Multi-shot Pedestrian Re-identification via Sequential Decision Making

Jianfu Zhang[1], Naiyan Wang[2], and Liqing zhang[1]

[1]Department of Computer Science and Engineering, Shanghai Jiao Tong University
[2]Tusimple
c.sis@sjtu.edu.cn,winsty@gmail.com,zhang-lq@cs.sjtu.edu.cn

Abstract

Multi-shot pedestrian re-identification problem is at the core of surveillance video analysis. It matches two tracks of pedestrians from different cameras. In contrary to existing works that aggregate single frames features by time series model such as recurrent neural network, in this paper, we propose an interpretable reinforcement learning based approach to this problem. Particularly, we train an agent to verify a pair of images at each time. The agent could choose to output the result (same or different) or request another pair of images to see (unsure). By this way, our model implicitly learns the difficulty of image pairs, and postpone the decision when the model does not accumulate enough evidence. Moreover, by adjusting the reward for unsure action, we can easily trade off between speed and accuracy. In three open benchmarks, our method are competitive with the state-of-the-art methods while only using 3% to 6% images. These promising results demonstrate that our method is favorable in both efficiency and performance.

1. Introduction

Pedestrian Re-identification (re-id) aims at matching pedestrians in different tracks from multiple cameras. It helps to recover the trajectory of a certain person in a broad area across different non-overlapping cameras. Thus, it is a fundamental task in a wide range of applications such as video surveillance for security and sports video analysis. The most popular setting for this task is single shot re-id, which judges whether two single frames are the same person. This setting has been extensively studied in recent years[8, 1, 16, 27, 17]. On the other hand, multi-shot re-id (or a more strict setting, video based re-id) is a more realistic setting in practice, however it is still at its early age compared with single shot re-id task.

Currently, the main stream of solving multi-shot re-id task is first to extract features from single frames, and then aggregate these image level features. Consequently, the key lies in how to leverage the rich yet possibly redundant and noisy information resides in multiple frames to build track level features from image level features. A common choice is pooling[36] or bag of words[37]. Furthermore, if the input tracks are videos (namely, the temporal order of frames is preserved), optical flow[5] or recurrent neural network (RNN)[23, 38] are commonly adopted to utilize the motion cues. However, most of these methods have two main problems: the first one is that it is computationally inefficient to use all the frames in each track due to the redundancy of each track. The second one is there could be noisy frames caused by occlusion, blur or incorrect detections. These noisy frames may significantly deteriorate the performance.

Figure 1: Examples to demonstrate the motivation of our work. For most track pairs, several even only one pair of images are enough to make confident prediction. However, when there exists occlusions, blur or other hard cases, it is necessary to use more pairs, and alleviate the influences of these samples of bad quality.

To solve the aforementioned problems, we formulate multi-shot re-id problem as a sequential decision making task. Intuitively, if the agent is confident enough about existing evidences, it could output the result immediately. Otherwise, it needs to ask for another pair to verify. To model such human like decision process, we feed a pair of images from the two tracks to a verification agent at each

time step. Then, the agent could output one of three actions: *same*, *different* or *unsure*. By adjusting the rewards of these three actions, we could trade off between the number of pairs used and final accuracy. We depict several examples in Fig. 1. In case of easy examples, the agent could decide using only one pair of images, while when the cases are hard, the agent chooses to see more pairs to accumulate evidences. In contrast to previous works that explicitly deduplicate redundant frames[6] or distinguish high quality from low quality frames[21], our method could implicitly consider these factors in a data driven end-to-end manner. Moreover, our method is general enough to accommodate all single shot re-id methods as image level feature extractor even those non-deep learning based methods.

The main contributions of our work are listed as following:

- We are the first to introduce reinforcement learning into multi-shot re-id problem. We train an agent to either output results or request to see more samples. Thus, the agent could early stop or postpone the decision as needed. Thanks to this behavior, we could balance speed and accuracy by only adjusting the rewards.
- We verify the effectiveness and efficiency on three popular multi-shot re-id dataset. Along with the deliberately designed image feature extractor, our method could outperform the state-of-the-art methods while only using 3% to 6% images without resorting to other post-processing or additional metric learning methods.
- We empirically demonstrate that the Q function could implicitly indicate the difficulties of samples. This desirable property makes the results of our method more interpretable.

2. Related Work

Pedestrian re-identification for single still images has been explored extensively in these years. These researches mainly focused on two aspects: the first one is to extract features that are both invariant and discriminative from different viewpoints to overcome illumination changes, occlusions, blurs issues, etc. Representative works before deep learning age include [29, 14, 35]. However, these handcrafted features are subverted by the rapidly developed Convolutional Neural Networks (CNN) in recent years. CNN has become *de facto* standard for feature extraction. The second aspect is metric learning. Metric learning embeds each sample to a latent space that preserves certain relationships of samples. Popular methods including Mahalanobis distance metric (RCA)[2], Locally Adaptive Decision Function (LADF)[18] and Large Margin Nearest Neighbor (LMNN)[30].

