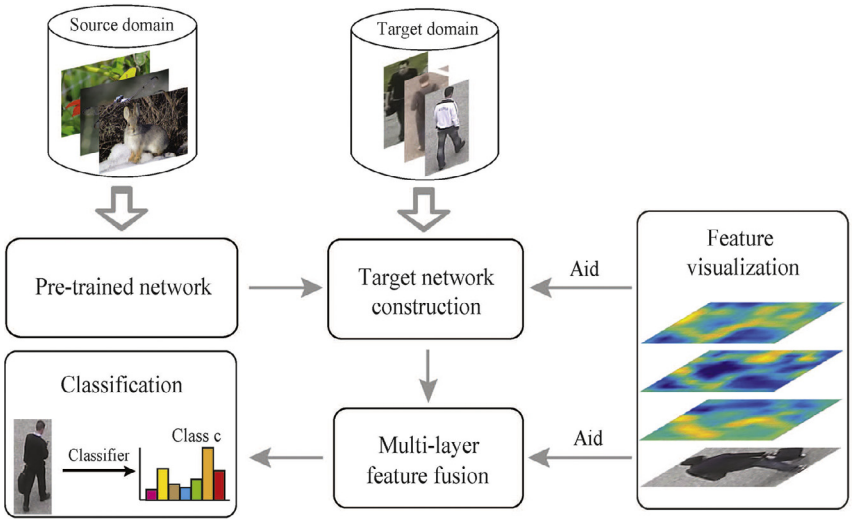


Fig. 2. Overview of our proposed approach.

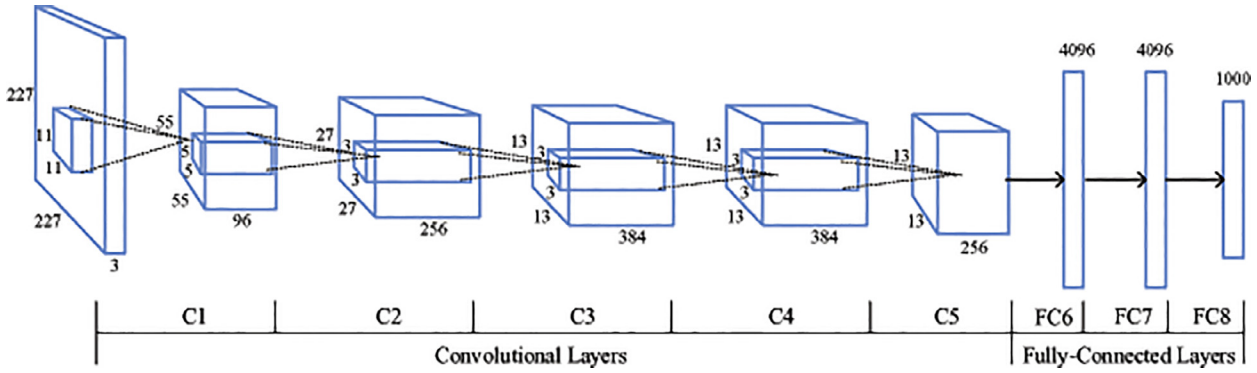


Fig. 3. Architecture of the pre-trained network. Each layer further consists of several sub-layers. Sequentially, C1 consists of conv1, relu1, pool1 and norm1; C2 consists of conv2, relu2, pool2 and norm2; C3 consists of conv3 and relu3; C4 consists of conv4 and relu4; C5 consists of conv5, relu5 and pool5; FC6 consists of fc6 and relu6; FC7 consists of fc7 and relu7.

3.2. Pre-trained network

Most publicly available pedestrian datasets are relatively small, for example, the WARD dataset [40] contains only 4786 images of 70 pedestrians. It is difficult to train an effective deep neural network on such small-scale datasets, since millions of parameters have to be learned. Therefore, instead of training a network from scratch on the small-scale target domain directly, we pre-train an initial network on the large-scale source domain and transfer its parameters to help network training on the target domain. Specifically, we leverage the classical convolution neural network proposed by Krizhevsky et al. [19], which consists of 5 convolutional layers and 3 fully-connected layers, with a final 1,000-way softmax. To mitigate the vanishing gradient problem in gradient based learning methods, we follow [19] and use rectified linear units (ReLUs) as alternatives to traditional saturating nonlinearities in the networks. We also employ dropout in the first 2 fully-connected layers as a regularization method to avoid over-fitting.

The architecture of our pre-trained network is illustrated in Fig. 3. Specifically, the first convolutional layer filters the $224 \times 224 \times 3$ input image, with 96 kernels of size $11 \times 11 \times 3$ with a stride of 4 pixels. After nonlinearly activating, pooling, and local response normalization, the convolutional results are sent into the second convolutional layer. The second convolutional layer filters the input with 256 kernels of size $5 \times 5 \times 96$. After nonlinearly activating, pooling, and local response normalization, the convolutional results are sent into the next convolutional layer. The third convolutional layer filters the input with 384 kernels of size $3 \times 3 \times 256$, and the forth convolutional layer took the activated output of the third convolutional layer as input and filters it with 384 kernels of size $3 \times 3 \times 384$. The fifth convolutional layer takes the activated output of the forth convolutional layer as input and filters it with 256 kernels of size $3 \times 3 \times 384$. This is followed by 2 fully-connected layers, both of which have 4096 neurons, and the final fully-connected layer is a 1000-way softmax. Following the method proposed by Krizhevsky et al. [19], we use a subset of ImageNet, which contains 1.2 million training images, 50,000 validation images, and 150,000 testing images of 1,000 categories to learn the pre-trained network.

3.3. Feature visualization of the pre-trained network

The motivation to visualize the hierarchical outputs of the pre-trained network is to examine the applicability of the pre-trained network on the pedestrian recognition task, since the pre-trained network is trained on the ImageNet database with the original objective to classify the images of 1000 diversified categories (such as human, animal, plant, and car). In this paper, however, the task is to recognize different pedestrians, i.e., to distinguish between individuals. This belongs to fine-grained recognition.

