**QUEEN'S
UNIVERSITY
BELFAST**

Multi-level Deep Learning Vehicle Re-identification using Ranked-based Loss Functions

Kamenou, E., Martinez-del-Rincon, J., Miller, P., & Devlin Hill, P. (2021). Multi-level Deep Learning Vehicle Re-identification using Ranked-based Loss Functions. In *International Conference on Pattern Recognition: 10/01/2021 → 15/01/2021 Milan, Italy* (International Conference on Pattern Recognition (ICPR): Proceedings). Institute of Electrical and Electronics Engineers Inc.. <https://doi.org/10.1109/ICPR48806.2021.9412415>

Published in:
International Conference on Pattern Recognition

Document Version:
Peer reviewed version

Queen's University Belfast - Research Portal:
[Link to publication record in Queen's University Belfast Research Portal](#)

Publisher rights
Copyright 2020 IEEE. This work is made available online in accordance with the publisher's policies. Please refer to any applicable terms of use of the publisher.

General rights
Copyright for the publications made accessible via the Queen's University Belfast Research Portal is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
The Research Portal is Queen's institutional repository that provides access to Queen's research output. Every effort has been made to ensure that content in the Research Portal does not infringe any person's rights, or applicable UK laws. If you discover content in the Research Portal that you believe breaches copyright or violates any law, please contact openaccess@qub.ac.uk.

Multi-level Deep Learning Vehicle Re-identification using Ranked-based Loss Functions

Eleni Kamenou*, Jesus Martinez del Rincon*, Paul Miller* and Patricia Devlin-Hill†

*Centre of Secure Information Technologies (CSIT)

Queen's University, Belfast, United Kingdom

†Thales UK, Belfast, Northern Ireland

Abstract—Identifying vehicles across a network of cameras with non-overlapping fields of view remains a challenging research problem due to scene occlusions, significant inter-class similarity and intra-class variability. In this paper, we propose an end-to-end multi-level re-identification network that is capable of successfully projecting same identity vehicles closer to one another in the embedding space, compared to vehicles of different identities. Robust feature representations are obtained by combining features at multiple levels of the network. As for the learning process, we employ a recent state-of-the-art structured metric learning loss function previously applied to other retrieval problems and adjust it to the vehicle re-identification task. Furthermore, we explore the cases of image-to-image, image-to-video and video-to-video similarity metric. Finally, we evaluate our system and achieve great performance on two large-scale publicly available datasets, CityFlow-ReID and VeRi-776. Compared to most existing state-of-art approaches, our approach is simpler and more straightforward, utilizing only identity-level annotations, while avoiding post-processing the ranking results (re-ranking) at the testing phase.

I. INTRODUCTION

Recently, distributed video camera networks have been widely deployed in urban areas for surveillance and security purposes. In smart city scenarios, there are numerous applications that require re-acquisition of vehicles as they move across different cameras with non overlapping fields of view. This problem, known as vehicle Re-identification (ReID), has recently gained much attention in the computer vision research community. In particular, it can be considered as a retrieval task, where, given a query image of a vehicle, we need to search over numerous gallery images to find the same vehicle captured by other cameras.

Typical challenges of vehicle ReID data include poor resolution, scene occlusions, extreme variations of viewpoints and illumination conditions. In addition, there are several inherent properties of the data that render the ReID task even more difficult. For instance, there is the inter-class similarity, which occurs when different vehicles of the same colour and model are present and the intra-class variability, where varying viewpoints induce large differences in a vehicle's appearance (see Fig. 1).

The vehicle ReID problem has been extensively studied by the computer vision research community in recent years. Most existing methods leverage transfer learning using additional training data, either from vehicle classification datasets [2] due to the higher image resolution, or generate more images

Figure 1: Vehicle images selected from the VeRi [1] dataset. The same vehicle captured under different camera-views.

from existing data. Other works [3] take advantage of detecting vehicle parts that carry valuable information e.g. wheels, windows, etc. Furthermore, current state-of-the-art methods rely on extra information (e.g. vehicle type, tracklet timestamps) and apply re-ranking techniques [4] in order to refine the initial ReID results provided by the corresponding metric learning paradigm. This results in overly complex systems that are difficult to update and maintain in real-time applications. Instead, we focus our efforts on improving the underlying ReID framework and generate powerful embeddings, in order to overcome the challenges of vehicle ReID data.

