Re-Ranking Person Re-Identification with Forward and Reverse Sorting Constraints

Meibin Qi, Yonglai Wei(✉), Kunpeng Gao, Jianguo Jiang, and Jingjing Wu

Hefei University of Technology, Hefei 230009, Anhui, China
qimeibin@163.com, wyl_hfut@163.com, gkp_hfut@163.com, jgjiang@hfut.edu.cn, hfutwujingjing@mail.hfut.edu.cn

Abstract. Person re-identification task aims at matching pedestrian images across multiple camera views. Extracting more robust feature of the pedestrian images and finding more discriminative metric learning are the main research directions in person re-identification. The achieved results are provided in the form of a list of ranked matching persons. It often happens that the true match which should be in the first position is not ranked first. In order to correct some false matches and improve the accuracy of person re-identification, this paper proposes a re-ranking method with forward and reverse sorting constraints. The forward sorting constraint makes the image, which is in the front position of one forward sorting list, be backward in the position of other forward sorting lists; The reverse sorting constraint makes two images of the same pedestrian be in the front position of each other's sorting list. Experiments on four public person re-identification datasets, VIPeR, PRID450S, CUHK01 and CUHK03 confirm the simplicity and effectiveness of our method.

Keywords: Person re-identification · Re-ranking
Forward and reverse sorting constraints

1 Introduction

With the popularity of video surveillance, more and more video surveillance systems are employed in the public areas, such as shopping malls, airports and hospitals. The control center of this system is usually connected with multiple cameras which are distributed in different areas. The control center operator can find and track a specific pedestrian (such as a criminal suspect) by observing the cameras. However, with the increasing number of cameras in the surveillance

Supported by organization by the National Natural Science Foundation of China Grant 61632007 and Key Research and Development Project of Anhui Province, China 1704d0802183.

R. Hong et al. (Eds.): PCM 2018, LNCS 11165, pp. 768–779, 2018.
https://doi.org/10.1007/978-3-030-00767-6_71

system, it becomes increasingly difficult to manually find and track pedestrians. Therefore, monitoring system needs to automatically find and track pedestrians.

The key of this monitoring system is to match the same pedestrian under different cameras, which is known as person re-identification. The problem of person re-identification can be expressed as follows: supposed that the existing n pedestrians pass through camera A and camera B in turn, the pedestrian images captured by camera A and camera B are called the probe images and the gallery images, respectively. Each probe image could determine a sort of the gallery images. The top-ranked gallery images are considered to be more similar to the probe images, in other words, they are more likely to be the same pedestrian.

However, many factors affect the performance of the person re-identification, such as various camera viewpoints, illumination, occlusion and the limitation of metric function. In recent years, there are two key points of person re-identification methods, i.e. extracting robust features and learning discriminative metrics. For feature representation, Liao et al. [10] propose an efficient feature representation called Local Maximal Occurrence (LOMO), using color and Scale Invariant Local Ternary Pattern (SILTP) histograms to represent picture appearance in a high dimension. In [15], a method called Gaussian of Gaussian (GOG) descriptor is proposed based on a hierarchical Gaussian distribution of pixel features. Lisanti et al. [12] propose a kernel descriptor to encode person appearance and project the data into common subspace using Kernel Canonical Correlation Analysis (KCCA). Chen et al. [1] propose an Spatially Constrained Similarity function on Polynomial feature map as SCSP to divide an image into four non-overlapping horizontal stripe regions, and each stripe region can be described by four visual cues which are organized as HSV1/HOG, HSV2/SILPT, LAB1/SILPT and LAB2/HOG. For distance metric, methods are designed to maximize the inter-class similarity and minimize the intra-class similarity. KISS Metric Learning (KISSME) [5], Crossview Quadratic Discriminant Analysis (XQDA) [10], Metric Learning with Accelerated Proximal Gradient (MLAPG) [11], Top-push Distance Learning model (TDL) [21] are representative methods.

