End-to-End Comparative Attention Networks for Person Re-identification

Hao Liu, Jiashi Feng, Meibin Qi, Jianguo Jiang and Shuicheng Yan, *Fellow, IEEE*

arXiv:1606.04404v2 [cs.CV] 28 Apr 2017

***Abstract*—Person re-identification across disjoint camera views has been widely applied in video surveillance yet it is still a challenging problem. One of the major challenges lies in the lack of spatial and temporal cues, which makes it difficult to deal with large variations of lighting conditions, viewing angles, body poses and occlusions. Recently, several deep learning based person re-identification approaches have been proposed and achieved remarkable performance. However, most of those approaches extract discriminative features from the whole frame at one glimpse without differentiating various parts of the persons to identify. It is essentially important to examine multiple highly discriminative local regions of the person images in details through multiple glimpses for dealing with the large appearance variance.**

In this paper, we propose a new soft attention based model, *i.e.*, the end-to-end Comparative Attention Network (CAN), specifically tailored for the task of person re-identification. The end-to-end CAN learns to selectively focus on parts of pairs of person images after taking a few glimpses of them and adaptively *comparing* their appearance. The CAN model is able to learn which parts of images are relevant for discerning persons and automatically integrates information from different parts to determine whether a pair of images belongs to the same person. In other words, our proposed CAN model simulates the human perception process to verify whether two images are from the same person. Extensive experiments on four benchmark person re-identification datasets, including CUHK01, CHUHK03, Market-1501 and VIPeR, clearly demonstrate that our proposed end-to-end CAN for person re-identification outperforms well established baselines significantly and offer new state-of-the-art performance.

***Index Terms*—Person re-identification, Comparative Attention Network, Multiple glimpses.**

I. Introduction

RECENTLY, person re-identification (re-id), *i.e.*, person or pedestrian re-identification across multiple cameras without overlapping view, has received increasing attention [1]–[38]. It aims to re-identify a person that has been captured by one camera in another camera at any new location. Person re-identification has many important applications in security systems and video surveillance of public scenarios such as stores and shopping malls. Different from the video-based person re-identification methods, such as [20], [39], we investigate the problem of image-based person re-identification in this paper. Therefore, one of the major challenges lies in the lack of spatial and temporal cues. Moreover, it is also challenging to obtain satisfactory results in terms of accuracy in real-world scenarios due to the large appearance variance across multiple cameras. For example, people usually pose differently in two different views. Besides, other factors such as variations in color, illumination, occlusion as well as low-resolution of the captured frames also increase difficulties of the realistic person re-identification.

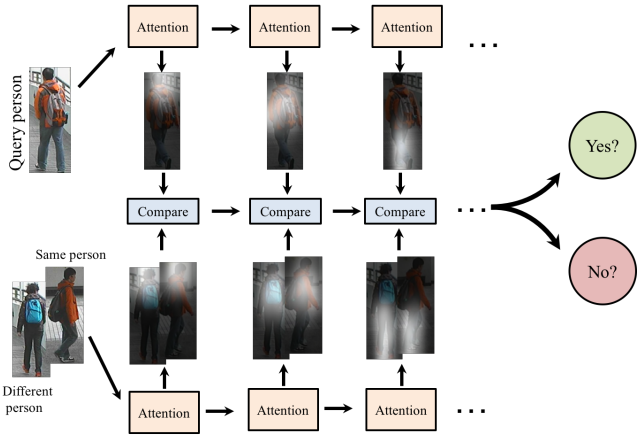


Fig. 1. Illustration of motivation of our method. The top image is the image of a query person while the bottom gives two images containing a same and a different person with the query. Through repeatedly comparing person pairs, a series of different local parts (*e.g.,* head parts, torso parts and leg parts) of persons are focused on (highlighted by the white regions), and then the information from the different parts is integrated to discern whether the image pairs belong to the same person.

Recently, several research attempts have been made for solving the re-identification problem through effectively reasoning over the person appearance. Some approaches for person re-identification [13], [14], [16], [17], [27], [35] utilize the Convolutional Neural Networks (CNNs) to learn effective representation of person appearance and achieve remarkable performance. Some of them [14], [16], [27], [35] exploit part or patch matching-based architecture to learn discriminative feature representation in local regions of persons. However, the parts in most of these methods are all pre-defined. For example, [35] splits the input person image into three square overlapping patches from top to bottom, and applies CNN architecture on them to learn the discriminative features of each patch. The performance of re-identification may be

H. Liu, M. Qi and J. Jiang are with School of Computer and Information, Hefei University of Technology, Hefei, Anhui 230009 China (e-mail: hfut.haoliu@gmail.com; qimeibin@163.com; jgjiang@hfut.edu.cn).

J. Feng is with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore 119077 (e-mail: elefjia@nus.edu.sg).

S. Yan is with the Artificial Intelligence Institute, Qihoo 360 Technology Company, Ltd., Beijing 100015, China (e-mail: eleyans@nus.edu.sg).

Manuscript received June, 2016; revised 2016. This work was supported in part by the National Natural Science Foundation of China (Grant Nos. 61371155, 61174170, 61632007.) and China Scholarship Council (Grant No. 201506690007).

influenced by the split way of regions. Additionally, the above methods learn the local discriminative representation from local regions once for all. As a consequence, the performance of such approaches may still suffer from factors such as illumination variance and occlusion if the learned local feature representation is not so robust to the factors. Considering the problems above, we propose a model which can adaptively find *multiple* local regions with more discriminative information in person images in a recurrent way and integrate them to further improve the person re-identification performance.

According to related research [1]–[17] and our daily experience, in the process of a human discerning another in a crowd, the human often abstracts the discriminative features of all the individuals and then compares the similarity and difference of them to find the specific one correctly, and this process can be repeated many times (i.e. multiple glimpses of each person). At the end of the process, the information gathered from glimpses is integrated as the comprehensive information to help the discerning. The motivation is illustrated in Fig.1. The focused parts in the repeated comparison process are highlighted by the white regions, corresponding to heads, torsos, and legs, which can provide discriminative information to identify persons. For example, whether they are wearing the same jackets or carrying the same backpack. Inspired by the observation, we propose an attention based model with inherent comparative components to solve the person re-identification problem.

With the recent development of Recurrent Neural Networks (RNNs) based on Long Short-Term Memory (LSTM) [40], the attention based models have demonstrated outstanding performance on several challenging sequential data recognition and modeling tasks, including caption generation [41], machine translation [42], as well as action recognition [43]. Briefly, similar to human visual processing, attention-based algorithms tend to selectively concentrate on a part of the information, and at the same time ignore other perceived information. Such a mechanism is usually called *attention* and can be employed to adaptively localize discriminative parts or regions of person images. Thus it is helpful to solve the person re-identification problem, which however has been rarely considered in the literatures.

In this work, we go beyond the standard LSTM based attention models and propose an end-to-end Comparative Attention Network (CAN). The proposed end-to-end CAN framework simulates the re-identification process of human visual system by learning a comparative model from raw person images to recurrently localize some discriminative parts of person images via a set of glimpses. At each glimpse, the model generates different parts without any manual annotations. It takes both the raw person images and the locations of a previous glimpse as inputs, and produces the next glimpse local region features as the outputs. These features can be regarded as a kind of dynamical pooling feature, and we show that exploiting these features generated by our CAN model for person re-identification performs better than conventional pooling features, which is used by many existing models [13], [14], [16], [17].

Compared with the work of attention-based action recognition [43] using video sequence, our work is more related to the attention-based image caption generation [41] which also applies the attention model on the still images to learn a series of different local attention features in a recurrent way. However, our proposed CAN framework end-to-end learns the attention regions from raw images while the model in [41] generates attention regions based on the pre-extracted CNN features. Furthermore, our model has a comparing ability in the generation of local attention regions as it is a three-branch architecture taking a triplet of images as input for each branch while [41] only has one branch structure taking one image as input. In contrast, our approach is also able to achieve comparatively better performance compared to other methods, as validated by experimental results.

In summary, we make following contributions to person re-identification:

- We propose a new attention model that dynamically generates discriminative features in a recurrent way of "seeing" and "comparing" person images for automatically localizing the most discriminative parts of persons.
- We develop a comparative network that can efficiently seek discriminative parts of person image pairs by incorporating an on-line triplet selection method. Moreover, our CAN framework is able to generate attention parts directly from raw person image pairs in an end-to-end way.
- Finally, we quantitatively validate the good performance of our end-to-end CAN framework by comparing it to the state-of-the-art performance on four benchmark datasets: CUHK01 [9], CUHK03 [13], Market-1501 [15] and VIPeR [44].

The paper is organized as follows. Sec. II reviews the related work briefly. In Sec. III, the framework is described in details. Then, the experimental results on several public benchmark datasets are shown and the analyses are given in Sec. IV. Finally, a conclusion is presented in Sec. V.

II. Related Work

Typically, extracting features from input images and seeking a metric for comparing these features across images are two main components of person re-identification. The basic thought of searching for better feature representation is to find features that are partially invariant to lighting, pose, and viewpoint changes. A part of existing methods primarily employ hand crafted features such as color and texture histograms. Some studies have obtained more discriminative and robust feature representation, such as Symmetry-Driven Accumulation of Local Features (SDALF) [4] exploiting both symmetry and asymmetry color and texure information. Local Maximal Occurrence (LOMO) [24] analyzes the horizontal occurrence of local features, and maximizes the occurrence to make a stable representation against viewpoint changes. In [23], the authors proposed the reference descriptors (RDs) generated with the reference set to improve the matching rate. To utilize complementary information from different feature descriptors, a multiple hypergraph fusion (multi-HG) method was proposed in [22] to learn multiple feature descriptors. In [37], a ranking method fusing the dense invariant features