In this paper, we propose an end-to-end multi-level embedding learning vehicle ReID system that is based on a structured metric learning loss function, while also exploring how our system responds to both image-based and video-based testing data. Thus, we demonstrate for the first time the effectiveness of Ranked-List Loss (RLL) function to vehicle ReID. Moreover, we employ a simple multi-level network training configuration, while using only basic data pre-processing and avoiding post-processing of the ReID results. In order to show the strong generalization ability of our method, we train our model by jointly using data from two datasets, without further fine-tuning on any of them. Finally, and to the best of our knowledge, our approach is the first to apply video-to-video similarity metric, resulting in a fully video-based vehicle ReID system.

II. RELATED WORK

As an emerging research topic, vehicle ReID has attracted great efforts [4]–[8] by researchers. In this section, relevant works are reviewed from three different aspects: feature extraction approaches, metric learning loss functions and the impact of data format (i.e. image and video data) on vehicle ReID.

Several former methods of vehicle ReID were based on visual features of license plate detection extracted from vehicle

Figure 2: Training and testing phases of the proposed method.

images [1], [7], or used semantic information about the date, time and camera position of a vehicle observation [9]. Recent research focuses on using deep convolutional neural networks (CNNs) to create embeddings that aim to map vehicles of the same identity close to one another in the embedding feature space, while keeping projections of different vehicles far apart [10], [11], usually through siamese and triple network configurations. Multi-level embeddings, i.e. the generation of embedding by pooling information at different stages of the CNN, have also been used for extracting more robust feature representations in metric learning methods [12], [13]. However, in the vehicle ReID field, they have not been fully explored, despite the apparent benefit of multi-level embeddings to the problem. By extracting information at different levels of resolution, both global appearance (general shape and model, colour) and small crucial details (stickers, brand logos, frontal grid) can be potentially combined in the final feature representation. To this end, recent works [14]–[16] propose multi-granularity networks (MGNs) for vehicle ReID at the cost of a considerably more complex network architecture.

In terms of metric learning loss functions, contrastive [17], triplet loss [11], [18], [19] and variations of them [10], [20], have been widely used in the ReID literature. Given triplets of samples projected to an embedding space, the purpose of triplet loss is to make the distance between the anchor and the negative point larger than the distance between the anchor and the positive point. Recently, group-based metric learning loss functions have been proposed [13], [18], [21], and achieve superior results in retrieval problems, since they incorporate more than three samples, and take into consideration valuable information among multiple data points. In several works [8], [14], classification loss is also used as a complement to similarity metric objectives, in order to increase the discriminative power of the network.

Image-based ReID has been widely studied, and there are numerous works [5], [8], [22], [23] that exploit only the visual information of the data. On the other hand, temporal information has also proved useful [3], [4], [24] in obtaining more accurate ReID results. In order to extract a more powerful video-based feature representation, feature

pooling [24] among frames of the same tracklet is proposed. Such pooling is usually used at inference time rather than during training. Due to the format of most vehicle ReID datasets, existing works mostly employ image-to-image and image-to-tracklet similarity measuring, neglecting the case of comparison between tracklets. However, given the existence of effective multi-target tracking systems, it is expected that both query and gallery candidates will be represented as time-series, or tracklets, rather than single images.

III. PROPOSED APPROACH

In this section, we describe the algorithmic design of our proposed ReID framework. An overview flow diagram of the proposed system is presented in Fig. 2. During the training phase, the connections at three different levels allow us to extract strong features that are fed into the loss function. As seen in the diagram, at testing time, input samples are organized in tracklets. The feature representation for each frame is composed by aggregating the features produced at each level, and then temporal pooling is applied across the frame features in order to obtain video feature representations.

A. Loss Function

Our objective is to learn a discriminative metric loss function such that the similarity between positive pairs is higher than the similarity between negative pairs in the feature space. At the end, given a query from any class identity, the system must successfully identify its matching samples from all other gallery samples. As for the similarity metric, we use the Euclidean distance between the feature vectors.

