Person Re-Identification Using Multi-region Triplet Convolutional Network

Bogdan Kwolek
AGH University of Science and Technology
Adama Mickiewicza 30
30-059, Krakow, Poland
bkw@agh.edu.pl

ABSTRACT

Person re-identification is a difficult task due to variations of person pose, scale changes, different illumination, occlusions, to name a few important factors usually diminishing identification performance across different views. In this work, we train a siamese and triplet convolutional neural networks and show that they can achieve promising recognition ratios. In order to cope with spatial transformations and scale changes across multi-view images we employ deformable convolutions in a triplet convolutional neural network. We propose an unified neural network architecture consisting of three triplet convolutional neural networks to jointly learn both the local body-parts features and full-body descriptors. We demonstrate experimentally that it achieves comparable results with results achieved by state-of-the-arts methods.

CCS CONCEPTS

•**Computing methodologies** → *Matching;* **Neural networks;** *Biometrics;*

KEYWORDS

Distributed smart cameras, person re-identification, deep learning, convolutional neural networks.

1 INTRODUCTION

As a core technology in distributed smart cameras, person re-identification attracts considerable research interest in both industrial and academic communities. The aim of person identification is matching individuals in a network of spatially non-overlapping surveillance or monitoring cameras [1]. It is a hard problem, which is still unsolved due to several difficulties, particularly because of difficulties caused by different camera views, complications arising due to different illumination and scale as well as background clutter and occlusions [2]. Such difficulties result in unlike person appearances in images acquired by different cameras. In addition, different subjects may share similar visual appearance, which in turn leads to additional difficulties in the person re-identification. In order to identify the same person in different cameras, the re-identification system must be robust against illumination changes, scale variations, outdoor clutter and occlusions.

Vision-based people re-identification is emerging as a very interesting research field with plenty of potential applications in soft-biometric technology, long-term surveillance or other security-related applications. In order to address these challenges several methods have been developed to describe visual appearance of persons and/or to measure the visual similarity among images with pedestrians undergoing monitoring. The research focuses on appearance descriptors [3, 4] and algorithms for matching across different views [5–7]. Most of existing methods usually consists of two parts: (i) extracting discriminative features, (ii) applying a distance metric for feature comparison. Discriminative feature-based methods concentrate on discovering robust descriptors that are resistant to occlusions as well as changes in pose and lighting while preserving the identity information, whereas the distance metric-based methods aim at minimizing the intra-class distance while maximizing the inter-class distance. The basic idea behind metric learning is to seek a mapping function from the feature space to a distance space, in which feature vectors representing the same person are closer than those from different ones. Most of the research in this area is devoted to developing or improving suitable hand-crafted features [8, 9] or good metric for multi-view feature comparison and person re-identification [2, 10–13], or both of them [6, 14].

Recently, promising approaches for person re-identification using neural networks have been proposed. Most of the methods uses convolutional neural networks (CNNs), which integrate the feature extraction and metric learning in single framework. CNNs build a hierarchy of features through extracting low-level features at the bottom layers and discovering higher level features such as the object parts or more general texture patterns at the mid-level. In [15] a filter pairing neural network (FPNN) has been proposed to jointly deal with occlusions and background clutter as well as to handle misalignment, photometric and geometric transforms.

In the field of deep learning-based person identification, in addition to work focusing on improving convolutional neural networks, several methods that use multiple neural networks have been proposed. Initially applied to signature verification [16], the Siamese neural network has since then been used in many applications, among them in learning complex similarity metrics for face verification [17] and dimensionality reduction [18]. Siamese neural networks optimize a loss function that drives the similarity metric to be large for feature pairs from different classes and small for feature pairs from the same category. The advantage of the siamese network over typical MLPs is that is capable of learning on data pairs instead of labeled instances. This means that it is particularly useful if the access to the labeled training data is limited or use of labeled data is too costly.