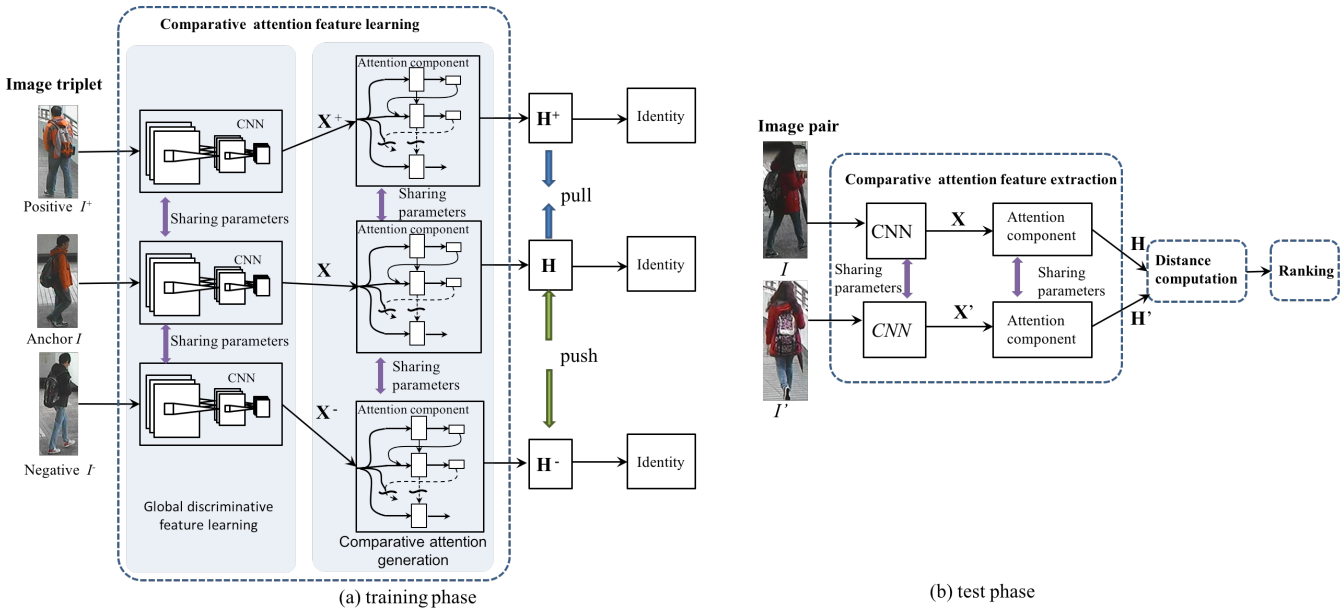


Fig. 2. The architecture of the proposed Comparative Attention Network (CAN). It consists of two parts, i.e., global discriminative features learning (CNNs) and local comparative visual attention generation. In the training phase, the model utilizes the weight-shared CNN to learn global features from a triplet of images and then passes them to the comparative attention component to compare positive pair and negative pair in a triplet to obtain the discriminative local visual attention features $\mathbf{H}$, $\mathbf{H}^+$ and $\mathbf{H}^-$, which makes positive pairs closer whereas negative pairs further away from each other in each triplet. These comparisons performed among positive pairs and negative pairs are achieved by introducing the step-wise triplet loss and identification loss on top of the network (refer to text for more details). And the whole architecture is trainable in an end-to-end way. In the test phase, the trained model is applied to each pair of persons and distances between each pair are computed for ranking the pairs of query and candidate frames.

(DIF) was proposed to model the relationship between an image pair across different camera views.

Additionally, some saliency-related methods [10], [11] have been proposed to enhance the ability of representation and discrimination of the feature for person re-identification. In [10], the authors presented a method of adjacency constrained patch matching to build dense correspondence between image pairs in an unsupervised way. Moreover, an approach called SalMatch [11] integrated both salience matching and patch matching based on the RankSVM framework. And mid-level filters (MidLevel) [12] was learned from patch clusters with coherent appearance obtained by pruning hierarchical clustering trees to get view-invariant and discriminative features. However, the above methods pre-extract the low-level or mid-level features of pre-defined local regions and then generate saliency maps. The feature extraction and saliency map generation are two separate processes, which would affect the person re-identification performance. Compared with the mentioned saliency-based methods, our attention-based CAN model is also a kind of saliency method. However, it is able to automatically learn the attention maps from raw person images in an end-to-end way.

Compared with the methods where complicated hand-crafted feature representation is designed, there are some approaches using metric learning, which formulate the person re-identification as a supervised distance metric learning problem where a transformation matrix is learned so that the Mahalanobis distance is relatively small when extracted features represent a same person and big otherwise. To achieve this goal, Large Margin Nearest Neighbor (LMNN) [5], Logistic Discriminant Metric Learning (LDM) [3], Information Theoretic Metric Learning (ITML) [1], Kernelized Relaxed Margin Components Analysis (KRMCA) [45], Robust Canonical Correlation Analysis (ROCCA) [21], Metric Learning with Accelerated Proximal Gradient (MLAPG) [31] and transfer Local Relative Distance Comparison (t-LRDC) are representative methods. Inspired by the thought of comparison in [5], [3], [1] and [45], we exploit the multi-task loss including triplet loss and identification loss in this paper, forcing the similarities of instances "more similar" and diversities "more different". Besides, in [6], Pairwise Constrained Component Analysis (PCCA) learns distance metrics from sparse pairwise similarity/dissimilarity constraints in the high dimensional input space. And large scale metric learning from equivalence constraint (KISSME) [7] considers a log-likelihood ratio test of two Gaussian distributions. In [25], a transfer local relative distance comparison (t-LRDC) model was formulated to address the open-world person re-identification problem by one-shot group-based verification.To match people from different views in a coherent subspace, [21] proposed a robust canonical correlation analysis (ROCCA) method. And in [31], the Metric Learning with Accelerated Proximal Gradient (MLAPG) was proposed to solve the person re-identification problem. In [8], the authors presented an effective structured learning based approach by combining multiple low-level hand-crafted and high-level visual features.

Despite the hand-crafted features based methods aforementioned, there are several deep learning based person re-identification approaches proposed [13], [14], [16], [17], [27], [35]. [13] proposed a method learning a Filter Pairing

Neural Network (FPNN) to encode and model photometric transforms by using the patch matching layers to match the filter responses of local across-view patches for person re-identification. In [14], a Siamese CNN, which is connected by a cosine layer, jointly learns the color feature, texture feature and metric in a unified framework. Moreover, [17] improved the performance by increasing the depth of layers and using very small convolution filters. In [27], [35], the authors proposed parts-based CNN model to learn the discriminative representations. Different from the part-based CNN methods [14], [16], [27], [35], our method is able to learn the local discriminative regions rather than pre-defining or splitting the local parts.

Recently, LSTMs have shown good performance in the domain of speech recognition [46] and image description [41]. More recently, [43] developed recurrent soft attention based models for action recognition and analysed where they focus their attention. Especially, the authors suggested that it is quite difficult to interpret internal representations learned by deep neural networks. Therefore, attention models add a dimension of interpretability by capturing where the model is focusing its attention when performing a particular task. Inspired by the above work, in this paper, we employ the recurrent attention model to generate different attention location information by comparing image pairs of persons and then integrate them together. As far as we know, our work is the first one applying the attention model to the person re-identification problem. Similar to saliency-related [10], [11], [12] methods mentioned above, the attention model is also a kind of saliency to certain extent, but our attention model can directly obtain the saliency-like attention maps from raw person image due to the end-to-end training pattern. Moreover, different from other attention models, our attention model generates attention maps based on the comparison over image triplets of people. Consequently, our network outperforms all previous approaches on benchmark person re-identification datasets.

III. Model Architecture

In this paper, we propose an end-to-end Comparative Attention Network (CAN) based architecture that formulates the problem of person re-identification as discriminative visual attention finding and ranking optimization. Fig. 2 illustrates our network architecture (III-A). For a given triplet of raw person images, we apply end-to-end Comparative Attention Network (CAN) at each one to learn comparative attention features. The global discriminative features are learned by CNNs, and then passed to the LSTM-based (III-B) comparative attention components (III-C) to obtain the discriminative attention masked features at different time steps. To combine these different time step features and make them more discriminative, a triplet selection method (III-D) is utilized after concatenating different time step features. Each of these components is explained in the following subsections.

A. End-to-End Comparative Attention Network Architecture

Fig. 2 illustrates the architecture of the proposed end-to-end Comparative Attention Network (CAN). The CAN network can localize and compare multiple person parts using the comparative attention mechanism. In this section, we describe how our comparative attention network works in the training phase and the test phase individually.

1) Training Phase: During training, the model starts from processing a triplet of raw images. Here, we denote the images of a triplet as I, I^+ and I^-, corresponding to the anchor sample, the positive sample and the negative sample respectively. I and I^+ come from the same class (positive pair), while I^- is from a different class (negative pair). The objective of CAN is to learn effective feature representation and to generate discriminative visual attention regions. Thus, in terms of the features extracted from the attention regions, the truly matched images are closer than the mismatched images by training the model on a set of triplets $\langle I, I^+, I^- \rangle$. Fig. 2 (a) shows the overall architecture used for training. The comparative attention network consists of following two parts: global discriminative feature learning components and comparative attention components.

Comparative Attention Feature Learning: In this paper, we adopt the truncated CNN such as Alexnet [47] and VGG [48] for global discriminative feature learning, and the learned feature map is denoted as $\mathbf{X} = \phi_{CNN}(I)$. Before end-to-end training the whole CAN, we pre-train the CNN with softmax classification model, which contains several convolutional feature learning layers with three fully-connected classification layers followed. After the pre-training of this network, the last three fully-connected layers are replaced with our proposed comparative attention model. The truncated CNN are used to learn global discriminative appearance features. Then, they are passed to the comparative attention components our proposed CAN to generate the comparative visual attention regional features, which are denoted as $\mathbf{H} = \beta(\mathbf{X})$. Here, β denotes the comparative attention generation part of our model, and $\mathbf{H}$ correspond to local comparative attention features of persons. Note that all the person samples in a triplet share the same parameters in feature learning and comparison, as shown in Fig. 2 (a). Details of the comparative attention model will be given in Section III-B and III-C.

Multi-task Loss: As mentioned above, our goal is to learn discriminative feature representation and visual attention regions through comparing the similarity and difference of positive and negative pairs in each triplet. Therefore, similar to [20], [49], we adopt the multi-task loss including triplet loss [50] and identification loss as the final loss function.

Within a triplet of $\langle \mathbf{H}_n, \mathbf{H}_n^+, \mathbf{H}_n^- \rangle$, we expect features of the positive sample $\mathbf{H}_n^+$ is more similar to $\mathbf{H}_n$ than the features of the negative sample:

$$\left\|\mathbf{H}_n - \mathbf{H}_n^+\right\|_2^2 + \alpha < \left\|\mathbf{H}_n - \mathbf{H}_n^-\right\|_2^2. \tag{1}$$

Here α is a margin that is introduced to enhance the discriminative ability of learned features between positive and negative pairs. Therefore, for N triplets, one loss function that CAN is going to minimize is:

$$\mathcal{L}_{trip} = \frac{1}{N}\sum_{n}^{N}\left[\left\|\mathbf{H}_n - \mathbf{H}_n^+\right\|_2^2 - \left\|\mathbf{H}_n - \mathbf{H}_n^-\right\|_2^2 + \alpha\right]_+, \tag{2}$$

where $[\cdot]_+$ truncates the involved variable at zero.

