DEEPAGENT: AN ALGORITHM INTEGRATION APPROACH FOR PERSON RE-IDENTIFICATION

Fulong Jiao, Bir Bhanu

Center for Research in Intelligent Systems
University of California, Riverside, Riverside, CA 92521, USA

ABSTRACT

Person re-identification(RE-ID) has played a significant role in the fields of image processing and computer vision because of its potential value in practical applications. Researchers are striving to design new algorithms to improve the performance of RE-ID but ignore the advantages of existing approaches. In this paper, motivated by *deep reinforcement learning*, we propose a Deep Agent which can integrate existing algorithms and enable them to complement each other. Two Deep Agents are designed to integrate algorithms for data augmentation and feature extraction parts separately for RE-ID. Experiment results demonstrate that the integrated algorithms can achieve a better accuracy than using each one of them alone.

Index Terms— Identification, Reinforcement learning, Deep Agent, Algorithm integration

1. INTRODUCTION

The goal of person re-identification is to match individuals who appear in different locations at different times in different camera views that do not overlap. Due to the emergence of the large-scale multi-camera tracking systems and the tremendous amount of surveillance and monitoring data, it is challenging to track a person among camera networks by manual monitoring. Hence, an automatic and accurate RE-ID (re-identification) system is required for public safety. However, matching people across cameras is intrinsically difficult [1, 2] due to the visual ambiguities:

- Low resolution: Multi-camera networks cannot capture high resolution images or videos of humans mainly due to (1) the limitation of camera hardware and cost of camera systems and each human subject occupies only a small part of an image, (2) the uncontrollable distance between cameras and human subjects.
- Arbitrary poses: Since the cameras in a public area are located in different locations with different views, and human's behaviors and direction of motion could be arbitrary, the poses of an individual captured by different cameras are usually different.
- Changing illumination: At different times and locations, the images' quality of cameras may be affected seriously by lighting, shadow or weather conditions, so the appearance of an individual may vary due to the change in illumination.

In order to solve the RE-ID problems accompanying with difficulties above-mentioned, a lot of efforts have been made. These efforts mainly focus on three parts: data augmentation, pedestrian description and similarity measurement. But because of the visual ambiguities, the quality of probe and gallery images varies significantly among different datasets. Therefore, one RE-ID algorithm that brings the best matching to a certain class of probe images may be inferior to others [3]. Figure 1 shows an example of matching probe and gallery images with/without image random erasing [4] on Market-1501 dataset. The first column contains two different samples of probe images. In gallery set part, top 10 ranking results are shown to evaluate the matching accuracy. Gallery images marked in green are the correct matchings. The 1st and 3rd rows are with random erasing, while 2nd and 4th rows are without random erasing. This figure shows that random erasing algorithm works well for probe A but not works for probe B.

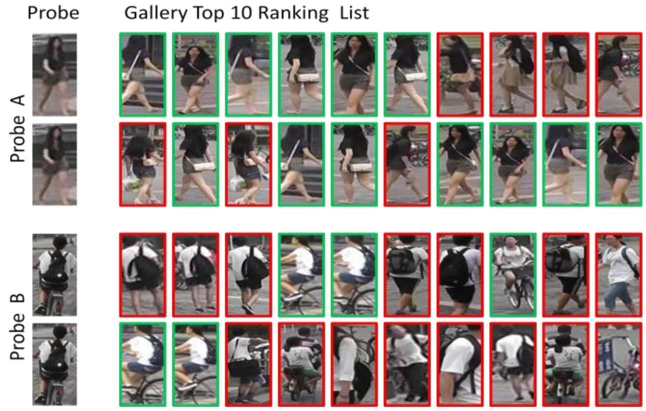

Fig. 1. Example of probes and top 10 ranked results (green indicates the correct matching, red indicates the wrong matching).

