Vehicle Re-Identification: Pushing the limits of re-identification

Abner Ayala-Acevedo, Akash Devgun, Sadri Zahir, Sid Askary
Huawei Technologies - R&D
2330 Central Expy, Santa Clara, CA 95050
{ayala.acevedo, akashdevgun, sadrizahir, sid.askary} @gmail.com

Abstract

In this paper, we present a series of techniques which help push the limits of vehicle re-identification. First, we establish a strong baseline by using one of the best person re-identification models and applying them to vehicle re-identification. Secondly, we show improvements in four key components of re-identification: 1) detection, 2) tracking, 3) model, 4) loss function. Finally, our improvements lead to the state-of-the-art in the vehicle re-identification dataset VeRi-776, with 85.20 mean Average Precision (mAP) and 96.60% Rank-1 accuracy. This represents a +17.65 mAP and +6.37 Rank-1 improvement over the literature.

1. Introduction

The field of re-identification (ReID) has grown in popularity in the last couple of years. The Conference on Computer Vision and Pattern Recognition (CVPR) has accepted 26 publications related to ReID in 2019 alone. Vehicle re-identification can help understand, monitor, and reduce traffic flow patterns [1]. However, to this day it is still considered a challenging task in computer vision; mainly due to different camera angles, occlusion, different weather conditions and the difference between vehicles viewpoints can be significantly greater than the difference between vehicles of the same color, make, model [2]. Hence, vehicle ids have high intra-class variance and low inter-class variance.

ReID is one of the main components of a bigger problem, Multi-Target Multi-Camera (MTMC) tracking. MTMC can be divided into Multi-Target Single-Camera (MTSC) detection and tracking, follow by re-identification to associate identities changing between cameras. By combining these we can obtain an end-to-end solution for MTMC tracking. However, for the remainder of this publication, we will focus on the task of image-to-video ReID, and assume our input is the output of an MTSC system.

Person ReID can be addressed using face recognition algorithms. Similarly, vehicle re-identification can benefit from license plate recognition since a license plate is a unique identifier for a given vehicle. However, these can be considered personally identifiable data and can lead to a breach of privacy such as the driver's privacy protection act [3]. Therefore, our approach focus on full body vehicle re-identification without the use of license plate information.

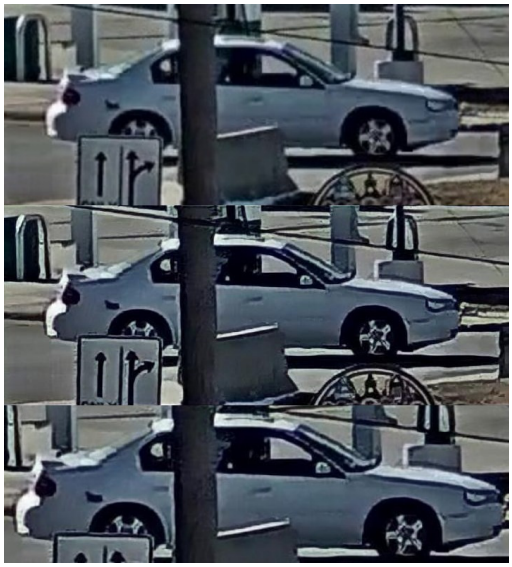

Figure 1. Image Pre-Processing: original image (top), super-resolution (middle) and cropped (bottom).

In order to push the limits of re-identification, we take an exploration and application approach. We explore the possibility of applying several techniques which have proven successful in other areas of computer vision and applied them to vehicle re-identification. Many tips and tricks in the field of ReID or related were tested before our final contributions. Below are our main findings and our key contributions:

- Show the important of pre-processing techniques such as super-resolution and a second pass vehicle detection and its effects on ReID.
- Introduce person re-identification networks such as

Multi Granularity Network (MGN) to vehicle re-identification.

- Enhanced network design and loss function with the addition of two new layers.
- Use tracklets information for a more robust query to test associations without the need for re-ranking.

2. Methodology

An in-depth description of our approach is described below. It is divided into pre-processing, modeling and post-processing features associations.