Additionally, for each δ-dimension feature vector $\mathbf{H}$ output by our CAN, the identity of the person is predicted using the standard softmax function, which is defined as follows:

$$\Omega(\mathbf{H}) = P(z = v|\mathbf{H}) = \frac{\exp(S_v^\top \mathbf{H})}{\sum_g \exp(S_g^\top \mathbf{H})}, \tag{3}$$

where G is the total number of identities, z is the predicted identity for the input person, and $S \in \mathbb{R}^{\delta \times G}$ is the weight matrix used in the softmax function, and $S_v \in \mathbb{R}^\delta$ and $S_g \in \mathbb{R}^\delta$ represent the v^{th} and g^{th} column of it, respectively. Then, for N triplets, the corresponding softmax loss function is defined as follows:

$$\mathcal{L}_{iden} = \frac{1}{3N} \sum_n^N \left(-\log(\Omega(\mathbf{H}_n)) - \log(\Omega(\mathbf{H}_n^+)) - \log(\Omega(\mathbf{H}_n^-)) \right). \tag{4}$$

Finally, we jointly end-to-end train our architecture with both triplet loss and identification loss. We can now define the overall multi-task training loss function $\mathcal{L}_{multi}$ which jointly optimizes the triplet cost and the identification cost as follows:

$$\begin{aligned} \mathcal{L}_{multi}(\mathbf{H}_n, \mathbf{H}_n^+ + \mathbf{H}_n^-) =& \mathcal{L}_{trip}(\mathbf{H}_n, \mathbf{H}_n^+ + \mathbf{H}_n^-) \\ &+ \mathcal{L}_{iden}(\mathbf{H}_n, \mathbf{H}_n^+, \mathbf{H}_n^-). \end{aligned} \tag{5}$$

Here, we give equal weights for the triplet cost and identification cost terms. The above comparative attention network (including CNNs and attention components) can be trained end-to-end using back-propagation from raw person images (details of our training parameters can be found in Sec. IV-B). Next, we proceed to introduce how to apply the network trained for testing.

2) Test Phase: As shown in Fig. 2 (b), in the test phase, we pass a set of person image pairs in the testing set into the trained CAN, where the Euclidean distance of them is computed. Then the ranking unit directly outputs the final ranking results. Here, we adopt average CMC (Cumulative Matching Characteristics) [44] and the accuracy at top ranks as the evaluation metrics, as in [1]–[17]. The detailed definition of CMC will be given in the Sec. IV-A. It is worth to mention that we also examine these items to see the performance of the whole network on the validation dataset during the training phase. This is because the training loss can only reflect the tendency of performance variance on the training set while the output evaluations on the validation set can directly indicate the true ranking performance. That is, we can train the network through directly optimizing the ranking results on the validation set.

B. Long Short-Term Memory Networks

In our CAN model, we use a long short-term memory (LSTM) network to produce an attention map over a local region at every time step conditioned on the input CNN feature maps, the previous hidden states and the generated attention map in the previous step. We implement the LSTM by following [51], [41] and [43], which is also illustrated in Fig. 3. At the time step t, LSTM takes a masked CNN feature map $\mathbf{A}_t$ and the previous hidden state $\mathbf{h}_{t-1}$ as inputs. The attention map $\mathbf{l}_{t-1}$ is predicted from the previous hidden state $\mathbf{h}_{t-1}$ using the learned parameters $W_{i,h}$. The predicted attention map is then used to mask the input feature map $\mathbf{X}_t$ of size $K \times K \times D$, giving a filtered feature maps where only attended regions are preserved. The formulations are shown as follows:

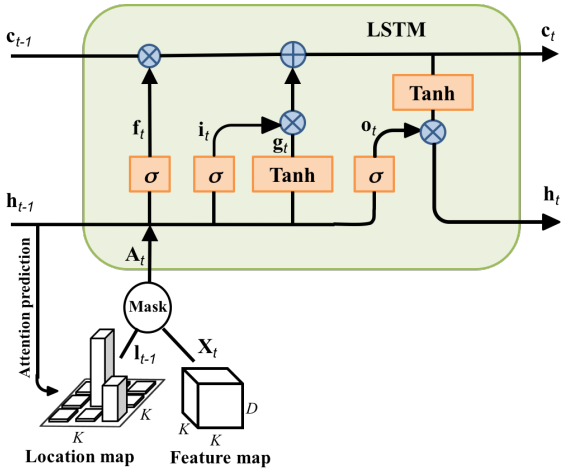


Fig. 3. A single time step LSTM unit which takes masked CNN feature map $\mathbf{A}_t$ and previous hidden state $\mathbf{h}_{t-1}$ as input. The location map or attention map $\mathbf{l}_{t-1}$ masking on the input feature map $\mathbf{X}_t$ is predicted by the previous hidden state $\mathbf{h}_{t-1}$. Each component of LSTM learns how to cooperate to weigh the input information (input gate $\mathbf{i}_t$), *i.e.* to remember the useful one (memory state $\mathbf{c}_t$) and to erase the unnecessary one (forget gate $\mathbf{f}_t$). Finally, the output gate $\mathbf{o}_t$ controls how the filtered memory should be emitted.

$$\begin{aligned} \mathbf{i}_t &= \sigma\left(\mathbf{M}_i\left[\mathbf{h}_{t-1}, \mathbf{A}_t\right] + b_i\right), \\ \mathbf{f}_t &= \sigma\left(\mathbf{M}_f\left[\mathbf{h}_{t-1}, \mathbf{A}_t\right] + b_f\right), \\ \mathbf{o}_t &= \sigma\left(\mathbf{M}_o\left[\mathbf{h}_{t-1}, \mathbf{A}_t\right] + b_o\right), \\ \mathbf{g}_t &= \tanh\left(\mathbf{M}_g\left[\mathbf{h}_{t-1}, \mathbf{A}_t\right] + b_g\right), \\ \mathbf{c}_t &= \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t, \\ \mathbf{h}_t &= \mathbf{o}_t \odot \tanh\left(\mathbf{c}_t\right), \end{aligned} \tag{6}$$

where $\mathbf{i}_t, \mathbf{f}_t, \mathbf{c}_t, \mathbf{o}_t$ and $\mathbf{h}_t$ are the input gate, forget gate, cell state, output gate and hidden state [40] respectively. $\mathbf{M}_\sim$ ($\mathbf{M}_\sim = \{\mathbf{M}_i, \mathbf{M}_f, \mathbf{M}_o, \mathbf{M}_g\}$) and $b_\sim$ ($b_\sim = \{b_i, b_f, b_o, b_g\}$) denote learnable weight parameters inside the gates and $\odot$ is the Hadamard product.

To produce the attention map, at each time step t, the softmax location map (*i.e.*, the attention map) $\mathbf{l}_{t-1}$ of size $K \times K$ is predicted from the previous hidden state $\mathbf{h}_{t-1}$ by the learnable parameters $W_{i,h}$ as follows:

$$l_{t-1,i} = \frac{\exp(W_{i,h}^\top \mathbf{h}_{t-1})}{\sum_{j=1}^{K \times K} \exp(W_{j,h}^\top \mathbf{h}_{t-1})}, \quad i = 1, \ldots, K^2, \tag{7}$$

where the weight parameters $W_{i,h}$ to generate softmax locations (attention) are learned together with the gate parameters $\mathbf{M}_\sim$ and $b_\sim$ in the end-to-end model training. Then, the masked feature $\mathbf{A}_t$ (through weighted average pooling) produced at the time step t is computed as follows:

$$\mathbf{A}_t = \mathbb{E}_{p(\mathbf{l}_{t-1}|\mathbf{h}_{t-1})}[\mathbf{X}_t] = \sum_{i=1}^{K^2} l_{t-1,i} \mathbf{X}_{t,i}, \tag{8}$$

where $\mathbf{X}_t$ and $\mathbf{X}_{t,i}$ correspond to the CNN feature maps and the i^{th} slice of the feature cube at time step t. Here $\mathbf{X}_t$ is a tensor of $K \times K \times D$. Each location out of $K \times K$ locations is described by a D-dimensional feature. The dimension of $\mathbf{A}_t$ is $1 \times 1 \times D$. Note, for each time-step t, our model takes the same CNN feature map $\mathbf{X}$ as input, so the CNN features $\mathbf{X}_t$ are the same for all the time steps. We adopt the following initialization method for memory state and hidden state:

$$\mathbf{c}_0 = f_{\text{init},c}\left(\frac{1}{K^2}\sum_{i=1}^{K^2}\mathbf{X}_{0,i}\right), \tag{9}$$

$$\mathbf{h}_0 = f_{\text{init},h}\left(\frac{1}{K^2}\sum_{i=1}^{K^2}\mathbf{X}_{0,i}\right), \tag{10}$$

where $f_{\text{init},c}$ and $f_{\text{init},h}$ are two-layer perceptrons consisting of two Fully Connected (FC) layers which can be learned end-to-end with other model components. And $\mathbf{X}_{0,i}$ represents the i^{th} slice of the feature map $\mathbf{X}_0$ output by CNN (corresponding to $\mathbf{X}_0$ in Fig. 4). These values are used to calculate the initial softmax attention location $\mathbf{l}_0$ which is applied on CNN features $\mathbf{X}_1$ to get the initial input $\mathbf{A}_1$ as shown in Fig. 3 and Fig. 4. Note that different from [43], our CAN model is trained end-to-end, so the CNN is trained together with the attention model of our CAN model in the training process, while in [43] the CNN features are pre-extracted off-line to initialize the memory state and hidden state for the attention model.

