Person Re-Identification by Multi-Channel Parts-Based CNN with Improved Triplet Loss Function

De Cheng, Yihong Gong, Sanping Zhou, Jinjun Wang, Nanning Zheng
Institute of Artificial Intelligence and Robotics
Xi'an Jiaotong University,Xi'an, Shaanxi, P.R. China

Abstract

Person re-identification across cameras remains a very challenging problem, especially when there are no overlapping fields of view between cameras. In this paper, we present a novel multi-channel parts-based convolutional neural network (CNN) model under the triplet framework for person re-identification. Specifically, the proposed CNN model consists of multiple channels to jointly learn both the global full-body and local body-parts features of the input persons. The CNN model is trained by an improved triplet loss function that serves to pull the instances of the same person closer, and at the same time push the instances belonging to different persons farther from each other in the learned feature space. Extensive comparative evaluations demonstrate that our proposed method significantly outperforms many state-of-the-art approaches, including both traditional and deep network-based ones, on the challenging i-LIDS, VIPeR, PRID2011 and CUHK01 datasets.

1. Introduction

Person re-identification is the problem of matching the same individuals across multiple cameras, or across time within a single camera. It is attracting rapidly increased attentions in the computer vision and pattern recognition research community due to its importance for many applications such as video surveillance, human-computer interaction, robotics, content-based video retrieval, etc. Despite years of efforts, person re-id remains challenging due to the following reasons: 1) dramatic variations in visual appearance and ambient environment caused by different viewpoints from different cameras; 2) significant changes in human pose across time and space; 3) background clutter and occlusions; and 4) different individuals that share similar appearances. Moreover, with little or no visible faces, in many cases the use of biometric and soft-biometric approaches is not applicable. Figure 1 illustrates some examples of the matched pairs in four challenging person re-id benchmark datasets i-LIDS [38],VIPeR[13],PRID2011[17] and CUHK01[24]. Images in each red bounding box are from the same person.

Figure 1. Matched examples in datasets i-LIDS, VIPeR, CUHK01 and PRID2011. Each row shows matched examples from the same dataset. Images in a red bounding box contain the same person.

Given a query person's image, in order to find the correct matches among a large set of candidate images captured by different cameras, two crucial problems must be addressed. First, good image features are required to represent both the query and the gallery images. Second, suitable distance metrics are indispensable to determine whether a gallery image contains the same individual as the query image. Many existing studies consider the two problems separately and have focused more on the first one, that is, developing more discriminative and robust feature representations to describe a person's visual appear-

ance [14, 32, 10, 40, 19, 7, 2, 4, 8]. Once the feature extraction is completed, these methods usually choose a standard distance measure such as l_1-norm based distance (L1norm) [40], Bhattacharyya distance (Bhat)[14], or Mahalanobis distance (Maha)[33] to determine the similarity between pairs.

The situation has motivated us to consider the feature and distance metric learning problems jointly to improve the person re-id performance. To extract better features for raw person images, we propose a new, multi-channel CNN model that learns features for both the input person's full body and the body parts. The full body and body parts features are concatenated together and fed into the top full-connection layer to produce the final representation of the input person. We also borrow the idea from Wang's *et al*. [39] and the FaceNet work [34] to use triplet training examples and the improved triplet loss function to further enhance the discriminative power of the learned features. In contrast to the original triplet loss function that only requires the intra-class feature distances to be less than the inter-class ones, the improved loss function further requires the intra-class feature distances to be less than a predefined margin. Our experimental evaluations show that the use of the improved triplet loss function alone can improve the person re-id accuracy by up to 4%, compared to the same DCNN model using the original triplet loss function.

Given a person's image, the proposed CNN model outputs an 800 dimension feature representation of the input image. The proposed CNN model together with the improved triplet loss function can be considered as learning a mapping function that maps each *raw image* into a feature space where the difference between images of the same person is less than that of different persons. Therefore, the proposed framework can learn the optimal feature and distance metric jointly for the person re-id task.

The main contributions of this paper are twofold: 1) a novel, multi-channel CNN model that learns both the global full-body and the local parts features, and integrates them together to produce the final feature representation of the input person; 2) an improved triplet loss function that requires the intra-class feature distances to be less than not only the inter-class ones, but also a predefined threshold. Experimental evaluations results show that the proposed method achieves the state-of-the-art performances on several widely adopted person re-id benchmark test datasets.

2. Related Work

Typical person re-id systems consist of two major components: a feature extraction method to describe the query image and the gallery images, and a distance metric for comparing those features across images. Research on person re-id problems usually focuses either on constructing robust and discriminative features, or finding an improved similarity metric for comparing features, or a combination of both.

There are a great amount of research efforts for developing better features that are at least partially invariant to lighting, pose, and viewpoint variations. Features that have been used for the person re-id task include color histograms and their variants [41, 20, 21, 28, 23, 46], local binary patterns(LBP) [41, 20, 21, 28, 23, 46], Gabor features [23], color names [44], and other visual appearance or contextual cues [3]. Quite some works have also investigated combinations of multiple visual features, including [41, 20, 23].

A large number of metric learning and ranking algorithms have also been applied to the person re-id problem [43, 31]. The basic idea behind metric learning is to find a mapping function from the feature space to the distance space with certain merits, such as feature vectors from the same person being closer than those from different ones. These metric learning methods mainly include Mahalanobis metric learning(KISSME) [21], Local Fisher Discriminant Analysis(LFDA) [41], Marginal Fisher Analysis(MFA) [41], large margin nearest neighbour (LMNN)[41], Locally Adaptive Decision Functions(LADF) [26], and attribute consistent matching [20].