Yosinski et al. [41] has experimentally quantified the transferability of features from each layer of a neural network in the

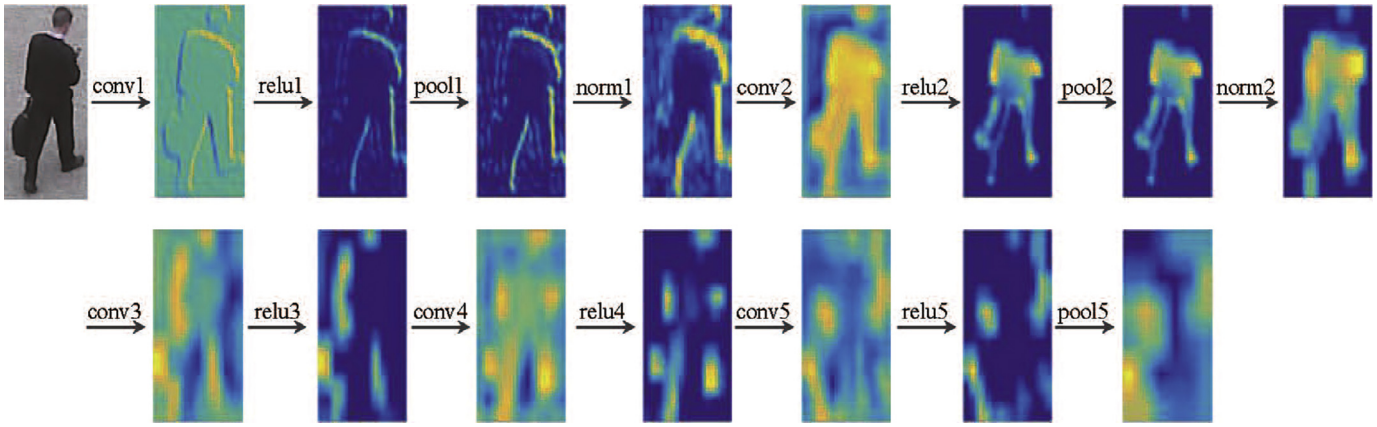


Fig. 4. Feature visualization results of each layer of the pre-trained network.

cross-task situation. Inspired by this work, many recent works has intuitively transferred the convolutional features of a ImageNet-pre-trained model for tasks such as object detection [42] and person re-identification [29,31,32]. Although initializing a target network with an ImageNet-pre-trained model and fine-tuning for the target task has become a common practice, further examination regarding whether such network is suitable for our pedestrian recognition task is required.

Towards this aim, we fed pedestrian images of target datasets into the pre-trained network and visualized its hierarchical outputs, so as to interpret the extracted features of each layer qualitatively and test the rationality of the network structure. Fig. 4 shows the visualization results of the feature maps extracted from the first 5 layers of the pre-trained network. For the sake of perspicuity, we merely present one feature map for each convolutional layer. We observe that after the convolutional operation of the first layer, the silhouette information of the human body is intensified. Furthermore, after the following ReLU non-linear activation units, only a small portion of the input signal is activated, and a large number of signals are shielded. Such sparse activation of neurons help speed up the sparse features extraction. After the max-pooling operation of the first layer, salient features are preserved (see the yellow part in Fig. 4), and after the local response normalization of the first layer, salient features are further enhanced. We also find that the convolutional operation of the second layer further explores the texture information of images. Additionally, after the following non-linear activation, max-pooling, and local response normalization of the second layer, salient features are preserved and further enhanced. Convolutional layers 3–5 unearths more abstract features, while for the fifth pooling layer (pool5, see bottom row of Fig. 4), we find it difficult to make out the silhouette of the human body. However, we can see that higher activation signals appear at the locations of handbag and cell phone in the visualized feature maps of higher layers, which indicates that the network learned to locate the discriminative parts of the pedestrian automatically.

From the hierarchical visualization results of the network, we can observe that the original input is transformed into shallow-level, middle-level, and high-level features, layer-by-layer in the deep network. Furthermore, the deeper the network, the more abstract features it outputs. By means of feature visualization, we are able to intuitively understand the features extracted from each layer of the network. In the meantime, the feasibility of transferring the pre-trained network to the target domain can be qualitatively validated. To better interpret the features, more advanced visualization methods need to be explored, such as the prediction difference analysis method [43] and a hybrid visualization method to disclose the multiple facets of each neuron and the interactions between them [44]. We leave it to future work. In the following, we will introduce the details of training a new network on the target domain use transfer learning.

3.4. Learning networks on target domain

In terms of data distribution, the target domain is quite different from the source domain, namely, the target domain contains only pedestrian images while the source domain contains images of diversified categories such as human, car, animal and plant. Therefore, we can not directly apply the pre-trained network on the target domain. Inspired by transfer learning, we adjust the network architecture on the basis of the pre-trained network, and update the network parameters adaptively by using the training samples of the target dataset.

From the feature visualization analysis of the pre-trained network, we qualitatively verified the effectiveness of the first 5 convolutional layers for pedestrian feature representation. Therefore, we reserve the parameters of the first 5 layers while transferring the pre-trained network to the target domain. Fig. 5 illustrates the process of transfer learning in this paper. We construct 3 target network models with different depths. One is an 8-layer network model that is the same as the pre-trained network. *Secondly*, by discarding the FC 7 layer of the pre-trained network model, we obtain a new network model with 7 layers. *Thirdly*, by further discarding the FC6 layer, we obtain another new network model with 6 layers. The last layer of the target network model is a C-way softmax, where C represents the number of identities in the target dataset. The parameters of this layer must be re-learned by the training samples of the target dataset. To distinguish it with the 1,000-way FC8 layer of the pre-trained network, we renamed it as *FC8_Target*.

We construct 3 target network models with different depths, so as to directly connect the last softmax layer with different layers and enable the softmax to take features from different layers as inputs for classification. Then, we evaluate the recognition (or classification) performance of the different models to determine the network depth suitable for the pedestrian recognition task. The comparison results of target network models with different depths will be presented in Section 4.2.

3.5. Extracting and visualizing deep features

Conventionally, the last layer of the CNN is a softmax classifier, which can only take the output of the preceeding layer as the input feature for classification, and thus can not take full advantage of

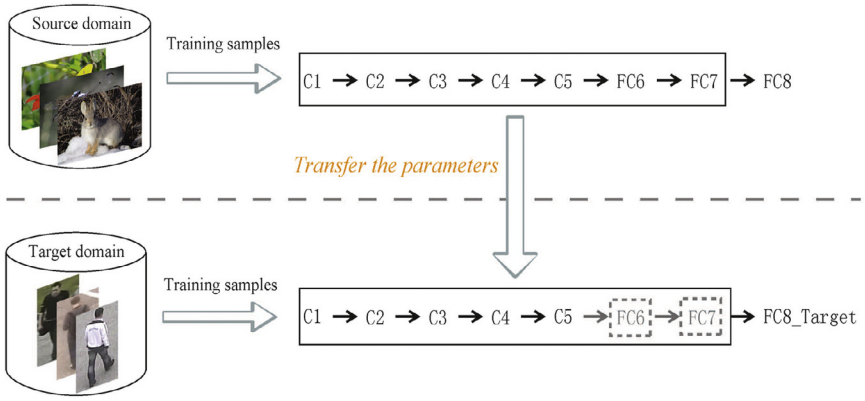


Fig. 5. Transferring parameters from the pre-trained network to the target network.