In this paper, instead of applying an algorithm for all images and RE-ID datasets, we propose a new Deep Agent which can integrate several algorithms. It can decide which algorithm should be applied to what kinds of images in a

978-1-4799-7061-2/18/$31.00 ©2018 IEEE

dataset. In other words, the proposed Deep Agent builds a mapping mechanism that connects a probe image to an algorithm which can help it in getting a correct match with a higher rate. The Deep Agent is motivated by deep reinforcement learning [5] which could learn a policy to adaptively select the best algorithm.

2. RELATED WORK AND CONTRIBUTIONS

The process of RE-ID can be divided into three parts: data augmentation, pedestrian description and similarity measurement. For data augmentation, Zhong *et al.* [4] proposed a random erasing mechanism to enlarge the dataset and overcome the occlusion problems. McLaughlin *et al.* [6] addressed the issue of dataset bias and improved the cross-dataset generalization by changing the image background. Apart from data augmentation, the focus of most of works has been along two directions: pedestrian description and similarity measurement.

Pedestrian description aims at extracting a set of robust features to describe a person's appearance and making it easier to distinguish from others. In early works, algorithms were designed to extract hand-crafted features as descriptors, such as color, texture and salient regions [7, 8, 9]. In recent years, many researchers have preferred deep learning for feature extraction because of its successful application for image classification. Yi *et al.* [10] employed the Siamese neural network, a symmetry structure with two sub-networks which are connected by a cosine layer, to determine whether a pair of input images belong to the same person. McLaughlin *et al.* [11] combined the Siamese neural network with recurrent convolutional neural network (RCNN) as a new architecture for video based person re-identification. Cheng *et al.* [12] proposed a triplet framework which uses three images as input. The three channels with the same network parameters extract features from three input images where two images are for the same person and the third image is for a different person.

After getting features from an image, we need to measure the similarity between features to decide whether a feature pair represents the same person. Euclidean distance, cosine distance, correlation measurement could be used for measuring the similarity. But due to the visual ambiguities among individuals, a mechanism is needed to keep all the feature vectors of the same person close together while pushing vectors from different person further apart. Motivated by this idea, lots of distance metric learning methods were proposed. Koestinger *et al*. [13] introduced a metric learning method which is called KISSME. It defines the similarity between a pair of images based on a likelihood ratio test. The difference between two individuals is employed and difference space is assumed to be characterized by a zero mean Gaussian distribution. Finally, principal component analysis (PCA) was applied for dimensionality reduction. Shengcai *et al.* [14] proposed a cross-view quadratic discriminant analysis (XQDA) method for mapping the feature vectors in a subspace and reducing the dimension of feature vectors.

In this paper, we propose an algorithm integration framework based on *deep reinforcement learning* which can automatically choose a proper algorithm for specific probe images in the dataset. We show both the competitive results and the improvements in accuracy of the proposed framework for the RE-ID problem.

3. TECHNICAL APPROACH

3.1. Overview

In RE-ID community, most of the researchers strive to design a better algorithm to improve the performance but ignore the fact that a new algorithm usually cannot benefit all the images in a dataset (or across various datasets), and it may be even inferior to some classes of images. In this section, we present how the proposed Deep Agent integrates algorithms by selecting a proper algorithm for each image.

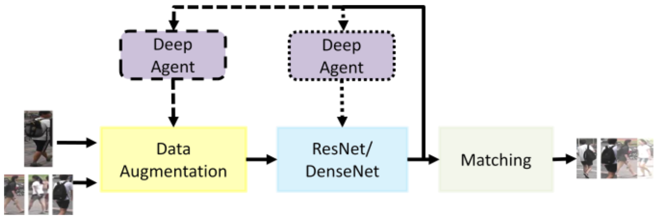

Fig. 2. The system diagram for the Deep Agent RE-ID system

Figure 2 shows the framework of our system, the system is composed of four parts: data augmentation, feature extraction, matching (similarity measurement) and the Deep Agent.