Inspired by the great success of deep learning networks in various computer vision and pattern recognition tasks [22, 11, 36, 37, 16], it becomes increasingly popular to apply deep convolution neural network(DCNN) models to the person re-id problem. It is worth noting that, recent state-of-the-art performances on widely used person re-id benchmark datasets, such as i-LIDS, VIPeR, CUHK01, etc, are all obtained by DCNN-based methods. In the following, we briefly introduce those deep learning based approaches related to, or to be compared with our work. Wang *et al*. [39] used triplet training examples and the triplet loss function to learn fine grained image similarity metrics. FaceNet [34] and Ding *et al*. [6] applied this triplet framework to the face and person re-identification problems, respectively. In this paper, we also borrow the idea from [39] and propose an improved triplet loss function for the person re-id task. DeepReID[25] proposed a novel Filter Pairing Neural Network (FPNN) that jointly handles the problems of misalignment, photometric and geometric transforms, occlusion and black cluster, etc, by using the patch matching layers to match the filter responses of local patches across views, and other convolution and max-pooling layers to model body parts displacements. mFilter [48] also used the local patch matching method that learns the mid-level filters to get the local discriminative features for the person re-id task. Ahmed *et al*. [1] proposed an improved deep learning architecture which takes pair-wise images as its inputs, and outputs a similarity value indicating whether the two input images depict the same person or not. Novel elements in their model include a layer that

computes cross-input neighborhood differences to capture local relationships between the two input images based on their mid-level features, and a patch summary layer to get high-level features. Yi *et al*. [45] constructed a siamese neural network (denoted as DeepM in our paper) to learn pairwise similarity, and also used body parts to train their CNN models. In their work, person images are cropped into three overlapped parts which are used to train three independent networks. Finally the three networks are fused at the score level.

Our CNN model differs from the above deep network based approaches in both the network architecture and the loss function. More specifically, We use a single network that consists of multiple channels to learn both the global full-body and local body-parts features. We use different convolution kernel sizes in different types of channels to look at full-body and body-parts with different resolutions, which is similar to the idea of the root/part filters in a DPM model [9]. In addition, we use an improved triplet loss function to make the features from the same person closer, meanwhile features from different persons farther away from each other. In Section 4, performance comparisons with some of the above methods will be made in our experimental evaluations.

3. The Proposed Person Re-Id Method

In this section, we present the proposed person re-id method in details. We first describe the overall framework of our person re-id method, then elaborate the network architecture of the proposed multi-channel CNN model. Finally, we present the improved triplet loss function used to train the proposed CNN model.

3.1. The Overall Framework

As illustrated in figure 2, similar to the works in [39, 34], the proposed person re-id method uses triplet examples to train the network. Denote by $I_i =< I_i^o, I_i^+, I_i^- >$ the three input images forming the i-th triplet, where I_i^o and I_i^+ are from the same person, while I_i^- is from a different person. Through the three CNNs that share the parameter set $\mathbf{w}$, i.e., weights and biases, we map triplets I_i from the raw image space into a learned feature space, where I_i is represented as $\phi_{\mathbf{w}}(I_i) =< \phi_{\mathbf{w}}(I_i^o), \phi_{\mathbf{w}}(I_i^+), \phi_{\mathbf{w}}(I_i^-) >$. Each CNN in the figure is a proposed multi-channel CNN model that is able to extract both the global full-body and local body-parts features. When the proposed CNN model is trained using the improved triplet loss function, the learned feature space will have the property that the distance between $\phi_{\mathbf{w}}(I_i^o)$ and $\phi_{\mathbf{w}}(I_i^+)$ is less than not only the distance between $\phi_{\mathbf{w}}(I_i^o)$ and $\phi_{\mathbf{w}}(I_i^-)$, but also a predefined margin. The improved loss function aims to pull the instances of the same person closer, and at the same time push the instances belonging to different persons farther from each other in the learned feature space.

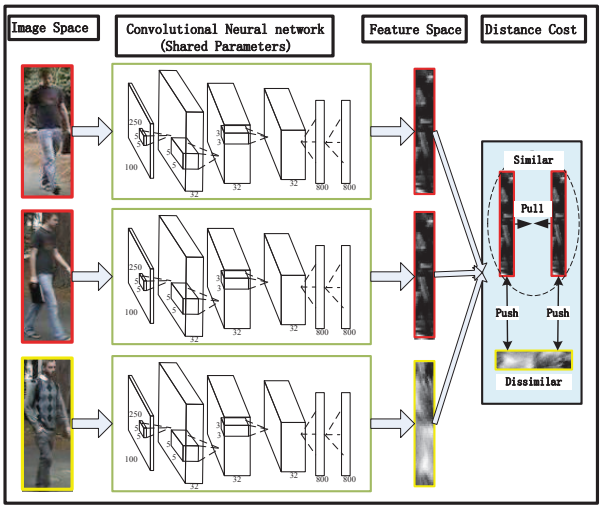

Figure 2. Triplet training framework. Triplet training images are fed into three network models with the shared parameter set. The triplet loss function is used to train the network models, which makes the distance between the matched pairs less than not only a predefined threshold, but also that of the mismatched pairs in the learned feature space.

3.2. Multi-Channel Parts-based CNN Model

The proposed multi-channel CNN model mainly consists of the following distinct layers: one global convolution layer, one full-body convolution layer, four body-part convolution layers, five channel-wise full connection layers, and one network-wise full connection layer. As shown in Figure 3, the global convolution layer is the first layer of the proposed CNN model. It consists of 32 feature maps with the convolution kernel of $7 \times 7 \times 3$ and the stride of 3 pixels. Next, this global convolution layer is divided into four equal parts $P_i, i = \{1, \ldots, 4\}$, and each part P_i forms the first layer of an independent body-part channel that aims to learn features for the respective body part. A full-body channel with the entire global convolution layer as its first layer is also established to learn global full-body features of the input persons. The four body-part channel together with the full-body channel constitute five independent channels that are trained separately from each other.

