Vehicle Re-Identification based on Ensembling Deep Learning Features including a Synthetic Training Dataset, Orientation and Background Features, and Camera Verification.

Marta Fernández, Paula Moral, Álvaro García-Martín, and José M. Martínez
Video Processing and Understanding Lab
Universidad Autónoma de Madrid, Madrid, Spain
marta.fernandezd@uam.es, paula.moral@uam.es,alvaro.garcia@uam.es,josem.martinez@uam.es

Abstract

Vehicle re-identification has the objective of finding a specific vehicle among different vehicle crops captured by multiple cameras placed at multiple intersections. Among the different difficulties, high intra-class variability and high inter-class similarity can be highlighted. Moreover, the resolution of the images can be different, which also means a challenge in the re-identification task. Intending to face these problems, we use as baseline our previous work based on obtaining different deep learning features and ensembling them to get a single, stable and robust feature vector. It also includes post-processing techniques that explode all the information provided by the CityFlowV2-ReID dataset, including a re-ranking step. Then, in this paper, several newly included improvements are described. Background and orientation similarity matrices are added to the system to reduce bias towards these characteristics. Furthermore, we take into account the camera labels to penalize the gallery images that share camera with the query image. Additionally, to improve the training step, a synthetic dataset is added to the original one.

1. Introduction

Vehicle Re-identification (Vehicle reID) is a computer vision task whose relevance has been increasing during the last years due to the growing emergence of smart cities that make use of this technology. Moreover, it is gaining prominence in Intelligent Transport Systems (ITS) since this new technology allows, for example, obtaining knowledge about the traffic flow, which makes it possible to adapt the traffic lights or provide useful information to the autonomous driving field [26]. The main objective of these systems is to identify a particular vehicle (query) recorded by a camera among a set of gallery images that have been recorded by different cameras. Ideally, the result is a ranked list in which the first images have the same identity as the query image. However, the ReID task is composed of different challenges. Firstly, the small variability between different vehicles is notable due to their similar orientation, colour or model, among other characteristics. In fact, similar background and orientation often generate a severe bias on the final distance matrix, since it reduces the distance between different vehicles. At the same time, there is a large variability between frames of the same vehicle because of different illumination conditions, resolutions or points of view. Furthermore, the number of labelled images is limited, which can lead to poor results. However, the creation of synthetic datasets with a large number of images and labels, such as VehicleX [23], helps to reduce this problem.

Regarding this information, we base the work presented in this paper in [14], which consists of a feature ensembling method that uses four feature vectors: three features are obtained from three different appearance extraction networks [12], whilst the fourth is the output of another extraction network [11, 1] that combines appearance and structure information. Then, they are concatenated to obtain a more discriminant and robust single vector. To avoid possible errors, a post-processing step is carried out, which includes a re-ranking technique and the use of tracking information.

The work presented in this paper proposes to add several additional techniques to the baseline system to improve the results. Firstly, we propose to follow the idea included in [30] which consists of using a convolutional neural network to obtain orientation and background features. By using them, orientation and background distance matrices are calculated to reduce the network bias towards these two characteristics. Moreover, we explode the idea that a query image and the correct candidates can not have been recorded by the same camera. Finally, a synthetic dataset is added to improve the training step.

The main dataset is CityFlowV2-ReID, which is a subset of CityFlow [19]. It consists of 3.58 hours of synchronized

HD videos from 46 cameras across 16 intersections with a maximum distance between cameras of 2.5 km. In total, it contains 200k annotated bounding boxes with different characteristics such as viewing angles or vehicle models. In particular, CityFlowV2-ReID contains a total of 85058 images of which 52717 are part of the training set, 31238 are part of the test set and, finally, the query set is composed of 1103 images. In total, there are 440 identities for both test and training sets. As commented before, in addition to CityFlowV2-ReID, we introduce the use of VehicleX [23] to increase the training set. It is a synthetic dataset composed of 192150 images that, apart from cameraID and vehicleID, has available numerous labels such as orientation, colour or type.

The paper is organized as follows. After this introductory section, Section 2 describes the related work and Section 3 provides a full explanation of our approach. Then, section 4 includes the experimental evaluation description and results provided by the 2021 AI City challenge server [15] and our evaluation environment. Finally, the conclusions are stated in Section 5.