These two streams have intersected in the deep learning age: Numerous work focus on learning discriminative features by the guide of metric learning based loss funcions. The earliest work was proposed by Chopra *et al.* in [4]. They presented a method based on Siamese architecture to learn similarity for face verification task with CNN. Schroff *et al.* [25] proposed FaceNet model to learn embeddings by comparing the triplet loss to maximize the relative distance for matched pairs and the mismatched pairs. Inspired by these methods for face verification, deep learning methods for image based re-identification also have shown a great progress in recent years[8, 16, 1]. Recently, some methods utilized domain knowledge to improve the performance. In [33, 34] pedestrian landmarks are included to handle body part misalignment problem. On the other hand, many multi-task methods are proposed based on deep learning and reported promising performance. Wang *et al.* [27] proposed a joint learning framework by combining the patch matching and metric learning methods. Li *et al.* [17] proposed a multi-loss model combining metric learning and global classification to discover the local and global features.

Compared with image based re-id task, multi-shot re-id problem is a more realistic setting, since the most popular application of re-id problem is surveillance video. It at least provides several representative frames after condensation, or even the entire videos are stored. Consequently, how to utilize such multi-frame information is at the core of multi-shot re-id. Flow Energy Profile[19] is proposed to detect walking cycles with flow energy profile to extract spatial and temporal invariant features. In [37], Bag-of-words are adopted with learned frame-wised features to generate a global feature. Not surprisingly, deep learning also expressed its power in multi-shot re-id problem. A natural choice for temporal model in deep learning is Recurrent Neural Network (RNN). In the pioneering work [23], McLaughlin *et al.* first extracted features with CNN from images and then use RNN and temporal pooling to aggregate those features. Similarly, Chung *et al.* [5] presented a two stream method using Siamese network for each stream and use RNN and temporal pooling for each stream. Recently, this idea is extended with spatial and temporal attention in [38, 32] to automatically pick out discriminative frames and integrate context information. Another interesting work is [21]. In [21], a CNN model learns the quality for each image. And then the video is aggregated with the image features weighted by the quality.

The goal of Reinforcement Learning (RL) is to learn policies based on trial and error in a dynamic environment. In contrast to traditional supervised learning, reinforcement learning trains an agent by maximizing the accumulated reward from environment. Additional to its traditional applications in control and robotics, recently RL has been successfully applied to a few computer vision tasks by treating

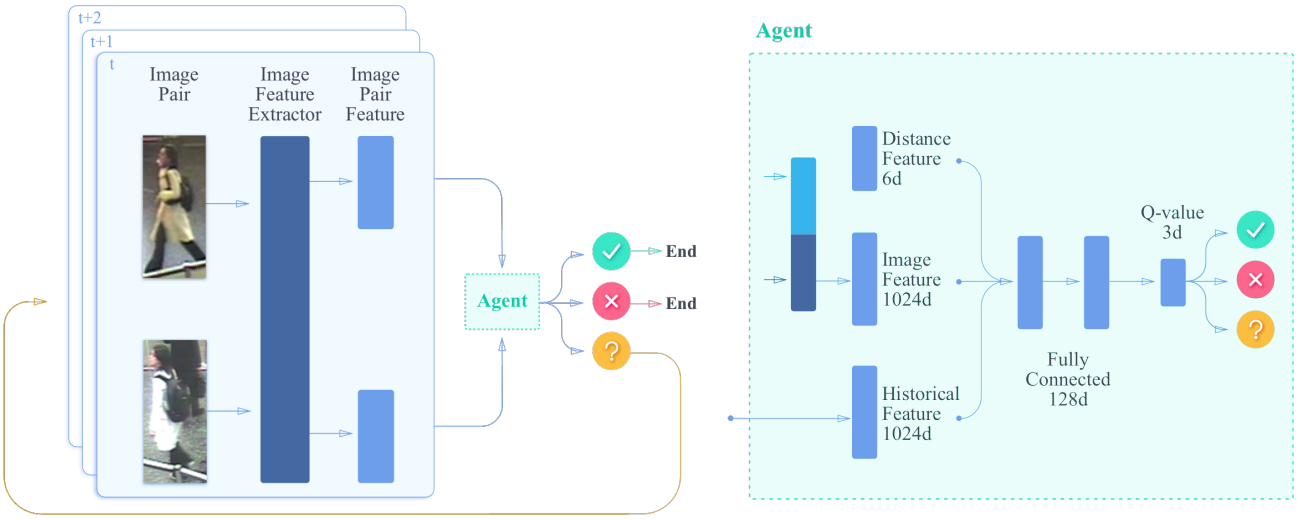

Figure 2: An illustration of our proposed method.

them as a decision making process[3, 22, 11, 15]. In [11], the features for visual tracking problem are organized according to their costs, and then an agent is trained to decide current features are good enough to make accurate prediction or need to proceed to next level of features. By this way, the agent saves unnecessary computation of expensive features. In [15], RL is applied to learn attention for detected bounding boxes by iteratively detecting and removing irrelevant pixels in each image. Our method shares the same spirit as these works, but tailored for multi-shot re-id problem.

3. Method

In this section, we will introduce our approach to multi-shot re-id problem. First, we will start with a formal formulation of this problem, and then present each component of our method. The overview of our method is depicted in Figure 2.