3.2. Deep Agent

The idea of Deep Agent is motivated by deep reinforcement learning [5, 15]. We utilize the main framework of reinforcement learning but make changes for solving RE-ID problems.

Reinforcement learning is the algorithm that enables an agent to learn optimal behavior through trial and error interaction with a dynamic environment [16]. More specifically, an agent tries different actions at each time step *t* to interact with the dynamic environment. After an action is done, the state of the agent will be changed. If the state after an action is better than the previous state, which means it executes a good action, then the agent will receive a positive reward. Otherwise, it will receive a negative reward. The goal of reinforcement learning is to maximize the total reward by operating a series of good actions. The Q function is the key to describe the reinforcement learning:

$$Q_{t+1}(s_t, a_t) = Q_t(s_t, a_t) + \alpha'.(r_{t+1} + \gamma.max_{\forall a \in A} Q_t(s_{t+1}, a_t) - Q_t(s_t, a_t)) \quad (1)$$

An experience tuple $\{s, a, r, s_{t+1}\}$ is defined to summarize a single transition in the environment between two time steps. Parameter s is the agent's state at time t, a is the action executed during transition, r is the reward, s_{t+1} is the state after transition, α' is the learning rate, γ is the discount factor of future rewards, A is the action set. The function above presents the principal idea about the expected maximal sum of rewards that an agent will get if it starts in an initial state and executes the optimal policy thereafter [15].

By the definition of reinforcement learning, state s, action a and reward r are the three most crucial components of an agent. Therefore, we need to design a set of experience tuple $\{s, a, r, s_{t+1}\}$ for the RE-ID system.

State: Since our goal is to design a Deep Agent to improve the RE-ID system, we need to select proper algorithms based on the state of each image. So, the state should be a representation of an image in the dataset. Taking the advantage of deep learning of its successful application in image representation, we employ ResNet-50 network for feature extraction. The output of the network is a 2048 dimensional feature vector.

Action: In this paper, we design two Deep Agents for algorithm selection as two cases, separately. Each agent can select one proper algorithm for the probe image among two candidate algorithms. The first Deep Agent in case one is employed for deciding whether the image should be augmented by random erasing. So, there are two actions the first agent can execute, do random erasing or not. The second Deep Agent in case two is for selecting a pre-trained deep network for feature extraction. The action to be chosen is ResNet-50 [25] or DenseNet-121 [26].

Reward: The reward are calculated by the function (2):

$$r = \frac{1}{position\ (1^{st}\ True\ Matching)} \tag{2}$$

Training images are divided into 3 parts as 'training', 'probe' and 'gallery' to train the Deep Agent. 'Probe' and 'gallery' data are used for evaluating the trained neural networks. After evaluation, we can get the ranking results for each 'probe' images and its feature vectors. Based on the function (2), we find the position of the first true matching in ranking list and set the reciprocal of the position as the reward r.

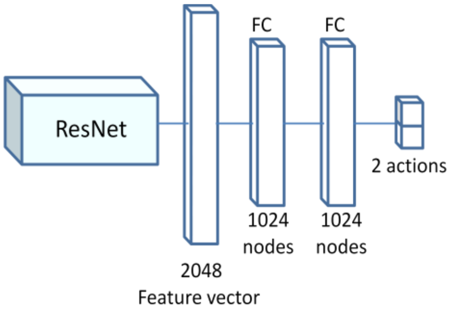

Fig. 3. Architecture of Deep Agent

The architecture of Deep Agent is shown in Figure 3. A pre-trained ResNet-50 is employed as the feature extractor. During training, we execute two different actions for every image and compare the rewards gained after executing the two actions. Deep Agent will label the action which results in more rewards as a good action. For each image, the 2048 dimensional feature vector and its good action label will be fed into the final two fully connected layers to train the Deep Agent. Finally, the Deep Agent will have ability to predict the good action based on an input image's feature vector.