2. Related Work

Object re-identification has gained a lot of popularity during recent years due to its important role in some fields. In particular, numerous works have focused on person ReID due to its importance in some areas such as group behaviour analysis [20], or long-term person tracking [4], among others. These researches have served as a basis for other object re-identification techniques, e.g. vechicle-ReID. For instance, [29] has had an important role in the progress of ReID systems due to its proposal of a re-ranking method which has subsequently been used numerous times. Moreover, [9] introduces a triplet loss variant which has been notable because of its successful results. As a consequence, other researchers have based their methods on it. Finally, another notable work is [28], which proposes a learning framework that joins the ReID learning and data generation end-to-end. However, vehicle ReID is still a challenging task because of the intra-class and inter-class difficulties caused by vehicles orientation, illumination or resolution changes, among others.

The following subsections cover the related work regarding the techniques used in our proposed ReID system.

2.1. Re-identification features

The increasing success of deep learning has contributed to making the convolutional neural networks one of the most widely used schemes. In fact, [30, 26, 8] have achieved top results in the current state of the art by exploiting different deep learning techniques. One of the main advantages these methods have is the resulting features, which are more discriminant than the traditional extractors.

Depending on the input to the system and the information to be obtained, the ReID approaches can be classified into two groups, image-based and video-based. Image-based methods obtain features without taking into account temporal information. In this line of work, [12] makes use of DenseNet121 [10] to extract different features related to appearance. The features depend on the configuration of the network which varies between the use of label smoothing regularization (LSR) [18] and triplet loss with hard or soft margin [9]. Additionally, [30] trains a convolutional neural network to obtain appearance, background and orientation features. It is also important to mention that in order to get more discriminant vectors, other works propose the utilisation of vehicle keypoints. For example, [11] infers orientation information by extracting 36 keypoints through the method described in [1].

On the other hand, video-based techniques obtain the features from video clips, *i.e.*, a set of consecutive images of the same vehicle. It includes temporal information which contributes to face some challenges such as scale variations. Some researches, like [6], test the application of temporal pooling and temporal attention models. On the basis of these proposals, [11] proposes a viewpoint-aware temporal attention model that uses deep learning features extracted from video clips. Another notable work is [24], which introduces the integration of a top-push distance learning model (TDL) for matching video features. Finally, a Spatial and Temporal Attention Pooling Network (ASTPN) is presented in [22].

This variety of possibilities leads to several works that propose to generate different features and ensemble them. Feature ensemble is a technique that is a widely utilised method in the ReID field. It consists of combining the resulting features from different extractors to obtain a more discriminative and robust representation. A great number of works take advantage of this technique [27, 26, 12]. In particular, [27] proposes to ensemble different features extracted from eight trained models. Based on this scheme, [26] proposes the ensemble of twelve features. Finally, as commented before, [12] obtains three different features that are also ensembled.

2.2. Camera and Orientation ReID

As commented before, deep learning techniques are being extensively used in the ReID tasks. However, the functioning of these methods can lead to bias issues. In particular, [30] has shown that networks often learn information that can result in subsequent errors. For example, if images contain a high amount of background, this information will be encoded in the feature vectors. Thus, it can cause the network to identify two different vehicleID as the same because they have similar backgrounds and, therefore, similar features. Analogously, this also happens with the orientation of vehicles. Due to these facts, [30] proposes to train a network focused on orientation and background to prevent this bias from appearing in the results.

2.3. Re-Ranking

The re-ranking step is a post-processing method used to improve the initial ranking without needing any additional training. Specifically, it consists of re-estimating the distances between the query and the gallery images by taking into account the likeness of their neighbourhood. This idea comes from the fact that the similarity between two images should not only be calculated by the distance between them, but also by the distance between their neighbours. Based on this statement, [2] proposes to encode in a vector the local distribution of an image and compare it with another image by using the Jaccard distance. Besides, [13] applies the idea that a true match should be similar to the query image in different baseline methods. Another remarkable work is [17], which introduces a re-ranking technique based on the k-nearest neighbours of the query. Based on these ideas, [16] applies the re-ranking by accumulating the distances of the immediate two-level neighbours for a pair of images. Moreover, a notable method used during the last years is the one proposed by [29], which exploits the k-reciprocal nearest neighbours concept. Two images are k-reciprocal nearest neighbours if both of them are in the top rank (top-k) when the other image is the query. Following this idea, they encode this information in a vector and then they calculate the Jaccard distance between them. Works such as [26, 30, 8], which have achieved remarkable results, make use of this proposal.

3. Proposed Method

The aim of this section is to give a detailed description of the proposed method for vehicle ReID, which is represented in Figure 1.

Firstly, regarding the baseline system [14], we can divide the system into two main groups, the image-based, which works with individual frames, and the video-based, whose input is a set of consecutive images of the same vehicle. Specifically, the image-based group consists of a CNN (Convolutional Neural Network) with three different configurations while the video-based block includes a CNN module and a keypoint and structure estimator. The test step infers the four feature vectors (image and video-based) for all the images. Then, they are concatenated to obtain a more robust vector, and, to refine it, query expansion and temporal pooling are carried out. Once we have the final features, it is possible to calculate the distances between query and gallery images. Finally, a re-ranking step and the use of the trajectory information provide the final results.