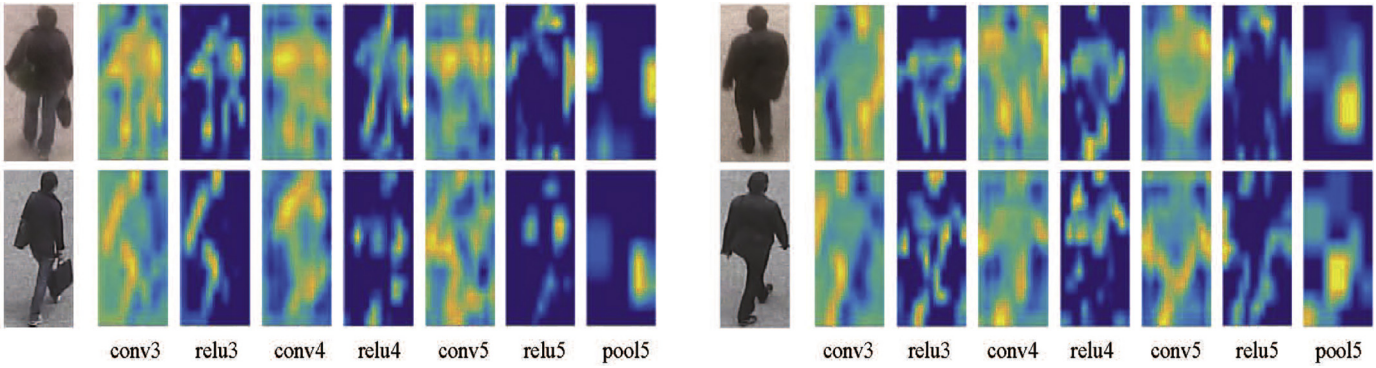


Fig. 6. Comparison of feature maps of different pedestrians by feature visualization.

deep features from different layers. In fact, the deep networks can be deemed as feature extractors that extracts multiple-level features, and we can replace the softmax with other classifiers, such as SVMs, for pedestrian recognition.

Due to the complexity of the network architecture, we can extract multi-level deep features, and the discriminative ability of these features is varied. Therefore, we draw support from the feature visualization approach to choose features suitable for the pedestrian recognition task and with strong discriminative ability. Fig. 6 shows the visualization results of the third to fifth layers of the network. The first 8 columns of the first and second rows represent the image sample and corresponding feature maps of pedestrian A under camera 1 and camera 2, respectively. Similarly, the latter 8 columns of the first and second rows represent the image sample and corresponding feature maps of pedestrian B under camera 1 and camera 2, respectively.

For convenience, we denote the 4 pedestrian images in row 1 column 1, row 2 column 1, row 1 column 9, and row 2 column 9, as *A-cam1, A-cam2, B-cam1, and B-cam2*, respectively. Obviously, A-cam1 is similar to B-cam1 in visual appearance, but relatively dissimilar to A-cam2 that with the same identity. As shown in Fig. 6, columns 2, 4, 6, 10, 12, and 14 have much yellow color scattered across these feature maps. Note that the yellow parts represent high response values while the blue parts represent low response values. As for columns 3, 5, 7, 11, 13, and 15, although only a small portion of the input signals is activated after applying the ReLU nonlinearity, the distinction between pedestrian A and pedestrian B is not obvious from these visualized feature maps. Furthermore, columns 8 and 16 (i.e., the feature maps of the pool5 layer) retain only a small amount of yellow color, which are located exactly on the discriminative parts of the pedestrian. For example, the locations of the handbag belonging to pedestrian A and the backpack belonging to pedestrian B display high response values. By comparing the 4 feature maps of pool5, we find that the response feature maps of the same identity are similar, while those of different identity are dissimilar, i.e., the features extracted from pool5 are robust and discriminative. Based on the above analysis, we chose outputs of the pool5 layer to construct pedestrian feature representation.

In addition to the outputs of pool5 layer, we also extract the outputs of its following 2 fully-connected layers, i.e., fc6 and fc7. We implement an effective feature fusion method on the multi-level deep features to obtain the final feature representation for a pedestrian image.

3.6. Multi-level feature fusion

So far, we have used the CNNs for extracting multi-level deep features. The multiple CNN layer features need to be further fused in order to get a single feature representation for each image sample. The most common way for feature fusion is concatenating all extracted feature vectors after ℓ_1 or ℓ_2 normalization, as in [11,16,24,25]. But the concatenated features have rather high dimensions. Another way is to fuse features by adding together, which is used in [45]. But it is inapplicable in this paper due to the different feature dimensions of different CNN layers. Given these reasons, in the following, we formulate a linear weighted sum approach to conduct feature fusion in the decision-making process of SVMs.

We employ the one-vs.-rest method to construct multiple binary SVM classifiers with multi-level deep features. Assuming that the number of classes in the target domain is k, then k binary SVM classifiers are trained. Meanwhile, the i-th classifier separates the i-th class from other classes. To train the i-th binary classifier, the training samples of the i-th class are taken as positives and the

training samples of other classes are taken as negatives. Assume that the decision function of the i-th classifier is g_i, $i = 1, 2, \ldots, k$. Then, the predicted label of sample x can be expressed as

$$C = \arg\max_i (g_i(x)) \tag{1}$$

where $i = 1, 2, \ldots, k$. We construct multiple sets of binary SVM classifiers with training samples represented by multi-level deep features based on the one-vs.-rest method. Let g_{li} denote the decision function of the i-th classifier trained by the l-th layer features ($l = 5, 6, 7$ in our experiments); and let x_l denote sample x expressed by the l-th layer features. For a test sample x, the decision values of k binary classifiers trained by the l-th layer features can be combined as a decision vector

$$G_l = [g_{l1}(x_l), g_{l2}(x_l), \ldots, g_{lk}(x_l)]^T \tag{2}$$

where k represents the number of classes in the target dataset. To effectively fuse the multi-level deep features, we formulate a linear weighted sum approach to transfer the decision value outputs of all binary classifiers into a final decision vector, i.e., the decision vector corresponding to our fusion features can be written as

$$V = \sum_l \alpha_l G_l \tag{3}$$

where $V \in \mathbb{R}^k$, and α_l represents the weight of the l-th layer features in the fusion features. The weights of different level features are determined by cross-validation in our experiments. Then, the predicted label of sample x can be further expressed as

$$C' = \arg\max_i \left(\sum_l \alpha_l G_{li}(x_l) \right) = \arg\max_i (v_i) \tag{4}$$

where v_i represents the ith element of vector V.