Based on [13], we employ RLL loss function, which aims to keep the distance between the query and a group of positive samples below a certain threshold, while separating the query from the negative samples by a margin. In our work, we explore and evaluate two variations of this loss function, that can be observed in Fig. 3. In the first case, the loss is computed over pairs of samples, similarly to other well-known loss functions such as contrastive and triplet loss, whereas in the second case we take into account a list of samples, which includes multiple positives and negatives, in order to compute the loss. By selecting multiple samples, we build a

Figure 3: Illustration of Single-Sampling and Ranked-List Loss. For the sake of simplicity, only 3 classes are shown. In the single-sampling case, the anchor is compared with only one positive and one negative sample. On the contrary, ranked-list loss not only exploits more than one positive samples, but also makes use of multiple negative samples from more than one identity classes.

more informative structure compared to pair samples, and use it to learn more discriminative embeddings.

1) *Single-Sampling Loss*: In the Single-Sampling Loss (SSL) case, we consider two samples, $x_{c_1}^i$ and $x_{c_2}^j$, which belong to c_1 and c_2 identity classes, respectively. When $x_{c_1}^i$ and $x_{c_2}^j$ belong to different c_1 and c_2 identity classes, we aim to ensure their separation is greater than a distance a . At the same time, when both samples belong to the c_1 identity class, we aim to ensure their separation is less than $a - m$. Thus, m is the margin between the two boundaries. Mathematically, we define 2 different losses for the positive and the negative case, and the final loss is the sum of the two.

$$l_p(x_{c_1}^i, x_{c_1}^j) = \max(d_{ij} - (a - m), 0) \quad (1)$$

$$l_n(x_{c_1}^i, x_{c_2}^j) = \max(a - d_{ij}, 0) \quad (2)$$

$$L_{SSL} = l_p(x_{c_1}^i, x_{c_1}^j) + l_n(x_{c_1}^i, x_{c_2}^j) \quad (3)$$

d_{ij} denotes the euclidean distance between two samples, $d_{ij} = \|x^i - x^j\|_2$.

2) *Ranked-List Loss*: For the above equations to work in a group-based way, we consider a list of samples, where there exists more than one instance per class. First, we sort this list of samples according to their similarity to the query. Then, for the loss computation we consider only the samples that produce a non-zero loss value according to Eq. 1 and 2. Those are the positive or negative samples whose distance from the query is larger than $a - m$ or less than a , respectively.

Let K be the number of sampled images per class and C be the number of vehicle identity classes in one batch. Then, the loss computation is made over a batch of size $K \times C$. Each image in the batch acts as an anchor, having $K - 1$ positive pair samples and $(C - 1) \times K$ negative pair samples in total. If the group of positive and negative pair samples that produce

non-zero loss for an anchor sample x_c^i according to Equation 1 and 2, are $P_{c,i} = x_c^j \mid j \neq i, d_{ij} > (a - m)$ and $N_{c,i} = x_k^j \mid k \neq c, d_{ij} < a$ respectively, then the list-based loss equations are:

$$L_p(x_c^i) = \frac{1}{|P_{c,i}|} \sum_{x_c^j \in P_{c,i}} l_p(x_c^i, x_c^j) \quad (4)$$

$$L_n(x_c^i) = \frac{1}{|N_{c,i}|} \sum_{x_k^j \in N_{c,i}} l_n(x_c^i, x_k^j) \quad (5)$$

$$L_{RLL}(x_c^i) = L_p(x_c^i) + L_n(x_c^i) \quad (6)$$

3) *Classification Loss*: As a complement to the similarity learning, we also add a classification loss L_{ID} , by adding a classification Fully Connected (FC) layer where each vehicle identity is considered as a separate class and the number of classes is equal to the number of vehicle identities in the training set. Adding classification increases the discriminative power of the model and leads to faster convergence [14]. For the classification loss function we use cross entropy loss [4].