3.2.1 Training: There are two parts in training. The first part is to train the Deep Agent. After preprocessing (see Section 4) the images, training data are split into three parts as 'training', 'probe' and 'gallery'. The 'training' images are fed into a pre-trained deep neural network for tuning parameters. To train one Deep Agent, we need to repeat training the network several times and each time with a different algorithm. The number of training times depends on how many algorithms we provide the Deep Agent for selection. For example, we provide the Deep Agent with two options in case one, do random erasing for data augmentation or not. So, the action set A = {random erasing, no random erasing}. We feed the neural network with original data at first time. For the second training time, we feed the network with data augmented by random erasing. Then, the 'probe' and 'gallery' images are used for evaluating training results twice and calculating the matching results and the rewards r based on equation (2). By comparing the rewards r for every probe image, we label the action which maximized the rewards as a good action. An arbitrary action is labeled if two rewards are the same. The recorded actions are ordered in the same sequence as the probe images, we call it action label list. Now, we have got two sets of probe images' feature vectors and choose one of them as its state. Finally, the probe's state and its corresponding action label list are fed into Deep Agent for training. The trained Deep Agent will enable to select the algorithm which can get more rewards for each probe image.

During the second part, we use the entire training data to train the network twice again, feeding data with random erasing during the first time, and feeding data without random erasing in the second time, then new action set A is created for the testing stage.

3.2.2 Testing: After the training stage, we will get one Deep Agent and two trained networks which are trained by random erased data and trained by original data as two actions. The trained Deep Agent in testing stage could first predict which action will enable the probe image to get more rewards, then feeding the probe image into one of them, random erased data network or the other one. So, the two trained networks will be integrated as one feature extractor for the dataset. The final result will be calculated by measuring the similarity of the features of probe and gallery images.

4. EXPERIMENTAL RESULTS

4.1. Dataset

In this section we validate our approach with two different public benchmark datasets, Market-1501 [17] and DukeMTMC-reID [18]. The two datasets are chosen because they provide a large scale image data with sufficient labeled information.

Market-1501 is an image-based RE-ID dataset with 32668 labeled bounding boxes of 1501 identities captured by 6 cameras in different viewpoints. Deformable Part Model (DPM) [19] is employed as the pedestrian detector. The dataset is split into two parts: 12936 images of 751 identities for training and 19732 images of 750 identities for testing. A subset which contains 3368 images of 750 identities from testing images is collected as a probe set to evaluate the RE-ID matching result.

DukeMTMC-reID is a large scale image-based RE-ID dataset which has been extracted from an 85-minute high resolution video data taken by 8 different cameras. It contains 16522 images of 702 identities for training, 2228 images of other 702 identities as probe set and 17661 images for gallery set.

Evaluation Criteria: In the experiment, we use cumulative match characteristic (CMC) [20] to evaluate our method, which validates the RE-ID accuracy by checking how many correct matches are there in the top n ranks.

4.2.Experimental Setup

The size of images are varying for both of the datasets. So, we resize all images to 256×128 in *data pre-processing stage.* As shown in Figure 3, the Deep Agent consists of 2 fully connected layers of 1024 dimension and a 2 dimensional output layers. ReLU as the activation function is added between fully connected layers. The dropout rate is set to 0.2. We choose a CNN model in [21] as the baseline RE-ID algorithm. During the first case, we train the Deep Agent for selecting random erasing or not and we use ResNet-50 as feature extractor. The learning rate is 0.01 and 0.1 for base layers and new fully connected layers, respectively. In the second case, we aim to train the Deep Agent for selecting feature extractors among ResNet-50 and DenseNet-121, they have the same parameters as above. In this case, we don't do any data augmentation. All the networks are trained for 60 epochs. We choose Euclidean distance to measure the similarity between probe and gallery images.