Yi et al. [19] employed body parts to train a neural network. In their method, images containing persons are cropped into three

ICDSC 2017, Stanford, CA, USA

overlapped parts, which are then employed to train three independent networks. Such networks are fused at the score level. In more recent work [20], an improved deep learning architecture with cross-input neighborhood differences, which capture local relationships between the two input images on the basis of mid-level features from image pairs has been proposed. In contrast to previously evoked work [19], which used siamese neural network with two convolutional layers, their siamese network comprises four convolutional layers. Gated siamese convolutional neural network was introduced in [21] to compare local features along a horizontal stripe for an input image pair and to adaptively boost them for enhancing the discriminative capability of the propagated features. Recently, Deng et al. [22] proposed a deep second-order siamese network for pedestrian re-identification, which consists of a convolutional neural network and a second-order similarity model. The convolutional network is used to learn comprehensive features, whereas the similarity model takes advantages of second-order information. Both models are jointly trained over one unified large margin objective.

Recently, triplet neural network-based algorithms for person re-identification were proposed [23, 24]. A CNN model proposed in [23] consists of multiple channels to jointly learn both the global full-body and local body-parts features of the pedestrians. In [24] a multi-scale triplet convolutional neural network, which is capable of capturing visual appearance of a person at various scales has been proposed. The discussed multi-scale network architecture consists of both deep and shallow neural networks.

Although various algorithms for person re-identification have been proposed, which exploit state-of-the-art methods for feature extraction and advanced metric learning algorithms, the performance of the best algorithms on commonly utilized person re-identification benchmarks [25], e.g. VIPeR, CUHK01, is still far from the performance needed for surveillance applications.

In this work we train a siamese and triplet convolutional neural networks and show that they can achieve promising results, particularly in comparison to results achieved by methods relying on hand-crafted features. We propose an extended triplet convolutional neural network to learn features of the pedestrians seen in multi-view multiple-scale and multiple channel images. In order to cope with spatial transformations and scale changes across multi-view images we employ recently proposed deformable convolutions in a triplet convolutional neural network. We demonstrate that on widely used benchmark dataset such an extended network can achieve promising performance, particularly in case of scale changes of pedestrian seen in multi-view images. We propose an unified neural network architecture consisting of three triplet convolutional neural networks to jointly learn both the local body-parts features and full-body descriptors. We demonstrate experimentally that it achieves promising results in comparison to results achieved by the siamese/triplet neural network.

2 METHOD

At the beginning of this section we overview convolutional neural networks. In the next section we outline the triplet neural networks. Afterwards, we present recently proposed deformable convolutional neural networks. In the last part of this section we present an enhanced neural network architecture for person re-identification.

2.1 Convolutional Neural Networks

Different from regular neural network, which only permits the input as vectors, convolutional neural networks (CNNs) allow 2 or 3-dimensional arrays at input layer. What makes convolutional neural networks distinct from classical MLPs is that the weights are shared, that is, being different with respect to the position relative to the center pixel they are identical for different pixels in the image [26]. Thus, it is straightforward to view a CNN as hierarchy of organized into layers a collection of local filters whose weights should be updated in a learning process. Every network layer acts as a detection filter for the presence of specific features or patterns present in the original data. The convolutions are usually followed by a non-linear operations after each layer since cascading linear convolutions would lead to a linear system. Besides, max-pooling is a mechanism that provides a form of translation invariance, which contributes towards the position independence. The CNNs are typically trained like a standard NNs using back-propagation.

2.2 Siamese and Triplet Networks

A siamese network is a symmetric architecture consisting of two networks [17], which share the same set of parameters. In contrast to ordinary MLPs, which employ loss functions comparing the neural network outputs with target values, the siamese networks use an objective that compares the feature vectors of pairs of the exemplars. The objective is constructed such that the distance between features representing instances from the same class is smaller in comparison to distances between features representing exemplars from different classes. By the use of such an objective there is no need to provide the labels for the classified exemplars. Moreover, in contrast to classical MLP in which the number of outputs is usually equal to number of the considered classes, the dimension of the target space can be specified with respect to the problem.

