Comparison performance of AlexNet and residual network of Vehicle re-identification task on VehicleX dataset

Research School of Computer Science,

Australian National University

Chenyang Pan

u6534059@anu.edu.au

Abstract

VehicleX dataset is the state-of-the-art vehicle simulation data set provided by [1] in 2020. This paper performed vehicle re-identification tasks and vehicle type prediction tasks using the data from VehicleX.

In the paper, a special neural network adopted from AlexNet was designed for the tasks performed in VehicleX, which achieved better results compared to the normal neural network structure and residual network. For the vehicle id prediction task, 62% accuracy was achieved on the test data using my own network, which is higher than 35.33% in [1] or 20% using residual network.

Pruning was implemented during the training process [2]. The implementation of pruning had little influence upon the accuracy of the mode;l but it significantly increased training time due to computation of angles between weight vectors.

This paper concludes that the data set is suitable for vehicle ID prediction and pruning is not useful for improving training time in this dataset. However, angle calculation in pruning process would be helpful to determine whether the hidden neurons are redundant and can assist hyperparameter tuning.

1. Introduction

Nowadays, there are many traffic surveillance cameras but most of them are not equipped with number plate identification hardwares. One of the best solutions for vehicle re-identification problems is through the vehicle's number plate. However, the number plate can be removed or modified. Therefore, re-identifying vehicles through its appearance is still worth researching, and will have functionality in criminal tracking and suspicious vehicle tracking. There are numbers of researches focusing on person re-identification [3,4,7,9,10]. However, the vehicle re-identification is much harder than person re-identification; because if there are two cars with same model and same color, even human eyes cannot tell the differences.

The vehicle re-identification problem is identifying same vehicles through a bunch of vehicle images. I use the VehicleX [1] data set without the number plates and try to identify vehicles purely through its appearance. The training neural network has two main layers, convolution layer and fully connected layer. The convolution layer is adopted from AlexNet; and the fully connected layer has two hidden layers and one output layer. I also implement pruning method [2] in order to improve the training and testing speed, and downsize the neural network.

- **Two tasks and goals**

I implemented two tasks to determine the performance of AlexNet on the dataset: ID prediction and type prediction. ID prediction is also a vehicle re-identification task and the goal of this task is to predict the vehicle ID number given a vehicle image. Type prediction is a subtask of vehicle re-identification task. The goal is to predict the vehicle type (sedan, SUV, etc) given an image of a vehicle.

2.Data

The original data are jpg images with three channel RGB values. The size of each image is 256*256 pixels. Each pixel has three values indicating the intensity of Red, Green and Blue. There are 45,438 images for training, 14,936 images

for validation and 15,142 images for testing. It contains 1,362 cars and each car has average 30-35 photo for training and 10-15 photos for testing.

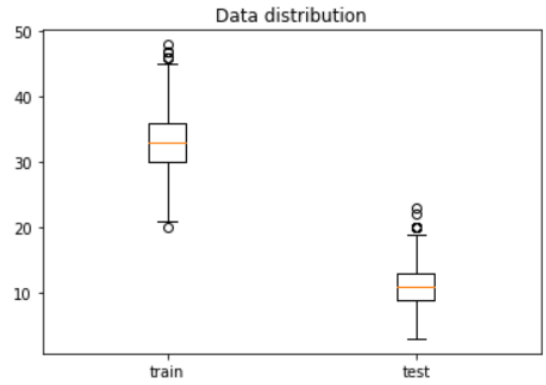

Figure 2.1 Data Distribution for Data Set

As shown in figure 2.1, the data distribution is quite normalized which does not have extremely high or low numbers of data on a single class.