4.3. Experimental Results on Market-1501

We first evaluate our approach on Market-1501 dataset. Results of several related methods are shown in Table 1. Our Deep Agent in case one integrates the Baseline + Res. (87.3%) and Baseline + Res. + Er (88.9%) and gets 91.1% recognition rate. The Deep Agent in case two gets 89.0% recognition rate by integrating ResNet-50 (87.3%) with DenseNet-121 (88.7%). So, both Deep Agents outperform the other state-of-the-art RE-ID algorithms in Table 1.

Methods	Rank 1	Rank 5	Rank 10
LSRO [18]	88.4	-	-
SVDNet [23]	82.3	92.3	95.2
SVDNet + Er. + Re. [22]	89.1	-	-
PAN [24]	82.8	93.5	-
Baseline + Res. [21]	87.3	95.0	97.0
Baseline + Res. + Er. [4]	88.9	96.1	97.4
Baseline + Dense. [21]	88.7	**96.2**	**97.8**
Ours 1	**91.1**	95.2	96.5
Ours 2	89.0	95.9	97.1

Table 1. The comparison of the top ranked recognition rates (%) on Market-1501 dataset. Er.: random erasing [4], Re.: re-rank [22], Res: ResNet-50, Dense: DenseNet-121. Ours 1: the Deep Agent in case one that integrates two ResNet-50 which are trained with and without random erasing. Ours 2: the Deep Agent in case two that integrates ResNet-50 and DenseNet-121.

4.4. Experimental Results on DukeMTMC-reID

Table 2 shows the experiment results on DukeMTMC-reID dataset. We achieve 67.0% rank-1 recognition rate in the first case and 67.2% rank-1 recognition rate in the second case. The accuracy is lower than other related state-of-the-art algorithms, but we still get a consistent improvement as compared to the baseline algorithms that we integrate. From the results, we can get that the accuracy of the Deep Agent depends heavily on the baseline's accuracy.

Methods	Rank 1	Rank 5	Rank 10
LSRO [18]	67.6	-	-
SVDNet [23]	76.7	**86.4**	**89.9**
SVDNet + Er. + Re. [22]	**84.0**	-	-
PAN [24]	71.5	83.8	-
Baseline + Res. [21]	65.7	80.6	86.2
Baseline + Res. + Er. [4]	65.3	80.5	85.6
Baseline + Dense. [21]	64.5	78.9	84.1
Ours 1	67.0	83.7	86.5
Ours 2	67.2	82.6	87.7

Table 2. The comparison of the top ranked recognition rates (%) on DukeMTMC-reID dataset. The name explanations are same as Table 1.

5. CONCLUSIONS

In this paper, we proposed a Deep Agent, an algorithm integration approach for person re-identification. We take advantage of the synergetic contribution of integrating multiple RE-ID methods and design a new framework for integrating data augmentation or feature extraction algorithms. Based on the principle of deep reinforcement learning, the Deep Agent can adaptively select the algorithms depending on the different probe images. The experiment results show that our Deep Agent can achieve a competitive recognition rate with respect to the previous work.

6. ACKNOWLEDGMENT

This work was supported in part by NSF grant 1330110 and ONR grant N00014-12-1-1026. The contents of the information do not reflect the position or policy of US Government.

7. REFERENCES

[1] An, L., Kafai, M., Yang, S., & Bhanu, B. (2016). Person reidentification with reference descriptor. *IEEE Transactions on Circuits and Systems for Video Technology*, *26*(4), 776-787.

[2] Thakoor, N., & Bhanu, B. (2013, October). Context-aware reinforcement learning for re-identification in a video network. In *Distributed Smart Cameras (ICDSC), 2013 Seventh International Conference on* (pp. 1-6). IEEE.

[3] Yin, P. Y., Bhanu, B., Chang, K. C., & Dong, A. (2005). Integrating relevance feedback techniques for image retrieval using reinforcement learning. *IEEE transactions on pattern analysis and machine intelligence*, *27*(10), 1536-1551.