The full-body channel is configured as follows: The global convolution layer, max pooling, the full-body convolution layer, another max pooling, and a full-connection layer. The kernel size for max pooling is 3×3, and the full-connection layer generates an output of 400 dimensions. The four body-part channels have the same configuration as follows: The copy of one of the four equally divided parts of the global convolution layer, the body-part convolution

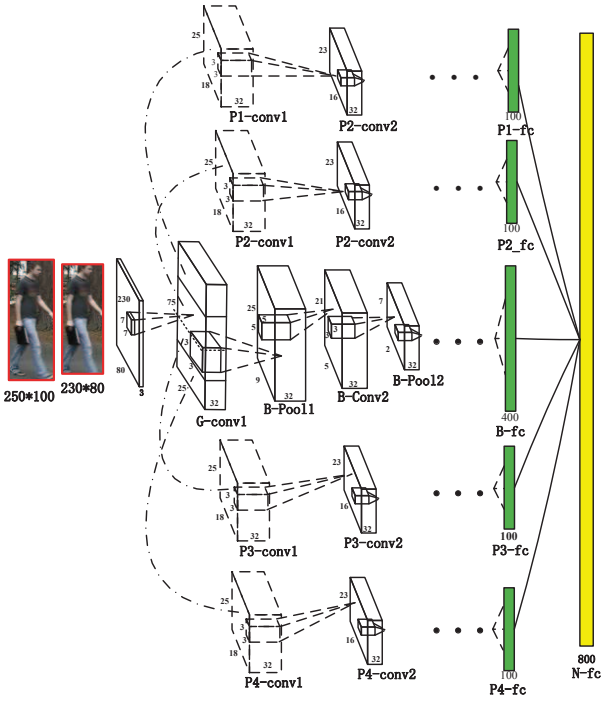

Figure 3. Network Architecture of the proposed multi-channel CNN model. The first layer is called the global convolution layer donated as G-conv1. It is then divided into four equal parts, denoted as Pi-conv1, where $i = \{1, \ldots, 4\}$. Each Pi-conv1 forms the first layer of an independent body-part channel, which is followed by a body-part convolution layer denoted as Pi-conv2, and a channel-wise full connection layer denoted as Pi-fc. The full-body channel consists of max pooling of G-conv1 denoted as B-pool1, full-body convolution layer denoted as B-conv2, another max pooling denoted as B-pool2, and a channel-wise fully connection layer denoted as B-fc. The network-wise full connection layer is denoted as N-fc.

layer, no max pooling, and a full-connection layer. The full-connection layer generates an output of 100 dimensions. Because the full-body convolution layer and the four body-part convolution layers aim to learn the global full-body and the local body-parts features, respectively, we use the convolution size of 5×5 for the former and a smaller size of 3×3 for the latter. This serves to learn finer grain local features for persons' body parts. Both types of convolution layers use the stride of 1. Note that all the convolution layers in our CNN model contain a relu layer to produce their outputs.

The above network configuration achieves state-of-the-art person re-id accuracies on relatively small benchmark datasets. In our experiments, we found that for some larger datasets such as CUHK01, constructing each of the five separate channels with two convolution layers lead to a much better result. Therefore, we use two network configurations to handle small and large benchmark datasets, respectively. The two network configurations are mostly the same except for the number of convolution layers (one or two) in each separate channel.

At the final stage, the outputs of the channel-wise full connection layers from the five separate channels are concatenated into one vector, and is fed into the final network-wise full connection layer. The multi-channel structure described above enables learning of the global full-body and local body-parts features jointly, and the fusion of these two types of features at the final stage leads to remarkable improvements of person re-id accuracies.

3.3. Improved Triplet Loss Function

As described in 3.1, we use triplet examples to train the network model. Given a triplet $I_i =< I_i^o, I_i^+, I_i^- >$, the network model maps I_i into a learned feature space with $\phi_{\mathbf{w}}(I_i) =< \phi_{\mathbf{w}}(I_i^o), \phi_{\mathbf{w}}(I_i^+), \phi_{\mathbf{w}}(I_i^-) >$. The similarities between the triplet images I_i^o, I_i^+, I_i^- are measured by the L_2-norm distances between $\phi_{\mathbf{w}}(I_i^o), \phi_{\mathbf{w}}(I_i^+), \phi_{\mathbf{w}}(I_i^-)$. The original triplet loss function requires that distance of the pair $(\phi_{\mathbf{w}}(I_i^o), \phi_{\mathbf{w}}(I_i^-))$ be larger than that of the pair $(\phi_{\mathbf{w}}(I_i^o), \phi_{\mathbf{w}}(I_i^+))$ by a predefined margin, and uses the following equation to enforce this requirement:

$$d^n(I_i^o, I_i^+, I_i^-, \mathbf{w}) = \\ d(\phi_{\mathbf{w}}(I_i^o), \phi_{\mathbf{w}}(I_i^+)) - d(\phi_{\mathbf{w}}(I_i^o), \phi_{\mathbf{w}}(I_i^-)) \leq \tau_1. \quad (1)$$

In the equation τ_1 is negative. However, since this loss function does not stipulate how close the pair $(\phi_{\mathbf{w}}(I_i^o), \phi_{\mathbf{w}}(I_i^+))$ should be, as a consequence, instances belonging to the same person may form a large cluster with a relatively large average intra-class distance in the learned feature space. Clearly, this is not a desired outcome, and will inevitably hurt the person re-id performance.

Based on the above observation, we add a new term to the original triplet loss function to further require that distance of the pair $(\phi_{\mathbf{w}}(I_i^o), \phi_{\mathbf{w}}(I_i^+))$ be less than a second margin τ_2, and that τ_2 be much smaller than $|\tau_1|$. Translating this statement into equation, we have:

$$d^p(I_i^o, I_i^+, \mathbf{w}) = d(\phi_{\mathbf{w}}(I_i^o), \phi_{\mathbf{w}}(I_i^+)) \leq \tau_2. \quad (2)$$

The improved loss function aims to pull the instances of the same person closer, and at the same time push the instances belonging to different persons farther from each other in the learned feature space. This is more consistent with the principal used by many data clustering and discriminative analysis methods.