3.1. Formulation

In multi-shot re-id task, we are given two sequences $(\mathcal{X}, \mathcal{Y}) = (\{x_1, \ldots, x_n\}, \{y_1, \ldots, y_n\})$, where x_1 represents the first image in $\mathcal{X}$. We compute the distance (or similarity) of these two sequences by:

$$D(\mathcal{X}, \mathcal{Y}) = \|g(f(x_1), \ldots, f(x_n)) - g(f(y_1), \ldots, f(y_n))\|_2^2. \quad (1)$$

Here $f(\cdot)$ is a feature extractor that extracts discriminative feature for each frame, and $g(\cdot)$ is an aggregation function that aggregates image level features to sequence level feature. We then use it to rank all the queries.

In the sequel, we will first present the details of our single image feature extractor $f(\cdot)$ in Sec. 3.2. It is built with a CNN trained with three different loss functions. Next, we elaborate our reinforcement learning based aggregation method $g(\cdot)$ in Sec. 3.3.

3.2. Image Level Feature Extraction

For single image feature extractor, a CNN is trained to embed an image into a latent space that preserves the relative relationships of samples. To achieve this goal, we train the CNN with combination of three different kinds of loss functions: classification loss, pairwise verification loss [4] and triplet verification loss [25]. According to a recent work [31], multiple loss functions could better ensure the structure of the latent space and margins between samples. Particularly, we optimize large margin softmax loss[20] instead of softmax loss, since it demonstrates extraordinary performance in various classification and verification tasks.

Implementation details: We use two well-known network structures Inception-BN[12] and AlexNet [13] pre-trained on Imagenet[13] as the base networks. We choose these two networks with different capacity and expression power to demonstrate the universality of our proposed aggregation method that will be presented next. In specific, we use the flattened features of the last pooling layer from base networks. we set the margin in triplet loss to 0.9. For large margin softmax, we set $\beta = 1000$, $\beta_{min} = 3$, and the margin as 3. For more details of these parameters, please refer to [20]. We optimize the network by momentum SGD optimizer with 320000 iterations, the learning rate is 0.01

and multiplied by 0.1 after 50000 and 75000 iterations.

As an important baseline, we use simple average pooling of the l2-normalized features generated from all the images as the feature for a sequence. Namely, the aggregation function is defined as:

$$g(\mathcal{X}) = \sum_{i}^{n} \frac{f(x_i)}{n} \tag{2}$$

3.3. Sequence Level Feature Aggregation

We train an agent to learn the aggregation function $g(\cdot)$. It jointly decides the results and whether additional image pairs are needed to make the final decision. The problem is formulated as a Markov Decision Processes (MDP), described by $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R})$ as the states, actions, transitions and rewards. Each time step t, the agent will get a random image pair from the two input sequences to observe a state $s_t \in \mathcal{S}$ and choose an action $a_t \in \mathcal{A}$ from the experience it has learned. Then the agent will receive a reward $r_t \in \mathcal{R}$ from the environment in training and determine the next state s_{t+1}. We will elaborate the details of them in the sequel.

Actions: We have three actions for the agent: *same*, *different* and *unsure*. The first two actions will terminate current episode, and output the result immediately. We anticipate when the agent has collected enough information and is confident to make the decision, it stops early to avoid unnecessary computation. If the agent chooses to take action *unsure*, we will feed the agent with another random image pair from the two input sequences.

Rewards: We define the rewards as follows:

1. $+1$, if a_t matches gt.
2. -1, if a_t differs from gt, or when $t = t_{max}$, a_t is still *unsure*.
3. r_p, if $t < t_{max}$, a_t is *unsure*.

Here t_{max} is defined as the maximum time step for each episode. gt is the ground truth. r_p is defined as a penalty (negative reward) or reward for the agent seeking for another image pair. If r_p is negative, it will be penalized if requesting more pairs; on the other hand, if r_p is positive, we encourage the agent to gather more pairs, and stop gathering when it has collected t_{max} pairs to avoid a penalty of -1. The value of r_p may strongly affect the agent's behavior, we will discuss its impact in Sec. 4.3.

States and Deep Q-learning: We use Deep Q-Learning[24] to find the optimal policy. For each state and action (s_t, a_t), $Q(s_t, a_t)$ represents the optimal discounted accumulated rewards for the state and action. In training, we could iteratively update the Q function by:

$$Q(s_t, a_t) = r_t + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}). \tag{3}$$

The state s_t for time step t in the episode consists of three parts. The first part is the image features of current pair $(f(x), f(y))$ which are generated by the image feature extractor mentioned in Section 3.2. The second part is a weighted average of historical image features. This part makes the agent be aware of the previous image pairs it has already seen before. In specific, for each observation o_t the weight w_t is defined as:

$$w_t = 1.0 - \frac{e^{Q_u}}{e^{Q_s} + e^{Q_d} + e^{Q_u}} \tag{4}$$

where Q_u is short for $Q(s_t, a_t = unsure)$, and vice versa. The weight decreases as Q_u increases, as higher Q_u may indicate that current pair of images are hard to distinguish. The aggregated features should be affected as small as possible. As a result, h_t is the weighted average of the historical features for $t > 1$:

$$h_t = \frac{\sum_{i=1}^{t-1} w_i \times o_i}{\sum_{i=1}^{t-1} w_i}. \tag{5}$$

$h_t = o_t$ when $t = 1$. Note that though the Q function is not specifically trained for sample weighting, it still reflects the importance of each frame. We leave end-to-end learning of the weights as our future work.