[4] Zhong, Z., Zheng, L., Kang, G., Li, S., & Yang, Y. (2017). Random Erasing Data Augmentation. *arXiv preprint arXiv:1708.04896*.

[5] Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., ... & Petersen, S. (2015). Human-level control through deep reinforcement learning. *Nature*, *518*(7540), 529.

[6] McLaughlin, N., Del Rincon, J. M., & Miller, P. (2015, August). Data-augmentation for reducing dataset bias in person reidentification. In *Advanced Video and Signal Based Surveillance (AVSS), 2015 12th IEEE International Conference on* (pp. 1-6). IEEE.

[7] Gray, D., & Tao, H. (2008, October). Viewpoint invariant pedestrian recognition with an ensemble of localized features. In *European conference on computer vision* (pp. 262-275). Springer, Berlin, Heidelberg.

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

[9] Zhao, R., Ouyang, W., & Wang, X. (2013, June). Unsupervised salience learning for person re-identification. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on* (pp. 3586-3593). IEEE.

[10] Yi, D., Lei, Z., Liao, S., & Li, S. Z. (2014, August). Deep metric learning for person re-identification. In *Pattern Recognition (ICPR), 2014 22nd International Conference on* (pp. 34-39). IEEE.

[11] McLaughlin, N., Martinez del Rincon, J., & Miller, P. (2016). Recurrent convolutional network for video-based person reidentification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1325-1334).

[12] Cheng, D., Gong, Y., Zhou, S., Wang, J., & Zheng, N. (2016). Person re-identification by multi-channel parts-based cnn with improved triplet loss function. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1335-1344).

[13] Koestinger, M., Hirzer, M., Wohlhart, P., Roth, P. M., & Bischof, H. (2012, June). Large scale metric learning from equivalence constraints. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on* (pp. 2288-2295). IEEE.

[14] Liao, S., Hu, Y., Zhu, X., & Li, S. Z. (2015). Person reidentification by local maximal occurrence representation and metric learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2197-2206).

[15] Sutton, R. S., & Barto, A. G. (1998). *Reinforcement learning: An introduction* (Vol. 1, No. 1). Cambridge: MIT press.

[16] Kaelbling, L. P., Littman, M. L., & Moore, A. W. (1996). Reinforcement learning: A survey. *Journal of artificial intelligence research*, *4*, 237-285.

[17] Zheng, L., Shen, L., Tian, L., Wang, S., Wang, J., & Tian, Q. (2015). Scalable person re-identification: A benchmark. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 1116-1124).

[18] Zheng, Z., Zheng, L., & Yang, Y. (2017). Unlabeled samples generated by gan improve the person re-identification baseline in vitro. *arXiv preprint arXiv:1701.07717*, *3*.

[19] Felzenszwalb, P. F., Girshick, R. B., McAllester, D., & Ramanan, D. (2010). Object detection with discriminatively trained part-based models. *IEEE transactions on pattern analysis and machine intelligence*, *32*(9), 1627-1645.

[20] Moon, H., & Phillips, P. J. (2001). Computational and performance aspects of PCA-based face-recognition algorithms. *Perception*, *30*(3), 303-321.

[21] Hermans, A., Beyer, L., & Leibe, B. (2017). In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*.

[22] Zhong, Z., Zheng, L., Cao, D., & Li, S. (2017, July). Reranking person re-identification with k-reciprocal encoding. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on* (pp. 3652-3661). IEEE.

[23] Sun, Y., Zheng, L., Deng, W., & Wang, S. (2017). Svdnet for pedestrian retrieval. *arXiv preprint*.

[24] Zheng, Z., Zheng, L., & Yang, Y. (2017). Pedestrian alignment network for large-scale person re-identification. *arXiv preprint arXiv:1707.00408*.

[25] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

[26] Huang, G., Liu, Z., Weinberger, K. Q., & van der Maaten, L. (2017, July). Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (Vol. 1, No. 2, p. 3).