Algorithm 1 summarizes the detailed procedure of feature fusion and classification.

Algorithm 1: feature fusion and classification

Input: Training set $\mathbf{X} = \left\{ \left(x^{(i)}, y^{(i)} \right) | i = 1, 2, \ldots, n \right\}$, test set $\mathbf{T} = \left\{ x^{(\tau)} | \tau = 1, 2, \ldots, m \right\}$, feature weight parameters α_l;
Output: Classification results $\mathbf{Y} = \left\{ y^{(\tau)} | \tau = 1, 2, \ldots, m \right\}$ of test set.

1. Input training samples $\left\{ x^{(i)} \right\}_{i=1}^{n}$ to the target network and extract the outputs of l-th layer to construct the deep features of l-th layer $\left\{ x_l^{(i)} \right\}_{i=1}^{n}$;
2. Take $\left\{ \left(x_l^{(i)}, y^{(i)} \right) | y^{(i)} = j \right\}$ as positives and $\left\{ \left(x_l^{(i)}, y^{(i)} \right) | y^{(i)} \neq j \right\}$ as negatives for training binary SVM classifier: SVM_{lj};
3. Input test samples $\left\{ x^{(\tau)} \right\}_{\tau=1}^{m}$ to the target network and extract the outputs of l-th layer to construct the deep features of l-th layer $\left\{ x_l^{(\tau)} \right\}_{\tau=1}^{m}$;
4. Compute decision vectors of $\left\{ x_l^{(\tau)} \right\}_{\tau=1}^{m}$ according to Eq. (2);
5. Compute decision vectors of fusion features according to Eq. (3);
6. Obtain classification result $\mathbf{Y}$ according to Eq. (4).

4. Experiments

We conducted extensive experiments on the WARD [40], PRID [46] and MARS datasets [47] to evaluate our proposed approach. The reasons behind selecting these 3 datasets are that they contain real-world pedestrian images collected from multiple camera views and each pedestrian posses multiple image samples.

Next, we will evaluate the classification performance of our proposed 8-layer, 7-layer, and 6-layer networks, respectively. In addition, we replace the softmax with SVMs to implement multi-level feature fusion in the decision-making process. We implemented our methods utilizing the Caffe deep learning framework, and all experiments were conducted using the following hardware configuration: core i7-4790 processor, 32GB memory, and NVIDIA Quadro K620 graphics card. Our codes and extracted features are made publicly available on the Internet[1] to facilitate further research.

4.1. Target datasets

The WARD dataset contains 4786 images of 70 pedestrians, captured by 3 non-overlapping cameras. Each pedestrian appears in all camera views. Fig. 1 shows some samples of WARD. We can see that there exists great cross-camera variations in viewpoint, illumination, and background clutter, which can cause large intra-class variations. This poses a higher demand for the robustness of pedestrian feature representation.

In contrast, the PRID dataset consists of pedestrian images captured from 2 cameras. Fig. 7 shows example pairs of images from the PRID dataset. Each pedestrian has at least 5 image samples under each camera view. In total, 385 pedestrians were captured by camera A, while 749 were captured by camera B. The first 200 persons appeared in both cameras. Thus, we used 40,033 images of these 200 pedestrians for experiments.

The MARS dataset is a more challenging person re-identification dataset that contains images captured by 6 cameras. In the dataset, there are 1261 different pedestrians, and each pedestrian posses images captured by at least two cameras. Moreover, images of the same person have significant variation in viewpoint, pose, occlusion and background. The original image resolutions are 128×48, 128×64, 256×128 of WARD, PRID and MARS, respectively. They were resized to a unified resolution of 256×256 to match with the input layer of our networks. For the purpose of data augmentation, a cropping operation was further applied to the input images, which result in a resolution of 227×227, as Fig. 3 depicted.

4.2. Classification performance of different networks

We took the WARD and PRID datasets as target datasets and tested the classification performance of CNNs with varied depths, respectively. For the 8-layer network, we fine-tuned the pre-trained network on the target network, and only the parameters of the last softmax layer need to be updated. The 7-layer network was obtained through removing the seventh layer (fc7) of the pre-trained network, and connecting the sixth layer (fc6) to the softmax classifier directly. Then, the 6-layer network was obtained through removing the seventh layer (fc7) and sixth layer (fc6) of the pre-trained network, and connecting the fifth pooling layer (pool5) to the softmax classifier directly. Likewise, only parameters of the last softmax layers need to be updated for the 7-layer and 6-layer networks.

For the WARD dataset, we randomly selected $\mathscr{R}$ images for training, and took the remaining images as test samples, where $\mathscr{R} = 500, 1,000, 1,500, \ldots, 3,000$. For the PRID dataset, we randomly selected $\mathscr{F}$ images for training, and took the remaining images as test samples, where $\mathscr{F} = 2000, 5000, 10,000, \ldots, 25,000$. To ensure images of each class has been included in train set, we further sample one image for each missing class. Note that after the random sampling process, not each class contains images from each camera in the train set—in fact, some classes may only contain training images from a single camera, which allows our approach to be more flexible. The classification accuracies of different networks with varied depths and different number of training samples on WARD are shown in Table 1, and experimental results on PRID are illustrated in Table 2. From the tables, we can observe that the 6-layer network (pool5 feature) and 7-layer network (fc6 feature) outperform 8-layer network (fc7 feature) for pedestrian recognition if we merely fine-tune the softmax layer.

[1] https://github.com/babyfisher/MultiCamera_Pedestrian_Recognition

Fig. 7. Example pairs of images from the PRID database.

Table 1
Classification accuracies of networks with different depths on the WARD database.

Network depth	Number of training samples					
	500	1000	1500	2000	2500	3000
8	83.54	93.58	96.7	97.24	98.7	99.16
7	85.58	95.4	**98.4**	**99.34**	**99.4**	**99.8**
6	**85.6**	**95.6**	98.26	99.26	99.1	99.12

Table 2
Classification accuracies of networks with different depths on the PRID database.

Network depth	Number of training samples					
	2000	5000	10,000	15,000	20,000	25,000
8	86.28	93.98	96.30	96.32	97.16	97.30
7	89.40	**97.66**	**98.74**	99.34	**99.54**	**99.62**
6	**89.42**	97.60	98.56	**99.48**	99.40	99.32

Table 3
Run time of networks with different depths.

Network depth	8	7	6
Fine-tune time(s)	3021	2650	1858
Test time of a single image (ms)	3.53	3.30	2.56

Table 4
Classification accuracy of different layer features on WARD dataset with SVM classifiers.