The final loss is given by the equations below:

$$L(x_c^i) = w_1 * L_{SSL}(x_c^i) + w_2 * L_{ID}(x_c^i) \quad (7)$$

or

$$L(x_c^i) = w_1 * L_{RLL}(x_c^i) + w_2 * L_{ID}(x_c^i) \quad (8)$$

where w_1 and w_2 are weighting factors.

B. Feature Extraction and Aggregation

For feature extraction, we use the ResNet50 [25] architecture, pretrained on ImageNet, as a backbone. An FC layer that outputs feature vectors of size 128 is inserted at the top of the convolutional part of the network. Moreover, in order to compute the classification loss, we also insert a classification

FC layer. This layer is only used during training and it is detached for the testing phase (see Fig. 2).

In order to create multi-level embeddings, we apply modifications to the network architecture. Inspired by [13], we extract low, medium and high-level features. As can be seen in Fig. 2, we insert a FC layer after layers 4c, 4f and 5c of ResNet50. Each of those FC layers outputs feature vectors of 128 dimensions. In this way, we capture useful visual information that lies at intermediate levels of the network. During training, we compute the losses at the three different levels of the network and sum them to get the final loss.

C. Image vs. Video ReID

At inference time, we evaluate our system in different data formats. Apart from the image-to-image similarity measuring, we examine the cases of image-to-video and video-to-video ReID. In vehicle ReID datasets, the vehicles appear in tracklets, and we obtain feature representation for a vehicle video, by computing the mean over the feature vectors of all the individual frames in the tracklet. Temporal pooling provides us with robust video feature representations and extends our system to a fully video-based approach for vehicle ReID.

IV. EXPERIMENTS

A. Vehicle ReID Datasets

In vehicle ReID, datasets are crucial to comprehensively and fairly evaluate the performance of different ReID methods. In order to train and evaluate our system, we used two publicly available large-scale benchmarks: VeRI-776 [1] and CityFlow-ReID [26].

VeRI-776 has been widely used by most recent research in vehicle ReID, as it provides multiple views of vehicles in an urban environment. It contains over 50K images along with identity, vehicle colour and type annotations for 776 vehicles captured by 20 cameras in traffic scenes. The first part includes $\approx 38K$ images corresponding to 576 vehicles and it is used for training. The second part, which is used as test set, includes $\approx 11K$ images from 200 vehicle identities. Each vehicle is captured by 2-18 cameras with various viewpoints and varying illumination conditions.

CityFlow-ReID is a more recent dataset, but also widely used in vehicle ReID literature. It consists of approximately 56K images, where 37K of them from 333 vehicle identities form the training set, while the test set consists of 18K images from the other 333 identities. The total number of 666 vehicles are captured by 40 cameras placed across a real-world city environment and each vehicle has on average 84.5 image signatures from 4.55 camera views.

B. Experimental Details

For the training, first we combine training images from both datasets, resulting in 909 vehicle identities in total. The learned model is then evaluated on the testing set of each dataset separately. Note that we avoid specific fine-tuning to any of the two datasets to ensure the generalization of our system.

In each batch there are images from $C = 21$ vehicle classes with $K = 3$ images per class, leading to a batch size equal to 63. This means that for each query sample, there are potentially 2 positive and 60 negative pair samples, given that they satisfy the criteria to be sampled described in III-A. After experimental analysis, we set the a and m parameters in the loss function to 1.2 and 0.8, respectively (see Table IV). In equations 7 and 8, w_1 and w_2 are set to 1.0 and 0.2 for the single-level embeddings settings and to 0.5 and 0.2 for the multi-level embeddings settings. As pre-processing steps, we only resize and normalize all images in training and testing sets, so that they fit to the pre-trained ResNet50 architecture requirements, and we avoid any data augmentation techniques to keep simple and efficient training settings and focus on the effects of the different loss functions.