In summary, the improved triplet loss function is defined

as follows:

$$L(I,\mathbf{w}) = \frac{1}{N}\sum_{i=1}^{N}(\underbrace{\max\{d^n(I_i^o, I_i^+, I_i^-, \mathbf{w}), \tau_1\}}_{inter-class-constraint} + \underbrace{\beta \max\{d^p(I_i^o, I_i^+, \mathbf{w}), \tau_2\}}_{intra-class-constraint}), \quad (3)$$

where N is the number of triplet training examples, β is a weight to balance the inter-class and intra-class constraints. In our implementation, the distance function $d(.,.)$ is defined as the L_2-norm distance,

$$d(\phi_{\mathbf{w}}(I_i^o), \phi_{\mathbf{w}}(I_i^+)) = ||\phi_{\mathbf{w}}(I_i^o) - \phi_{\mathbf{w}}(I_i^+)||^2. \quad (4)$$

3.4. The Training Algorithm

We use the stochastic gradient decent algorithm to train the proposed CNN achitecture model with the improved triplet loss function. The derivatives of Eq.(3) can be computed as follows:

$$\frac{\partial L(I,w)}{\partial w} = \frac{1}{N}\sum_{i=1}^{N} h_1(I_i, w) + \frac{1}{N}\sum_{i=1}^{N} h_2(I_i, w) \quad (5)$$

$$h_1(I_i,\mathbf{w}) = \begin{cases} \dfrac{\partial d^n(I_i^o, I_i^+, I_i^-, \mathbf{w})}{\partial \mathbf{w}} & d^n(I_i^o, I_i^+, I_i^-, \mathbf{w}) > \tau_1 \\ 0 & d^n(I_i^o, I_i^+, I_i^-, \mathbf{w}) \leq \tau_1. \end{cases} \quad (6)$$

$$h_2(I_i,\mathbf{w}) = \begin{cases} \beta\dfrac{\partial d^p(I_i^o, I_i^+, \mathbf{w})}{\partial \mathbf{w}} & d^p(I_i^o, I_i^+, \mathbf{w}) > \tau_2 \\ 0, & d^p(I_i^o, I_i^+, \mathbf{w}) \leq \tau_2. \end{cases} \quad (7)$$

By the definitions of $d^n(I_i^o, I_i^+, I_i^-, \mathbf{w})$ and $d^p(I_i^o, I_i^+, \mathbf{w})$, we can obtain their gradients as follows,

$$\frac{\partial d^n}{\partial \mathbf{w}} = 2(\phi_{\mathbf{w}}(I_i^o) - \phi_{\mathbf{w}}(I_i^+))\frac{\partial \phi_{\mathbf{w}}(I_i^o) - \partial \phi_{\mathbf{w}}(I_i^+)}{\partial \mathbf{w}} - 2(\phi_{\mathbf{w}}(I_i^o) - \phi_{\mathbf{w}}(I_i^-))\frac{\partial \phi_{\mathbf{w}}(I_i^o) - \partial \phi_{\mathbf{w}}(I_i^-)}{\partial \mathbf{w}}. \quad (8)$$

$$\frac{\partial d^p}{\partial \mathbf{w}} = 2(\phi_{\mathbf{w}}(I_i^o) - \phi_{\mathbf{w}}(I_i^+))\frac{\partial \phi_{\mathbf{w}}(I_i^o) - \partial \phi_{\mathbf{w}}(I_i^+)}{\partial \mathbf{w}}, \quad (9)$$

From the above derivations, it is clear that the gradient on each input triplet can be easily computed given the values of $\phi_{\mathbf{w}}(I_i^o)$, $\phi_{\mathbf{w}}(I_i^+)$, $\phi_{\mathbf{w}}(I_i^-)$ and $\frac{\partial \phi_{\mathbf{w}}(I_i^o)}{\partial \mathbf{w}}$, $\frac{\partial \phi_{\mathbf{w}}(I_i^+)}{\partial \mathbf{w}}$, $\frac{\partial \phi_{\mathbf{w}}(I_i^-)}{\partial \mathbf{w}}$, which can be obtained by separately running the standard forward and backward propagations for each image in the triplet examples. As the algorithm needs to go though all the triplets in each batch to accumulate the gradients for each iteration, we call it the triplet-based stochastic gradient descent algorithm. Algorithm 1 shows the main procedures of the training algorithm.

Algorithm 1 Triplet-based stochastic gradient descent training algorithm

Input
  Training samples $\{I_i\}$
Output
  The network parameters $\{\mathbf{w}\}$
while $t < T$ **do**
  $t \leftarrow t + 1$
  $\frac{\partial L(I,\mathbf{w})}{\partial \mathbf{w}} = 0$
  for all training triplet samples I_i **do**
    Calculate $\phi_{\mathbf{w}}(I_i^o), \phi_{\mathbf{w}}(I_i^+), \phi_{\mathbf{w}}(I_i^-)$ by forward propagation;
    Calculate $\frac{\phi_{\mathbf{w}}(I_i^o)}{\partial \mathbf{w}}, \frac{\partial \phi_{\mathbf{w}}(I_i^+)}{\partial \mathbf{w}}, \frac{\partial \phi_{\mathbf{w}}(I_i^-)}{\partial \mathbf{w}}$ by back propagation;
    Calculate $\frac{\partial d^p}{\partial \mathbf{w}}$ and $\frac{\partial d^n}{\partial \mathbf{w}}$ according to Eq. 9 and 8;
    Calculate $\frac{\partial L(I,\mathbf{w})}{\partial \mathbf{w}}$ according to Eq. 5, 7, and 6.
  end for
  Update the parameters $\mathbf{w}^t = \mathbf{w}^{t-1} - \lambda_t \frac{\partial L(I,\mathbf{w})}{\partial \mathbf{w}}$.
end while

4. Experiments

4.1. Setup

Data augmentation: Data augmentation is an important mean for increasing the volume of training data, and for alleviating the over-fitting problem. In our implementation, we resize all the images into 100×250 pixels. During the training process, we crop a center region of 80×230 pixels with a small random perturbation from each image to augment the training data.