While the first image of the list is usually not the true matching image in existing person re-identification methods. In order to solve this problem and improve the accuracy of person re-identification, a large number of researches called re-ranking have emerged for person re-identification [2,3,6,7,13,20,22]. Some of these researches [13] require the supervision of tag information, while this paper prefers unsupervised automatic re-ranking research. The method proposed in [3] learns an unsupervised re-ranking model by jointly considering content and context information in a sorting list for effectively eliminating fuzzy samples and improving the performance of person re-identification. Lend et al. [20] propose a bidirectional ranking method which combines the similarity of the content with context, and uses the new similarity to correct the initial sorting list. Recently, it is increasingly popular to correct the initial sorting by using the k-reciprocal neighbors [23]. However, the method in [23] has no effect if there

is only one positive for each identity in the gallery (single-shot). In order to solve this problem, this paper proposes a forward and reverse sorting constraints algorithm under single-shot to significantly improve pedestrian re-identification performance.

2 Proposed Approach

2.1 Problem Definition

Given a probe set with M pedestrian images $P = \{p_i | i = 1, 2, ..., M\}$ and a gallery set with N pedestrian images $G = \{g_j | j = 1, 2, ..., N\}$, the original distance between the pedestrian images p_i and g_j is represented as $d(p_i, g_j)$. Therefore, the initial ranking list $L(p_i,G) = \{g^i_{k,j} | i \in (1, 2, ...M); j, k \in (1, 2, ...N)\}$ can be obtained by calculating the distances between the probe image p_i and all gallery images G, where $g^i_{k,j}$ denotes the gallery g_j which is ranked k-th in the list of p_i. The goal of re-ranking is to modify the initial ranking list $L(p_i, G)$ so that more positive samples are ranked at the top of the list to improve the performance of person re-identification.

2.2 Forward Sorting Constraint

The matching of person re-identification is to calculate the distances between a probe image and all gallery images, then we sort the image of the gallery based on the distances. The first image of the list is considered to be the true match of the probe image. This sorting method is called forward sorting in this paper. However, the true matching results are not ranked first in the sorting list after the forward sorting. Inspired by [16], we propose the algorithm of forward sorting constraint.

As shown in Fig. 1(a), the left images are the probe images, and the right images are the sorting of gallery images for them. The values in parentheses of the figure indicate the distance values between the corresponding gallery images

Fig. 1. Forward sorting constraint: (a) Initial ranking results. (b) Re-ranking results.

and the probe images. It can be seen from the figure, the gallery image g_1 is respectively rank-1 in the sorting list of p_1, rank-1 in the sorting list of p_2, and rank-2 in the sorting list of p_3. Therefore, according to the ranking, the possibilities that the gallery image g_1 and the probe images p_1, p_2 are the same pedestrian are higher than that of p_3. But it still cannot be judged that whether p_1 or p_2 is more likely to be the same pedestrian as g_1. Therefore, this paper combines the distance information with the constraint of the sorting information to judge. From the Fig. 1(a), because of $d(p_1, g_1) < d(p_2, g_1)$, the possibility that g_1 and p_1 are the same pedestrian is higher than the possibility that g_1 is the same pedestrian as p_2.

In summary, the possibility that the gallery image g_1 and the probe image p_1 are the same pedestrian is the highest. We assume that g_1 and p_1 are the same pedestrian, so g_1 and p_2, p_3 should be different pedestrian. Under the situation, the matching accuracy can be improved by "punishing" the distance between g_1 and p_2, p_3 (increasing the distance between them). The result of re-ranking after "punishment" is shown in Fig. 1(b). The ranking of correct matching is improved in the forward sorting list of p_2, p_3.

According to the above analysis, how to "punish" the distances of false matches becomes the primary problem. The algorithm of this paper gives the appropriate "punishment" based on the original distance value of mismatches. In short, the smaller (larger) original distance, the smaller (heavier) "punishment". In this way, some excessive punishment can be effectively avoided. The detailed algorithm is summarized in Algorithm 1.