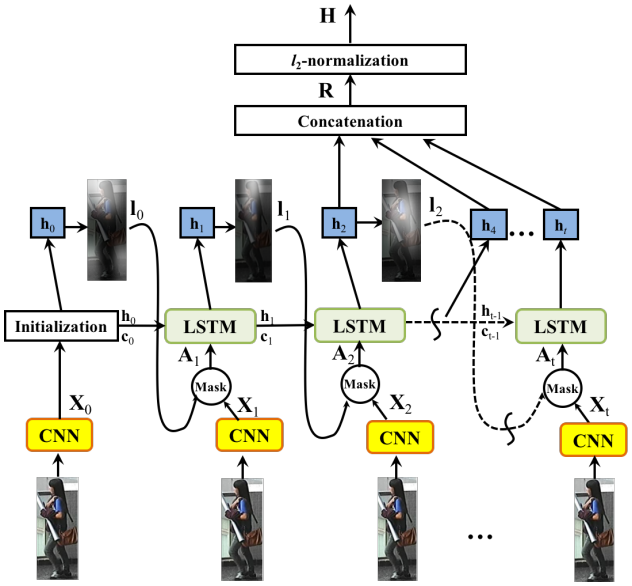


Fig. 4. The comparative attention component composed of several time step LSTMs takes CNN feature maps of a same person image as its input at each time step and outputs the concatenated hidden states, which are utilized every several time steps, as the features sent to the triplet loss layer. Here, we simply show one attention component, but there still exist another two weight-sharing attention components working simultaneously to compare positive and negative pairs in triplets and generate the comparative attention maps in training phase.

C. The Comparative Attention Component

In our comparative attention network, the LSTM-based comparative attention components take the feature maps output by CNN as inputs. Generally, CNN is used to obtain the global discriminative features from raw person images while the comparative attention components generate more local attention regions through comparing different person images. Intuitively, this mechanism has somewhat similarity with the function of the human visual system. In this subsection, we look inside the comparative attention components. The detailed composition shown in Fig. 3 corresponds to the attention components in the training phase and the test phase in Fig. 2. The brief working process of the comparative attention components is illustrated in Fig. 4. Note that although we simply show one attention component here, there exist three weight-sharing attention components working simultaneously in the training phase, among which two work in the test phase in practice, as shown in Fig. 2. At each time step, the attention component takes CNN features learned from a triplet of raw person images as its inputs. Each triplet of person images is "seen" once by our attention component in one glimpse.

Then some information is "remembered" and some "forgotten", decided by the LSTM unit, in order to generate attention location maps and hidden states for the next time step or "glimpse". For the hidden states output by each time step of attention components, they contain the memory in the process of comparing person images, and are exploited to obtain the local attention maps, as introduced in III-B. Therefore, to combine all the generated attention parts information and utilize it as holistic discriminative features for comparison, a concatenation layer is applied to concatenate a few time steps of hidden states along channel axis. The concatenation layer can be defined as follows:

$$\mathbf{R} = [\mathbf{h}_{\omega_1}; \mathbf{h}_{\omega_2}; ...; \mathbf{h}_{\omega_m}], \omega_m \in [1, 2, ..., t], \tag{11}$$

where $\mathbf{h}_{\omega_m} \in \mathbb{R}^{q\times 1}$ represents the hidden state with q channels from different time steps. t is the number of all time steps while m is the number of chosen time steps. We only use hidden steps at m chosen steps to form the matrix $\mathbf{R}$. And $\mathbf{R} \in \mathbb{R}^{mq\times 1}$ is the mq-channel output of the concatenation layer. The concatenation layer plays a crucial role in our CAN model because it integrates different time step features produced by repeatedly comparing person samples so that our model can judge which channels of features from which time step are more important to discerning persons. Through this way, our CAN model can attend different local regions at different time steps. In the Sec. IV-D, we will investigate the effect of choosing features from different time steps to concatenate.

Additionally, our CAN framework employs the triplet loss as one loss of our multi-mask loss and the whole framework is a three-weights-sharing-branch as well as recurrent network, so the loss value would fluctuate wildly in the training phase. To ensure that the distance derived from each triplet should not easily exceed the margin α so that more triplet constraints can take effect for the triplet loss function (Eqn. (2)), the concatenated features $\mathbf{R}$ are passed to the ℓ_2-normalization layer as the output of attention components:

$$\mathbf{H} = \frac{\mathbf{R}}{\sqrt{\sum_{d=1}^{mq} R_d^2}}, \tag{12}$$

where R_d represents the d^{th} entry of $\mathbf{R}$ and $\mathbf{H}$ is also the mq-dimension vector.

As aforementioned, our attention-based CAN model is a kind of saliency method. However, compared with the previous saliency-based methods extracting the saliency of pre-defined regions of persons based on low-level or mid-level features, such as eSDC [10], SalMatch [11] and MidLevel [12], our method can automatically learn the attention maps from raw person images in an end-to-end way. And the performance of the previous saliency-based methods would be affected by the low-level features because the low-level feature and saliency maps generation are two separate processes. Besides, the previous saliency-based methods get only one saliency map for each person image while our method can learn multiple attention maps which highlight different local regions in a recurrent style. And the superiority of our method is validated in Sec. IV-E.

D. Triplet Selection

It is crucial to select triplets that violate the constraint given in Eqn. (1). In particular, given $\mathbf{H}_n$, we want to select a positive sample $\mathbf{H}_n^+$ satisfying $\text{argmax}_{\mathbf{H}_n^+} \|\mathbf{H}_n - \mathbf{H}_n^+\|_2^2$ while a negative sample satisfying $\text{argmin}_{\mathbf{H}_n^-} \|\mathbf{H}_n - \mathbf{H}_n^-\|_2^2$. However, it is difficult and unrealistic to compute the argmin and argmax for the whole training set. Furthermore, our model needs to compare pair-wise images and to generate a series of attention locations for every image of each person. Therefore, it requires an efficient way to compute the argmin and argmax. There are two methods to be chosen as mentioned in [50]:

- Off-line triplets selection. The triplets are generated every few steps, and the most recent network checkpoint is employed to compute the argmin and argmax.
- On-line triplets selection. The selection can be done within a mini-batch.

Obviously, generating all possible triplets would result in overwhelming many triplets that are feasible for the constraint in Eqn.1. But some of these triplets would not contribute to the training and slow down the convergence of model training. Besides, they would still be passed through the network, which cause large unnecessary resource consumption. Different from off-line triplets selection method, on-line triplets selection approach selects triplets that are active and can contribute to improving the model within a mini-batch, so it is of higher efficiency and lower resource consumption. Therefore, we adopt the on-line triplets selection method in this paper. Specifically, instead of picking the hard positive, we adopt all positive pairs and randomly sample negative samples added to each mini-batch. In practice, we find that using all positive pairs makes the model more stable and converge faster than selectively using hard positive pairs in a mini-batch.

IV. Experiments

A. Datasets and Evaluation Protocol

There exist several challenging benchmark data sets for person re-identification. In this paper, we use CUHK01 [9], CUHK03 [13] Market-1501 [15], and VIPeR [44], which are four public benchmarks available, to conduct experiments. In experiments, for each pedestrian, the matching of his or her probe image (captured by one camera) with the gallery images (captured by another camera) is ranked. To reflect the statistics of the ranks of true matches, the Cumulative Match Characteristic (CMC) curve is adopted as the evaluation metric. Specifically, to create a CMC curve, the Euclidean distances between probe samples and those of gallery samples are computed firstly. Secondly, for each sample, a rank order of all the samples in the gallery is sorted from the sample with the smallest distance to the biggest distance. In the end, the percentage of true matches founded among the first m ranked samples is computed and denoted as rank$m$. Note, all the CMC curves for CUHK01, CUHK03 and VIPeR datasets are computed with single-shot setting. And the experiments on the Market-1501 dataset are under both the single-query and multi-query evaluation settings. In addition, for the Market-1501 dataset, the mean average precision (mAP) as in [15] is also employed to evaluate the performance since there are on average 14.8 cross-camera ground truth matches for each query. To construct the validation set, for the CUHK03 dataset, 100 persons are extracted as the validation set with the similar setting of [13]. For the CUHK01, VIPeR and Market-1501 datasets, the five-fold cross-validation is applied on the training set of each dataset, one as the validation set and the other four as the training set. And the samples in the validation set have no overlap with training set and testing set.

1) CUHK01: The CUHK01 [9] dataset contains 971 persons captured from two camera views, and each of them has two images in each camera view. Camera A captures the individuals in frontal or back views while camera B captures them in side views.

We randomly divide the dataset under two settings. The first setting contains a training set of 871 people and a test set of 100 people. In the second setting, 485 persons are randomly extracted for training while the left 486 persons compose the test set. For each setting, the training/test split is repeated 10 times and the average of CMC curves is reported.

2) CUHK03: There are 13,164 images of 1,360 identities contained in the CUHK03 dataset [13]. All pedestrians are captured by six cameras, and each person is only taken from two camera views. It consists of manually cropped person images and images automatically detected by the Deformable-Part-Model (DPM) detector [52]. This is a more realistic setting considering the existence of misalignment, occlusions, body part missing and detector errors. We evaluate the performance of CAN with the similar setting of [13]. That is, the dataset is partitioned into three parts: 1,160 persons for training, 100 person for validation and 100 persons for testing. The experiments are conducted with 20 random splits for computing averaged performance.

3) VIPeR: The VIPeR dataset [44] is considered the most challenging person re-identification dataset. It contains images of 632 persons, and each person has two images captured by two non-overlapping cameras with different viewpoints and illumination conditions. The dataset is randomly split into two subsets of 316 persons each, for training and test respectively. And the procedure is repeated 10 times to get an average

performance.

4) Market-1501: Market-1501 [15] is currently the largest public available re-identification dataset, containing 32,668 detected bounding boxes of 1,501 persons, with each of them captured by six cameras at most and two cameras at least. Similar to the CUHK03 dataset, it also employs DPM detector [52]. We use the provided fixed training and test set, containing 750 and 751 identities respectively, to conduct experiments.

B. Implementation Details

Pre-training of CNN: As introduced in Sec. III-A, our proposed CAN includes CNN and attention components to learn comparative attention features. Specifically, CNN is exploited to learn global discriminative features passed to attention components later. In this paper, we employ two classic CNNs: Alexnet [47] and VGG-16 [48]. And the difference of performance of our CAN by using different CNNs is validated in Sec.IV-D. Before end-to-end training our CAN, we pre-train the CNN part. In details, we use the training set of each person re-identification dataset except for VIPeR to fine-tune the standard softmax classification CNN trained on ImageNet [53] dataset. After pre-training, we remove the fully-connected classification layers and use the pre-trained CNN to initialize the CNN of our CAN model. Then the end-to-end training is performed on the specific training set of each person re-identification dataset. For the VIPeR dataset, considering the size of the training set is small, we use the training set of Market-1501 to pre-train the CNN and then use the training set of VIPeR to end-to-end train our CAN.