Setting training parameters: The weights are initialized from two zero-mean Gaussian distributions with the standard deviations of 0.01 and 0.001, respectively. The bias terms are set to 0. We generate the triplets as follows: For each batch of 100 instances, we select 5 persons and generate 20 triplets for each person in each iteration. In each triplet, the matched reference is randomly selected from the same class, and the mismatched one is also randomly selected, but from the remaining classes. In our experiments, the parameters τ_1, τ_2, β in Eq.(3) are set to -1, 0.01 and 0.002, respectively.

Datasets: We use four popular person re-id benchmark datasets, i-LIDS, PRID2011, VIPeR and CUHK01, for performance evaluations. All the datasets contain a set of persons, each of whom has several images captured by different cameras. The following is a brief description of these four datasets:

i-LIDS dataset: It is constructed from video images shooting a busy airport arrival hall. It contains 479 images from 119 persons, which are normalized to 128×64

pixels. Each person has four images in average. These images are captured by non-overlapping cameras, and are subject to large illumination changes and occlusions.

PRID2011 dataset: This dataset consists of images recorded by two static surveillance cameras. Camera view A and B contain 385 and 749 persons, respectively, with 200 persons appearing in both views.

VIPeR dataset: This dataset contains two views of 632 persons. Each pair for a person is captured by different cameras with different viewpoints, poses, and lighting conditions. It is one of the most challenging datasets for the person re-id task due to its huge variance and discrepancy.

CUHK01 dataset: This is a larger dataset for the person re-id task, which contains 971 persons captured from two camera views in a campus environment. Camera view A captures frontal or back views of a person while camera B captures the person's profile views. Each person has four images with two from each camera.

Evaluation protocol We adopt the widely used cumulative match curve (CMC) metric for quantitative evaluations. For each dataset, we randomly select about half of the persons for training, and the remaining half for testing. For datasets with two cameras, we randomly select one image of a person from camera A as a query image and one image of the same person from camera B as a gallery image. For multi-camera datasets, two images of the same individual are chosen: one is used as a query and the other as a gallery image. The gallery set comprises one image for each person. For each image in the query set, we first compute the distance between the query image and all the gallery images using the L2 distance with the features produced by the trained network, and then return the top n nearest images in the gallery set. If the returned list contains an image featuring the same person as that in the query image at k-th position, then this query is considered as success of rank k. We repeat the procedure 10 times, and use the average rate as the evaluation result.

4.2. Experimental Evaluations

Our proposed person re-id method contains two novel ingredients: 1) the multi-channel CNN model that is able to learn both the global full-body and the local body-parts features, 2) the improved triplet loss function that serves to pull the instances of the same person closer, and at the same time push the instances belonging to different persons farther from each other in the learned feature space. To reveal how each ingredient contributes to the performance improvement, we implemented the following four variants of the proposed person re-id method, and compared them with a dozen of representative methods in the literature:

Table 1. Experimental evaluations on i-LIDS dataset.

Method	Top1	Top5	Top10	Top15	Top20	Top30
Adaboost[14]	29.6	55.2	68.1	77.0	82.4	92.1
LMNN[41]	28.0	53.8	66.1	75.5	82.3	91.0
ITML[5]	29.0	54.0	70.5	81.0	86.7	95.0
MCC[12]	31.3	59.3	75.6	84.0	88.3	95.0
Xing's[42]	27.0	52.3	63.4	74.8	80.7	93.0
PLS[35]	22.1	46.0	60.0	70.0	78.7	87.5
L1norm[40]	30.7	55.0	68.0	75.0	83.0	90.0
Bhat.[14]	28.4	51.1	64.3	72.0	78.8	89.0
PRDC[49]	37.8	63.7	75.1	82.8	88.4	95.0
Sakrapee[31]	50.3	−−	−−	−−	−−	−−
Ding[6]	52.1	68.2	78.0	83.6	88.8	95.0
OursT	43.2	64.9	74.9	84.4	86.1	93.3
OursTC	47.3	69.8	80.1	88.6	90.4	95.3
OursTP	57.2	80.7	**90.9**	96.4	97.1	98.9
OursTCP	**60.4**	**82.7**	90.7	**96.4**	**97.8**	**99.3**

Table 2. Experimental evaluations on PRID2011 dataset.

Method	Top1	Top10	Top20	Top50	Top100
KISSME [21]	15.0	39.0	52.0	68.0	80.0
EIML[18]	16.0	39.0	51.0	68.0	81.0
LMNN[41]	10.0	30.0	42.0	59.0	73.0
LMNN-R[41]	9.0	32.0	43.0	60.0	76.0
ITML[5]	12.0	36.0	47.0	64.0	79.0
LDML[15]	2.0	6.0	11.0	19.0	32.0
Maha[33]	16.0	41.0	51.0	64.0	76.0
Euclidean[33]	3.0	10.0	14.0	28.0	45.0
Descr[17]	4.0	24.0	37.0	56.0	70.0
DeepM[45]	17.9	45.9	55.4	71.4	−−
Sakrapee[31]	17.9	−−	−−	−−	−−
OursT	17.0	39.0	46.0	49.0	55.0
OursTC	15.0	41.0	47.0	53.0	58.0
OursTP	22.0	43.0	55.0	67.0	78.0
OursTCP	**22.0**	**47.0**	**57.0**	**76.0**	**83.0**

Variant 1 (denoted as OursT): We remove the four body-part channels from the proposed CNN model and use the original triplet loss function to train the network.

Variant 2 (denoted as OursTC): We use the same network model as OursT, but use the improved triplet loss function to train the network instead.