During training, for the single-level embeddings, we keep the first layers of the network fixed and update the weights after the 5a layer, while for the multi-level embeddings we further unfreeze layers and train the network from the 4a to 5c layers. For the multi-level embeddings, we sequentially insert a max-pooling layer, a feature extraction FC layer and a classification FC layer right after each of 4c, 4f and 5c layers of ResNet50, in order to extract low, medium and high-level features. Feature extraction FC layers output feature vectors of size 128. Moreover, the Adam optimiser is used and we set the learning rate for the backbone layers to 0.0001. Similar to [27], the new layers are randomly initialised and optimised with a learning rate ten times larger than the backbone layers for faster convergence.

C. Evaluation Protocol

In our experiments, we follow the evaluation protocol proposed in [28] measuring the mean Average Precision (mAP) and the rank-k accuracy. For mAP, we compute the mean of all queries' average precision, i.e., the area under the Precision-Recall curve. The rank-k hit rate denotes the possibility that at least one true positive is ranked within the top k positions. In addition, rank-k mAP, introduced by the creators of CityFlow-ReID dataset [26], measures the mean of average precision for each query considering only the top k matches. Therefore, we use this metric when evaluating on CityFlow-ReID, where $k = 100$.

D. Results

In this section, we present the results of our system on VeRI-776 and CityFlow-ReID datasets, and compare our system's performance with existing state-of-art methods. As a baseline, we consider our ReID system simply using triplet loss combined with identity classification. All loss functions listed in Tables I and II are combined with the identity classification loss. I2I, I2T and T2T represent image-to-image, image-to-tracklet and tracklet-to-tracklet similarity measuring at the testing phase. Also, L, M and H stand for Low Medium and High-level features aggregated in the multi-level embeddings network. Note that CityFlow-ReID does not provide information about queries' tracklets, therefore for this dataset we are

	I2I			I2T			T2T		
	mAP	R-1	R-5	mAP	R-1	R-5	mAP	R-1	R-5
[5]	58.05	-	-	-	-	-	-	-	-
[10]	59.47	96.24	98.97	-	-	-	-	-	-
[26]	67.60	90.20	-	-	-	-	-	-	-
[29]	61.83	88.50	94.46	-	-	-	-	-	-
[11]	67.55	90.23	96.42	-	-	-	-	-	-
[4]	-	-	-	69.08	-	-	-	-	-
[8]	53.45	83.49	92.55	-	-	-	-	-	-
[14]	82.14	96.25	-	85.20	96.6	-	-	-	-
[30]	64.60	92.3	95.7	-	-	-	-	-	-
[16]	81.90	96.20	97.90	-	-	-	-	-	-
[31]	66.06	91.78	96.42	-	-	-	-	-	-
Baseline	49.52	80.87	92.25	58.02	80.51	81.94	67.60	87.72	88.08
SSL	54.60	78.24	91.06	62.13	80.51	81.10	68.00	85.69	86.17
RLL	56.64	79.43	91.17	63.37	78.96	79.55	69.24	87.00	87.42
RLL (L,M,H)	62.48	88.64	94.75	68.25	87.90	88.31	73.34	91.06	91.41

Table I: Evaluation results on VeRi-776.

	I2I				I2T			
	mAP@100	R-1	R-5	R-10	mAP@100	R-1	R-5	R-10
[30]	23.6	48.9	59.7	-	-	-	-	-
[14]	-	-	-	-	46.31	-	-	-
[32]	-	-	-	-	25.05	-	-	-
[24]	-	-	-	-	31.57	42.21	42.21	42.78
[26]	32.0	48.4	65.2	-	-	-	-	-
[3]	-	-	-	-	35.02	49.24	49.52	-
[31]	23.1	-	-	-	-	-	-	-
Baseline	18.48	40.19	58.29	66.10	33.42	44.39	44.39	45.25
SSL	19.51	42.67	61.90	70.10	33.27	41.14	41.14	42.10
RLL	21.85	40.19	59.43	69.17	34.18	44.57	44.76	45.33
RLL (L,M,H)	24.59	47.07	62.36	69.77	36.44	46.48	46.48	47.43

Table II: Evaluation results on CityFlow-ReID.

not able to compute tracklet-to-tracklet similarity. For all of the works that post-process their results by using re-ranking methods we report their results prior to re-ranking for fair comparison.