Parameter Setting: We implement our network using Caffe [54] deep learning framework. The training of the CAN converges in roughly 8-10 hours on NVIDIA GeForce GTX TITAN X GPU. And it takes roughly 5-10 minutes for one split testing. In all of experiments, the dimensionality of the LSTM hidden state, the cell state, and the hidden layer are set to 512 for CUHK01, CUHK03, VIPeR and Market-1501. The dimensionality of two layers within the MLP model ($f_{\text{init},c}$ and $f_{\text{init},h}$) in Eqn. (9) and (10) are also set as 512 to initialize the LSTM memory states and hidden states. When the AlexNet is adopted, the images in all the datasets are resized to 227 $\times$ 227 while the images are resized to 224 $\times$ 224 when the VGG-16 is adopted to train our model. At the stage of pre-training of CNN, we perform stochastic gradient descent [55] to update the weights. We start with a base learning rate of $\eta^{(0)} = 0.01$ and gradually decrease it along with the training process using an inverse policy: $\eta^{(k)} = \eta^{(0)}(1+\gamma \cdot k)^{-p}$ where $\gamma = 10^{-4}$, $p = 0.75$, and k is index of the current mini-batch iteration. We use a momentum of $\mu = 0.9$ and weight decay $\lambda = 5 \times 10^{-4}$. After the CNN feature learning network is pre-trained, we use the pre-trained model to initialize our end-to-end Comparative Attention Network (CAN). Here, we use the weight update parameter settings similar to those in the pre-training stage except that the initial learning rate is set to $\eta^{(0)} = 0.001$. As mentioned above, we adopt the on-line triplet selection method. Determined by cross-validation on CUHK03 dataset, the batchsize is set to 134 when CAN uses AlexNet and it is set to 66 when the VGG-16 is adopted. We chose the value of the margin parameter as $\alpha = 0.3$ by cross-validation on CUHK03 dataset with labeled setting. In the experiments on other datasets, we also adopt the above parameter settings. As mentioned in Sec. III-B, hidden states of LSTM at different time steps are concatenated as the final features passed to the normalization layer. Thus, we use 8 time steps and the extracted hidden states of the 2^{nd}, 4^{th} and 8^{th} time step in all experiments. It is illustrated in Fig. 6 (e) and Fig. 5, and is validated in Sec. IV-D.

C. Data Augmentation

In the training set, there exist much more negative pairs than positive pairs, which can lead to data imbalance and overfitting. To overcome this issue, we artificially augment the data by performing random 2D translation, similar to the processing in [13]. For an original image of size $w \times h$, we sample ten same-sized images around the image center, with translation drawn from a uniform distribution in the range $[-0.05w, 0.05w] \times [-0.05h, 0.05h]$. For all the datasets, we horizontally flip each image. In addition, because we use the on-line triplet selection method (see III-D), we randomly shuffle the dataset in terms of their labels. Through this shuffle strategy, more triplets can be produced in a mini-batch. Specifically, we perform this operation ten rounds for each dataset.

D. Analysis of the Proposed Model

1) Ablation Study: In Sec. III, we introduce our model architecture using CNN to learn global discriminative features. To investigate the feature learned from which layer is more effective for our CAN model, we use the CAN with AlexNet to conduct several experiments on the CUHK03 labeled dataset. We compare the performance of our model by using features learned from two different layers: $Conv5$ and $Max5$, which represent features from the 5^{th} convolutional layer and from the 5^{th} max pooling layer of AlexNet, respectively. Note, if $Conv5$ is used as a feature, the shape of the feature cube is 13 $\times$ 13 $\times$ 256, while if $Max5$ is used as a feature, the shape of the feature cube is 6 $\times$ 6 $\times$ 256. The experimental results on CUHK03 dataset are shown in Table I. From it, we observe that using $Max5$ can achieve better performance than using $Conv5$ as features. This may be because the $Max5$ can represent more abstract information for each person image, and provide more effective information for the subsequent comparative attention components.

Moreover, different from [43], our attention-based model is end-to-end trainable, which means that the comparative attention location maps can be obtained directly from the raw person images. So in Table I, we also compare the performance of end-to-end CAN with the non-end-to-end CAN using pre-extracted CNN features. In the non-end-to-end CAN, the CNN features are extracted from the $Conv5$ and $Max5$ layers of pre-trained standard softmax classification CNN by using the pre-training method described in Sec. IV-B. It shows that end-to-end CAN ("end-to-end CAN using Conv5", "end-to-end CAN using Max5") can achieve better results than the

TABLE I
RANK1, RANK5, RANK10 AND RANK20 RECOGNITION RATE (IN %) OF VARIOUS METHODS ON CUHK03 DATASET WITH LABELED SETTING.

Model	Rank1	Rank5	Rank10	Rank20
non-end-to-end CAN using Conv5	39.2	68.6	86.8	89.2
non-end-to-end CAN using Max5	46.3	72.1	92.3	95.1
end-to-end Avg pooled LSTM using Conv5	55.1	86.2	91.1	94.5
end-to-end Max pooled LSTM using Conv5	54.4	85.1	91.7	93.8
end-to-end Avg pooled LSTM using Max5	58.3	89.2	94.1	96.6
end-to-end Max pooled LSTM using Max5	57.9	88.9	93.3	95.2
end-to-end FC using Max5	53.8	86.5	90.1	93.5
end-to-end CAN using Conv5	63.8	91.0	95.2	97.1
end-to-end CAN using Max5	**72.3**	**93.8**	**98.4**	**99.2**

one using pre-extracted CNN features ("non-end-to-end CAN using Conv5", "non-end-to-end CAN using Max5"). This is because the global feature learning and comparative attention components of the end-to-end version both participate in the process of comparing person images and updating the parameters. That is to say, the training loss would back-propagate not only the attention components but also the CNN part of our CAN. Otherwise, if the features are off-line pre-extracted and sent to the attention model, the CNN features may not contain enough comparative information since the CNN model is pre-trained using the network for the classification task.

To further demonstrate the effectiveness of the comparative attention components in CAN, we also compare the performance of the proposed CAN with that of a similar architecture with masked input $\mathbf{A}_t$ of each LSTM replaced by the simple average pooling ("end-to-end Avg pooled LSTM using Conv5", "end-to-end Avg pooled LSTM using Max5") or max pooling ("end-to-end Max pooled LSTM using Conv5", "end-to-end Max pooled LSTM using Max5") over the CNN feature $\mathbf{X}_t$ in Fig. 4. In other words, we use the same architecture illustrated in Fig. 4 except that none of the attention prediction mechanisms is contained in the model, and thus there is no softmax location map (attention map) $\mathbf{l}_t$ produced and all locations in a feature map have the same weight. Note that the LSTM model used here is also in an end-to-end form. From the results given in Table I, it is obvious that comparing positive pair and negative pair of each person triplet and staying focusing on those more discriminative parts or locations can perform better than using the complete feature cube to discern different persons. Then we replace the LSTM part with two fully-connected layers with 512 dimensionality. From the results ("end-to-end FC using Max5"), we can observe that learning features in a recurrent way can indeed achieve better performance for person re-identification.

Additionally, we perform experiments with different time step numbers varying from 5 to 14. The performance is evaluated using the rank1 recognition rate. Here, we use the hidden states of all the time steps. The results are shown in the Fig. 5. We observe that the performance is gradually improved when the time step number increases from 5 to 8. However, further increasing step number from 8 to 14 does not bring significant improvement but increases the computational cost. So we choose the 8 time steps as it gives the best trade-off between performance and computational cost.

In the end, we also conduct a series of experiments to evaluate which time steps are chosen to be concatenated can achieve the best performance for our model. We use the following three settings: i) all the time steps (all 8 steps); ii) last time step (the 8^{th} step); iii) the 2^{nd}, 4^{th} and 8^{th} time steps. The experiment shown in Fig. 6 (e) illustrates concatenating step-2/4/8 within our proposed CAN model gives best performance, rather than using all time steps. This is because discriminative information offered by the hidden states of adjacent time step may have redundancy. They are not very distinguishable from adjacent ones. Thus, combing all the hidden states may lead to over-smoothed features and lose discriminative information. In contrast, selecting features from the time step using our discovered interval can avoid over-smoothing and keep the necessary discriminative information at early steps. Moreover, using all the time steps can also cause the large dimensionality of the feature output by the concatenation layer (Eqn. (11)), which increases the computation cost. Under the second setting, our proposed CAN also can not achieve good performance because only using the last time step can not provide sufficient discriminative information and there is no integration of multiple local region features included.

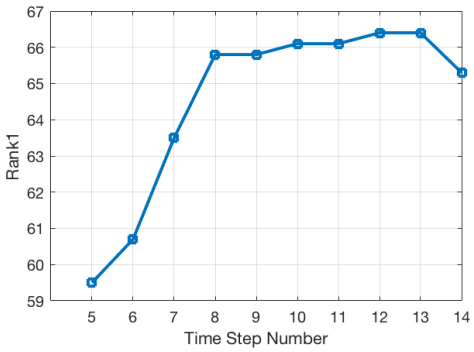


Fig. 5. The Rank1 recognition rate (in %) of our proposed CAN using different time step numbers, varying from 5 to 14, on CUHK03 dataset with labeled setting.

2) Effect of Different CNNs and Losses: In this subsection, we first compare the performance of CANs using two different CNN models (*i.e.*, AlexNet and VGG-16 respectively). When the VGG-16 is used, the feature learned from the 5^{th} max pooling layer is passed to the attention components of CAN.

Therefore, the dimension of the passed feature cube is 7 $\times$ 7 $\times$ 512. From the curves shown in Fig. 6 (f), one can observe that the proposed CAN can achieve better performance if it uses VGG-16 instead of AlexNet. This is because the VGG-16 model is a deeper network and is able to learn more discriminative representations for person re-id. In Fig. 6 (f), for both the CANs using AlexNet and VGG-16, one can also observe that using the multi-task loss can perform better than using either triplet loss or identification loss individually. This implies that both the identification and the ranking information (conveyed in the triplet loss) are important for learning discriminative features through comparative attention. More importantly, the most effective features come from combining the power of the two loss functions.

E. Comparison with State-of-the-Art Methods

We compare our model with the following state-of-the-art methods: SDALF [4], LMNN [5], ITML [1], KRMCA [45], LDM [3] , eSDC [10], Metric Ensembles (Ensembles) [8], KISSME [7], JointRe-id [16], FPNN [13], PersonNet [17], LOMO+XQDA [24], FT-JSTL+DGD [26], TCP [27], Embedding DM [35], LSSCDL [30] GOG [29], SalMatch [11], multi-HG [22], DNS [28], TMA [34], ROCCA [21], LMLF [12], CS [33], SCSP [36], RDs [23], MLAPG [31], DSVR_FSA [37].