2.1. Vehicle Detection (Pre)

In the field of computer vision pre-processing techniques are still to this day an important step in object detection and object recognition. For example, in face recognition pre-processing of face detection and face alignment are crucial components of the face recognition pipeline. Most state-of-the-art face algorithms, such as ArcFace, CosFace, and SphereFace *et al*. [4, 5, 6], use Multi-Task Cascaded Convolutional Networks (MTCNN) [7], or equivalent networks, as their pre-processing step. Most image-based ReID datasets, such as Market-1501, VeRi-776 [8], and CityFlow-ReID [1], already loosely crop the original images into the desired targets (i.e. person or vehicle). This step, whether is manual or automated, can be considered as the first pass of detection during the MTSC stage. However, based on initial observations, we notice CityFlow-ReID dataset, described in Section 3.1, introduced a lot of background artifacts which made the rankings biased to images with a similar background. Hence, to reduce background noise and to allow the network to focus on the vehicle of interest, we propose to do a second pass of vehicle detection to tightly crop the vehicle of interest even after the initial bounding boxes. Secondly, as mentioned in *et al*. [9, 10], super-resolution (SR) has been used effectively to improve the mean average precision (mAP) scores of object detection. For this reason, before applying vehicle detection, we apply super-resolution on every image to ensure we don't introduce an extra source of error in our pipeline.

To summarize, given an input image we 1) apply super-resolution with the formula $sr\_image = min(2x, 4x, 1024)$, where x represents a multiplier of the original image size; 2) get vehicle object detection from the SR image. For super-resolution, a U-Net network with resnet34 backbone and a Feature Loss was used. The AI City Challenge 2019 Track 2 train data, called CityFlow-ReID, was used to train the network. Our SR implementation was based on the excellent FastAI notebook [1].

[1] SR: https://github.com/fastai/course-v3/blob/master/nbs/dl1/lesson7-superres.ipynb

Algorithm 1: AdaptiveConcatPool2d

```
def init(self, sz):
  self.ap = nn.AdaptiveAvgPool2d((sz, sz))
  self.mp = nn.AdaptiveMaxPool2d((sz, sz))

  def forward(self, x):
  return torch.cat([self.mp(x), self.ap(x)], 1)
```

Algorithm 2: Cosine

```
def forward(self, x):
  return F.linear(F.norm(x), F.norm(self.weights))
```

For vehicle detection, a recent state-of-the-art detection network called CenterNet was used, mainly for its simplicity, speed and mAP scores [11]. Figure #1, shows the steps from the original image, to super-resolution, to the cropped image after vehicle detection. As it can be seen it aims to produce tighter bounding boxes that allow the network to focus more on the vehicle and less on the background.

2.2. ReID Model & Loss (CMGN)

The core of this paper revolves around a strong vehicle re-identification baseline. To do so we explored both the pedestrian and vehicle re-identification literature; and Multi Granularity network (MGN) method *et al*. [12] was among the strongest ReID models, and generic enough that can be applied to any ReID task. MGN uses a ResNet50 [13] as the base model to learn both global features and the local features based on part-based horizontal strides. To the best of our knowledge, this is the first publication which applies MGN to vehicle re-identification.

Besides a strong baseline, two key modifications are made to the MGN model to improve the baseline implementation. First, we improved the reduction block also known as costume head. We first replaced the pooling layer $MaxPool2d$ with an $AdaptiveConcatPool2d$, see Algorithm #1 for pseudocode implementation. This pooling layer is a concatenation of the average and max pooling layers and was first implemented on fastai deep learning framework [14]. Besides changing our pooling layers we also added a BN layer both before and after each 2D convolution, and the weights are share for each of the eight features. Hence one costume head with shared weights produces 8x256 features, as opposed 8 reduction blocks in the original MGN implementation.

Secondly, inspired by its success in the task of face recognition, we used an angular margin layer, similar to ArcFace, CosFace, and SphereFace, which we denote as $Cosine$. This layer will replace all eight linear classification layers on MGN. Contrary to these previous implementations, [4, 5, 6], we do not use any margin hyper-

parameters in order to reduce complexity. Hence, we mainly focus on transforming the feature space by applying the dot product of the L2 norm of the features and the L2 norm of the weights, see Algorithm #2 for a pseudo-implementation. As opposed to the literature our loss function is now a combination of Triplet Loss and Cosine CrossEntropy Loss, which lead to faster convergence (i.e. fewer epochs) and better results; see section 3.3. We call our solution CMGN short for Cosine-MGN.