Features	Number of training samples					
	500	1000	1500	2000	2500	3000
fc7	85.18	95.54	97.99	99.03	99.39	99.78
fc6	88.24	97.28	98.87	99.57	99.74	99.89
pool5	89.36	98.02	**99.45**	99.89	**99.87**	**100**
pool5_fc6	89.52	98.10	**99.45**	**99.93**	**99.87**	**100**
pool5_fc7	89.52	97.99	**99.45**	99.89	99.83	**100**
fc6_fc7	88.12	97.31	98.87	99.61	99.74	99.89
pool5_fc6_fc7	89.59	98.15	**99.45**	99.89	**99.87**	**100**
Ours	**89.90**	**98.42**	**99.45**	**99.93**	**99.87**	**100**

For example, when using the same 2000 images from WARD for training, the classification accuracies of 6-layer and 7-layer networks can surpass 99%, while the accuracy of the 8-layer network is around 97%. A reasonable explanation is that the pool5 and fc6 layer can capture fine-grained spatial details such as discriminative parts of pedestrians, while the fc7 layer is more closely related to category-level semantics. And for pedestrian recognition, which belongs to fine-grained recognition, the spatial details are more efficient than semantics. We also tried to fine-tune all 8 layers, but it turned out to have an over-fitting issue on extremely small target datasets. For instance, when fine-tuning on the WARD dataset with 500 randomly selected training images, the classification accuracy of the network fine-tuned only with the last softmax layer is 83.54%, while the accuracy of the network fine-tuned with all layers is 81.25%

For training target networks with different depths, we set the number of iterations for fine-tuning as 10,000 and the test interval as 1000. We started with a base learning rate of 0.001 and used a momentum of 0.9, and the weight decay is set to 0.0005 on both WARD and PRID datasets. Table 3 shows the fine-tune time of networks with different depths and the test time of a single image. From Table 3, we can see the high efficiency of fine-tuning the pre-trained network. Also, obviously, the fewer the layers of the network, the less time required for fine-tuning and testing.

4.3. Training SVMs with multi-layer deep features

To further verify the validity and extensibility of deep features, we extracted multi-layer outputs of the target networks to construct feature vectors, and replaced the softmax with SVMs to test the recognition performance of different level deep features. More specifically, we input all images of the target dataset to a trained network, and extracted their multi-level deep features. Each layer of deep features of training samples were used to train a multi-class SVM classifier, and test samples expressed by these layer features were used to evaluate the trained SVM classifier. We compared experimental results of the output features of fc7, fc6, and pool5 on both WARD and PRID datasets (see the first 3 rows in Tables 4 and 5). Tables 4 and 5 show that for the compared single layer features, the output features of pool5 achieved the best classification accuracy rates on multi-class SVMs.

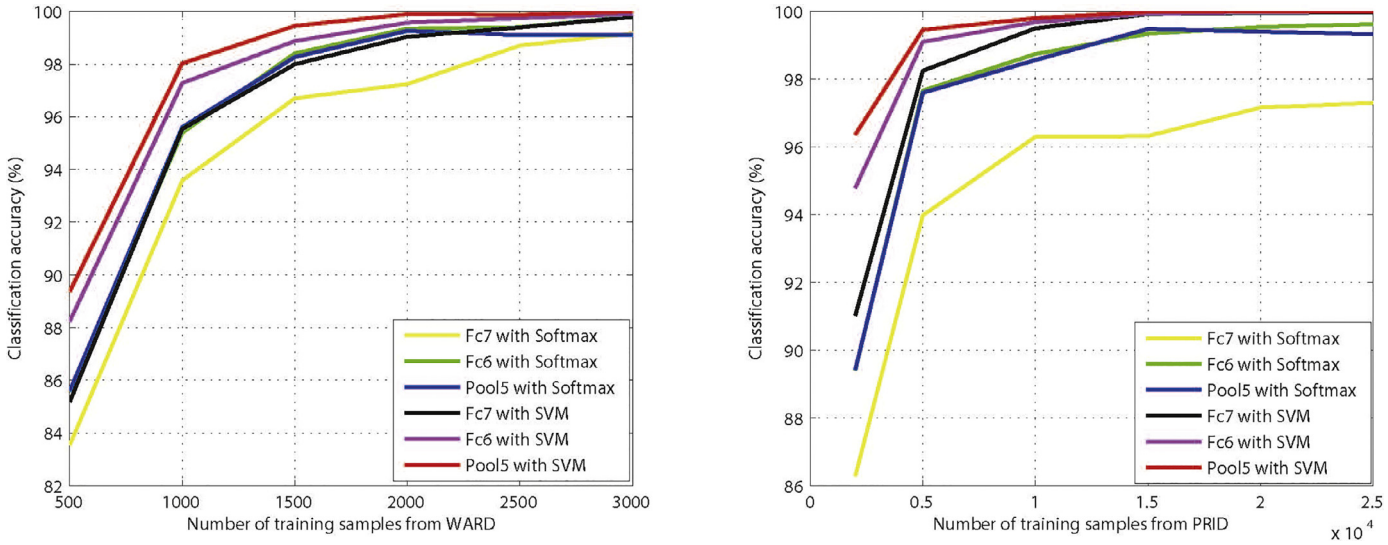


Fig. 8. Classification performance of different layer features and different classifiers.

Table 5
Classification accuracy of different layer features on PRID dataset with SVM classifiers.

Features	Number of training samples					
	2000	5000	10,000	15,000	20,000	25,000
fc7	91.02	98.25	99.50	99.92	99.95	99.98
fc6	94.79	99.10	99.68	99.93	99.98	99.99
pool5	96.36	99.45	**99.80**	**99.99**	**100**	**100**
pool5_fc6	96.41	99.45	**99.80**	**99.99**	**100**	**100**
pool5_fc7	96.37	99.45	**99.80**	99.98	**100**	**100**
fc6_fc7	94.68	99.10	99.69	99.94	99.98	99.99
pool5_fc6_fc7	96.44	99.46	**99.80**	**99.99**	**100**	**100**
Ours	**96.71**	**99.52**	99.79	**99.99**	**100**	**100**

As shown in Fig. 8, we compared the combinational classification performance of different layer features and different classifiers, i.e., end-to-end softmax and off-the-shelf SVMs. Note fc7+softmax, fc6+softmax, and pool5+softmax correspond to our 8-layer, 7-layer, and 6-layer networks, respectively. We found that for the same layer feature, the classification accuracy of SVM was higher than softmax. This observation is consistent with the claims by Alalshekmubarak et al. [48] and Tang [49] that SVM is a superior approach in terms of classifier. In our case, we think it's because SVM maps its input (CNN features) to some high dimensional space where (hopefully) the fine-grained differences between pedestrains can be revealed. We also observed that the combination of the pool5 feature and SVM classifier achieved the best performance among all combinations.