Then, this work proposes new enhancements. Firstly, we add the VehicleX dataset [23] to the training set to train the commented networks. Besides, in order to face the intra-class variability and the inter-class similarity, we penalize the gallery images that share the recording camera with the query image, since it is a scenario that is not possible in vehicle ReID. With the same objective, we penalize the initial distances using orientation and background features to avoid possible bias towards these characteristics. This additional information is obtained through the use of another CNN block [30].

3.1. Feature Extraction

3.1.1 Image-based feature extractors

Following [14] and [12], the chosen network for this task is DenseNet121 [10] which has been trained with ImageNet [3]. In particular, triplet loss and cross-entropy loss are used to train this feature extractor. As commented before, this block outputs three different features that depend on different variations of the loss functions. Specifically, all of them include label smooth regularisation (LSR) [18] and triplet loss that varies between the use of hard or soft margin [9]. Another difference is the utilization of Jitter augmentation [12], a data augmentation technique.

3.1.2 Video-based features extractor

This part of the system receives as input a set of images of the same vehicle that are consecutive in time (*track information file*) with the objective of representing the spatial structure of a vehicle. It uses ResNet50 [7], a convolutional neural network trained on ImageNet [3], to get appearance features. As commented before, the appearance information is not enough to perform vehicle ReID properly. Due to this fact, it is proposed to obtain also structure features. Based on the ideas explained in [1] and [11], 36 vehicle keypoints are located to define 18 vehicle orientation surfaces, which allows inferring the orientation of the car. One example is shown in Figure 2, where the yellow arrow indicates the driving direction [11]. Then, both appearance and structure information are concatenated following a temporal attention model [11]. The resulting video feature is calculated for all the gallery video clips and query images (video clip with a single image). Concerning the training step, it is proposed to use triplet loss with hard margin and cross-entropy. Similar to the previous section, this block is part of [14].

3.2. Feature Ensemble

This block consists of the concatenation of the four different features provided by the image-based and the video-based extractors. To concatenate them, it is necessary to normalize the vectors using L_2 normalization [14].

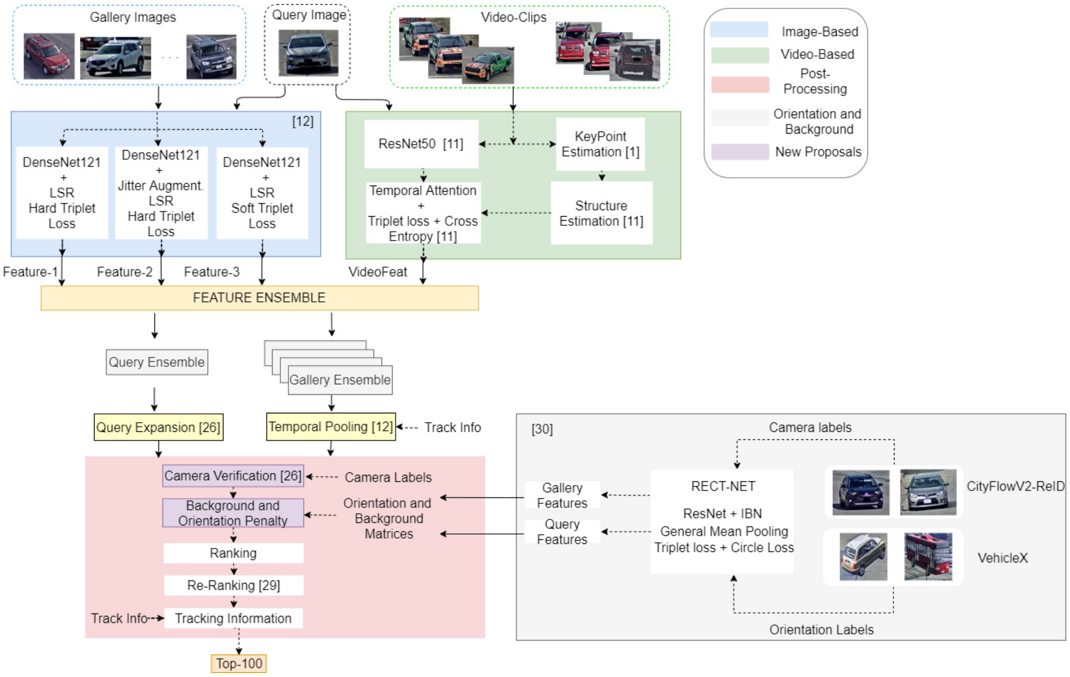

Figure 1. Proposed system overview. It is based on [14]. It is composed of two extraction blocks. The image-based module extracts the appearance features with different configurations while the video-based module extracts appearance and structure data by using temporal information. Then, all the features are ensembled followed by query expansion and temporal pooling. Afterwards, a post-processing module performs several techniques: firstly, it includes camera verification to avoid false negatives; in parallel, orientation and background similarity matrices are obtained to apply them as penalty to the distances matrices; and, finally, a re-ranking step and the trajectory information are applied.