We also augment the image features with hand-crafted features for better discrimination. For each time step t, we calculate the distance $\|f(x_i) - f(y_j)\|_2^2$ and inner product $f(x_i) \cdot f(y_j)$ for all $1 \leq i, j \leq t-1$, and then add the maximum, minimum and mean of them to the input, which results in 6 dimension extra features.

The network structure is shown in Fig.2. We simply use a two layer fully connected network as the Q function. Each fully connected layer has 128 outputs.

Testing: For each query video sequences we play one episode and take the difference of the Q-value of action *same* and *different* at the terminal step as the final score. Then we rank the gallery identities by this value.

Implementation details: In training phase, for each episode we randomly choose positive or negative sequence pairs with the ratio $1:1$. We feed the weighted historical features, features of current step and hand-crafted distance features into the Q-Net. The whole net along with the single image feature extractor is trained end-to-end except for fixing the first two stages of the base networks.

We train the Q-Net for 20 epochs by momentum SGD optimizer, 100000 iterations for each epoch. We use ϵ-learning[26] as the exploration strategy and anneal ϵ linearly from 1 to 0.1 in the first 10 epochs. Learning rate is

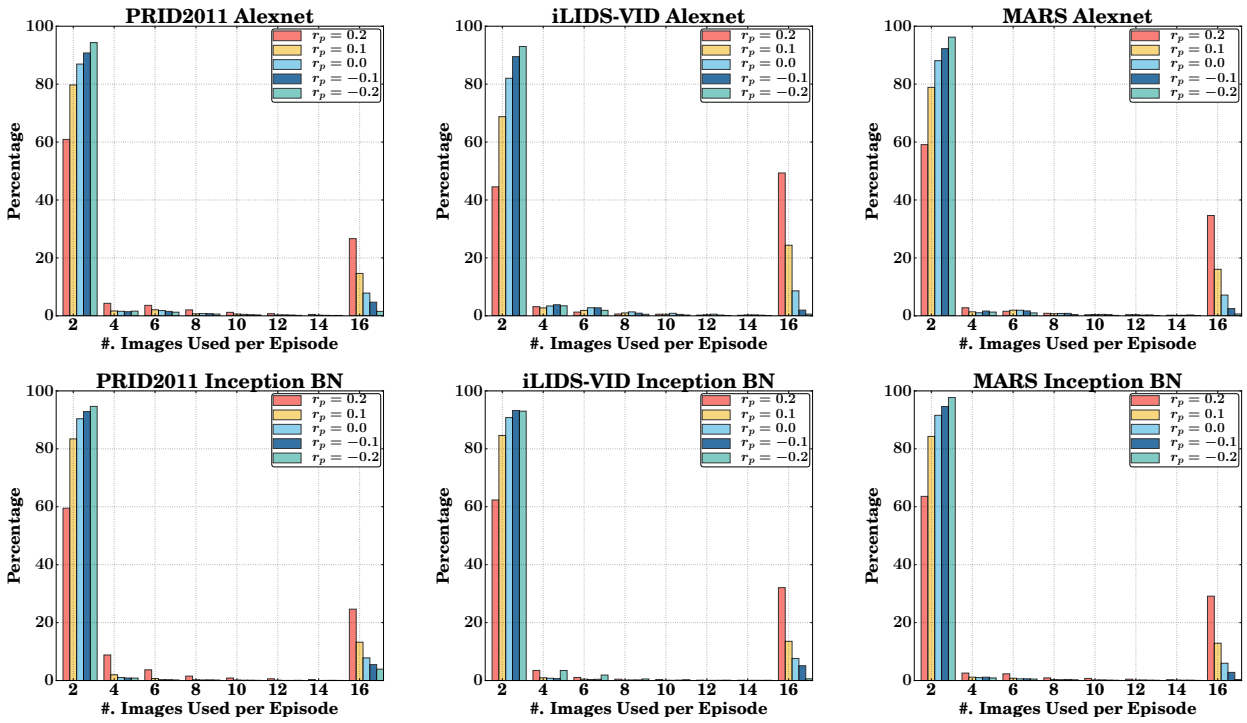

Figure 3: Statistics of the number of images used in each episode for our model with different reward for action *unsure*.

Dataset	PRID2011		iLIDS-VID		MARS	
Settings	CMC1	#.of Images	CMC1	#.of Images	CMC1	#.of Images
Baseline	84.3	200.000	60.0	146.000	68.3	111.838
$r_p = 0.2$	85.2	6.035	60.2	6.681	71.2	6.417
$r_p = 0.1$	84.6	3.970	60.3	3.966	70.5	3.931
$r_p = 0$	83.7	3.162	55.4	3.134	69.0	2.952
$r_p = -0.1$	81.9	2.835	54.0	2.789	68.2	2.507
$r_p = -0.2$	80.8	2.605	50.7	2.307	67.5	2.130

Table 1: Test results for our model based on Inception BN image feature extractor.

set to 0.0001, discount factor $\gamma = 0.9$ and batch size is 16. Experience replay is used and the memory buffer size is set to be 5000. In our implementation, we set the maximum time step for each episode $t_{max} = 8$.