Algorithm 1: Forward sorting constraint

Input: Initial distance matrix $dist(N, M)$
Output: Re-ranking distance matrix $outdist(N, M)$
for $j = 1, 2..., N$
  for $i = 1, 2, ..., M$
    $K(i) = g_j$ position in $L(p_i, G)$
  end
  $array\_k = argmin(K(i))$
  $k = array\_k$ lets $min(dist(p_{array\_k}, g_j))$
  for $i = 1, 2, ..., M$
    if $(i! = k)$ $outdist(j, i) = dist(j, i) + \lambda * dist(j, i)$
  end
end

2.3 Reverse Sorting Constraint

The forward sorting uses the distance values between the probe images and the gallery images to sort the gallery images. In turn, the distance values between them can also sort the probe images, and the sorting method is called the reverse sorting in this paper.

In Fig. 2, we show an example of the reverse sorting constraint. The true match (green box) of the probe image p_1 is incorrectly ranked, shown in Fig. 2(a). Figure 2(b) shows an reverse sorting list of the first two gallery images from the forward sorting list in Fig. 2(a). It is not difficult to find that the probe image p_1 is located in the third and first positions of reverse sorting list of the gallery images g_1 and g_2, respectively. There are some distinctions between Fig. 2(a) and 2(b). Figure 2(a) shows that the position of g_2 is relatively low-ranking in the forward sorting list of p_1. In Fig. 2(b), in the reverse sorting list of g_2, the position of p_1 is relatively high-ranking. It inspires us to use the reverse sorting information to constrain the initial sorting.

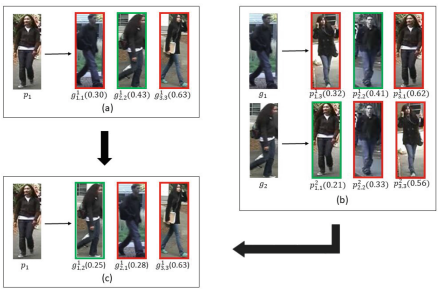


Fig. 2. Reverse sorting constraint: (a) Probe image and its forward sorting list. (b) Gallery images g_1, g_2 and their reverse sorting lists. (c) Re-ranking results.

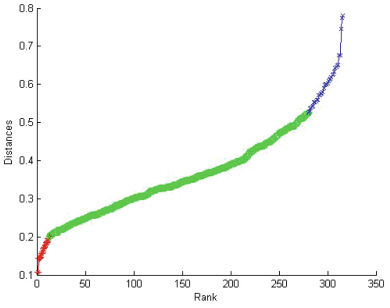


Fig. 3. An example of the results obtained by applying clustering algorithm to the distance computed between a probe and all the gallery images.

According to the above analysis, in order to efficiently use the reserve sorting, this paper does not implement reverse sorting constraint on all gallery images, while applies to several top gallery images in the forward sorting list. The top-ranked gallery images have a higher probability of containing true matches, so they are called content sets. In this paper, the reverse sorting constraint algorithm selects the content sets according to the dynamic method proposed by Jorge García et al. [2]. Figure 3 shows the relationship of the position of the gallery images in the forward sorting list of a probe image and the distance between the gallery image and the probe image: (1) at first ranks, the distance between the gallery image and the probe image increases abruptly, then flattens(first elbow); (2) from the first elbow, distances grow linearly till reaching high ranks, and at last, distances start increasing significantly again (second elbow). According to such trend, the gallery images can be divided into three classes: (1) the similar appearance class (C_{sa}), which corresponds to the gallery images whose positions are before the first elbow; (2) the difference appearance class (C_{da}), which corresponds to the gallery images whose positions are between the first elbow and the second elbow; (3) the opposite appearance class (C_{oa}), which corresponds to the gallery images whose positions are after the second elbow.