TABLE II
COMPARISON OF OUR END-TO-END CAN METHOD'S PERFORMANCE ON THE CUHK01 DATASET WITH 100 TEST IDS TO THE STATE-OF-THE-ART MODELS.

Method	Rank1	Rank5	Rank10	Rank20
JointRe-id [16]	65.0	88.7	93.1	97.2
FPNN [13]	27.9	58.2	73.5	86.3
ITML [1]	17.1	42.3	55.1	71.7
LMNN [5]	21.2	49.7	62.5	78.6
KRMCA [45]	31.2	57.7	73.6	86.1
LDM [3]	26.5	57.7	72.1	84.7
SDALF [4]	9.9	41.2	56.0	66.4
eSDC [10]	22.8	43.9	57.7	69.8
KISSME [7]	29.4	57.7	72.4	86.1
LOMO+XQDA [24]	63.2	83.9	90.1	94.2
PersonNet [17]	71.1	90.1	95.0	98.1
Embedding DM [35]	86.6	-	-	-
end-to-end CAN (AlexNet)	82.8	97.0	99.6	100.0
end-to-end CAN (VGG-16)	**87.2**	**98.2**	**99.8**	**100.0**

1) Results on CUHK01 and CUHK03: These two datasets consist of thousands of training samples. Table II and Fig. 6 (a) show results on CUHK01 with 100 test IDs. Our method beats all compared methods at low ranks. For the CUHK01 with 486 test IDs, our method can also beat other compared methods. Although our attention-based CAN is also a kind of saliency method, it beats the other saliency-based methods [10]–[12] by a large margin. As for CUHK03, there are two settings: manually cropped person images and person images produced by DPM detector. Obviously, the performance on the latter one appears lower than that on the former, as shown in Fig. 6 (c), Fig. 6 (d), Table IV and Table V. However, the images produced by the detector can also reflect the algorithms in the real world. It can be seen from Fig. 6 (c) and Table IV that, as expected, on this large dataset, some other deep learning based methods, such as FT-JSTL+DGD [26], can achieve similar performance with millions of parameters or become much more competitive. However, with the detector boxes, our method is less affected, especially for the Rank1, and outperforms other approaches including deep learning based ones by a large margin. We suppose that the performance is not affected too much, possibly because our model could accurately attend to different discriminative parts of images and integrate their information which is robust to the influence brought by the detector.

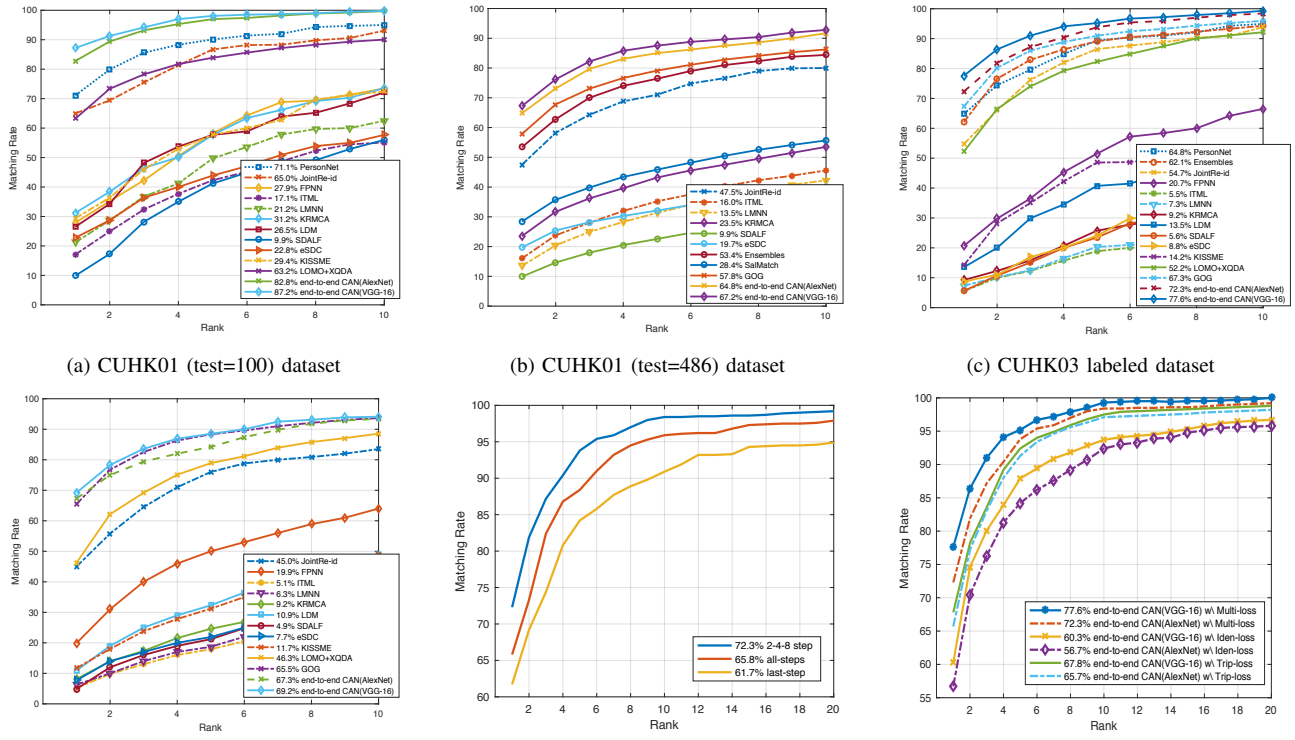
TABLE III
COMPARISON OF OUR END-TO-END CAN METHOD'S PERFORMANCE ON THE CUHK01 DATASET WITH 486 TEST IDS TO THE STATE-OF-THE-ART MODELS.

Method	Rank1	Rank5	Rank10	Rank20
FT-JSTL+DGD [26]	66.6	-	-	-
SDALF [4]	9.9	22.6	30.3	41.0
ITML [1]	16.0	35.2	45.4	59.8
LMNN [5]	13.5	31.3	42.3	54.1
KRMCA [45]	23.5	43.2	53.5	63.2
eSDC [10]	19.7	32.7	40.3	50.6
LSSCDL [30]	66.0	-	-	-
DSVR_FSA [37]	33.5	50.9	61.0	71.0
MidLevel [12]	34.3	55.1	65.0	75.0
TCP [27]	53.7	84.3	91.0	96.3
Ensembles [8]	53.4	76.4	84.4	90.5
JointRe-id [16]	47.5	71.0	80.0	-
SalMatch [11]	28.4	45.8	55.7	67.9
ROCCA [21]	29.8	-	67.8	77.0
GOG [29]	57.8	79.1	86.2	92.1
multi-HG [22]	64.4	-	90.6	94.6
DNS [28]	**69.1**	86.9	91.8	95.4
end-to-end CAN (AlexNet)	64.8	84.7	91.7	96.8
end-to-end CAN (VGG-16)	67.2	**87.3**	**92.5**	**97.2**

TABLE IV
COMPARISON OF OUR END-TO-END CAN METHOD'S PERFORMANCE ON THE CUHK03 DATASET WITH LABELED SETTING TO THE STATE-OF-THE-ART MODELS.

Method	Rank1	Rank5	Rank10	Rank20
Ensembles [8]	62.1	89.1	94.3	97.8
JointRe-id [16]	54.7	86.4	91.5	97.3
FPNN [13]	20.7	51.5	68.7	83.1
ITML [1]	5.5	18.9	30.0	44.2
LMNN [5]	7.3	21.0	32.0	48.9
KRMCA [45]	9.2	25.7	35.1	53.0
LDM [3]	13.5	40.7	52.1	70.8
SDALF [4]	5.6	23.5	36.1	52.0
eSDC [10]	8.8	24.1	38.3	53.4
KISSME [7]	14.2	48.5	52.6	70.0
LOMO+XQDA [24]	52.2	82.2	92.1	96.3
PersonNet [17]	64.8	89.4	94.9	98.2
GOG [29]	67.3	91.1	96.0	98.8
Embedding DM [35]	61.3	-	-	-
DNS [28]	62.6	90.1	94.8	98.1
FT-JSTL+DGD [26]	75.3	-	-	-
end-to-end CAN(AlexNet)	72.3	93.8	98.4	99.2
end-to-end CAN(VGG-16)	**77.6**	**95.2**	**99.3**	**100.0**

2) Results on Market-1501: Market-1501 is a large and realistic dataset since it was captured in a scene of crowded supermarket with complex environment. Besides, it contains several natural detector errors as the person images were



(a) CUHK01 (test=100) dataset (b) CUHK01 (test=486) dataset (c) CUHK03 labeled dataset

(d) CUHK03 detected dataset (e) CUHK03 labeled dataset (f) CUHK03 labeled dataset

Fig. 6. Performance comparison with state-of-the-art approaches using CMC curves on CHK01, CUHK03 labeled and CUHK03 detected datasets. (a) and (b) show comparisons of our method with previous methods on CUHK01 with 100 test IDs and 486 IDs, respectively. (c) and (d) show comparisons of our method with previous methods on CUHK03 labeled and detected, respectively. (e) gives comparison of our method with our own variations of concatenated time steps on CUHK03 labeled. (f) compares the performance of our method using different CNNs (AlexNet and VGG-16) and different losses (Multi-task loss, Triplet loss and Identity loss). Rank-1 matching rates are shown in the legend next to the method name. Our method beats the state-of-the-arts by a large margin, and gets the best performance when the 2^{nd}, 4^{th} and 8^{th} concatenated time steps are exploited. Our method can achieve the best performance by using VGG-16 as CNN and multi-task loss as training loss. See IV-D and IV-E for details. **Best viewed in 2 zoomed-in color pdf file.**

TABLE V
COMPARISON OF OUR END-TO-END CAN METHOD'S PERFORMANCE ON THE CUHK03 DATASET WITH DETECTED SETTING TO THE STATE-OF-THE-ART MODELS.