4. Experiments

In this section, we will present the results of our method on three open benchmarks, and compare it with other state-of-the-art methods. We will first introduce the datasets and evaluation metric used, and then present the ablation analyses of our method. After comparisons with other methods, we will also present some qualitative results to interpret the mechanism of our methods.

4.1. Datasets

The iLIDS-VID dataset[28] contains 300 identities with 600 image sequences from two cameras. The length for each image sequence ranges from 23 to 192 frames. The challenge of this dataset is mainly due to severe occlusion. The bounding boxes are human annotated.

The PRID2011[10] dataset consists two cameras with 385 identities in camera A and 749 identities in camera B. 200 identities appear in both camera. The length for each image sequence varies from 5 to 675 frames. Same as iLIDS-VID, the bounding boxes are labeled by human.

The Motion Analysis and Re-identification Set (MARS)[36] is a recently released large scale dataset

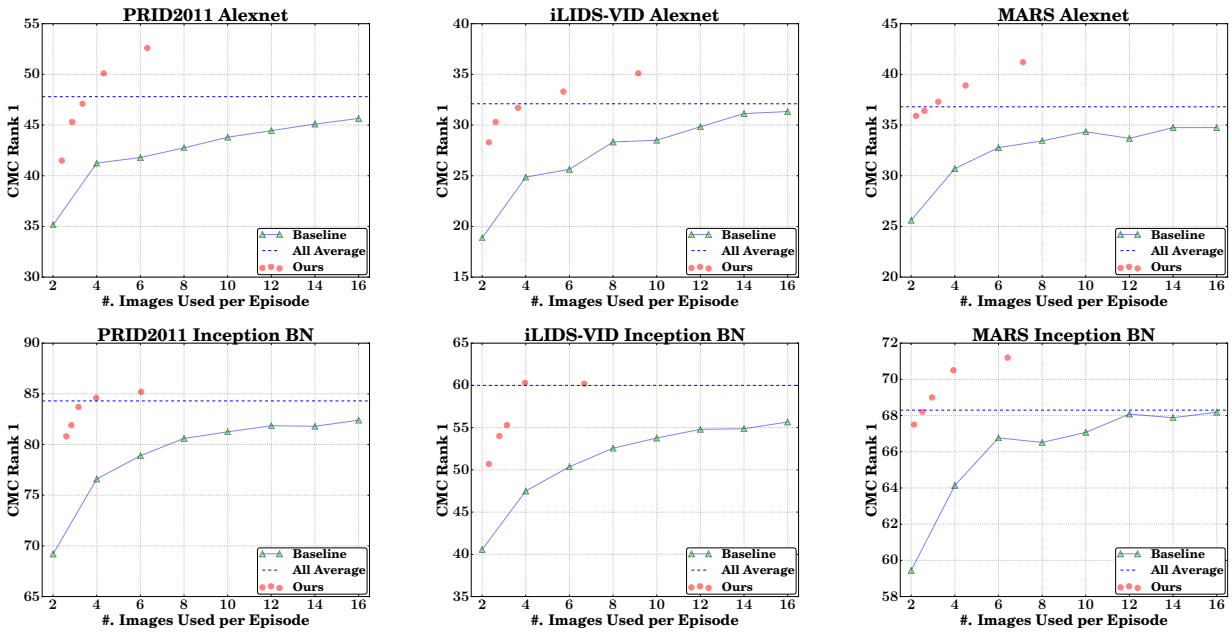

Figure 4: CMC Rank 1 result for our model compared with baseline.

Dataset	PRID2011		iLIDS-VID		MARS	
Settings	CMC1	#.of Images	CMC1	#.of Images	CMC1	#.of Images
Baseline	47.8	200.000	32.1	146.000	36.8	111.838
$r_p = 0.2$	52.6	6.316	35.1	9.154	41.2	7.119
$r_p = 0.1$	50.1	4.317	33.3	5.722	38.9	4.491
$r_p = 0$	47.1	3.349	31.7	3.637	37.3	3.238
$r_p = -0.1$	45.3	2.870	30.3	2.614	36.4	2.604
$r_p = -0.2$	41.5	2.394	28.3	2.307	35.9	2.221

Table 2: Test results for our model based on Alexnet image feature extractor.

containing 1261 identities and 20715 tracklets under 6 different camera views. Bounding boxes are generated by GMMCP[7] tracker and Deformable Part Model (DPM)[9] pedestrian detector, which makes it quite noisy yet close to real applications.

4.2. Evaluation Settings

For iLIDS-VID and PRID2011 dataset, we randomly split the dataset half-half for training and testing. We average the results of 10 runs to make the evaluation stable. For MARS dataset, we follow the setting by the authors of the dataset. 625 identities are used for training, and the rest are used for testing. In testing, 1980 tracklets are preserved for query sets, while the rests are used as gallery sets.

To evaluate performance for each algorithm, we report the Cumulative Matching Characteristic (CMC) metric. It represents the expectation of the true matching hits in the first top-n ranking. Here we use $n \in \{1, 5, 10, 20\}$ in the evaluations.