In order to find the positions of the three elbows and effectively divide the three types of the gallery set, this paper uses the k-means clustering algorithm to divide the original gallery set. The specific approach is as follows. First, defining the cluster mean value: $\mu_{sa} = d(p_i, g_1^i)$, $\mu_{da} = d(p_i, g_{(N/2)}^i)$ and $\mu_{oa} = d(p_i, g_N^i)$ where g_k^i denotes a gallery image ranked k-th in the forward sorting list of probe image p_i. Then the cluster mean is substituted into the following cost function:

$$\sum_{k \in \{sa,da,oa\}} \sum_{g_j^i \in C_k} ||d(p_i, g_j^i) - \mu_k||^2, j = 1, 2, ..., N \tag{1}$$

After continuously iterating and optimizing, the above cost function will converge. The similar appearance class will contain the first m gallery images of the forward sorting list, forming the content sets $B_i^{cn} = \{g_1^i, g_2^i, ..., g_m^i\}$. The reverse sorting constraint algorithm will be described in detail.

Given a probe image p_i and its forward sorting list $L(p_i, G)$. First of all, according to the method described above, we find the content set B_i^{cn} of the forward sorting list and then calculate the reverse sorting list $L(g_1^i, P)$ and $L(g_m^i, P)$ of gallery images g_1^i and g_m^i on the content set. Finally, we give certain "rewards" to the initial distance between p_i and g_1^i and the distance between p_i and g_m^i (decreasing the distance between them). The "reward" here is calculated by a reward function that gives more "rewards" to the distance between the top-ranked probe image and the gallery image of the reverse sorting list. According to the above properties, this paper proposes a reward function called Reward, as shown in the Eq. (2). It is worth noting that other reward functions that meet the above properties can also be used here.

$$Reward(p_i, g_j) = 2 - e^{0.01 * rank(g_j, p_i)} \tag{2}$$

Algorithm 2: Reverse sorting constraint

Input: Initial distance matrix $dist(N, M)$
Output: Re-ranking distance matrix $outdist(N, M)$
for $i = 1, 2..., M$
  for $k = m, m-1, ..., 2$
    $dist(p_i, g_k^i) -= Reward(p_i, g_k^i)$
    $dist(p_i, g_1^i) -= Reward(p_i, g_1^i)$
    Update sorting list.
  end
  $outdist = dist$
end

The $rank(g_j, p_i)$ indicates the position of the probe image p_i in the reverse sorting list of the gallery image g_j. Firstly, the "reward" is given to the distances between p_i and g_1^i, g_m^i, then we update the sorting list. Secondly, the "reward" operation is repeated on the distances between p_i and new g_1^i, g_{m-1}^i, then we

update the sorting list again. Finally, we repeat the operation until "reward" the distances between p_i and g_1^i, g_2^i. The complete algorithm is summarized in Algorithm 2.

3 Experiments

3.1 Results of Different Datasets

We apply the proposed re-ranking algorithm on four different datasets of one baseline method to prove the universality of the method. The baseline method extracts hierarchical Gaussian descriptor [15] and uses cross-view quadratic discriminant analysis (XQDA) as distance metric. The four standard datasets include VIPeR [4], CUHK01 [8], PRID450S [18] and CUHK03 [9]. The test protocol of each dataset is the same as literature [15].