Variant 3 (denoted as OursTP): We use the full version of the proposed multi-channel CNN model and train it with the original triplet loss function.

Variant 4 (denoted as OursTPC): We use the same network model as OursTP, but train it with the improved triplet

Table 3. Experimental evaluations on VIPeR dataset.

Method	Top1	Top5	Top10	Top15	Top20	Top30
MtMCML[29]	28.8	59.3	75.8	83.4	88.5	93.5
SDALF[8]	19.9	38.4	49.4	58.5	66.0	74.4
eBiCov[28]	20.7	42.0	56.2	63.3	68.0	76.0
eSDC[47]	26.3	46.4	58.6	66.6	72.8	80.5
PRDC[49]	15.7	38.4	53.9	63.3	70.1	78.5
aPRDC[27]	16.1	37.7	51.0	59.5	66.0	75.0
PCCA[30]	19.3	48.9	64.9	73.9	80.3	87.2
KISSME[21]	19.6	48.0	62.2	70.9	77.0	83.7
SalMatch[46]	30.2	52.3	66.0	73.4	79.2	86.0
LMLF[48]	29.1	52.3	66.0	73.9	79.9	87.9
Ding[6]	40.5	60.8	70.4	78.3	84.4	90.9
mFilter+LADF[48]	43.4	−−	−−	−−	−−	−−
Sakrapee[31]	45.9	−−	−−	−−	−−	−−
OurT	34.3	55.6	65.1	71.7	74.4	81.7
OurTC	37.2	55.6	67.1	76.5	75.3	83.9
OurTP	43.8	69.5	79.7	81.0	85.4	90.2
OurTCP	**47.8**	**74.7**	**84.8**	**89.2**	**91.1**	**94.3**

Table 4. Experimental evaluations on CUHK01 dataset.

Method	Top1	Top5	Top10	Top15	Top20	Top30
mFilter[48]	34.3	55.0	65.3	70.5	−−	−−
SalMatch[46]	28.5	46.3	57.2	64.1	−−	−−
PatMatch[46]	20.4	34.1	41.0	47.3	−−	−−
genericM[24]	20.0	44.1	57.1	64.3	−−	−−
ITML[5]	16.0	28.5	45.3	53.5	−−	−−
LMNN[41]	13.5	31.2	41.8	48.5	−−	−−
eSDC[47]	19.7	33.1	40.5	46.8	−−	−−
FPNN[25]	27.9	−−	−−	−−	−−	−−
Ejaz[1]	47.5	−−	−−	−−	−−	−−
Sakrapee[31]	53.4	76.4	84.4	−−	90.5	−−
Ours3T	46.0	67.7	78.7	85.3	88.7	90.3
Ours3TC	49.3	76.5	86.6	93.7	94.7	98.0
Ours3TP	52.3	82.1	90.3	**94.0**	95.6	**98.4**
Ours3TCP	**53.7**	**84.3**	**91.0**	93.3	**96.3**	98.3

loss function.

Note that, since the CUHK01 dataset is much larger than the other three datasets, we choose to model it using a larger configuration with an additional convolution layer in each of the five channels. The derived models corresponding to Variant 1 to 4 are denoted as Ours3T, Ours3TC, Ours3TP, and Ours3TPC, respectively.

Table 1, 2, 3, and 4 show the evaluation results on the four benchmark datasets, respectively, using the top 1, 5, 10, 15, 20, and 30 ranking accuracies. Each table includes 11 to 14 representative methods that have reported evaluation results on the corresponding dataset. Some of the works in these tables, such as Ding's method [6], FPPN [25], DeepM[45], mFilter[48] and Ejaz's[1] all used DCNN models to learn features for the person re-id task, and their performance accuracies are near the top in the list. Among these works, DeepM also used body parts to train their CNN models. In contrast to our single network with multiple channels, this work divides person images into three overlapped parts, and uses them to train three independent networks. The three networks are fused at the score level. There are also some works, such as Sakrapee's method [31], mFilter+LADF [48], that combine several different approaches to boost the performance accuracies. These ensemble methods have achieved state-of-the-art performances so far.

Table 5. Analysis the parameter β on VIPeR dataset.

β	Top1	Top5	Top10	Top15	Top20	Top30
0	43.8	69.5	79.7	81.0	85.4	90.2
0.001	45.9	73.4	81.9	87.0	**93.0**	**95.6**
0.002	**47.8**	74.7	84.8	**89.2**	91.1	94.3
0.003	45.6	**75.3**	**85.4**	87.6	90.5	94.6
0.004	43.7	73.1	81.5	87.9	91.1	93.4

Compared to the above representative works, the OursTCP model has achieved the top performances on all the four datasets, with all the six ranking measurements. The evaluation results shown in the four tables can be summarized as follows.

- Compared to Sakrapee's ensemble-based method, which is the state-of-the-art method so far, the OursTCP model is slightly better on CUHK01 dataset, but remarkably outperforms the former on the remaining three datasets by a margin of 2% to 10%.
- The improved triplet loss function is able to improve the performance accuracies for both the single and multi-channel models. Training a model with this loss function can get up to 4% performance improvement compared to the same model trained with the original triplet loss function.
- The multi-channel model that explores both the global full-body and local body-parts features is very powerful and effective for improving the performance accuracies. Compared to the model with no parts information in the structure, it can boost the person re-id accuracy by up to 13%.

As defined by Eq.(3), the improved triplet loss function contains two terms: the intra-class and the inter-class constraints. To investigate the effect of the parameter β on the performance accuracy, we conducted experiments using cross validation method on the VIPeR dataset, and the results are shown in Table 5. We can clearly see that our proposed person re-id method yields the best performances

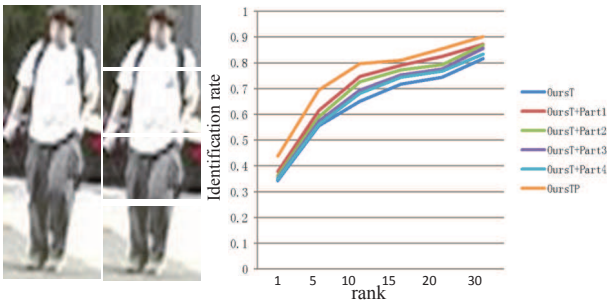

Figure 4. Analysis of different body parts on VIPeR dataset.

when β is in the range of 0.001 to 0.003. Based on this observation, we set β to 0.002 in all our experimental evaluations.