Let us assume that we have in disposal a training set consisting of images with pedestrians $\{\mathbf{x}_i, y_i\}_{i=1}^{N}$, where $\mathbf{x}_i \in \mathbb{R}^n$ and y_i are class labels. The siamese network produces a feature embedding $f(\mathbf{x}, \theta_f)$ that is defined as $f : \mathbb{R}^n \times \mathbb{R}^k \rightarrow \mathbb{R}^m$, where $\theta_f \in \mathbb{R}^k$ stands for parameters of the network. The pairwise loss can be defined as follows:

$$J_1^s(\mathbf{x}_i, \mathbf{x}_j, \theta_f) = \delta(y_i - y_j)\|f^{(1)}(\mathbf{x}_i, \theta_f) - f^{(2)}(\mathbf{x}_j, \theta_f)\|_2 - (1 - \delta(y_i - y_j))\|f^{(1)}(\mathbf{x}_i, \theta_f) - f^{(2)}(\mathbf{x}_j, \theta_f)\|_2 \quad (1)$$

where $\delta(\cdot)$ denotes Dirac delta function, whereas $f^{(1)}(\mathbf{x}, \theta_f)$ is constrained to be equal to $f^{(2)}(\mathbf{x}, \theta_f)$. Alternatively, the pairwise loss can be expressed as:

$$J_2^s(\mathbf{x}_i, \mathbf{x}_j, \theta_g) = \delta(y_i - y_j)(1/(\kappa + g(\mathbf{x}_i, \mathbf{x}_j, \theta_g))) + (1 - \delta(y_i - y_j))g(\mathbf{x}_i, \mathbf{x}_j, \theta_g) \quad (2)$$

where a convolutional network $g(\mathbf{x}_i, \mathbf{x}_j, \theta_g)$ returns the similarity between the features $\mathbf{x}_i$ and $\mathbf{x}_j$, where κ is a small positive constant.

It is worth to noting that the classical MLPs and the siamese neural networks have similar gradient formulations, which permits the use of standard back-propagation algorithm for training.

Training a siamese network is almost the same to the training of a standard MLP or a CNN. The only difference is that a siamese network operates on pairs of data samples, whereas the standard MLP operates on single data samples.

A triplet network consists of three instances of the same network that share parameters [27]. More specifically, the network is trained using set of triplets, which are processed by the mentioned above three instances of the same network with shared parameters. In person re-identification tasks each triplet contains three images. i.e. a query image, a matched image, i.e. an image of the same person as that in the query image, and the mismatched image. The network discovers features such that for every triplet the L2 distance between the matched pair and the mismatched pair tend to be as large as possible. Subsequently, the distances between matched image pairs assume smaller values than those between the mismatched image pairs.

The triplet loss can be expressed in the following manner:

$$J_1^t(\mathbf{x}, \mathbf{x}^+, \mathbf{x}^-, \theta_f) = \max\left(0, 1 - \frac{\|f^{(1)}(\mathbf{x}, \theta_f) - f^{(3)}(\mathbf{x}^-, \theta_f)\|_2}{\|f^{(1)}(\mathbf{x}, \theta_f) - f^{(2)}(\mathbf{x}^+, \theta_f)\|_2 + m}\right) \tag{3}$$

where m denotes the margin, $\mathbf{x}^-$ and $\mathbf{x}$ are from different classes, $\mathbf{x}^+$ and $\mathbf{x}$ are from the same class, and $f^{(1)}(\cdot)$, $f^{(2)}(\cdot)$ and $f^{(3)}(\cdot)$ are constrained to be the same neural network.