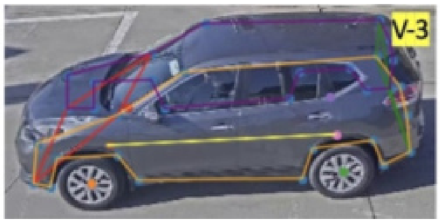

Figure 2. Example of vehicle keypoint detection and surfaces and direction estimation. [11].

3.3. Query Expansion and Temporal pooling

This section explains the methods incorporated to improve the feature representation. With this purpose, and as proposed in [14], we apply query expansion and temporal pooling.

- **Query expansion.** This method aims to improve the representation of the query images. It consists of using DBSCAN, a clustering method [5], to find the most similar samples. Then, the query vectors are replaced by the mean of the vectors that are in the same cluster. To improve the results, the images with low resolution are not involved [26].

- **Temporal Pooling** This module intends to improve the gallery representation. Taking into account the trajectory information *(Track information file)*, the gallery ensembles are replaced by the average calculated between the T-1 consecutive image vectors [12].

3.4. Post-processing

- **Background and Orientation penalty**. Following the scheme explained in [30], we propose to obtain orientation and background distances between images. The reason for this is that sometimes ReID methods identify two identities as the same because they have the same orientation or background, *i.e.*, they could imply a bias. Therefore, to reduce it, we use RECT-Net, a network proposed in [30]. It consists of a combination of ResNet, with Generalized-mean Pooling, Circle loss and Triplet loss. Moreover, to improve the training, [30] proposes to augment the training data. Then, a weakly supervised detection method is applied to obtain images with a more tightly cropping of the vehicle. This process doubles the dataset since the network is trained with both original and cropped sets.

  Background distance matrix. To obtain this information, we train RECT-Net [30] with the cameraID as labels since the background is directly related to the recording camera. After doing this, we obtain feature vectors that encode the background information. Then, the background distances are calculated using the cosine distance.

Orientation distance matrix. Similar to the background, this module has as objective to remove the orientation bias. As before, RECT-Net [30] is trained with the orientation as labels. Since the CityFlowV2-ReID dataset [19] does not include this information, it is necessary to train it with VehicleX [23], a synthetic dataset that has orientation labels available. The angles (0-360) are divided into 36 bins, where each one is a different label [30]. Again, once the features are obtained, it is possible to compute the orientation distance (cosine distance) between vehicles.

Then, these matrices are used as penalty to get the final distances between images. After several experiments, we conclude that the best trade-off to apply the penalty is as indicated in eq.1. The initial distance matrix is indicated by d_i while d_f is the final distance matrix. The background and orientation distance matrices are d_b and d_o, respectively.

$$d_f = d_i + 0.05d_b + 0.05d_o \tag{1}$$

- **Camera verification**. This step has as objective to improve the results by using the camera information. Specifically, it is proposed to make use of the fact that if two vehicles have been recorded by the same camera, they can not be the same ID [26]. Therefore, the similarity between vehicles that have the same camera label is set to 0. In fact, there is a relation between this step and the background bias removal. The vehicles that are recorded by the same camera have the same background, so this step contributes to avoiding the background bias.

- **Re-ranking with k-reciprocal encoding**. As commented before, the re-ranking step is a post-processing method that aims to improve the initial ranking. We propose to use the idea explained in [29], which states that if a gallery image is close to the query image in the k-reciprocal nearest neighbours, it is more likely to be a true positive [14].

- **Trajectory information**. This step prevents the system from including some false positives in the final top-100 ranking by using the tracking information provided by CityFlowV2-ReID [19]. Following [14], the tracks are sorted depending on their first appearance in the top-100 candidates. Then, all the track images are added to the final ranking until the top-100 list is completed.

4. Experimental Validation

This section shows and analyses the results obtained with the proposed ReID scheme in the evaluation system provided by the 2021 AI City Challenge [15]. Additionally, it includes the performance estimated by our own evaluation method which has guided the development of the ReID proposal.