4.3. Analysis of different body parts

To understand the contribution of different body regions to the person re-id performance accuracy, we trained four different network models that contains the full-body channel and one body-part channel which corresponds to the body-part 1, 2, 3 and 4, respectively. These four models are denoted as Ours-Part1, Ours-Part2, Ours-Part3, and Ours-Part4, respectively. We also included the models of OursT and OursTP for comparisons. The experiments are performed on the VIPeR dataset, and the performance accuracies are shown in Figure 4. It is interesting to observe that the body part 1, which includes the face and shoulder of a person, leads to the largest performance improvement. When we move down the body, the performance improvement gradually decreases, with the body part 4, which includes the legs and feet of a person, providing the least performance improvement. This result is not surprising, because legs and feet are the moving parts of a person, which change dramatically in shape and pose. Such parts provide the least reliable features, and hence contribute little to the person re-id task.

We have visualized the features learned by each convolution layer, which are shown in Figure 5. We can see that the second convolution layer of the full-body channel captures the global information of each person, while the second convolution layers of the four body-parts channels capture the detailed local body-parts features of a person. Therefore, such a joint representation and learning framework for the global full-body and local body-parts features can achieve superior performances.

5. Conclusion

In this paper, we present a novel multi-channel parts-based convolutional network for person re-identification problem, which is formulated under a triplet framework via an improved triplet loss function. In this framework, we

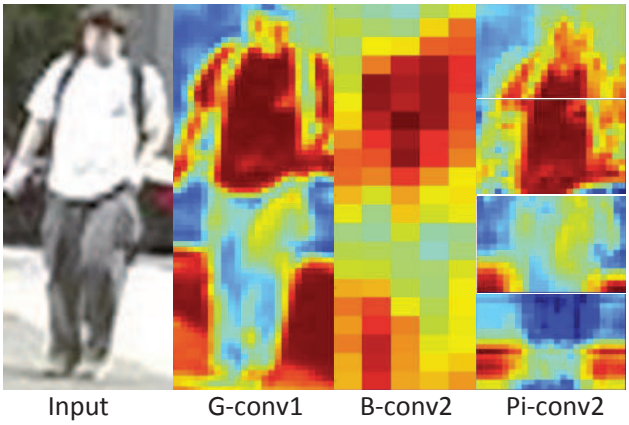

Figure 5. Learned feature maps of the network. G-conv1 shows features learned by the global convolution layer. B-conv2 represents the features learned by the full-body convolution layer which captures salient global full-body features, while Pi-conv2 capture salient local body-parts features learned by the body-part convolution layers.

constructed a CNN architecture including both global body convolution layer and local parts convolution layers. Thus the feature representations learned by our model can contain global information and local detailed properties. The architecture is trained by a set of triplets to produce features that aims to pull the instances of the same person closer, meanwhile push the instances belonging to different persons farther from each other in the learned feature space via the organized triplet samples. And our model got state-of-the-art performance on most benchmark datasets. In the future, we will extend our framework and approach to other task such as image and video retrieval problems.

Acknowledgement

This work was supported by the National Basic Research Program of China (Grant No.2015CB351705), the State Key Program of National Natural Science Foundation of China (Grant No.61332018).

References

[1] E. Ahmed, M. Jones, and T. K. Marks. An improved deep learning architecture for person re-identification. *CVPR*, 5:25, 2015.

[2] S. Bak, E. Corvee, F. Brémond, and M. Thonnat. Person re-identification using spatial covariance regions of human body parts. In *Advanced Video and Signal Based Surveillance (AVSS), 2010 Seventh IEEE International Conference on*, pages 435–440, 2010.

[3] X. Chang, Y. Yang, E. P. Xing, and Y.-L. Yu. Complex event detection using semantic saliency and nearly-isotonic svm. In *International Conference on Machine Learning (ICML)*, 2015.

[4] D. S. Cheng, M. Cristani, M. Stoppa, L. Bazzani, and V. Murino. Custom pictorial structures for re-identification. In *BMVC*, volume 1, page 6, 2011.

[5] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon. Information-theoretic metric learning. In *ICML*, pages 209–216, 2007.

[6] S. Ding, L. Lin, G. Wang, and H. Chao. Deep feature learning with relative distance comparison for person re-identification. *Pattern Recognition*, 2015.

[7] P. Dollár, Z. Tu, H. Tao, and S. Belongie. Feature mining for image classification. In *CVPR*, pages 1–8, 2007.

[8] M. Farenzena, L. Bazzani, A. Perina, V. Murino, and M. Cristani. Person re-identification by symmetry-driven accumulation of local features. In *CVPR*, pages 2360–2367, 2010.

[9] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(9):1627–1645, 2010.

[10] N. Gheissari, T. B. Sebastian, and R. Hartley. Person reidentification using spatiotemporal appearance. In *CVPR*, volume 2, pages 1528–1535, 2006.

[11] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, pages 580–587, 2014.

[12] A. Globerson and S. T. Roweis. Metric learning by collapsing classes. In *NIPS*, pages 451–458, 2005.

[13] D. Gray, S. Brennan, and H. Tao. Evaluating appearance models for recognition, reacquisition, and tracking. In *Proc. IEEE International Workshop on Performance Evaluation for Tracking and Surveillance (PETS)*, volume 3, 2007.

[14] D. Gray and H. Tao. Viewpoint invariant pedestrian recognition with an ensemble of localized features. In *ECCV*, pages 262–275. 2008.