2.3 Deformable Convolutional Networks

Recently, a new type of convolution and pooling, called deformable convolution and deformable RoI pooling has been proposed [28]. The deformable convolution contains two parts, namely regular convolution layer and another convolution layer that is devoted to learn 2D offset for each input. The deformable convolution can be perceived as a learnable dilated convolution, for which the dilated rate is learned and can be different for each input. Having on regard that offsets are not integer (fractional), a bilinear interpolation is employed to sample from the input feature map. The 2D offsets are then encoded in the channel dimension. The authors demonstrated that the introduced deformable convolution is capable of expanding the receptive fields for bigger objects. In context of person identification this is very desirable property, particularly in context of real-world person re-identification [24], since the identification system should cope with scale change over images from different camera views.

2.4 Parts-based Triplet Neural Network

Every instance of convolutional neural network processes color RGB images of size 64 x 64. The first convolutional layer consists of 16 filters of size 3 x 3. The second convolutional layer comprises 32 filters of size 3 x 3 and is followed by 2D maxpooling layer. The next layer is convolutional layer with 64 filters of size 3 x 3, which is followed by the 2D maxpooling layer. This layer is followed by convolutional layers with 64 and 128 filters of size 3 x 3, respectively, which in turn are followed by the 2D maxpooling layer. The output of this layer is flattened and then fed into a fully connected neural network, which is in turn followed by a dense output layer. Between the dense layers the dropout is executed. The convolution and dense layers apply ReLU activation function. The dimensionality of the output of such a base network is equal to 64. Thus, the loss of each triplet network is calculated on the basis of vectors consisting of 64 features. Since we divide the input images into two horizontally divided sub-images, as well we process the whole input image, the loss values produced by the triplet networks are summed and then averaged, see Fig. 1. This way the proposed neural network integrates three triplet networks. As far as we know, besides our network, only network proposed by [23] integrates neural networks in a single learning framework. For instance, [29] divides person images into three overlapped parts, but uses them to train three independent networks.

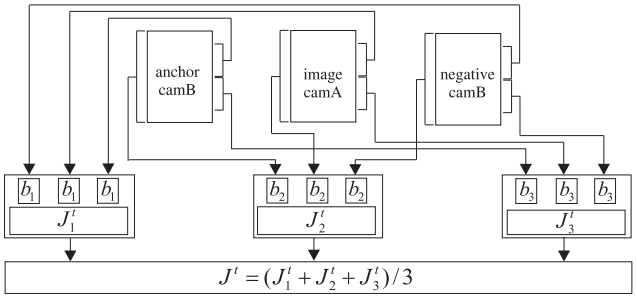

Figure 1: Parts-based triplet neural network.

3 EXPERIMENTS

At the beginning of this section we present datasets and discuss the employed evaluation protocol. Then, we present results that were obtained in experimental evaluations. Afterwards, we present details of the training of the proposed unified neural architecture. Finally we discuss the implementation details.

3.1 Datasets and Evaluation Protocol

The proposed framework has been evaluated on two publicly available benchmark datasets: VIPeR dataset and CUHK01 dataset. VIPeR dataset consists of 1264 images belonging to 632 subjects captured with two non-overlapping cameras. Each person pair was captured by different cameras with different viewpoints, poses, and lighting conditions. VIPeR dataset is one of the most challenging datasets for the person re-identification task due to vast variance and discrepancy. CUHK01 dataset contains 971 persons, captured from two camera views in a campus environment. The camera view A captures frontal or back views of a person, whereas camera B contains profile views of persons. For each person there are four images.

For each dataset we select half of the persons for training, and the remaining half for testing. The images from first camera are selected as query images, while the images from the second camera are selected as gallery images. The gallery set comprises single image for each person. For every image in the query set we calculate the distance between the query image and all the gallery images. We compute L2 distances for features (embeddings) produced by the trained networks and then select the most n nearest images from the gallery set. If in the selected set of images there is image representing the same individual as that in the query image at k-th position, then the recognition is achieved with rank k.