Table 1. Matching rates (%) of this algorithm on different datasets

Datasets	Method	r = 1	r = 5	r = 10
VIPeR	GOG	49.72	79.72	88.67
	GOG+OURS	**55.60**	**82.25**	**90.38**
	LOMO	40.00	68.13	80.51
	LOMO+OURS	**45.54**	**72.85**	**83.32**
CUHK01	GOG	57.91	79.14	86.27
	GOG+OURS	**63.68**	**83.60**	**89.29**
	LOMO	49.84	75.26	83.34
	LOMO+OURS	**55.16**	**78.16**	**85.66**
PRID450S	GOG	68.00	88.67	94.36
	GOG+OURS	**74.89**	**92.49**	**96.49**
	LOMO	57.42	81.07	88.31
	LOMO+OURS	**65.02**	**84.31**	**90.18**
CUHK03Labeled	GOG	68.47	90.69	95.84
	GOG+OURS	**77.08**	**94.59**	**97.20**
	LOMO	50.85	81.18	91.14
	LOMO+OURS	**56.90**	**84.83**	**93.69**
CUHK03 Detected	GOG	64.10	88.40	94.30
	GOG+OURS	**72.59**	**92.25**	**96.65**
	LOMO	44.10	78.70	87.70
	LOMO+OURS	**49.75**	**81.20**	**88.95**

The VIPeR dataset is a challenging dataset for person re-identification task, which contains 632 person image pairs, captured by different cameras in an outdoor environment. The CUHK01 dataset contains 971 persons and each person

has two images in each camera. The PRID450S dataset is an extension of the PRID2011 dataset. It contains 450 pedestrian image pairs captured by two outdoor cameras. The background, lighting and viewpoints of two cameras are very different. The CUHK03 dataset contains 13164 images of 1360 identities captured by six surveillance cameras. Each pedestrian in the dataset has an average of 4.8 images per view. It provides bounding boxes manually labeled pedestrian and bounding boxes automatically detected by the pedestrian detector.

As shown in the Table 1, experimental results on above four datasets demonstrate that our approach achieves obvious improvements on the baseline method. Among them, the rank-1 matching rates increase by 5.32% at least and 8.85% at most. At the same time, there are also improvements on rank-5 and rank-10. It can be concluded that the proposed algorithm can be applied to different datasets and will not be affected by the acquisition environment and magnitude of datasets.

3.2 Results of Different Person Re-Identification Algorithms

In order to verify the scalability of the proposed algorithm, we apply the proposed re-ranking algorithm on four different person re-id methods of the VIPeR dataset.

Table 2. Matching rates (%) of different pedestrian re-identification algorithms

Method	VIPeR (%)			
	Rank-1	Rank-5	Rank-10	Rank-20
KCCA	37.25	71.39	84.56	92.81
KCCA+OURS	**43.39**	**75.06**	**85.76**	**93.04**
LOMO	40.00	68.13	80.51	91.03
LOMO+OURS	**45.54**	**72.85**	**83.32**	**92.94**
GOG	49.72	79.72	88.67	94.53
GOG+OURS	**55.60**	**82.25**	**90.38**	**95.51**
SCSP	53.54	82.59	91.49	96.65
SCSP+OURS	**57.75**	**83.01**	**91.23**	**98.92**

As shown in Table 2, our re-ranking algorithm can significantly improve performances for different person re-identification algorithms. And our method improves the rank-1 matching rates by 4% at least.

3.3 Comparison with Other Re-Ranking Methods

In this section, we compare our method with other re-ranking algorithms, Prototype-Specific Feature Importance (PSFI) [14], Individual-Specific Feature Importance (ISFI) [14], and Probe-Specific re-ranking (PSR) [19]. The results are copied from their papers and recorded in Table 3 for comparison.

It is shown in Table 3 that all the re-ranking methods can achieve higher matching accuracies compared with their baseline algorithms. Compared with our baseline method GOG, our result increases by 5.88% at Rank-1, which is the best improvement at rank1 among the compared methods. It is shown that our method outperforms these existing methods. Besides, our final results are better than those of the compared methods.

3.4 Effect of Major Components

From Sects. 2.2 and 2.3, it can be observed that the proposed re-ranking algorithm contains two parts: forward sorting constraint and reverse sorting constraint. The evaluation of each part of the algorithm is performed with the Cumulative Matching Characteristics (CMC) curve on the VIPeR dataset.