4.3. Ablation Studies

Before comparing our models with prior works, we first conduct ablation studies of some important factors of our method. We compare the CMC Rank 1 results of our proposed models with baseline methods in Figure 4. For baseline results we calculate the average pooling feature mentioned in Equation 2. The dashed blue line in the figure denotes the results of average pooling of all pairs, while the green triangle denotes the setting that we randomly sample pairs from the tracks, and then average pool their features to a track level feature. We vary the number of pairs sampled to generate the curve.

Dataset	PRID2011				iLIDS-VID				MARS			
CMC Rank	1	5	10	20	1	5	10	20	1	5	10	20
RNN-CNN[23]	70	90	95	97	58	87	91	96	40	64	70	77
ASTPN[32]	77	95	99	99	62	86	94	98	44	70	74	81
Two-Stream[5]	78	94	97	99	60	86	93	97	-	-	-	-
CNN+XQDA[37]	77.9	93.5	-	99.3	53.0	81.4	-	95.1	65.3	82.0	-	89.0
Alexnet (All frames)	47.8	74.4	83.6	91.2	32.1	59.0	70.0	80.6	36.8	53.1	61.6	68.8
Alexnet + Ours	52.6	81.3	88.4	96.3	35.1	61.3	72.1	84.0	41.2	55.6	63.1	73.3
Inception-BN (All frames)	84.3	96.5	98.8	99.7	60.0	85.4	92.0	96.3	68.3	83.5	88.0	90.8
Inception-BN + Ours	85.2	97.1	98.9	99.6	60.2	84.7	91.7	95.2	71.2	85.7	91.8	94.3
QAN[21]	90.3	98.2	99.3	100	68.0	86.8	95.4	97.4	-	-	-	-
STRN[38]	79.4	94.4	-	99.3	55.2	86.5	-	97.0	70.6	90.0	-	97.6

Table 3: Comparisons with other state-of-the-art methods. Please note that some results are not directly comparable due to different setting. For more details, please refer to the text.

We first evaluate the choice of network for the image feature extractor. We use two different networks: Alexnet[13] and Inception-BN[12] to test our models. The results are listed in Table 1 and Table 2. In specific, we improve over the baseline of AlexNet remarkably, while we achieve a good performance compared with state-of-the-art methods with Inception-BN. It is worthy noting that our method also outperforms the baseline that uses all pairs. We owe the reason to that the average pooling of all the pairs may be easily contaminated by some imperfect frames. We then take a close look of the analysis of the number of pairs used in these two networks. Not surprisingly, our method uses notably less number of images. Particularly, *we achieve competitive or even better results than state-of-the-art models using only 3% to 6% image pairs.* Compared with Alexnet, Inception-BN provides more accurate image level discriminative features, therefore the agent tends to make a quick decision. The agent chooses more samples to stop at the end of a episode in AlexNet.

Another important parameter of our model is the reward for unsure action r_p. We show the statistics of the length of each episode in Figure 3 and corresponding CMC rank 1 in Figure 4. When r_p is small (negative), the agent will stop early and verify the identities with fewer images. When r_p is big (positive), the agent will be encouraged to be more cautious, requesting more image pairs for better performance. This will help the agent postpone its decision to avoid mistakes caused by imperfect quality like occlusions. Next, we compare across different datasets. There are tons of occlusions in iLIDS-VID and MARS datasets. Moreover, there are many mislabeled samples in MARS since the bounding boxes for MARS dataset are machine generated. PRID2011 dataset is much easier with few occlusions. Comparing among different datasets, we find that the agent tends to ask for more images in iLIDS-VID and MARS dataset than PRID2011 dataset under the same setting. These two findings coincide with our anticipated behavior of the agent.

4.4. Comparisons with State-of-the-art Methods

Table 3 summarizes the CMC results of our model and the state-of-the-art multi-shot re-id models. Here we use the setting of $r_p = 0.2$ since this setting is the best according to evaluations in previous section. CNN-RNN[23], ASTPN[32], STRN[38] and Two-Stream[5] are four different methods based on RNN time series model and more advanced attention mechanism. CNN-XQDA[37] and QAN[21] train discriminative embeddings of images and apply different pooling methods. Among them, CNN-RNN[23], ASTPN[32] and Two-Stream[5] use both image and explicit motion features (optical flow) as inputs for deep neural network.

Here QAN[21] uses their own extra data for training. STRN[38] uses MARS pre-trained model to train PRID2011 and iLIDS-VID. Therefore, their methods cannot be fairly compared with other methods. We just list their results for reference.

For PRID2011 dataset, our method outperforms all other methods, improves the CMC Rank 1 about 5% compared with best state-of-the-art methods. For iLIDS-VID and MARS dataset, our results are at least comparable or even better than the compared methods. For CMC Rank 5, 10 and 20, the trends are similar to Rank 1.

Note that all the other methods use all the images for each verification. Our proposed model uses only 3 to 4 image pairs for each track pairs on average to obtain this encouraging performance.