Method	Rank1	Rank5	Rank10	Rank20
JointRe-id [16]	45.0	76.0	83.5	93.2
FPNN [13]	19.9	50.0	64.0	78.5
ITML [1]	5.1	17.9	28.3	43.1
LMNN [5]	6.3	18.7	29.1	45.0
KRMCA [45]	8.1	20.3	33.0	50.0
LDM [3]	10.9	32.3	48.8	65.6
SDALF [4]	4.9	21.2	35.1	48.4
eSDC [10]	7.7	21.9	35.0	50.1
KISSME [7]	11.7	31.2	49.0	65.6
LOMO+XQDA [24]	46.3	78.9	88.6	94.3
GOG [29]	65.5	88.4	93.7	97.6
Embedding DM [35]	52.1	-	-	-
DNS [28]	54.7	86.8	94.8	95.2
end-to-end CAN(AlexNet)	67.3	84.2	93.4	95.2
end-to-end CAN(VGG-16)	**69.2**	**88.5**	**94.1**	**97.8**

collected by applying the automatic DPM detector. Each person is captured by by six cameras at most in Market-1501 — the number of cameras is significantly larger than the CUHK01 and CUHK03 datasets. Therefore, the relationships between person pairs are more complicated. We compare the performance of our proposed CAN model against state-of-the-art results under both the single query and multi-query settings and with both evaluation metrics in Table VI. The performance of baseline methods given in [15] is not very competitive, probably because BoW (Bag-of-Word) features the authors used are not robust enough. From Table VI, we can observe that the rank1 performance of our CAN model is slightly lower than the current best method (DNS [28]) under single query setting, but it still achieves the mAP as high as 35.9%. Under the multiple query setting, our method can provide the new state-of-the-art. This shows that our model performs comparable to other methods in more complex multi-camera person re-identification tasks, benefiting from the inherent attention and comparison mechanism.

3) Results on VIPeR: Compared with the aforementioned three datasets, VIPeR [44] is one of the most challenging, since it has 632 people but with various poses, viewpoints, image resolutions, and lighting conditions. And each person has only one image under each camera. From Table VII, one can see that our proposed method ("end-to-end CAN(VGG-16)") can beat most of the compared state-of-the-arts except SCSP [36] exploiting hand-crafted features. This is because the size of the training set of VIPeR is so small that our deep learning-based method is easy to overfit to the training set with millions of parameters. To overcome the overfitting problem, we perform the data augmentation introduced in Sec. IV-C

TABLE VI
COMPARISON OF OUR END-TO-END CAN METHODS PERFORMANCE ON THE MARKET-1501 DATASET WITH BOTH S ERY AND MULTIPLE QUERY SETTING TO THE STATE-OF-THE-ART MODELS.

Query	SingleQ		MultipleQ	
Method	**Rank1**	**mAP**	**Rank1**	**mAP**
SDALF [4]	20.5	8.2	29.2	13.8
eSDC [10]	33.5	13.5	42.5	18.4
LOMO+TMA [34]	47.9	22.3	-	-
Zheng et al. [15]	34.4	14.1	42.6	19.5
PersonNet [17]	37.2	18.6	-	-
SCSP [36]	51.9	26.4	-	-
DNS [28]	**61.1**	35.7	71.6	46.0
end-to-end CAN(AlexNet)	55.1	30.3	65.4	42.2
end-to-end CAN(VGG-16)	60.3	**35.9**	**72.1**	**47.9**

and fine-tune our proposed CAN based on the model pre-trained on other large person re-identification datasets such as Market-1501. However, the data are still insufficient to generate enough person triplets to train our CAN model. Compared with the experimental results on large datasets, such as CUHK03 and Market-1501, discussed in Sec.IV-E1 and Sec. IV-E2, the methods using low-level hand-crafted features are easier to achieve comparable results with deep learning-based methods on the small dataset. Therefore, we also conduct another experiment combining the output features of CAN and LOMO (LOcal Maximal Occurrence representation) [24] which is a kind of classic low-level feature containing both color and texture features. The combination is performed by simply concatenating the CAN feature vector and the LOMO feature vector. The effectiveness is verified by the results ("CAN(VGG-16)+LOMO") shown in Table VII.

TABLE VII
COMPARISON OF OUR END-TO-END CAN METHOD'S PERFORMANCE ON THE VIPER DATASET TO THE STATE-OF-THE-ART MODELS.

Method	**Rank1**	**Rank5**	**Rank10**	**Rank20**
ROCCA [21]	30.4	-	75.6	86.6
FT-JSTL+DGD [26]	38.6	-	-	-
JointRe-id [16]	34.8	64.5	78.5	89.1
PCCA [6]	19.3	48.9	64.9	80.3
SDALF [4]	19.9	38.4	49.4	66.0
eSDC [10]	26.3	46.4	58.6	72.8
SalMatch] [11]	30.2	52.3	66.0	73.4
KRMCA [45]	23.2	54.8	72.2	85.5
KISSME [7]	19.6	48.0	62.2	77.0
MidLevel+LADF [12]	43.4	73.0	84.9	93.7
Ensembles [8]	45.9	-	-	-
LMLF [12]	29.1	52.3	66.0	79.9
CS [33]	34.8	68.7	82.3	91.8
DSVR_FSA [37]	29.4	50.7	62.0	75.0
TCP [27]	47.8	74.7	84.8	91.1
SCSP [36]	53.5	82.6	91.5	**96.6**
multi-HG [22]	44.7	-	83.0	92.4
RDs [23]	33.3	-	78.4	88.5
TMA [34]	39.9	-	81.3	91.5
LOMO+XQDA [24]	40.00	68.9	80.5	91.1
GOG [29]	49.7	79.7	88.7	94.5
MLAPG [31]	40.7	-	82.3	92.4
DNS [28]	51.2	82.1	90.5	95.9
end-to-end CAN(AlexNet)	41.5	72.6	83.2	92.7
end-to-end CAN(VGG-16)	47.2	79.2	89.2	95.8
CAN(VGG-16)+LOMO	**54.1**	**83.1**	**91.8**	96.4

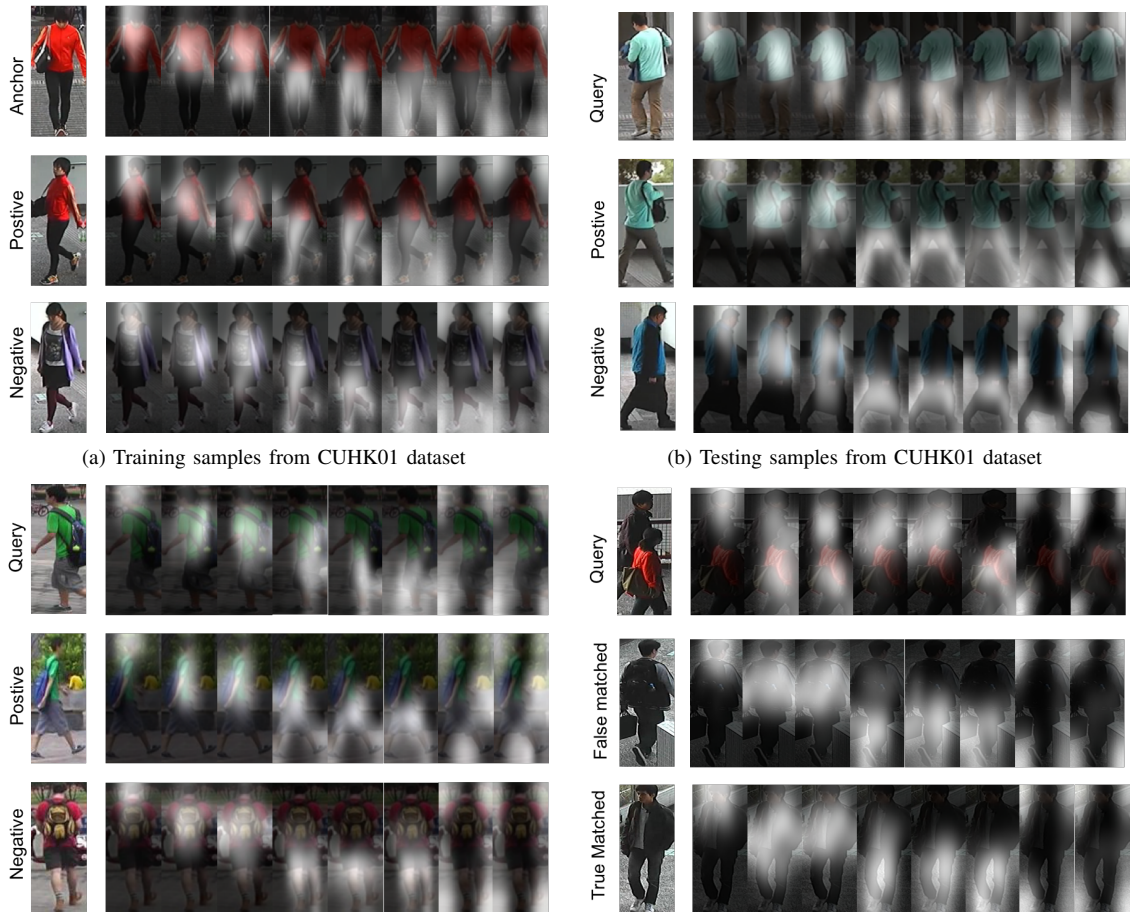
F. Visualization of Attention Maps and Discussions

In Fig. 7, we visualize some comparative attention maps produced by our network for both training samples (Fig. 7 (a)) and testing samples (Fig. 7 (b)) from CUHK01 dataset with 100 test IDs. In Fig. 7 (a), the triplet of training samples is randomly selected from one batch. And in Fig. 7 (b), the positive sample is ranked at top 1 in re-identification results while the negative sample is randomly selected from those ranked at bottom 10 in re-identification results.

In the attention maps generated on training samples (In Fig. 7 (a)), we can see that the model is able to focus on different parts of the person images at different time steps. The attention often starts from the head parts in the triplet sample images, and then is focused on the upper parts of the body at the second step. In the next several time steps, the attention gradually shifts on the lower parts of the images in a triplet. That is to say, our CAN model often focuses on the discriminative parts based on which the comparison can tell the persons are the same (for positive pairs) and are different (for negative pairs). Thus, the attention maps learned by our CAN model only represent which parts within the triplet are used by our CAN for comparison. Take the Fig. 7 (a) in which the query person wearing the red cloth for example. At the second step, the attention of CAN is focused on the upper body part. Thus intuitively our CAN compares what kinds of clothes the persons wearing to make decision: the positive pair of persons wear red cloth and the negative pairs of persons wear purple cloth. This part attended at this time step can tell whether the persons are the same or different. But the cloth information is not very reliable. Therefore, CAN proceeds to collect information with different attention in the following steps. At a single time step, if the attention does not focus on the corresponding local regions for both matched pair and unmatched pair, then the comparison is meaningless. For example, comparing the head part of one person with the leg part of other person cannot tell whether they are the same person or not. Note that, the model does not always attend to the foreground. We can see that some background is attended to at last two steps, which means the background can also provide information to assist matching persons correctly. The similar attention map changes of testing samples can also be observed in Fig. 7 (b).