Table 3. Matching rates (%) of different re-ranking methods

Method	VIPeR (%)				
	Rank-1	Rank-5	Rank-10	Rank-20	Improvement of Rank-1
RDC	12.15	27.78	38.94	54.46	4.97
RDC+ISFI	**17.12**	**38.96**	**52.94**	**67.34**	
RSVM	12.93	31.46	43.91	59.64	2.83
RSVM+PSFI	**15.76**	**38.70**	**51.36**	**66.84**	
AML	43.04	72.28	83.96	93.54	2.15
AML+PSR	**45.19**	**73.58**	**85.35**	**93.99**	
GOG	49.72	79.72	88.67	94.53	5.88
GOG+OURS	**55.60**	**82.25**	**90.38**	**95.51**	

As shown in Fig. 4, both parts of the re-ranking algorithm (red and blue curves) have achieved good performances compared with the original results of baseline (black curve), and the combination of both parts will have better performance.

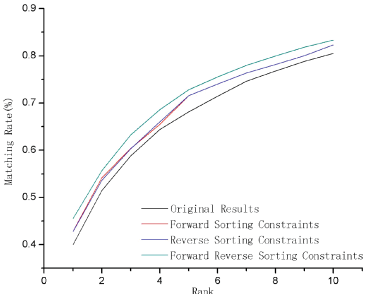
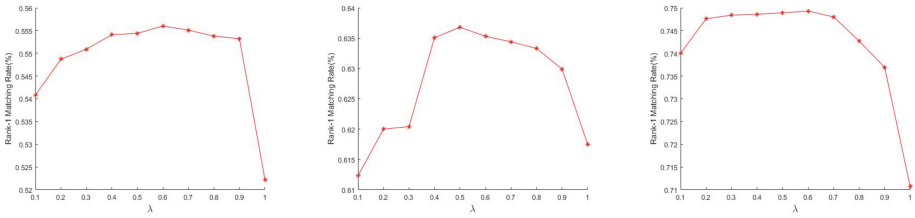


Fig. 4. Performance is obtained by separately considering the forward sorting constraint and reverse sorting constraint. Results have been computed considering the GOG baseline. (Color figure online)

3.5 Effect of Hyper-parameter on Algorithm Performance

Hyper-parameter λ is introduced to implement different levels of "punishment" for error matching in the algorithm 1. Figure 5 shows the influence of different λ values at rank-1 on three datasets. It is shown that the performances of person re-identification are best for VIPeR, CUHK01, and PRID450S, when λ values are 0.6, 0.5, and 0.5, respectively.



(a) (b) (c)

Fig. 5. The impact of the parameter λ on person re-identification performance on three datasets. (a) VIPeR dataset. (b) CUHK01 dataset. (c) PRID450S dataset.

4 Conclusions

In this paper, we use the implicit constraint information in the initial sorting list, which is formed by the existing person re-identification algorithm, and we conversely re-rank the initial sorting list to improve the performance of the original person re-identification algorithm. The experimental results show that the application of this algorithm on different datasets can significantly improve the performance of the original person re-identification algorithm, especially when

the dataset is relatively large, the effect is even more gratifying. It is worth mentioning that the forward and reverse sorting constraints algorithm proposed in this paper is fully automatic and unsupervised, and it can be easily applied to existing person re-identification algorithms.