To further exploit deep features, we employed feature fusion on multi-layer deep features to produce more robust feature representations. The classification accuracy rates of different fused features are displayed in Tables 4 and 5 (see the last 5 rows in each table). Pool5_fc6, pool5_fc7, fc6_fc7, and pool5_fc6_fc7 represent simply fused features of 2 or 3 single layer features (i.e., concatenating multiple feature vectors directly). Meanwhile "Ours" represents fused features of pool5, fc6, and fc7 using our proposed feature fusion method. We set $\alpha_5 = 0.7$, $\alpha_6 = 0.2$, and $\alpha_7 = 0.1$ in our experiments, which are optimized using a grid search algorithm. To be specific, we varied α_5 from 0.5 to 0.9, α_6 from 0 to 0.5 and α_7 from 0 to 0.5 with an interval of 0.1, while subject to the constraint that $\alpha_5 + \alpha_6 + \alpha_7 = 1$. Then search for the optimal configuration. From Tables 4 and 5, we can see that the fused features generally outperform single layer features. Furthermore, the performance of fused features are closely related to the single layer features they contained; for example, pool5_fc6 and pool5_fc6_fc7 features have the best performance among simply fused features. Through comparing classification performance of different fused features on both WARD and PRID datasets, we proved that our proposed feature fusion method could further improve classification accuracy effectively.

4.4. Comparing with other methods

To facilitate comparison with other methods, we selected a subset of the PRID database for experiments in this section. The PRID subset contains 200 pedestrians as described above, but each person contains at most 30 image samples under each camera view, and therefore, there are a total of 11,463 samples. We also conducted the comparison experiments on the MARS dataset [47], a more challenging person re-identification dataset that contains images of 1261 different pedestrians from 6 cameras. Among all the six cameras, the camera-1 and camera-2 captured the most and the second-most pedestrians and totally 833 pedestrians are captured by both of them. Besides, the same person captured by camera-1 and camera-2 undergoes significant variation in viewpoints, poses, occlusions and background (see Fig. 9), which reflect most of the challenges in real-world person re-identification applications. For each of these 833 pedestrians, 30 image sequences were chose at most under each of the two cameras, resulting a total of 49,494 image samples for experiments.

We compared our method with three competitive person re-identification methods, i.e., LOMO_XQDA [16], MLAPG [50] and CaffeNet_2stream [29]. We also compared our deep features with other feature representations, i.e., PCANet [51] and Color_LBP [52]. The abilities to express pedestrian images of all compared features were tested by SVM classifiers. Note that the modeling and testing methods adopted by LOMO_XQDA, MLAPG and CaffeNet_2stream are different from those of general classification problems. For fair comparison, we divided the experimental data into 3 parts: the training set, test set, and gallery set, where the training set consists of $\mathscr{X}$ ($\mathscr{X} \in \{1, 2, 3, 4\}$) samples of each pedestrian under each camera, the gallery set consists of 1 sample of each pedestrian under each camera, and the test set consists of all remaining samples. During the test phase of LOMO_XQDA, MLAPG and CaffeNet_2stream, we matched each test sample of camera 1 to the gallery set of camera 2, and computed the matching accuracy. We

Fig. 9. Example images captured by camera-1 (row 1) and camera-2 (row 2) from the MARS dataset. Images in the same column represent the same person.

Table 6
Recognition accuracy of different methods on the PRID dataset.

Method	Number of training samples of each pedestrian under each camera			
	1	2	3	4
LOMO_XQDA [16]	81.29	91.41	94.08	94.50
MLAPG [50]	79.26	89.91	93.32	93.94
CaffeNet_2stream [29]	75.32	92.03	95.97	**98.75**
PCANet [51]	70.57	85.63	91.99	94.85
Color_LBP [52]	52.16	65.62	74.86	79.37
pool5	80.10	92.20	96.31	97.97
pool5_fc6	80.41	92.33	96.35	97.98
pool5_fc6_fc7	80.55	92.39	96.43	98.02
Ours	**81.81**	**93.27**	**96.76**	98.10

Table 7
Recognition accuracy of different methods on the MARS dataset.

Method	Number of training samples of each pedestrian under each camera			
	1	2	3	4
LOMO_XQDA [16]	54.17	61.61	62.08	62.39
MLAPG [50]	61.41	77.78	83.11	86.02
CaffeNet_2stream [29]	72.15	86.08	90.25	93.12
PCANet [51]	60.71	76.72	84.48	88.52
Color_LBP [52]	47.80	64.19	71.88	77.85
pool5	76.86	88.36	93.33	95.42
pool5_fc6	77.17	88.58	93.48	95.50
pool5_fc6_fc7	77.25	88.61	93.52	95.53
Ours	**80.03**	**90.10**	**94.39**	**96.21**

also matched each test sample of camera 2 to the gallery set of camera 1, and computed the matching accuracy. Then, we reported the mean of these 2 accuracies. Following [29], we set batch size to 128 and initialized the learning rate as 0.001 to train the CaffeNet_2stream (i.e., two-stream CaffeNet) models with 155 training epochs. But the networks didn't converge using the above settings when the number of training samples of each pedestrian under each camera was less than four on PRID dataset. For these cases that training samples was extremely limited, the learning rate was set to 0.0001 and the number of training epochs was set to 250 for better convergence. The experimental results of all methods on the PRID and MARS datasets are displayed in Tables 6 and 7, respectively. Our method achieved higher recognition accuracy compared with other pedestrian recognition methods on both the PRID and MARS datasets. This is because LOMO_XQDA and MLAPG are metric learning based methods. The learned models embed only the mapping relationship between two cameras, while the appearance changes of pedestrians, especially pose and occlusion changes across different camera-views, are complex and diverse among different pedestrians. Thus a shared cross-camera mapping is insufficient in a sense. On the other hand, our classification method takes into consideration the feature distribution of all classes across camera views thus embed more discriminative information. Note that CaffeNet_2stream also use a pre-trained

CNN network but in a two-stream architecture that simultaneously computes the identification loss and verification loss. This method was suffered from limited training samples compared with ours. Furthermore, our deep features consistently outperforms other feature representation methods. This is because the PCANet model and the traditional Color_LBP extractor are incompetent at dealing with cross-view feature distortion, while the multi-layer nonlinear structure of the CNN model is conducive to learn discriminative information for producing more robust feature representations. Particularly, recognition accuracy was further improved through effectively fusing multi-layer deep features.