3.2 Experimental Evaluations

In the first part of the experiments we trained the siamese and triplet convolutional neural networks. The networks were trained on the whole images from VIPeR and CUHK01 benchmark datasets. The experimental results obtained on VIPeR and CUHK01 benchmark datasets are presented in Tab. 1 and Tab. 2. The presented results were achieved in ten experiments with different splits of data into train and test parts. As we can observe, the siamese and triplet convolutional neural network achieve better results in comparison to results achieved by methods that are based on hand-crafted features and achieve promising results in comparison to recent learning-based methods [8, 15, 20, 29]. Our triplet convolutional neural network is slightly outperformed by recently proposed multi-channel parts-based convolutional neural network with enhanced triplet loss [23].

Table 1: Performance of state-of-the-art algorithms and siamese/triplet convolutional neural networks on VIPeR dataset.

	Rank			
method	1	5	10	20
[10]	0.196	0.480	0.622	0.770
[7]	0.157	0.384	0.539	0.701
[8]	0.291	0.523	0.660	0.799
[12]	0.302	0.523	0.660	0.792
[8]	0. 434	-	-	-
[2]	0.459	-	-	-
[23]	0.478	0.747	0.848	0.911
Siamese	0.352	0.521	0.627	0.691
Triplet	0.440	0.700	0.810	0.820

Table 2: Performance of state-of-the-art algorithms and siamese/triplet convolutional neural networks on CUHK01 dataset.

	Rank			
	1	5	10	15
[8]	0.343	0.550	0.653	0.705
[12]	0.285	0.463	0.572	0.641
[15]	0.278	-	-	-
[20]	0.475	-	-	-
[2]	0.534	0.764	0.844	-
[23]	0.537	0.843	0.910	0.933
Siamese	0.445	0.720	0.769	0.851
Triplet	0.526	0.816	0.882	0.899

Afterwards, we evaluated the performance of the triplet convolutional neural network with deformable convolutions on the VIPeR dataset. The weights obtained in training of the triplet convolutional neural network were employed in the initialization of the triplet convolutional neural network with deformable convolutions. On VIPeR dataset the triplet convolutional neural network relying on deformable convolution achieves similar results in comparison to ordinary triplet convolutional neural network. For rank 1, 5, 10 and 20 the recognition accuracy was equal to 0.40, 0.74, 0.78 and 0.88, respectively, cf. results in Tab. 1. Next, we enlarged the test images using randomly selected scale from 1.0 to 1.5 and evaluated the recognition rate of the triplet convolutional network with and without the deformable convolutions. It turned out that on the rescaled images the recognition accuracy was almost 50% smaller for ordinary triplet network, whereas the recognition accuracy of the triplet network with deformable convolutions was smaller about 78%.

Finally, we trained a network consisting of three triplet convolutional neural networks. The first triplet CNN operates on whole images, whereas the remaining triplet networks operate on two non-overlapping sub-images, which were obtained by dividing the image into two horizontal stripes. The neural network was trained on VIPeR dataset, where first half of the images from both cameras was used in training, whereas the second part of the images was used for testing. Figure 2 demonstrates the averaged cumulative match characteristic (CMC) curve, which was obtained in ten runs with unlike initializations. For rank 1, 5, 10 and 20 the recognition rate is equal to 0.481, 0.692, 0.753 and 0.803, respectively.

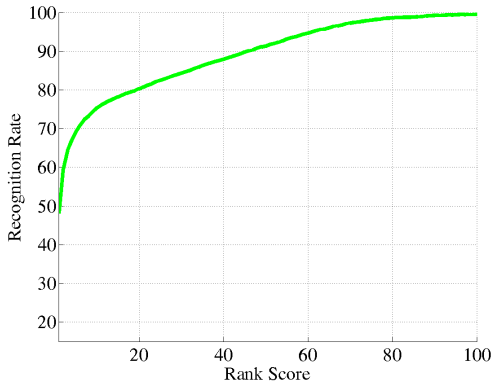

Figure 2: Averaged cumulative match characteristic (CMC) curve of 10 runs of the triplet network on VIPeR dataset. The first 316 images from both cameras were used to construct training triplets, whereas the remaining images were used in testing.