4.1. Dataset

This section describes the datasets that have been part of our study. Firstly, the main dataset is CityFlowV2-ReID dataset, which is a subset of CityFlow [19]. It contains 85058 images collected from 46 cameras where 52717 are part of the training set and 31238 constitute the test set. In total, both sets contain 440 identities. Besides, 1103 images are employed as queries. The training set includes vehicle and camera labels annotated, whereas the test and the query sets contain only the camera labels.

Moreover, the dataset provides trajectory information for both training and test sets. Specifically, this information consists of a file that contains all the tracks. A track includes images of the same vehicle captured by one camera. Since the camera labels are included in both sets, it is possible to know which camera has recorded each of the tracks. In addition, this information indicates that there are 2173 trajectories in the training set and 991 in the test set.

Finally, the challenge also provides a synthetic vehicle dataset generated by VehicleX [23]. It is a labelled dataset that includes a total of 192150 synthetic images that belong to 1362 identities. Apart from the camera and vehicle labels, the colour, orientation, type, light direction, light intensity, camera height and camera distance are also annotated.

4.2. Parametrization

The proposed methods have been trained in a NVIDIA GeForce GTX 1080Ti with a GPU RAM capacity of 11 GB. Moreover, the processor is a Xeon 4114 with 32 GB RAM Memory.

As proposed in [14], the appearance features resulting from the image-based block are extracted using Densenet121 [10] pre-trained on ImageNet [3]. A mini-batch SGD and a learning rate of 0.0001 are used to train 80 epochs and the input images are resized to 256x256. The video-based block also utilizes ResNet50 [7]. It is trained during 305 epochs and it applies the Adam optimizer with a learning rate of 0.0001. The images are resized to 224x224.

The orientation and background features are extracted from RECT-Net [30], which uses ResNet50-IBN-a [21], pre-trained on ImageNet [3], as backbone. Both models are trained during 12 epochs with a learning rate that decays from 3.5e-4 to 7.7e-7. The model employed to crop the images uses the same configuration. The orientation model is trained with VehicleX [23] and the background model with CityFlowV2-ReID dataset [19]. Moreover, regarding the camera verification step, the similarity between images that share camera is set to 0.

Finally, according to [14] and [11], the parameter T, used in temporal pooling, is set to 6. Similarly, the parameters involved in DBSCAN [5] are those proposed by the authors.

4.3. Experimental results

This section covers all the results obtained for all the experiments that have been carried out to analyze the performance of the proposed algorithm. The evaluation is based on two metrics: the mean Average Precision (mAP) [25] and the Cumulative Matching Curve (CMC). Regarding the CMC, the rank-1, rank-5, rank-10, rank-15, rank-20, rank-30 and rank-100 are also indicated. The evaluation is divided into the results provided by our evaluation method, which has been our main reference during development, and the AICITY21 challenge online server [15].

- **Proposed evaluation system.** As mentioned before, CityFlowV2-ReID is a subset of CityFlow [19]. Specifically, CityFlowV2-ReID test set is part of the validation set of CityFlow for multi-target multicamera vehicle tracking task. Then, it has all the necessary annotations to create our evaluation environment. To do this, we have compared the gallery and query images with the CityFlow validation set. However, CityFlowV2-ReID includes some cameras which are not annotated. Therefore, to evaluate properly the performance, we create a subset of query and gallery images without the samples recorded by the missing cameras. It is important to note that due to this fact the results are slightly different from those provided by the challenge server. However, they are a good reference for knowing how the system works.

- **2021 AI City Challenge evaluation server.** This server is provided to the challenge participants to allow them to evaluate their methods and compare them with the top-3 candidates. It is allowed to submit a maximum of 5 results per day up to a total of 20 results for the entire challenge. During the challenge, the results are evaluated over 50% of test data. Once the deadline is reached, the results achieved with 100% of the test data are shown.

4.3.1 Proposed evaluation system

Table 1 shows the results evaluated in our own evaluation system. To gain knowledge about the contribution of the different proposals, each row shows the result obtained by applying the different blocks included in the scheme illustrated in Figure 1. "Feature-1", "Feature-2" and "Feature-3" correspond to the three possible combinations of the appearance extractor. Similarly, "VideoFeat" corresponds to the feature extracted from the video-based scheme.

Concerning the performance of the baseline (white rows), the first noticeable result is that the ensemble of the four vectors (Ensemble 1-2-3 + VideoFeat), means an increase of 12.24% over the result of the best feature (Feature-3) and 3.42 % over the appearance ensemble (Ensemble 1-2-3). Moreover, adding to the ensemble the trajectory information increases the mAP to 0.3772.