5. Conclusions

In this paper, we have formulated pedestrian recognition as a classification problem, and proposed to improve classification accuracy from both feature modeling and classifier training. To overcome the problem of inadequate number of labeled samples in the real surveillance scene for training a deep network, we have adopted transfer learning to transfer and update the parameters of a pre-trained network to our target database. We have also constructed new target network structures to extract multi-level deep features for pedestrian recognition, and proposed to fuse multi-level deep features using feature visualization and our feature fusion method. By thoroughly evaluating classification performance of SVM and softmax classifiers on different features, we have identified the best single layer deep feature, i.e., the pool5 feature. Additionally, comprehensive experimental results have demonstrated that our feature fusion approach can effectively improve the robustness of feature representation and achieve higher classification performance. Note that the feature extraction and the off-the-shelf classifier training are two steps in our approach. For the future work, we will explore alternative feature fusion methods such as skipping connections to jointly improve the fused feature with other layers during the training stage and implement pedestrian recognition in an end-to-end manner. Furthermore, as we assume that the queried pedestrian is a known subject that has registered in the recognition system beforehand, it is a practical issue that the same person may change the dress in different days in the real application scenarios. In this case, the recognition system needs to be updated accordingly in order to maintain its performance.

Acknowledgments

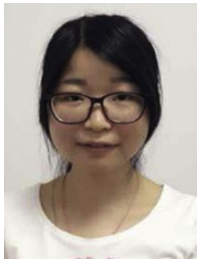
We thank LetPub for its linguistic assistance during the preparation of this manuscript. This research was supported by the Joint Funds of the National Natural Science Foundation of China under Grant No. U1609215, by the National Natural Science Foundation of China under Grant No. 61672460, by Key Program of Zhejiang Province under Grant No. LZ16F020002, and by the National Key R&D Program of China (2017YFB1401300, 2017YFB1401304).

References

[1] D. Gray, H. Tao, Viewpoint invariant pedestrian recognition with an ensemble of localized features, in: Proceedings of the European Conference on Computer Vision, 2008, pp. 262–275.
[2] M. Farenzena, L. Bazzani, A. Perina, V. Murino, M. Cristani, Person re-identification by symmetry-driven accumulation of local features, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2010, pp. 2360–2367.
[3] B. Ma, Y. Su, F. Jurie, Local descriptors encoded by fisher vectors for person re-identification, in: Proceedings of the European Conference on Computer Vision Workshops, 2012, pp. 413–422.
[4] B. Ma, Y. Su, F. Jurie, Covariance descriptor based on bio-inspired features for person re-identification and face verification, Image Vision Comput. 32 (6) (2014) 379–390.
[5] U. Park, A.K. Jain, I. Kitahara, K. Kogure, Vise: Visual search engine using multiple networked cameras, in: Proceedings of the International Conference on Pattern Recognition, 2006, pp. 1204–1207.
[6] N. Gheissari, T.B. Sebastian, R. Hartley, Person reidentification using spatiotemporal appearance, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2006, pp. 1528–1535.
[7] O. Hamdoun, F. Moutarde, B. Stanciulescu, B. Steux, Person re-identification in multi-camera system by signature based on interest point descriptors collected on short video sequences, in: Proceedings of the IEEE International Conference on Data Science in Cyberspace, 2008, pp. 1–6.
[8] D.S. Cheng, M. Cristani, M. Stoppa, L. Bazzani, V. Murino, Custom pictorial structures for re-identification, in: Proceedings of the British Machine Vision Conference, 2011, pp. 1–11.
[9] S. Bak, E. Corvee, F. Brémond, M. Thonnat, Person re-identification using haar-based and dcd-based signature, in: Proceedings of the 7th IEEE International Conference on Advanced Video and Signal Based Surveillance, 2010, pp. 1–8.
[10] R. Zhao, W. Ouyang, X. Wang, Unsupervised salience learning for person re-identification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013a, pp. 3586–3593.
[11] R. Zhao, W. Ouyang, X. Wang, Person re-identification by salience matching, in: Proceedings of the International Conference on Computer Vision, 2013b, pp. 2528–2535.
[12] A.S. Razavian, H. Azizpour, J. Sullivan, S. Carlsson, Cnn features off-the-shelf: an astounding baseline for recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2014, pp. 512–519.
[13] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, Y. LeCun, Overfeat: integrated recognition, localization and detection using convolutional networks, arXiv:1312.6229, (2013).
[14] B. Prosser, W. Zheng, S. Gong, T. Xiang, Q. Mary, Person re-identification by support vector ranking, in: Proceedings of the British Machine Vision Conference, 2010, pp. 21.1–11.
[15] W. Zheng, S. Gong, T. Xiang, Person re-identification by probabilistic relative distance comparison, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2011, pp. 649–656.
[16] S. Liao, Y. Hu, X. Zhu, S.Z. Li, Person re-identification by local maximal occurrence representation and metric learning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 2197–2206.
[17] X. Yang, M. Wang, D. Tao, Person re-identification with metric learning using privileged information, IEEE Trans. Image Process. 27 (2) (2018) 791–805.
[18] Q. Wang, J. Wan, Y. Yuan, Locality constraint distance metric learning for traffic congestion detection, Pattern Recognit. 75 (2018) 272–281.
[19] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Proceedings of the Conference on Neural Information Processing Systems, 2012, pp. 1–8.
[20] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: Proceedings of the International Conference on Learning Representations, 2015, pp. 1–14.
[21] Y. Yuan, Z. Xiong, Q. Wang, An incremental framework for video-based traffic sign detection, tracking, and recognition, IEEE Trans. Intell. Transp. Syst. 18 (7) (2017) 1918–1929.
[22] Y. Yuan, J. Fang, Q. Wang, Incrementally perceiving hazards in driving, Neurocomputing 282 (2018) 202–217.
[23] L.F. Teixeira, C.R. Luis, Video object matching across multiple independent views using local descriptors and adaptive learning, Pattern Recognit. Lett. 30 (2) (2009) 157–167.
[24] H. Wang, X. Wang, J. Zheng, J.R. Deller, H. Peng, L. Zhu, W. Chen, X. Li, R. Liu, H. Bao, Video object matching across multiple non-overlapping camera views based on multi-feature fusion and incremental learning, Pattern Recognit. 47 (12) (2014) 3841–3851.
[25] H. Wang, Y. Yan, J. Hua, Y. Yang, X. Wang, X. Li, J.R. Deller, G. Zhang, H. Bao, Pedestrian recognition in multi-camera networks using multilevel important salient feature and multicategory incremental learning, Pattern Recognit. 67 (7) (2017) 340–352.
[26] E. Ahmed, M. Jones, T.K. Marks, An improved deep learning architecture for person re-identication, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 3908–3916.
[27] D. Yi, Z. Lei, S. Liao, S.Z. Li, Deep metric learning for practical person re-identication, in: Proceedings of the IEEE International Conference on Pattern Recognition (ICPR), 2014, pp. 34–39.
[28] D. Chung, K. Tahboub, E.J. Delp, A two stream siamese convolutional neural network for person re-identification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 1983–1991.
[29] Z. Zheng, L. Zheng, Y. Yang, A discriminatively learned CNN embedding for person re-identification, ACM Trans. Multimedia Comput., Commun. Appl. 14 (1) (2017) 13.
[30] D. Cheng, Y. Gong, S. Zhou, J. Wang, N. Zheng, Person re-identification by multi-channel parts-based CNN with improved triplet loss function, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 1335–1344.
[31] A. Hermans, L. Beyer, B. Leibe, In defense of the triplet loss for person re-identification, arXiv:1703.07737, (2017).
[32] M. Geng, Y. Wang, T. Xiang, Y. Tian, Deep transfer learning for person re-identification, arXiv:1611.05244 (2016).
[33] Y. Sun, L. Zheng, W. Deng, S. Wang, Svdnet for pedestrian retrieval, in: Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017.