Figure 3 depicts the embedded features determined by our parts-based triplet convolutional network, which were projected into two dimensional space using t-SNE (t-distributed Stochastic Neighbor Embedding) algorithm [30]. t-SNE transforms similarities between data points into joint probabilities and tries to optimize the KL divergence between the joint probabilities of high-dimensional data and low-dimensional embedding. In the discussed plot, every projected feature into two dimensional space is represented by the corresponding image. The test images from the VIPeR dataset were processed by the base convolutional neural networks b_1, b_2, b_3, see Fig. 1, which determined 64-dimensional embeddings, which in turn were projected in two-dimensional space by the t-SNE algorithm. Having on regard that the part-based siamese network consists of three base networks the size of the embedded vector

was equal to 192. As we can observe, there is consistency among the projections of such embedded vectors onto 2d space, i.e. the 2d points representing the same person are usually closer than those from different ones.

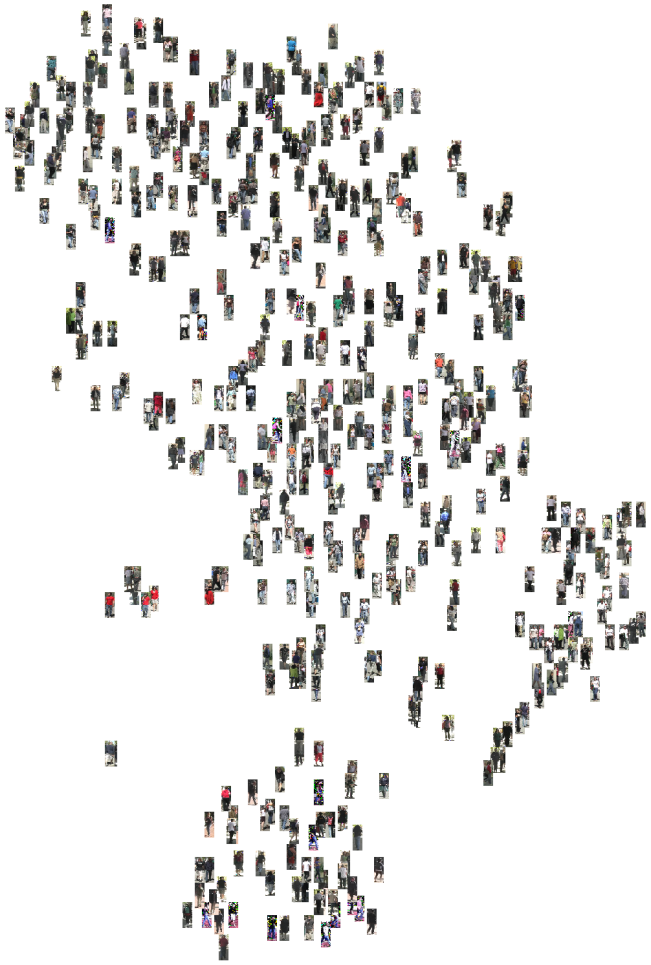

Figure 3: Thet-SNEmaps of triplet CNN features for VIPeR dataset.

3.3 Training of the Convolutional Neural Network

In order to alleviate the over-fitting we increased the volume of the training set through small perturbations of the location of the windows, which were used to crop the person images. The cropped images were resized to images of size 64 x 64 pixels. In the process of artificial augmentation of the data the sub-images were translated and flipped. The number of positive image pairs as well as negative image pairs for each subject has been increased approximately four times. Afterwards, the average image computed from all the training images has been subtracted from all images. In addition to shifting of the inputs to zero-mean the images were also normalized to unit variance. Such images were then rescaled to the range -1 … 1.

The siamese and triplet networks that were employed in this work, cf. Subsection 2.2, contain a large number of parameters, which means that it was necessary to provide a large number of pairs or triplets at the learning stage. The simplest solution to this problem is to employ sampling from training data. However, sampling all possible pairs or triplets from the training data can quickly become intractable since the majority of those samples may produce small costs in lost functions, which in turn leads to slow convergence [31]. Below we detail a smart data selecting strategy that has been applied to avoid overfitting, and particularly how we avoided focusing on the hard training samples.