[15] M. Guillaumin, J. Verbeek, and C. Schmid. Is that you? metric learning approaches for face identification. In *CVPR*, pages 498–505, 2009.

[16] J. Han, D. Zhang, S. Wen, L. Guo, T. Liu, and X. Li. Two-stage learning to predict human eye fixations via sdaes. 2015.

[17] M. Hirzer, C. Beleznai, P. M. Roth, and H. Bischof. Person re-identification by descriptive and discriminative classification. In *Image Analysis*, pages 91–102. 2011.

[18] M. Hirzer, P. M. Roth, and H. Bischof. Person re-identification by efficient impostor-based metric learning. In *Advanced Video and Signal-Based Surveillance (AVSS), 2012 IEEE Ninth International Conference on*, pages 203–208, 2012.

[19] W. Hu, M. Hu, X. Zhou, T. Tan, J. Lou, and S. Maybank. Principal axis-based correspondence between multiple cameras for people tracking. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(4):663–671, 2006.

[20] S. Khamis, C.-H. Kuo, V. K. Singh, V. D. Shet, and L. S. Davis. Joint learning for attribute-consistent person re-identification. In *ECCV*, pages 134–146, 2014.

[21] M. Koestinger, M. Hirzer, P. Wohlhart, P. M. Roth, and H. Bischof. Large scale metric learning from equivalence constraints. In *CVPR*, pages 2288–2295, 2012.

[22] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1097–1105, 2012.

[23] W. Li and X. Wang. Locally aligned feature transforms across views. In *CVPR*, pages 3594–3601, 2013.

[24] W. Li, R. Zhao, and X. Wang. Human reidentification with transferred metric learning. In *ACCV*, pages 31–44, 2012.

[25] W. Li, R. Zhao, T. Xiao, and X. Wang. Deepreid: Deep filter pairing neural network for person re-identification. In *CVPR*, pages 152–159, 2014.

[26] Z. Li, S. Chang, F. Liang, T. S. Huang, L. Cao, and J. R. Smith. Learning locally-adaptive decision functions for person verification. In *CVPR*, pages 3610–3617, 2013.

[27] C. Liu, S. Gong, C. C. Loy, and X. Lin. Person re-identification: What features are important? In *ECCV*, pages 391–401, 2012.

[28] B. Ma, Y. Su, and F. Jurie. Bicov: a novel image representation for person re-identification and face verification. In *BMVC*, pages 11–pages, 2012.

[29] L. Ma, X. Yang, and D. Tao. Person re-identification over camera networks using multi-task distance metric learning. *Image Processing, IEEE Transactions on*, 23(8):3656–3670, 2014.

[30] A. Mignon and F. Jurie. Pcca: A new approach for distance learning from sparse pairwise constraints. In *CVPR*, pages 2666–2672, 2012.

[31] S. Paisitkriangkrai, C. Shen, and A. v. d. Hengel. Learning to rank in person re-identification with metric ensembles. *arXiv preprint arXiv:1503.01543*, 2015.

[32] U. Park, A. K. Jain, I. Kitahara, K. Kogure, and N. Hagita. Vise: Visual search engine using multiple networked cameras. In *ICPR*, volume 3, pages 1204–1207, 2006.

[33] P. M. Roth, M. Hirzer, M. Köstinger, C. Beleznai, and H. Bischof. Mahalanobis distance learning for person re-identification. In *Person Re-Identification*, pages 247–267. 2014.

[34] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. *arXiv preprint arXiv:1503.03832*, 2015.

[35] W. R. Schwartz and L. S. Davis. Learning discriminative appearance-based models using partial least squares. In *Computer Graphics and Image Processing (SIBGRAPI), 2009 XXII Brazilian Symposium on*, pages 322–329, 2009.

[36] Y. Sun, Y. Chen, X. Wang, and X. Tang. Deep learning face representation by joint identification-verification. In *NIPS*, pages 1988–1996, 2014.

[37] A. Toshev and C. Szegedy. Deeppose: Human pose estimation via deep neural networks. In *CVPR*, pages 1653–1660, 2014.

[38] UK. Home office i-lids multiple camera tracking scenario definition. In *2008*.

[39] 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 *CVPR*, pages 1386–1393, 2014.

[40] X. Wang, G. Doretto, T. Sebastian, J. Rittscher, and P. Tu. Shape and appearance context modeling. In *ICCV*, pages 1–8, 2007.

[41] K. Q. Weinberger, J. Blitzer, and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. In *NIPS*, pages 1473–1480, 2005.

[42] E. P. Xing, M. I. Jordan, S. Russell, and A. Y. Ng. Distance metric learning with application to clustering with side-information. In *NIPS*, pages 505–512, 2002.

[43] F. Xiong, M. Gou, O. Camps, and M. Sznaier. Person re-identification using kernel-based metric learning methods. In *ECCV*, pages 1–16. 2014.

[44] Y. Yang, J. Yang, J. Yan, S. Liao, D. Yi, and S. Z. Li. Salient color names for person re-identification. In *ECCV*, pages 536–551. 2014.

[45] D. Yi, Z. Lei, S. Liao, and S. Z. Li. Deep metric learning for person re-identification. In *ICPR*, pages 34–39, 2014.

[46] R. Zhao, W. Ouyang, and X. Wang. Person re-identification by salience matching. In *ICCV*, pages 2528–2535, 2013.

[47] R. Zhao, W. Ouyang, and X. Wang. Unsupervised salience learning for person re-identification. In *CVPR*, pages 3586–3593, 2013.

[48] R. Zhao, W. Ouyang, and X. Wang. Learning mid-level filters for person re-identification. In *CVPR*, pages 144–151, 2014.

[49] W.-S. Zheng, S. Gong, and T. Xiang. Person re-identification by probabilistic relative distance comparison. In *CVPR*, pages 649–656, 2011.