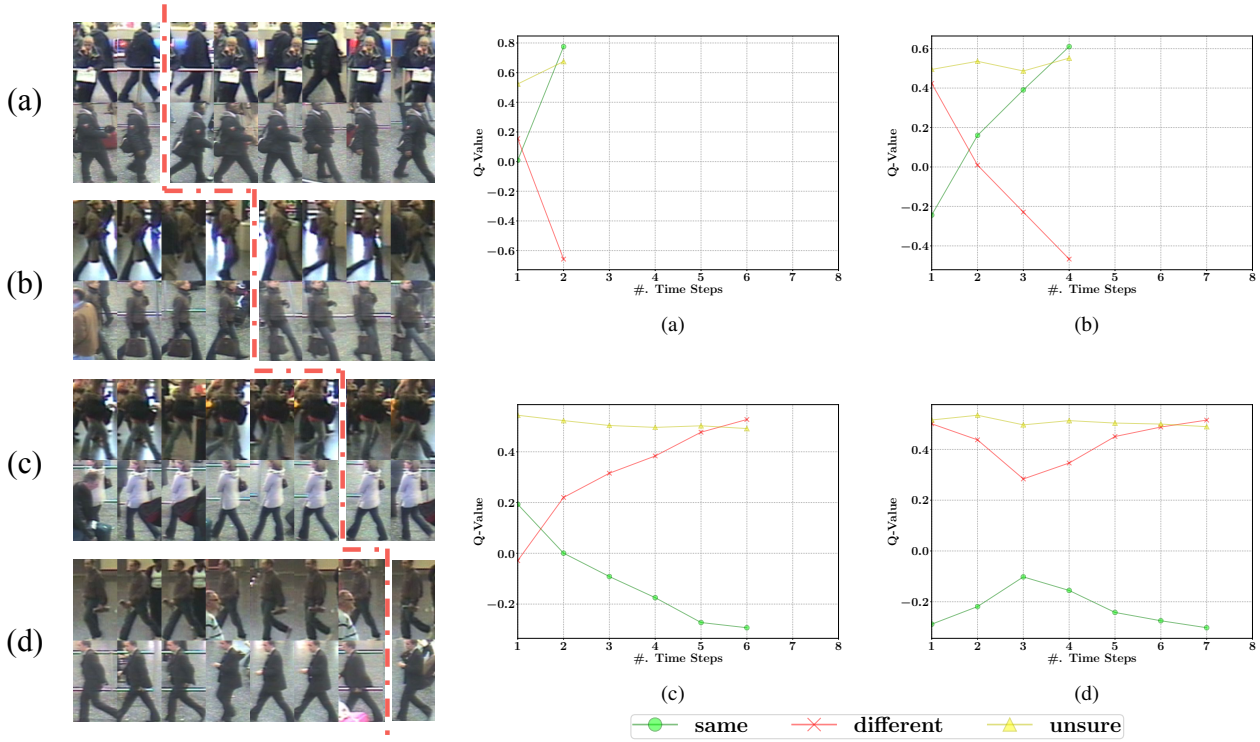

Figure 5: Some example episodes generated by our model. All the sampled images for each identity are listed on the left with a red dashed line split the end of the episode and the unused images. On the right side, Q values for each are shown.

4.5. Qualitative Results

In Figure 5, four representative episodes are shown. We can see the change of the Q value for the agent in dynamic environment. (a) and (b) are the same person, while (c) and (d) are different persons. These four episodes end with different length. Occlusions happen in the early pairs of (a), (b) and (c). The agent stops soon after the occlusions disappear. For (d), the these two persons have similar appearances but they are different persons, the agent had a high but not enough confidence to choose *different*. Then another person appears in the second and third pair makes the agent hesitate and reduce its confidence for the *different* action. After receiving more images, the agent has collected enough information and choose *different* eventually.

5. Conclusion

In this paper we have introduced a novel approach for multi-shot pedestrian re-identification problem by casting it as a pair by pair decision making process. Thanks to reinforcement learning, we train an agent to receive image pairs sequentially, and output one of the three actions: *same, different* or *unsure*. By early stop or decision postponing, the agent could adjust the budget needs to make confident decision according to the difficulty of the tracks.

We have tested our method on three different multi-shot pedestrian re-id datasets. We have shown our model can yield competitive or even better results with state-of-the-art methods using only 3% to 6% of images. Furthermore, the Q values outputted by the agent is a good indicator of the difficulty of image pairs, which makes our decision process is more interpretable by the help of Q values.

Currently, the weight for each frame is determined by the Q value heuristically, which means the weight is not guided fully by the final objective function. More advanced mechanism such as attention can be easily incorporated into our framework. We leave this as our future work.

References

[1] E. Ahmed, M. J. Jones, and T. K. Marks. An improved deep learning architecture for person re-identification. In *CVPR*, 2015. 1, 2

[2] A. Bar-Hillel, T. Hertz, N. Shental, and D. Weinshall. Learning a Mahalanobis metric from equivalence constraints. *Journal of Machine Learning Research*, 6:937–965, 2005. 2

[3] J. C. Caicedo and S. Lazebnik. Active object localization with deep reinforcement learning. In *ICCV*, 2015. 3

[4] S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *CVPR*, 2005. 2, 3

[5] D. Chung, K. Tahboub, and E. J. Delp. A two stream Siamese convolutional neural network for person re-identification. In *ICCV*, 2017. 1, 2, 7

[6] A. Das, R. Panda, and A. K. Roy-Chowdhury. Continuous adaptation of multi-camera person identification models through sparse non-redundant representative selection. *Computer Vision and Image Understanding*, 156:66–78, 2017. 2