The neural networks were initially trained on selected data from available datasets for person recognition and re-identification. In the first epochs of the training a data generator has been used to create balanced datasets. The aim of the generator was to shuffle the dataset and to produce training data with similar number of matched and unmatched observations. In the early stage of the training the persons in the training batch were selected such that the distance between feature vectors representing positive examples was small, whereas the distance between the negative samples was possibly large. In order to cope with the internal covariate shift we applied batch normalization, which allowed us to obtain faster learning and higher overall accuracy. By normalizing the data in each mini-batch it was possible to apply higher learning rates and to shorten the learning time. Having on regard that in the training of deep neural networks, small perturbation in the initial layers typically leads to large change in the later layers, we paid considerable attention to selecting the training samples in the first batches as well as to regularization the gradients in order to prevent their distraction by outliers.

After training the network such that it was capable of solve some hard cases, the data generator selected hard cases in order to keep up a high loss of the objective function. During learning all pair-wise distance as well as norms of the embeddings were automatically verified as well as stored in a log file and then manually inspected. On the basis of such information the learning has been resumed several times until achieving satisfactory values of the above mentioned values, and particularly the desirable values of the objective loss function.

3.4 Implementation

The data preprocessing has been realized in Matlab. The data preprocessed in such a way were then imported into python. The learning of the neural network has been performed in python using keras framework. The learning of the neural network has been accomplished with GPU support using tensorFlow backend. All computations were realized on a PC equipped with an NVIDIA graphics card and running Windows 7.

4 CONCLUSIONS

In this work an effective framework for person re-identification on RGB images acquired by multiple cameras has been proposed. We trained a siamese and a triplet convolutional network and demonstrated experimentally that they can achieve promising results, particularly in comparison to results achieved by methods relying

on hand-crafted features. We proposed an extended triplet convolutional neural network to learn features of the pedestrians seen in multi-view multiple-scale and multiple channel images. In order to cope with spatial transformations and scale changes across multi-view images we utilized deformable convolutions. We demonstrated that such an extended network can achieve promising performance, particularly in case of scale changes of pedestrian seen in multi-view images. We proposed an unified neural network architecture consisting of three triplet convolutional neural networks to jointly learn both the local body-parts features and full-body descriptors. We demonstrated experimentally that it achieves promising results in comparison to results achieved by the siamese/triplet neural network.

5 ACKNOWLEDGMENT.

This work was supported by Polish National Science Center (NCN) under a research grant 2014/15/B/ST6/02808.

REFERENCES

[1] R. Vezzani, D. Baltieri, and R. Cucchiara, "People reidentification in surveillance and forensics: A survey," *ACM Comput. Surv.*, vol. 46, no. 2, pp. 29:1–29:37, 2013.

[2] S. Paisitkriangkrai, C. Shen, and A. van den Hengel, "Learning to rank in person re-identification with metric ensembles," in *CVPR*. IEEE Computer Society, 2015, pp. 1846–1855.

[3] B. Ma, Y. Su, and F. Jurie, "Local descriptors encoded by fisher vectors for person re-identification," in *Proc. of the 12th Int. Conf. on Computer Vision - Part I*. Springer, 2012, pp. 413–422.

[4] ——, "Covariance descriptor based on bio-inspired features for person re-identification and face verification," *Image and Vision Computing*, vol. 32, no. 6...7, pp. 379 – 390, 2014.

[5] Z. Li, S. Chang, F. Liang, T. S. Huang, L. Cao, and J. R. Smith, "Learning locally-adaptive decision functions for person verification," in *IEEE Conf. on CVPR.*, June 2013, pp. 3610–3617.

[6] F. Xiong, M. Gou, O. Camps, and M. Sznaier, *Person Re-Identification Using Kernel-Based Metric Learning Methods*. Springer, 2014, pp. 1–16.