2.3. Tracklets Association (Track)

Most publications in the field of ReID [12, 15, 16], rely on re-ranking methods such as Zhong2017 k-reciprocal *et al*. [17] and Sarfraz2018 expanded cross neighbors *et al*. [18] to boost post-processing results. The idea is to use temporal information, query-to-query, and/or test-to-test correlation in order to take the query and/or test distribution into consideration when ranking the results. Although these techniques add value in competitions and/or image retrieval applications, they have little value in real-time MTMC and ReID applications. However, both datasets discuss in this publication contain tracklet information which is typical in Video ReID datasets. These tracklets (i.e. a sequence of images from the same camera) are obtained from the output of an MTSC system after successful tracking of the object in the same camera. Tracklets provides us with more robust features that will take into consideration the distribution of the test data. For the datasets discussed in section 3.1, we only have tracklet information for the train and test split but not for the query split. Hence, we do a query to test-tracklets cosine distance as our ranking metric. Each test tracklet feature is simply the mean of all the image-features of a given tracklet.

3. Experimental Results

3.1. Datasets

Veri-776 [19, 8] is one of the main datasets in vehicle re-identification to this date and consist of 776 identities for training and 200 identities for testing. The dataset was collected from 20 non-overlapping cameras of video traffic in China, obtaining a variety of camera angles vehicle viewpoints, illuminations, and occlusions. Every identity appears in 2-18 cameras. The dataset contains 37,778 training images, 1,678 query images, and 11,579 test images respectively.

CityFlow-ReID [1] is the proposed vehicle image-reidentification dataset for this competition. It is a subset of a full Multi-Target Multi-Camera (MTMC) vehicle tracking dataset called CityFlow collected from different locations in the USA. It contains 333 identities for training from 35 cameras and 333 identities for testing from 5 different cameras that are not used in the training dataset. Each vehicle identity has an average of 84.5 images and the vehicles are seen on 4.5 different cameras on average. The dataset contains 36,935 training images, 1,052 query images, and 18,290 test images, with both train and test tracklets information also being provided.

3.2. Experimental Setup

All experiments where run using Pytorch [20] and fastai [14] library on Ubuntu 16.04 and CUDA 10. A PK batch sampler strategy was used, where P=8 identities were sampled per batch and K=4 images per identity were sampled in order to create an online triplet loss with positive, negatives and anchor samples. Each of the 8 features vectors had 256 for a final predictor of 2048 output features per image. Adam optimizer with amsgrad was used and a learning rate=2e-4 and weight decay of 5e-4 for a total of 750 epochs. Remember each epoch is not an entire pass over the data but simply a pass of 4 samples for every id. Hence, depending on the dataset this is proportional to around 30 passes to the entire dataset.

For data augmentation we used the standard re-sizing to (224, 224, 3) images using ImageNet [21] pretrained weights for the base model of MGN, along with random horizontal flip and random erasing with .5 probability. For the loss function, a margin of 1.2 was used for the Triplet Loss and a margin of 0 was used for cosine. A one-to-one weight relationship between cosine cross-entropy loss and triplet loss was used.

3.3. Experiments

Many experiments were explored during this competition, for brevity we will focus on those that were fruitful. Table #1, shows three of the best results found in the literature on the VeRi-776 dataset and compares it with our contributions. The top block represents those 3 state-of-the-art methods in vehicle re-identification and compares it with the bottom block; which shows our contributions and their effects on the VeRi-776 dataset.

Baseline represents the usage of MGN on vehicle ReID without any additional modifications and shows the biggest incremental gain from previous work at 79.56 mAP (+12.01 mAP). It shows that MGN provides a strong baseline for both person and vehicle re-identification. Secondly, on the bottom block, we provide results for our modified Cosine-MGN we denoted as CMGN which lead to 79.70 mAP. This improvement may not seem significant but as can be seen in all the experiments CMGN always outperformed the MGN version. Additionally at epoch 50/750 CMGN had 61.41 mAP while MGN had 53.54 mAP. This shows CMGN ability to converge faster and has an easier time training the model.