As seen in Tables I and II, RLL performs better than SSL and Baseline in terms of mAP score. This could be explained by the fact that sampling groups of positive and negative samples for each anchor during training, better imitates the testing settings, where we aim to retrieve a group of positive samples among thousands of negative samples that compose the gallery set. Note that according to a previous study [28], mAP is a more robust metric than rank-k score and provides a general measure of ReID performance. This is because it considers whether all of the relevant samples tend to get highly ranked, whereas rank-k score only rewards the single highest-ranked relevant sample.

In addition, exploiting time information seems to boost our system's performance significantly, since the highest accuracy is obtained when time-pooling is applied to both query and gallery candidates (see Table I).

Our training procedure is simpler than most of the approaches presented in Tables I and II, with an efficient embedding dimension of 384 (3×128). In addition, none of these methods report results for the case of tracklet-to-tracklet

similarity measuring. Although [14] and [16] outperform our system's accuracy, they both employ a much more complex MGN architecture with an embedding size $5.3 \times$ bigger than ours. Furthermore, both methods use data augmentation and pre-processing techniques to increase the resolution of vehicle images [14]. Other works [4], [10], [31], using an embedding size $2.6 \times$ larger than ours, achieve better rank-1 and rank-5 accuracy, but in terms of mAP, our approach performs better indicating robustness at all ranks. Also, it is worth noting that our method outperforms several other works ([5], [30] on CityFlow-ReID) that utilize further annotations (e.g. colour, orientation). As opposed to the works reported in Tables I and II, our system is trained by combining two datasets and it is not being specifically fine-tuned on any of the two. Finally, the exact same model provides the reported results for the two datasets, indicating the generalization ability of our system.

E. Ablation Study

1) *Impact of a and m* : As explained in section III-A, for each query we only sample the negative and positive pairs that produce non zero loss. This means that we select the positive pairs whose distance from the query is larger than $a - m$, while for the negative pairs, we sample only those whose distance from the query is smaller than a . The sample

Figure 4: Qualitative results for 4 query images from VeRi-776 [1] dataset. For each query, there is a sequence of 10 gallery images corresponding to the ranking results produced by our ReID system, using (a) single-level and (b) multi-level embeddings. Vehicles surrounded by green box denote the same vehicle as the probe, otherwise it is surrounded by a red box.

mining range depends on the values of the parameters a and m . Therefore, we examine their influence on the ReID results according to evaluation on VeRi-776 for the case of single-level embeddings.

$m = 0.8$	mAP	rank-1	rank-5
$a = 1.0$	53.34	74.79	89.03
$a = 1.2$	56.64	79.43	91.17
$a = 1.4$	50.94	76.87	89.51
$a = 1.6$	51.59	78.42	91.47

Table III: The impact of a margin when evaluating on Veri-776 dataset.

$a = 1.2$	mAP	rank-1	rank-5
$m = 0.4$	51.66	78.60	90.82
$m = 0.6$	55.24	79.67	91.59
$m = 0.8$	56.64	79.43	91.17
$m = 1.0$	52.32	74.07	88.31

Table IV: The impact of m margin when evaluating on Veri-776 dataset.

2) *Impact of K parameter*: Another critical parameter when applying the RLL loss function is K , which represents the number of different vehicle identities per batch. For this experiment we used single-level embedding settings, while keeping $a = 1.2$ and $m = 0.8$. In loss computation, K value increases along with the number of positive and negative samples for every anchor. Table V demonstrates K effects on our system. It can be noticed that this parameter does not seem critical as it does not significantly affect the accuracy. However, a larger number of identities which results in more images per batch, potentially increases noise and makes the learning process more difficult.

	mAP	rank-1	rank-5
$K=3$	56.64	79.43	91.17
$K=4$	56.46	78.42	91.95
$K=5$	54.46	77.77	90.88

Table V: Impact of K parameter.

3) *Impact of multi-level embeddings*: At inference time for each sample we aggregate the features produced by low, medium and high-level layers of the network. To examine the

effectiveness of multi-level features we conduct experiments of using only one kind of features at testing time compared to concatenating them, while keeping $a = 1.2$, $m = 0.8$ and $K = 3$ in multi-level embedding training configuration. Table VI proves our hypothesis that multi-level embeddings provide richer information that lies in the intermediate network layers. This information increases significantly the discriminative power of our ReID system.