[7] W.-S. Zheng, S. Gong, and T. Xiang, "Reidentification by relative distance comparison," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 3, pp. 653–668, 2013.

[8] R. Zhao, W. Ouyang, and X. Wang, "Learning mid-level filters for person re-identification," in *IEEE Conf. on Computer Vision and Pattern Rec.*, 2014, pp. 144–151.

[9] T. Matsukawa, T. Okabe, E. Suzuki, and Y. Sato, "Hierarchical gaussian descriptor for person re-identification," in *IEEE Conf. on Computer Vision and Pattern Rec.*, 2016, pp. 1363–1372.

[10] M. Hirzer, "Large scale metric learning from equivalence constraints," in *CVPR*, 2012, pp. 2288–2295.

[11] W. Li and X. Wang, "Locally aligned feature transforms across views," in *CVPR*, 2013, pp. 3594–3601.

[12] R. Zhao, W. Ouyang, and X. Wang, "Person re-identification by salience matching," in *ICCV*, 2013, pp. 2528–2535.

[13] Z. Zhang, Y. Chen, and V. Saligrama, "Group membership prediction," in *ICCV*, 2015, pp. 3916–3924.

[14] M. Song, S. Gong, C. Liu, Y. Ji, and H. Dong, "Person re-identification by improved local maximal occurrence with color names," in *8th Int. Congress on Image and Signal Processing (CISP)*, 2015, pp. 675–679.

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

[16] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a "siamese" time delay neural network," in *Proc. of the 6th Int. Conf. on Neural Information Processing Systems*, 1993, pp. 737–744.

[17] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *CVPR*, 2005, pp. 539–546.

[18] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *CVPR*, 2006, pp. 1735–1742.

[19] D. Yi, Z. Lei, S. Liao, and S. Z. Li, "Deep metric learning for person re-identification," in *22nd Int. Conf. on Pattern Rec.*, 2014, pp. 34–39.

[20] E. Ahmed, M. Jones, and T. Marks, "An improved deep learning architecture for person re-identification," in *IEEE Conf. on Comp. Vision and Pattern Rec.*, 2015, pp. 3908–3916.

[21] R. R. Varior, M. Haloi, and G. Wang, *Gated Siamese Convolutional Neural Network Architecture for Human Re-identification*. Springer, 2016, pp. 791–808.

[22] X. Deng, B. Ma, H. Chang, S. Shan, and X. Chen, *Deep Second-Order Siamese Network for Pedestrian Re-identification*. Springer, 2017, pp. 321–337.

[23] D. Cheng, Y. Gong, S. Zhou, J. Wang, and N. Zheng, "Person re-identification by multi-channel parts-based CNN with improved triplet loss function," in *CVPR*, 2016, pp. 1335–1344.

[24] J. Liu, Z.-J. Zha, Q. Tian, D. Liu, T. Yao, Q. Ling, and T. Mei, "Multi-scale triplet CNN for person re-identification," in *Proc. of the ACM on Multimedia Conf.*, 2016, pp. 192–196.

[25] A. Bedagkar-Gala and S. K. Shah, "Editor's choice article: A survey of approaches and trends in person re-identification," *Image Vision Comput.*, vol. 32, no. 4, pp. 270–286, 2014.

[26] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," in *Proc. of the IEEE*, 1998, pp. 2278–2324.

[27] E. Hoffer and N. Ailon, *Deep Metric Learning Using Triplet Network*. Springer, 2015, pp. 84–92.

[28] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," 2017.

[29] D. Yi, Z. Lei, and S. Z. Li, "Deep metric learning for practical person re-identification," *CoRR*, 2014.

[30] L. van der Maaten and G. Hinton, "Visualizing high-dimensional data using t-SNE," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008.

[31] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face rec. and clustering," in *IEEE Conf. on Computer Vision and Pattern Rec.*, 2015, pp. 815–823.