Regarding the new proposals (grey rows), "Orientation" and "Background" refer to the use of the orientation and background distance matrices. It is demonstrated that applying both matrices (mAP = 0.4671) increase the result by 23.83% compared to the previous result achieved with the baseline (mAP = 0.3772). Besides, "Camera" refers to the result obtained by adding to the baseline the camera verification step. It is possible to check that its utilisation increases the mAP to a value of 0.4611. Finally, the application of the complete scheme increases the result up to mAP = 0.5108.

A visual result is represented in Figure 3. The upper rows illustrate the result achieved with the whole scheme while the lower rows show the result obtained by using the initial baseline [14]. This example illustrates the commented orientation bias since we can check how the baseline selects incorrectly, as a top-1 candidate, a vehicle that has the same orientation as the query. However, the proposal made in this work demonstrates to solve this problem, since, as we can see in this example, it avoids this bias.

4.3.2 Online evaluation server

Finally, this section provides the results obtained with the challenge evaluation server [15]. Firstly, Table 2 shows the results achieved with different combinations of the proposals. ”Baseline” refers to the reference system [14] without the addition of tracking information. "Camera", "Orientation" and "Background" indicate the application of the camera verification step and the orientation and background matrices, respectively. It is shown that, in coherence with our evaluation system, each of the proposed blocks improves the baseline score while the combination of all of them gives the best output. Finally, the addition of tracking information leads to the most successful outcome (0.4900).

Moreover, Table 3 provides the final ranking. The first three positions, our proposal and the last position are shown. It can be seen that the system proposed in this paper has finally obtained the 16th position.

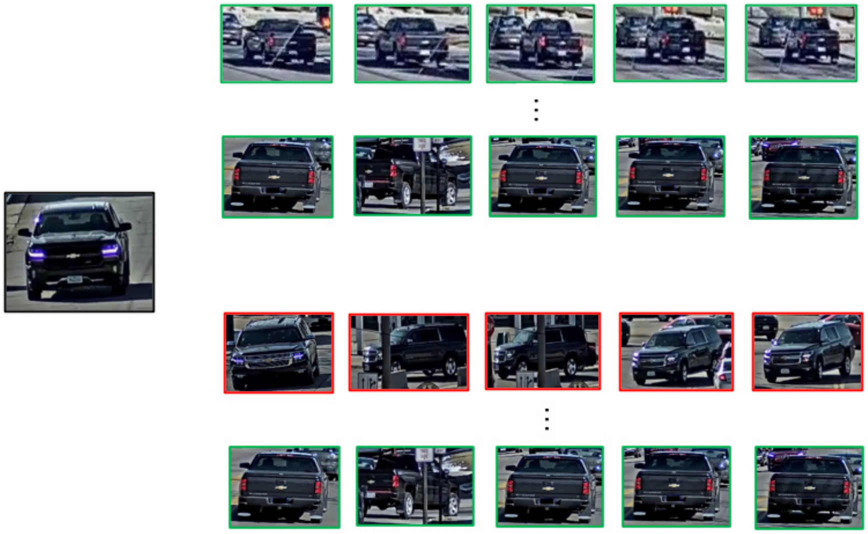

Figure 3. Example of the visual results for the proposed ReID system. The image on the left is the query while the images on the right are the candidates. The lower rows are the results obtained with the baseline. Then, the upper rows are the results of applying the proposed enhancements. Green boxes represent true positives and red boxes false positives.

	Rank-100 mAP	CMC-1	CMC-5	CMC-10	CMC-15	CMC-20	CMC-30	CMC-100
Feature-1	0.3140	0.5523	0.5683	0.6003	0.6074	0.6394	0.6678	0.8721
Feature-2	0.2684	0.5079	0.5328	0.5506	0.5577	0.5630	0.6021	0.8223
Feature-3	0.3201	0.5435	0.5559	0.5772	0.5843	0.6003	0.6518	0.8632
VideoFeat	0.2090	0.3161	0.3179	0.3214	0.3374	0.3658	0.4049	0.5488
Ensembe 1-2-3	0.3474	0.5541	0.5541	0.5612	0.5719	0.5879	0.6447	0.8206
Ensembe 1-2-3 + TrackInfo	0.3680	0.5541	0.5541	0.5612	0.5630	0.5719	0.6074	0.7388
Ensembe 1-2-3 + VideoFeat	0.3593	0.5310	0.5346	0.5541	0.5683	0.5825	0.6447	0.8170
Ensembe 1-2-3 + VideoFeat + TrackInfo	0.3772	0.5310	0.5346	0.5435	0.5523	0.5612	0.6039	0.7548
Ensembe 1-2-3 + VideoFeat + Background	0.4470	0.6447	0.6554	0.6660	0.6802	0.6944	0.7655	0.8969
Ensembe 1-2-3 + VideoFeat + Orientation	0.3984	0.5825	0.5914	0.6056	0.6145	0.6287	0.6909	0.8490
Ensembe 1-2-3 + VideoFeat + Orientation + Background	0.4671	0.6660	0.6714	0.6802	0.6909	0.7069	0.7655	0.9236
Ensembe 1-2-3 + VideoFeat + Camera	0.4611	0.6625	0.6731	0.6891	0.7015	0.7087	0.7602	0.9147
Ensembe 1-2-3 + VideoFeat + Orientation + Camera	0.4806	0.6802	0.6980	0.7033	0.7140	0.7264	0.8046	0.9253
Ensembe 1-2-3 + VideoFeat + Background + Camera	0.4871	0.6856	0.6944	0.7033	0.7122	0.7211	**0.8099**	0.9342
Ensembe 1-2-3 + VideoFeat + Camera + Orientation + Background	0.4956	0.6927	**0.7033**	**0.7104**	**0.7282**	**0.7388**	0.8081	**0.9449**
Ensembe 1-2-3 + VideoFeat + Camera + Orientation + Background + TrackInfo	**0.5108**	**0.6927**	0.6962	0.6980	0.7069	0.7087	0.7495	0.8650