[7] A. Dehghan, S. M. Assari, and M. Shah. GMMCP tracker: Globally optimal generalized maximum multi clique problem for multiple object tracking. In *CVPR*, 2015. 6

[8] S. Ding, L. Lin, G. Wang, and H. Chao. Deep feature learning with relative distance comparison for person re-identification. *Pattern Recognition*, 48(10):2993–3003, 2015. 1, 2

[9] P. F. Felzenszwalb, R. B. Girshick, D. A. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645, 2010. 6

[10] M. Hirzer, C. Beleznai, P. M. Roth, and H. Bischof. Person re-identification by descriptive and discriminative classification. In *SCIA*, 2011. 5

[11] C. Huang, S. Lucey, and D. Ramanan. Learning policies for adaptive tracking with deep feature cascades. In *ICCV*, 2017. 3

[12] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015. 3, 7

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *NIPS*, 2012. 3, 7

[14] I. Kviatkovsky, A. Adam, and E. Rivlin. Color invariants for person reidentification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(7):1622–1634, 2013. 2

[15] X. Lan, H. Wang, S. Gong, and X. Zhu. Identity alignment by noisy pixel removal. In *BMVC*, 2017. 3

[16] W. Li, R. Zhao, T. Xiao, and X. Wang. DeepReID: Deep filter pairing neural network for person re-identification. In *CVPR*, 2014. 1, 2

[17] W. Li, X. Zhu, and S. Gong. Person re-identification by deep joint learning of multi-loss classification. In *IJCAI*, 2017. 1, 2

[18] Z. Li, S. Chang, F. Liang, T. S. Huang, L. Cao, and J. R. Smith. Learning locally-adaptive decision functions for person verification. In *CVPR*, 2013. 2

[19] K. Liu, B. Ma, W. Zhang, and R. Huang. A spatio-temporal appearance representation for viceo-based pedestrian re-identification. In *ICCV*, 2015. 2

[20] W. Liu, Y. Wen, Z. Yu, and M. Yang. Large-margin softmax loss for convolutional neural networks. In *ICML*, 2016. 3

[21] Y. Liu, J. Yan, and W. Ouyang. Quality aware network for set to set recognition. In *CVPR*, 2017. 2, 7

[22] M. Malmir, K. Sikka, D. Forster, I. R. Fasel, J. R. Movellan, and G. W. Cottrell. Deep active object recognition by joint label and action prediction. *Computer Vision and Image Understanding*, 156:128–137, 2017. 3

[23] N. McLaughlin, J. M. del Rincón, and P. C. Miller. Recurrent convolutional network for video-based person re-identification. In *CVPR*, 2016. 1, 2, 7

[24] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. A. Riedmiller, A. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015. 4

[25] F. Schroff, D. Kalenichenko, and J. Philbin. FaceNet: A unified embedding for face recognition and clustering. In *CVPR*, 2015. 2, 3

[26] R. S. Sutton and A. G. Barto. Reinforcement learning: An introduction. *IEEE Transactions on Neural Networks*, 9(5):1054–1054, 1998. 4

[27] F. Wang, W. Zuo, L. Lin, D. Zhang, and L. Zhang. Joint learning of single-image and cross-image representations for person re-identification. In *CVPR*, 2016. 1, 2

[28] T. Wang, S. Gong, X. Zhu, and S. Wang. Person re-identification by video ranking. In *ECCV*, 2014. 5

[29] X. Wang, G. Doretto, T. Sebastian, J. Rittscher, and P. H. Tu. Shape and appearance context modeling. In *ICCV*, 2007. 2

[30] K. Q. Weinberger and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research*, 10:207–244, 2009. 2

[31] T. Xiao, H. Li, W. Ouyang, and X. Wang. Learning deep feature representations with domain guided dropout for person re-identification. In *CVPR*, 2016. 3

[32] S. Xu, Y. Cheng, K. Gu, Y. Yang, S. Chang, and P. Zhou. Jointly attentive spatial-temporal pooling networks for video-based person re-identification. In *ICCV*, 2017. 2, 7

[33] H. Zhao, M. Tian, S. Sun, J. Shao, J. Yan, S. Yi, X. Wang, and X. Tang. Spindle net: Person re-identification with human body region guided feature decomposition and fusion. In *CVPR*, 2017. 2

[34] L. Zhao, X. Li, Y. Zhuang, and J. Wang. Deeply-learned part-aligned representations for person re-identification. In *ICCV*, 2017. 2

[35] R. Zhao, W. Ouyang, and X. Wang. Learning mid-level filters for person re-identification. In *CVPR*, 2014. 2

[36] L. Zheng, Z. Bie, Y. Sun, J. Wang, C. Su, S. Wang, and Q. Tian. MARS: A video benchmark for large-scale person re-identification. In *ECCV*, 2016. 1, 6

[37] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian. Scalable person re-identification: A benchmark. In *ICCV*, 2015. 1, 2, 7

[38] Z. Zhou, Y. Huang, W. Wang, L. Wang, and T. Tan. See the forest for the trees: Joint spatial and temporal recurrent neural networks for video-based person re-identification. In *CVPR*, 2017. 1, 2, 7