Features	mAP	rank-1	rank-5
L	54.08	80.51	92.61
M	59.45	84.50	93.92
H	59.52	85.28	93.56
L, M, H	62.48	88.64	94.75

Table VI: Impact of multi-level embeddings on Veri-776 results. (L, M, H) stands for the concatenation of Low-level, Mid-level and High-level features.

Finally, we provide some qualitative results (see Fig. 4), to compare our system's performance, with single and multi-level embedding settings. It can be observed that our system has effectively learned to re-identify the vehicles and discard several false positives by paying attention not only to global appearance but also to small crucial details such as brand logos, frontal grid and stickers.

V. CONCLUSIONS

In this work, we build a robust end-to-end framework for vehicle ReID, which is able to effectively identify vehicles from both image and video data. Competitive results are achieved on two large-scale datasets while avoiding data augmentation, further fine-tuning and re-ranking, to ensure the generalization of our system across different datasets. Combining features from different levels of the network allows us to obtain stronger feature representations and improves our system's performance up to 6 % in mAP score and 9% in rank-1. Finally, when compared to the image-based approach, our video-based ReID framework, which uses temporal information, results in up to 12% mAP improvement.

REFERENCES

- [1] X. Liu, W. Liu, T. Mei, and H. Ma, "A deep learning-based approach to progressive vehicle re-identification for urban surveillance," in *Computer Vision - ECCV 2016, Proceedings, Part II*, ser. Lecture Notes in Computer Science, vol. 9906, 2016, pp. 869–884.
- [2] C. Wu, C. Liu, C. Chiang, W. Tu, and S. Chien, "Vehicle re-identification with the space-time prior," in *2018 IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops*, 2018, pp. 121–128.
- [3] K. Nguyen, T. Hoang, M. Tran, T. Le, N. Bui, T. Do, V. Vo-Ho, Q. Luong, M. Tran, T. Nguyen, T. Truong, V. Nguyen, and M. N. Do, "Vehicle re-identification with learned representation and spatial verification and abnormality detection with multi-adaptive vehicle detectors for traffic video analysis," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops*, 2019, pp. 363–372.
- [4] C. Liu, M. Lee, C. Wu, B. Chen, T. Chen, Y. Hsu, and S. Chien, "Supervised joint domain learning for vehicle re-identification," in *CVPR Workshops*, 2019, pp. 45–52.
- [5] N. Jiang, Y. Xu, Z. Zhou, and W. Wu, "Multi-attribute driven vehicle re-identification with spatial-temporal re-ranking," in *2018 IEEE International Conference on Image Processing, ICIP 2018*, pp. 858–862.
- [6] P. Khorramshahi, A. Kumar, N. Peri, S. S. Rambhatla, J. Chen, and R. Chellappa, "A dual path model with adaptive attention for vehicle re-identification," *CoRR*, vol. abs/1905.03397, 2019.
- [7] W. Liu, X. Liu, H. Ma, and P. Cheng, "Beyond human-level license plate super-resolution with progressive vehicle search and domain prior GAN," in *Multimedia Conference (MM) 2017*, pp. 1618–1626.
- [8] J. Zhu, H. Zeng, Z. Lei, S. Liao, L. Zheng, and C. Cai, "A shortly and densely connected convolutional neural network for vehicle re-identification," in *24th International Conference on Pattern Recognition, ICPR 2018*, pp. 3285–3290.
- [9] X. Liu, H. Ma, and S. Li, "PVSS: A progressive vehicle search system for video surveillance networks," *J. Comput. Sci. Technol.*, vol. 34, no. 3, pp. 634–644, 2019.
- [10] Y. Bai, Y. Lou, F. Gao, S. Wang, Y. Wu, and L. Duan, "Group-sensitive triplet embedding for vehicle reidentification," *IEEE Trans. Multimedia*, vol. 20, no. 9, pp. 2385–2399, 2018.
- [11] R. Kumar, E. Weill, F. Aghdasi, and P. Sriram, "Vehicle re-identification: an efficient baseline using triplet embedding," in *International Joint Conference on Neural Networks, IJCNN 2019*, pp. 1–9.