Table 1. Table of results obtained with the proposed evaluation environment. Grey rows are related to the contributions of this work while white rows are part of the baseline [14]. Bold refers to the best performance per metric.

5. Conclusions

This paper proposes a vehicle ReID system based on our previous year proposal. The baseline consists of a feature ensembling method that combines image-based and video-based features. Image-based features are focused on the appearance of the vehicles while video-based features combine appearance with structure information. Then, it applies query expansion and temporal pooling followed by a post-processing step which includes a re-ranking. Among the new proposals, we include a synthetic dataset [23] in the training set to improve the training by increasing the number of labelled images. Moreover, a penalty is applied on the gallery images that are taken with the same camera as the query image. Besides, background and orientation distance matrices are generated and applied as penalty to avoid the possible bias towards these characteristics. It has been demonstrated that the combination of all these proposals outputs the best result compared with different combinations of the individual blocks. As a result, this proposal has achieved the 16^{th} position in the challenge out of a total of 30 participants. As future work, it is suggested to make use of the available labels that are not used, such as colour or type, to continue solving several challenges. The use of other feature extractors and different deep learning techniques could also be beneficial to the system. Finally, we also think that an improvement of the synthetic images could lead to better training and, then, to better results.

	Score
Baseline	0.3240
Baseline + Camera	0.4002
Baseline + Orientation	0.3459
Baseline + Background	0.3653
Baseline + Camera + + Background	0.4033
Baseline + Camera + + Orientation	0.4099
Baseline + Orientation + + Background	0.3868
Baseline + Orientation + + Background + Camera	0.4172
Baseline + Orientation + + Background + Camera + + TrackInfo	**0.4900**

Table 2. Table of results provided by the challenge server [15].

Ranking	Team ID	Score
1	47	0.7445
2	9	0.7151
3	7	0.6650
16	**54**	**0.4900**
30	163	0.1121

Table 3. Final ranking of the challenge [15] (City-Scale Multi-Camera Vehicle Re-Identification). Bold indicates the final result of our proposal.

Acknowledgements

This work was partially supported by the Spanish Government under project TEC2017-88169-R (MobiNetVideo).

References

[1] Junaid Ahmed Ansari, Sarthak Sharma, Anshuman Majumdar, J Krishna Murthy, and K Madhava Krishna. The earth ain't flat: Monocular reconstruction of vehicles on steep and graded roads from a moving camera. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8404–8410, 2018.

[2] Song Bai and Xiang Bai. Sparse contextual activation for efficient visual re-ranking. *IEEE Transactions on Image Processing*, 25(3):1056–1069, 2016.

[3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[4] T. D'Orazio and G. Cicirelli. People re-identification and tracking from multiple cameras: A review. In *IEEE International Conference on Image Processing*, pages 1601–1604, 2012.

[5] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, volume 96, pages 226–231, 1996.

[6] Jiyang Gao and Ramakant Nevatia. Revisiting temporal modeling for video-based person reid. *ArXiv*, abs/1805.02104, 2018.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[8] Shuting He, Hao Luo, Weihua Chen, Miao Zhang, Yuqi Zhang, Fan Wang, Hao Li, and Wei Jiang. Multi-domain learning and identity mining for vehicle re-identification. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 582–583, 2020.

[9] Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*, 2017.

[10] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *IEEE conference on Computer Vision and Pattern Recognition*, pages 4700–4708, 2017.