Our second contribution focuses on the use of super-resolution and vehicle detection to tightly crop the images.

Methods	mAP	Rank-1
RAM [22]	61.5	88.6
QD DLF [23]	61.83	88.5
MoV1+BS [1, 24]	67.55	90.23
MGN (Baseline)	79.56	95.65
CMGN	79.70	95.83
MGN+Pre	81.69	96.42
CMGN+Pre	82.14	96.25
MGN+Track	82.88	96.07
CMGN+Track	83.27	96.31
MGN+Pre+Track	84.54	96.25
CMGN+Pre+Track (Ours)	**85.20**	**96.60**

Table 1. VeRi-776 results based on mAP and Rank-1. Top publications can be found in top block and lower block shows our contributions.

We refer to this as Pre-processing (Pre) in Table #1. As it can be seen we can get around 2-3% mAP increase by providing crop images which focus more on the foreground. Thirdly, we show the effect of using the tracklets information to provide a better re-ranking on the results in a real-time manner; we refer to this as Tracklets (Track) in Table #1. As discuss above tracklets are used to get the mean of all the features in a given tracklet, in order to calculate the distance between each query feature and each test tracklet mean feature; as oppose to query-to-test feature to feature distance. This allows us to simplify the problem since we already know this entire sequence of image belong to the same ID (i.e tracking). From Table #1, we can see this leads to a +3.57 mAP gain in addition to the CMGN results. Finally, we integrate all three main contributions of CMGN + Pre + Track which lead to a final mAP of 85.20 mAP and 96.60 Rank-1 accuracy. This represents a +17.65 mAP gain from the literature.

For the CityFlow-ReID competition, we finished in 19th place, with a 46.31 mAP, by using our final version of CMGN + Pre + Track. Figures #2-3 show a hard example with perfect retrieval and a simpler example with no correct retrievals, respectively. This can serve as a small qualitative sample of the current state of vehicle re-identification and its challenges. In Table #2 we compare our results with the leader-board. As it can be seen we got a +14.31 mAP from the baseline results obtain in [1]. This represents a similar gain from the one obtain on the VeRi-776 dataset, which shows the ability of our solution to perform well on other datasets. However, as it can be seen our results are significantly lower than the 1st place. Besides the obvious reason of a more challenging dataset, the main reason for the drop in mAP is that the test data from CityFlow-ReID is from a different set of cameras, and quite possibly a different location than those from the training dataset. For example, in VeRi-776 there are 20 cameras but all cameras

Methods	mAP
Baseline [1]	32.00
1st Place	**85.54**
2nd Place	79.17
3rd Place	75.89
Ours	46.31

Table 2. CityFlow-ReID results, based on mAP metric.

are known to both the train and the test dataset. The same observation can be seen in pedestrian ReID where Market-1501 MGN obtains high results of 86 mAP when both the train and test are from the same set of cameras and same distribution but when trained on Market-1501 and tested on a different dataset it's performance drops to 40-50 mAP at best. The CitiFlow-ReID uses 35 cameras for training and 5 cameras for testing. This can be considered to some degree as training on a different dataset than the test environment; because each camera provides a different background, camera angle and color filter.

4. Conclusion

We observed significant improvements of +17.65 mAP on the VeRi-776 dataset which was used as our validation dataset for all experiments. Most of the improvements came from pre-processing and the post-processing techniques discussed above. It shows the importance of focusing on the object of interest and reducing background noise in complex tasks such as ReID. Even though super-resolution and the second phase of vehicle detection may be computationally expensive our proposed solution can be implemented in near real-time solutions with strong retrieval and ReID results without the need for re-ranking.

Although we achieve state-of-the-art results on VeRi-776 datasets we ended up in 19th place on the CityFlow-ReID challenge. It is important to notice that our solution can work in the real world and do not rely on ensemble solutions or expensive data annotations. Our solution only uses ID annotation and tracklet association. Finally, we observe that our solution over-fits to the cameras it is trained on. Leading to great results when having access to all the cameras, but not generalizing well to newer cameras. This work could benefit from style transfer methods in order to emulate the style of the test cameras on the training dataset or a much bigger number of cameras for the features learned to become more camera invariant. We aim to learn from the leader-board and keep pushing the limit of vehicle re-identification.