[34] H. Liu, J. Feng, M. Qi, J. Jiang, S. Yan, End-to-end comparative attention networks for person re-identification, arXiv:1606.04404(2016a).
[35] J. Liu, Z.-J. Zha, Q. Tian, D. Liu, T. Yao, Q. Ling, T. Mei, Multi-scale triplet cnn for person re-identification, in: Proceedings of the 2016 ACM on Multimedia Conference, ACM, 2016b, pp. 192–196.
[36] C. Vondrick, A. Khosla, T. Malisiewicz, A. Torralba, Hoggles: visualizing object detection features, in: Proceedings of the IEEE International Conference on Computer Vision, 2013, pp. 1–8.
[37] C. Ma, J.B. Huang, X. Yang, M.H. Yang, Hierarchical convolutional features for visual tracking, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 3074–3082.
[38] M.D. Zeiler, R. Fergus, Visualizing and understanding convolutional networks, in: Proceedings of the European Conference on Computer Vision, 2014, pp. 818–833.
[39] D. Turcsany, A. Bargiela, T. Maul, Local receptive field constrained deep networks, Inf. Sci. 349–350 (7) (2016) 229–247.
[40] N. Martinel, C. Micheloni, C. Piciarelli, Distributed signature fusion for person re-identification, in: Proceedings of the 6th International Conference on Distributed Smart Cameras, 2012, pp. 1–6.
[41] J. Yosinski, J. Clune, Y. Bengio, H. Lipson, How transferable are features in deep neural networks? in: Advances in Neural Information Processing Systems, 2014, pp. 3320–3328.
[42] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: towards real-time object detection with region proposal networks, IEEE Trans. Pattern Anal. Mach. Intell. 39 (6) (2017) 1137–1149.
[43] L.M. Zintgraf, T.S. Cohen, T. Adel, M. Welling, Visualizing deep neural network decisions: prediction difference analysis, arXiv:1702.04595(2017).
[44] M. Liu, J. Shi, Z. Li, C. Li, J. Zhu, S. Liu, Towards better analysis of deep convolutional neural networks, IEEE Trans. Visual. Comput. Graphics 23 (1) (2017) 91–100.
[45] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2015, pp. 3431–3440.
[46] M. Hirzer, C. Beleznai, P.M. Roth, H. Bischof, Person re-identification by descriptive and discriminative classification, in: Proceedings of the Scandinavian Conference on Image Analysis, 2011, pp. 1–6.
[47] L. Zheng, Z. Bie, Y. Sun, J. Wang, C. Su, S. Wang, Q. Tian, Mars: a video benchmark for large-scale person re-identification, in: Proceedings of the European Conference on Computer Vision, 2016, pp. 868–884.
[48] A. Alalshekmubarak, L.S. Smith, A novel approach combining recurrent neural network and support vector machines for time series classification, in: Proceedings of the IEEE International Conference on Innovations in Information Technology (IIT), 2013, pp. 42–47.
[49] Y. Tang, Deep learning using linear support vector machines, arXiv:1306.0239(2013).
[50] S. Liao, S.Z. Li, Efficient PSD constrained asymmetric metric learning for person re-identification, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 3685–3693.
[51] T.H. Chan, K. Jia, S. Gao, J. Lu, Z. Zeng, Y. Ma, Pcanet: a simple deep learning baseline for image classification? IEEE Trans. Image Process. 24 (12) (2014) 5017–5032.
[52] F. Xiong, M. Gou, O. Camps, M. Sznaier, Person re-identification using kernel-based metric learning methods, in: Proceedings of the European Conference on Computer Vision, 2014, pp. 1–16.

Jing-Tao WANG was born in 1995. He is pursuing his Bachelor's degree in computer science and technology in Harbin Institute of Technology. His research interests include video/image processing and Visualization.



Guo-Li YAN was born in 1991. She is pursuing her Master's degree in computer science and technology in Zhejiang Gongshang University. Her interests include video/image processing and pattern recognition.

Hui-Yan WANG was born in Yantai, China. She received the M.S. degree in power engineering from Shandong University, Jinan, China and the Ph.D. degree in electrical engineering from Zhejiang University, Hangzhou, China, in 1999 and 2003, respectively. She was a postdoctoral research fellow (2003–2005) in clinical medicine from pharmaceutical informatics institute, Zhejiang University, Hangzhou, China. She is currently a professor of Computer Science and Technology in the school of Computer Science and Information Engineering, Zhejiang Gongshang University, China. Her research interests include biomedical signal processing, pattern recognition, and image processing.

Jing HUA is a professor of Computer Science and Technology in the school of Computer Science and Information Engineering. Dr. Hua received his Ph.D. degree (2004) in Computer Science from the State University of New York at Stony Brook. He received his M.S. degree (1999) in Pattern Recognition and Artificial Intelligence from the Institute of Automation, Chinese Academy of Sciences in Beijing, China and his B.S. degree (1996) in Electrical Engineering from the Huazhong University of Science & Technology in Wuhan, China. His research interests include Computer Graphics, Visualization, Image Analysis and Informatics, Computer Vision, etc. He has published over 100 peer-reviewed papers in the above research fields at top journals and conferences, such as IEEE Transactions on Visualization and Computer Graphics, IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Visualization, MICCAI, CVPR, ICDM, etc.