[11] Tsung-Wei Huang, Jiarui Cai, Hao Yang, Hung-Min Hsu, and Jenq-Neng Hwang. Multi-view vehicle re-identification using temporal attention model and metadata re-ranking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 434–442, 2019.

[12] Kai Lv, Heming Du, Yunzhong Hou, Weijian Deng, Hao Sheng, Jianbin Jiao, and Liang Zheng. Vehicle re-identification with location and time stamps. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2019.

[13] Yi Yu Zheng Wang Qingming Leng Chunxia Xiao Jun Chen Mang Ye, Chaeo Liang and Ruimin Hu. Person reidentification via ranking aggregation of similarity pulling and dissimilarity pushing. In *IEEE Transactions on Multimedia*, pages 2553–2566, 2016.

[14] Paula Moral, Alvaro Garcia-Martin, and Jose M. Martinez. Vehicle re-identification in multi-camera scenarios based on ensembling deep learning features. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020.

[15] Milind Naphade, Shuo Wang, David C. Anastasiu, Zheng Tang, Ming-Ching Chang, Xiaodong Yang, Liang Zheng, Anuj Sharma, Rama Chellappa, and Pranamesh Chakraborty. The 4th ai city challenge. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, page 2665–2674, June 2020.

[16] M Saquib Sarfraz, Arne Schumann, Andreas Eberle, and Rainer Stiefelhagen. A pose-sensitive embedding for person re-identification with expanded cross neighborhood re-ranking. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 420–429, 2018.

[17] Xiaohui Shen, Zhe Lin, Jonathan Brandt, S. Avidan, and Ying Wu. Object retrieval and localization with spatially-constrained similarity measure and k-nn re-ranking. In *IEEE Conference on Computer Vision and Pattern Recognition(CVPR) Workshops*, pages 3013–3020, 2012.

[18] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception ar-

chitecture for computer vision. In *IEEE conference on Computer Vision and Pattern Recognition*, pages 2818–2826, 2016.

[19] Zheng Tang, Milind Naphade, Ming-Yu Liu, Xiaodong Yang, Stan Birchfield, Shuo Wang, Ratnesh Kumar, David C. Anastasiu, and Jenq-Neng 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) Workshops*, pages 8797–8806, 2019.

[20] Shaogang Gong Wei-Shi Zheng and Tao Xiang. Group association: Assisting re-identification by visual context. In *Person Re-Identification*, pages 183–201. Springer, 2014.

[21] Jianping Shi Xingang Pan, Ping Luo and Xiaoou Tang. Two at once: Enhancing learning and generalization capacities via ibn-net. In *ECCV*, 2018.

[22] Shuangjie Xu, Yu Cheng, Kang Gu, Yang Yang, Shiyu Chang, and Pan Zhou. Jointly attentive spatial-temporal pooling networks for video-based person re-identification. In *IEEE International Conference on Computer Vision*, 2017.

[23] Yue Yao, Liang Zheng, Xiaodong Yang, Milind Naphade, and Tom Gedeon. Simulating content consistent vehicle datasets with attribute descent. arXiv:1912.08855, 2019.

[24] Jinjie You, Ancong Wu, Xiang Li, and Wei-Shi Zheng. Top-push video-based person re-identification. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1345–1353, 2016.

[25] Liang Zheng, Liyue Shen, Lu Tian, Shengjin Wang, Jingdong Wang, and Qi Tian. Scalable person re-identification: A benchmark. In *IEEE international conference on Computer Vision*, pages 1116–1124, 2015.

[26] Zhendong Zheng, Minyue Jiang, Zhigang Wang, Jian Wang, Zechen Bai, Xuanmeng Zhang, Xin Yu, Xiao Tan, Yi Yang, Shilei Wen, and Errui Ding. Going beyond real data: A robustvisual representation for vehicle re-identification. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR) Workshops*, pages 2550–2558, 2020.

[27] Zhedong Zheng, Tao Ruan, Yunchao Wei, Yi Yang, and Tao Mei. Vehiclenet: Learning robust visual representation for vehicle re-identification. In *CVPR Workshops*, pages 1–4, 2019.

[28] Zhedong Zheng, Xiaodong Yang, Zhiding Yu, Liang Zheng, Yi Yang, and Jan Kautz. Joint discriminative and generative learning for person re-identification. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2138–2147, 2019.

[29] Zhun Zhong, Liang Zheng, Donglin Cao, and Shaozi Li. Re-ranking person re-identification with k-reciprocal encoding. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1318–1327, 2017.

[30] Xiangyu Zhu, Zhenbo Luo, Pei Fu, and Xiang Ji. Voc-reid: Vehicle re-identification based on vehicle-orientation-camera. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 602–603, 2020.