References

1. Chen, D., Yuan, Z., Chen, B., Zheng, N.: Similarity learning with spatial constraints for person re-identification. In: Computer Vision and Pattern Recognition, pp. 1268–1277 (2016)
2. Garcia, J., Martinel, N., Gardel, A., Bravo, I., Foresti, G.L., Micheloni, C.: Discriminant context information analysis for post-ranking person re-identification. IEEE Trans. Image Process. **26**(4), 1650–1665 (2017)
3. Garcia, J., Martinel, N., Micheloni, C., Gardel, A.: Person re-identification ranking optimisation by discriminant context information analysis. In: IEEE International Conference on Computer Vision, pp. 1305–1313 (2015)
4. Gray, D., Tao, H.: Viewpoint Invariant Pedestrian Recognition with an Ensemble of Localized Features. Springer, Heidelberg (2008)
5. Hirzer, M.: Large scale metric learning from equivalence constraints. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2288–2295 (2012)
6. Leng, Q., Hu, R., Liang, C., Wang, Y., Chen, J.: Person re-identification with content and context re-ranking. Multimedia Tools Appl. **74**(17), 6989–7014 (2015)
7. Li, W., Wu, Y., Mukunoki, M., Minoh, M.: Common-near-neighbor analysis for person re-identification. In: IEEE International Conference on Image Processing, pp. 1621–1624 (2013)
8. Li, W., Zhao, R., Wang, X.: Human Reidentification with Transferred Metric Learning. Springer, Heidelberg (2013)
9. Li, W., Zhao, R., Xiao, T., Wang, X.: DeepReID: deep filter pairing neural network for person re-identification. In: Computer Vision and Pattern Recognition, pp. 152–159 (2014)
10. Liao, S., Hu, Y., Zhu, X., Li, S.Z.: Person re-identification by local maximal occurrence representation and metric learning. In: Computer Vision and Pattern Recognition, pp. 2197–2206 (2015)
11. Liao, S., Li, S.Z.: Efficient PSD constrained asymmetric metric learning for person re-identification. In: IEEE International Conference on Computer Vision, pp. 3685–3693 (2015)
12. Lisanti, G., Masi, I., Bimbo, A.D.: Matching people across camera views using kernel canonical correlation analysis, pp. 1–6 (2014)
13. Liu, C., Chen, C.L., Gong, S., Wang, G.: POP: person re-identification post-rank optimisation. In: IEEE International Conference on Computer Vision, pp. 441–448 (2014)
14. Liu, C., Gong, S., Chen, C.L.: On-the-fly feature importance mining for person re-identification. Patt. Recogn. **47**(4), 1602–1615 (2014)
15. Matsukawa, T., Okabe, T., Suzuki, E., Sato, Y.: Hierarchical Gaussian descriptor for person re-identification. In: Computer Vision and Pattern Recognition, pp. 1363–1372 (2016)
16. Nguyen, N.B., Nguyen, V.H., Ngo, T.D., Nguyen, K.M.T.T.: Person re-identification with mutual re-ranking. Vietnam J. Comput. Sci. **4**, 1–12 (2017)

17. Nguyen, V.H., Ngo, T.D., Nguyen, K.M.T.T., Duong, D.A., Nguyen, K., Le, D.D.: Re-ranking for person re-identification. In: Soft Computing and Pattern Recognition, pp. 304–308 (2015)
18. Roth, P.M., Hirzer, M., Köstinger, M., Beleznai, C., Bischof, H.: Mahalanobis distance learning for person re-identification, pp. 247–267 (2014)
19. Xie, Y., Yu, H., Gong, X., Levine, M.D.: Adaptive metric learning and probe-specific reranking for person reidentification. IEEE Sig. Process. Lett. **24**(6), 853–857 (2017)
20. Ye, M., et al.: Person reidentification via ranking aggregation of similarity pulling and dissimilarity pushing. IEEE Trans. Multimedia **18**(12), 2553–2566 (2016)
21. You, J., Wu, A., Li, X., Zheng, W.S.: Top-push video-based person re-identification, pp. 1345–1353 (2016)
22. Zheng, L., Wang, S., Tian, L., He, F., Liu, Z., Tian, Q.: Query-adaptive late fusion for image search and person re-identification. In: Computer Vision and Pattern Recognition, pp. 1741–1750 (2015)
23. Zhong, Z., Zheng, L., Cao, D., Li, S.: Re-ranking person re-identification with k-reciprocal encoding, pp. 3652–3661 (2017)