- [12] Y. Guo and N. Cheung, "Efficient and deep person re-identification using multi-level similarity," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018*, pp. 2335–2344.
- [13] X. Wang, Y. Hua, E. Kodirov, G. Hu, R. Garnier, and N. M. Robertson, "Ranked list loss for deep metric learning," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019*, pp. 5207–5216.
- [14] A. Ayala-Acevedo, A. Devgun, S. Zahir, and S. Askary, "Vehicle re-identification: Pushing the limits of re-identification," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2019*, pp. 291–296.
- [15] M. Chang, J. Wei, Z. Zhu, Y. Chen, C. Hu, M. Jiang, and C. Chiang, "AI city challenge 2019 - city-scale video analytics for smart transportation," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2019*, pp. 99–108.
- [16] Y. Chen, L. Jing, E. Vahdani, L. Zhang, M. He, and Y. Tian, "Multi-camera vehicle tracking and re-identification on AI city challenge 2019," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2019*, pp. 324–332.
- [17] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2005*, pp. 539–546.
- [18] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015*, pp. 815–823.
- [19] A. Hermans, L. Beyer, and B. Leibe, "In defense of the triplet loss for person re-identification," *CoRR*, vol. abs/1703.07737, 2017.
- [20] M. Chen, Y. Ge, X. Feng, C. Xu, and D. Yang, "Person re-identification by pose invariant deep metric learning with improved triplet loss," *IEEE Access*, vol. 6, pp. 68 089–68 095, 2018.
- [21] M. T. Law, R. Urtasun, and R. S. Zemel, "Deep spectral clustering learning," in *Proceedings of the 34th International Conference on Machine Learning, ICML 2017*.
- [22] E. Kamenou, J. Martinez del Rincon, P. Miller, and P. Devlin Hill, "A simple deep learning architecture for city-scale vehicle re-identification," *Irish Machine Vision and Image Processing Conference, IMVIP 2019*.
- [23] F. Wu, S. Yan, J. S. Smith, and B. Zhang, "Joint semi-supervised learning and re-ranking for vehicle re-identification," in *24th International Conference on Pattern Recognition, ICPR 2018*, pp. 278–283.
- [24] J. Spanhel, V. Bartl, R. Juránek, and A. Herout, "Vehicle re-identification and multi-camera tracking in challenging city-scale environment," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2019*, pp. 150–158.
- [25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*, pp. 770–778.
- [26] Z. Tang, M. Naphade, M. Liu, X. Yang, S. Birchfield, S. Wang, R. Kumar, D. C. Anastasiu, and J. Hwang, "Cityflow: A city-scale benchmark for multi-target multi-camera vehicle tracking and re-identification," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019*, pp. 8797–8806.

- [27] H. O. Song, Y. Xiang, S. Jegelka, and S. Savarese, “Deep metric learning via lifted structured feature embedding,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*, pp. 4004–4012.
- [28] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian, “Scalable person re-identification: A benchmark,” in *2015 IEEE International Conference on Computer Vision, ICCV 2015*, pp. 1116–1124.
- [29] J. Zhu, H. Zeng, J. Huang, S. Liao, Z. Lei, C. Cai, and L. Zheng, “Vehicle re-identification using quadruple directional deep learning features,” *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 1, pp. 410–420, 2020.
- [30] A. Kanaci, M. Li, S. Gong, and G. Rajamanoharan, “Multi-task mutual learning for vehicle re-identification,” in *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2019*, pp. 62–70.
- [31] A. Shankar, A. Poojary, V. Kollerathu, C. Yeshwanth, S. Reddy, and V. Sudhakaran, “Comparative study on various losses for vehicle re-identification,” in *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2019*, pp. 256–264.
- [32] E. Luna, P. Moral, J. C. SanMiguel, Á. García-Martín, and J. M. Martínez, “Vpulab participation at AI city challenge 2019,” in *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2019*, pp. 343–352.