The comparative attention maps of testing samples from Market-1501 dataset under single query setting are also visualized in Fig. 7 (c). Different from the CUHK01 dataset in which the person images are manually cropped, the person images are automatically detected by the DPM detector in the Market-1501 dataset. Therefore, more difficult cases, such as misalignment and body part missing, are contained due to the detector errors. Through the Fig. 7 (c), we can see that the attention maps of testing samples from Market-1501 also change in the similar way with that of CUHK01 dataset.

In Fig. 7 (d), we also visualize failure cases of our proposed CAN model in terms of the generated attention maps based on the comparison mechanism. We can observe that our model fails to focus on the same parts of positive person image pairs. This is partially because there is more than one person in



(a) Training samples from CUHK01 dataset

(b) Testing samples from CUHK01 dataset

(c) Testing samples from Market-1501 dataset

(d) Failure samples from CUHK03 labeled dataset

Fig. 7. Attention maps learned by our network for different training and testing person samples in CUHK01 (test=100) dataset and CUHK03 labeled dataset. (a) and (b) show the triplets of training and testing samples from CUHK01 (test=100) dataset, respectively. (c) shows the testing samples from Market-1501 dataset under single query setting. (d) shows some failure cases of our model for CUHK03 labeled dataset. In Fig. 7 (a), the triplet of training samples is randomly selected from one batch. And in Fig. 7 (b) and (c), the positive sample is ranked at top 1 in re-identification results while the negative sample is randomly selected from those ranked at bottom 10 in re-identification results. For both of triplet samples from training and testing set, the comparative attention is often focused on the discriminative parts based on which the comparison can tell the persons are the same (for positive pairs) and are different (for negative pairs).

the query image and this person is occluded heavily by other persons. This extremely hard scenario poses a big challenge to our CAN model as it can not exactly compare person image pairs and decide which local region should be selected. This phenomenon can also be observed from the generated attention maps of the query sample, false matched sample and true matched sample in Fig. 7 (d).

V. Conclusion

In this work, we introduced a novel visual attention model that is formulated as a triplet recurrent neural network which takes several glimpses of triplet images of persons and dynamically generates comparative attention location maps for person re-identification. We conducted extensive experiments on three public available person re-identification datasets to validate our method. Experimental results demonstrated that our model outperforms other state-of-the-art methods in most cases, and verified that our comparative attention model is beneficial for the recognition accuracy in person matching.

References

[1] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon, "Information-theoretic metric learning," in *Proceedings of the 24th international conference on Machine learning*. ACM, 2007, pp. 209–216.

[2] D. Gray and H. Tao, "Viewpoint invariant pedestrian recognition with an ensemble of localized features," in *Computer Vision–ECCV 2008*. Springer, 2008, pp. 262–275.

[3] M. Guillaumin, J. Verbeek, and C. Schmid, "Is that you? metric learning approaches for face identification," in *Computer Vision, 2009 IEEE 12th international conference on*. IEEE, 2009, pp. 498–505.

[4] M. Farenzena, L. Bazzani, A. Perina, V. Murino, and M. Cristani, "Person re-identification by symmetry-driven accumulation of local features," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 2360–2367.

[5] 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*. IEEE, 2012, pp. 203–208.

[6] A. Mignon and F. Jurie, "Pcca: A new approach for distance learning from sparse pairwise constraints," in *CVPR*. IEEE, 2012, pp. 2666–2672.

[7] M. Koestinger, M. Hirzer, P. Wohlhart, P. M. Roth, and H. Bischof, "Large scale metric learning from equivalence constraints," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 2288–2295.

[8] S. Paisitkriangkrai, C. Shen, and A. van den Hengel, "Learning to rank in person re-identification with metric ensembles," in *Proceedings of the*

IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1846–1855.

[9] W. Li and X. Wang, "Locally aligned feature transforms across views," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3594–3601.

[10] R. Zhao, W. Ouyang, and X. Wang, "Unsupervised salience learning for person re-identification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3586–3593.

[11] ——, "Person re-identification by salience matching," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 2528–2535.

[12] ——, "Learning mid-level filters for person re-identification," in *CVPR*, 2014, pp. 144–151.

[13] W. Li, R. Zhao, T. Xiao, and X. Wang, "Deepreid: Deep filter pairing neural network for person re-identification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 152–159.

[14] D. Yi, Z. Lei, S. Liao, and S. Z. Li, "Deep metric learning for person re-identification," in *2014 22nd International Conference on Pattern Recognition (ICPR)*. IEEE, 2014, pp. 34–39.

[15] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian, "Scalable person re-identification: A benchmark," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1116–1124.

[16] E. Ahmed, M. Jones, and T. K. Marks, "An improved deep learning architecture for person re-identification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3908–3916.

[17] L. Wu, C. Shen, and A. v. d. Hengel, "Personnet: Person re-identification with deep convolutional neural networks," *arXiv preprint arXiv:1601.07255*, 2016.

[18] S. M. Assari, H. Idrees, and M. Shah, "Re-identification of humans in crowds using personal, social and environmental constraints," *arXiv preprint arXiv:1612.02155*, 2016.

[19] ——, "Human re-identification in crowd videos using personal, social and environmental constraints," in *European Conference on Computer Vision*. Springer, 2016, pp. 119–136.

[20] N. McLaughlin, J. Martinez del Rincon, and P. Miller, "Recurrent convolutional network for video-based person re-identification," in *CVPR*, 2016.

[21] L. An, S. Yang, and B. Bhanu, "Person Re-Identification by Robust Canonical Correlation Analysis," *IEEE Signal Processing Letters*, vol. 22, no. 8, pp. 1103–1107, Jan. 2015.

[22] L. An, X. Chen, S. Yang, and X. Li, "Person re-identification by multi-hypergraph fusion," *IEEE Transactions on Neural Networks and Learning Systems*, 2016.

[23] L. An, M. Kafai, S. Yang, and B. Bhanu, "Person reidentification with reference descriptor," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 4, pp. 776–787, 2016.

[24] S. Liao, Y. Hu, X. Zhu, and S. Z. Li, "Person re-identification by local maximal occurrence representation and metric learning," in *CVPR*, 2015, pp. 2197–2206.

[25] W.-S. Zheng, S. Gong, and T. Xiang, "Towards open-world person re-identification by one-shot group-based verification," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 3, pp. 591–606, 2016.

[26] T. Xiao, H. Li, W. Ouyang, and X. Wang, "Learning deep feature representations with domain guided dropout for person re-identification," in *CVPR*, 2016.

[27] 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.

[28] L. Zhang, T. Xiang, and S. Gong, "Learning a discriminative null space for person re-identification," *CVPR*, 2016.

[29] T. Matsukawa, T. Okabe, E. Suzuki, and Y. Sato, "Hierarchical gaussian descriptor for person re-identification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1363–1372.

[30] Y. Zhang, B. Li, H. Lu, A. Irie, and X. Ruan, "Sample-specific svm learning for person re-identification," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[31] S. Liao and 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.

[32] C. Su, F. Yang, S. Zhang, Q. Tian, L. S. Davis, and W. Gao, "Multi-task learning with low rank attribute embedding for person re-identification," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 3739–3747.

[33] Y. Shen, W. Lin, J. Yan, M. Xu, J. Wu, and J. Wang, "Person re-identification with correspondence structure learning," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 3200–3208.

[34] N. Martinel, A. Das, C. Micheloni, and A. K. Roy-Chowdhury, "Temporal model adaptation for person re-identification," in *European Conference on Computer Vision*. Springer, 2016, pp. 858–877.

[35] H. Shi, Y. Yang, X. Zhu, S. Liao, Z. Lei, W. Zheng, and S. Z. Li, "Embedding deep metric for person re-identification: A study against large variations," in *European Conference on Computer Vision*. Springer, 2016, pp. 732–748.

[36] D. Chen, Z. Yuan, B. Chen, and N. Zheng, "Similarity learning with spatial constraints for person re-identification," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[37] S. Tan, F. Zheng, L. Liu, J. Han, and L. Shao, "Dense invariant feature based support vector ranking for cross-camera person re-identification," *IEEE Transactions on Circuits and Systems for Video Technology*, 2016.

[38] F. Zheng and L. Shao, "Learning cross-view binary identities for fast person re-identification." International Joint Conferences on Artificial Intelligence, 2016.

[39] H. Liu, Z. Jie, K. Jayashree, M. Qi, J. Jiang, S. Yan, and J. Feng, "Video-based person re-identification with accumulative motion context," *arXiv preprint arXiv:1701.00193*, 2017.

[40] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[41] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *Proceedings of The 32nd International Conference on Machine Learning*, 2015, pp. 2048–2057.

[42] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[43] S. Sharma, R. Kiros, and R. Salakhutdinov, "Action recognition using visual attention," *arXiv preprint arXiv:1511.04119*, 2015.

[44] 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)*, vol. 3, no. 5. Citeseer, 2007.

[45] H. Liu, M. Qi, and J. Jiang, "Kernelized relaxed margin components analysis for person re-identification," *Signal Processing Letters, IEEE*, vol. 22, no. 7, pp. 910–914, 2015.

[46] A. Graves, N. Jaitly, and A.-r. Mohamed, "Hybrid speech recognition with deep bidirectional lstm," in *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*. IEEE, 2013, pp. 273–278.

[47] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[48] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[49] Y. Sun, X. Wang, and X. Tang, "Deep learning face representation by joint identification-verification," in *in Advances in Neural Information Processing Systems*. Citeseer, 2014.

[50] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 815–823.

[51] W. Zaremba, I. Sutskever, and O. Vinyals, "Recurrent neural network regularization," *arXiv preprint arXiv:1409.2329*, 2014.

[52] 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*, vol. 32, no. 9, pp. 1627–1645, 2010.

[53] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 248–255.

[54] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," *arXiv preprint arXiv:1408.5093*, 2014.

[55] L. Bottou, "Stochastic gradient tricks," *Neural Networks, Tricks of the Trade, Reloaded*, pp. 430–445, 2012.