References

[1] Zheng Tang, Milind Naphade, Ming-Yu Liu, Xiaodong Yang, Stan Birchfield, Shuo Wang, Ratnesh Kumar, David

Anastasiu, and Jenq-Neng Hwang. Cityflow: A city-scale benchmark for multi-target multi-camera vehicle tracking and re-identification, 2019.

[2] Yan Em, Feng Gag, Yihang Lou, Shiqi Wang, Tiejun Huang, and Ling-Yu Duan. Incorporating intra-class variance to fine-grained visual recognition. *2017 IEEE International Conference on Multimedia and Expo (ICME)*, Jul 2017.

[3] The drivers privacy protection act (dppa) and the privacy of your state motor vehicle record, 2017.

[4] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. Cosface: Large margin cosine loss for deep face recognition. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun 2018.

[5] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition, 2018.

[6] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. Sphereface: Deep hypersphere embedding for face recognition. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul 2017.

[7] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10):14991503, Oct 2016.

[8] Xinchen Liu, Wu Liu, Tao Mei, and Huadong Ma. Provid: Progressive and multimodal vehicle reidentification for large-scale urban surveillance. *IEEE Transactions on Multimedia*, PP, 09 2017.

[9] Muhammad Haris, Greg Shakhnarovich, and Norimichi Ukita. Task-driven super resolution: Object detection in low-resolution images, 2018.

[10] Jacob Shermeyer and Adam Van Etten. The effects of super-resolution on object detection performance in satellite imagery, 2018.

[11] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Qi Tian. Centernet: Keypoint triplets for object detection, 2019.

[12] Guanshuo Wang, Yufeng Yuan, Xiong Chen, Jiwei Li, and Xi Zhou. Learning discriminative features with multiple granularities for person re-identification. *2018 ACM Multimedia Conference on Multimedia Conference - MM 18*, 2018.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2016.

[14] Jeremy Howard et al. fastai. `https://github.com/fastai/fastai`, 2018.

[15] Yifan Sun, Liang Zheng, Yi Yang, Qi Tian, and Shengjin Wang. Beyond part models: Person retrieval with refined part pooling (and a strong convolutional baseline). *Lecture Notes in Computer Science*, page 501518, 2018.

[16] Xing Fan, Wei Jiang, Hao Luo, and Mengjuan Fei. Spherereid: Deep hypersphere manifold embedding for person re-identification. *Journal of Visual Communication and Image Representation*, 60:5158, Apr 2019.

[17] Zhun Zhong, Liang Zheng, Donglin Cao, and Shaozi Li. Re-ranking person re-identification with k-reciprocal encoding. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul 2017.

[18] M. Saquib Sarfraz, Arne Schumann, Andreas Eberle, and Rainer Stiefelhagen. A pose-sensitive embedding for person re-identification with expanded cross neighborhood re-ranking. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun 2018.

[19] Xinchen Liu, Wu Liu, Huadong Ma, and Huiyuan Fu. Large-scale vehicle re-identification in urban surveillance videos. *2016 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6, 2016.

[20] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *NIPS-W*, 2017.

[21] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[22] Xiaobin Liu, Shiliang Zhang, Qingming Huang, and Wen Gao. Ram: A region-aware deep model for vehicle re-identification. *2018 IEEE International Conference on Multimedia and Expo (ICME)*, Jul 2018.

[23] Jianqing Zhu, Huanqiang Zeng, Jingchang Huang, Shengcai Liao, Zhen Lei, Canhui Cai, and LiXin Zheng. Vehicle re-identification using quadruple directional deep learning features, 2018.

[24] Ratnesh Kumar, Edwin Weill, Farzin Aghdasi, and Parthsarathy Sriram. Vehicle re-identification: an efficient baseline using triplet embedding, 2019.

Figure 2. Difficult query with bad light far away car and different viewpoint from test results, perfectly identified.

Figure 3. ”Easy” frontal view SUV in query mistaken by a vehicle of a different color, make, model