- **Data visualization and preprocessing**

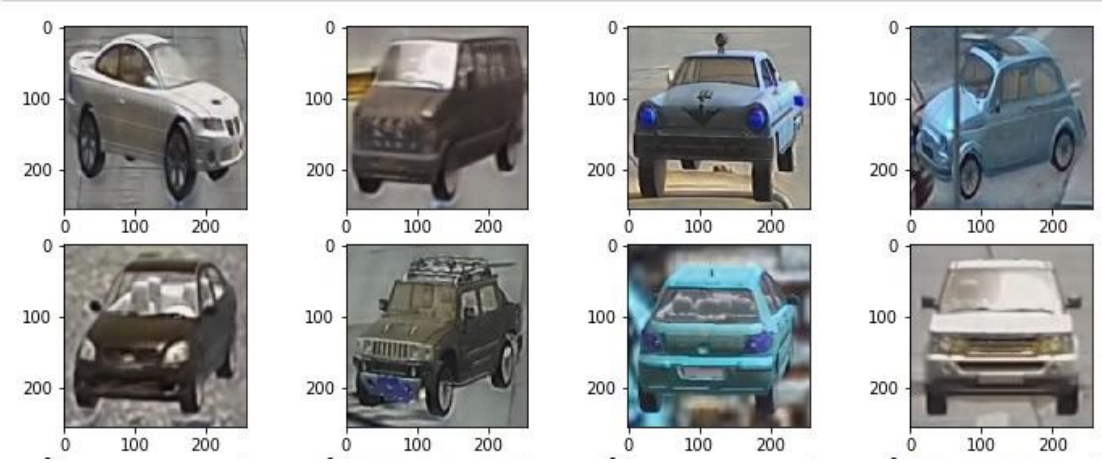

Figure 2.2 Original Data

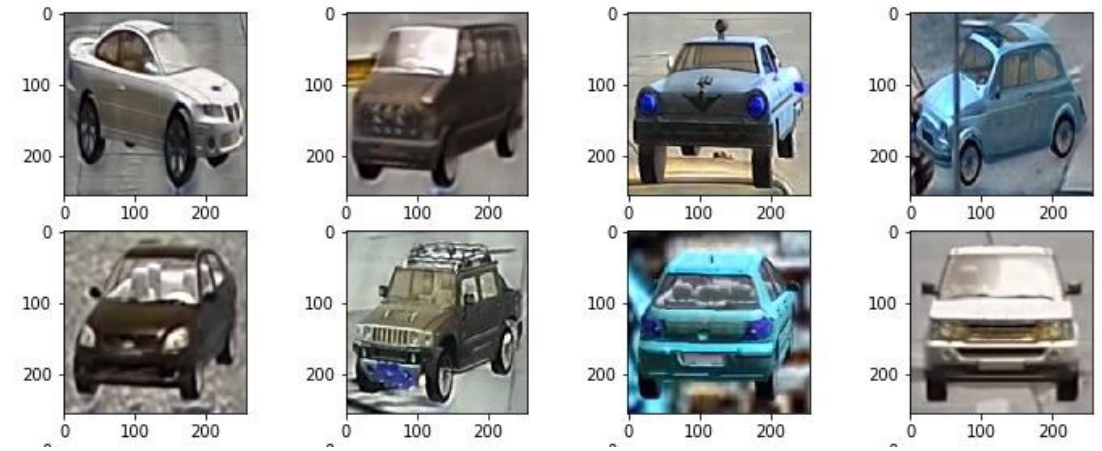

Figure 2.3 Data after Auto Brightness & Contrast Adjustment

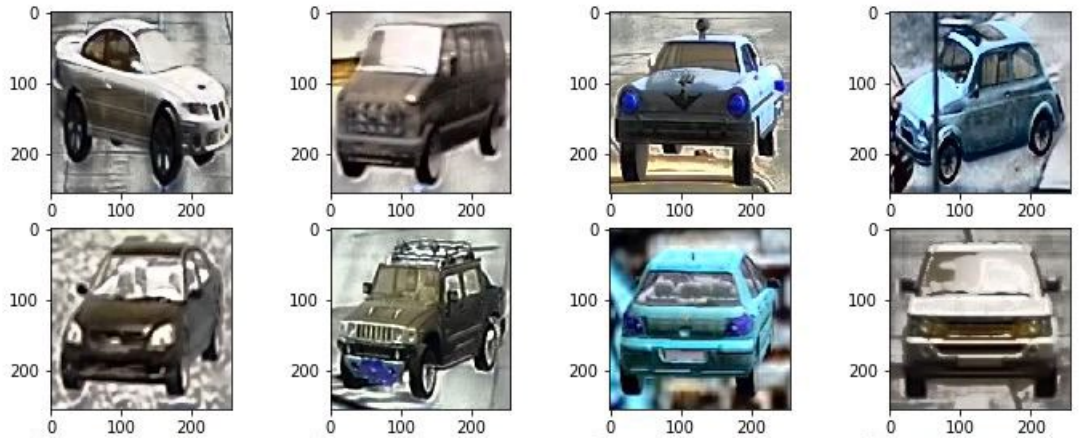

Figure 2.4 Data after Histogram Equalization

Fig2.2 shows eight original pictures from the data set. The original pictures are under different backgrounds, and the background itself does not contain any information and is pure noise. The only information in the images is the vehicles themselves. Therefore, in order to achieve higher accuracy from the training, I tried to perform background removing on these datasets. I used canny edge detector to detect the edges of the vehicles in these images and then perform erosion after dilation. However, the result of performing background removal on the original dataset directly is not very satisfactory. In many images, a large part of the vehicles itself is treated as background. This will cause a large amount of information loss as the vehicle itself is the most valuable information for training and testing. The reason of this is because the background removal algorithm uses fixed hyperparameters for detecting edges, and the edge detection largely depends on the lightness and contrast of the image.

In order to improve the background removing algorithm, I aligned the lightness and contrast of all the images among the dataset. The algorithm of lightness and contrast alignment come from [11]. Besides, I also tried histogram equalization in order to compare the performances of lightness and contrast alignment. Figure 2.3 shows the images after lightness and contrast alignment and figure 2.4 shows the histogram equalization. I performed background removal on both set of images; and the result showed that the images after lightness and contrast alignment perform better than histogram alignment. I think the reason is that histogram equalization only counts each individual image's histogram and perform equalization on it, which means that it does not have an overview over the entire data set. And the automatic lightness and contrast alignment algorithm aligned each individual to a fixed standard, which provides better processed images before the background removing. Besides, I also added mandatory dilation if the background is more than 20% of the image, which prevented the loss of major information.

Fig2.5 below shows the images after background removing, the background removal is not very effective, but it removes most background in most images without losing valuable information. For example, background of second and third images in first row are removed quite decently, while the first image in second row has some parts of the vehicle itself removed as well.

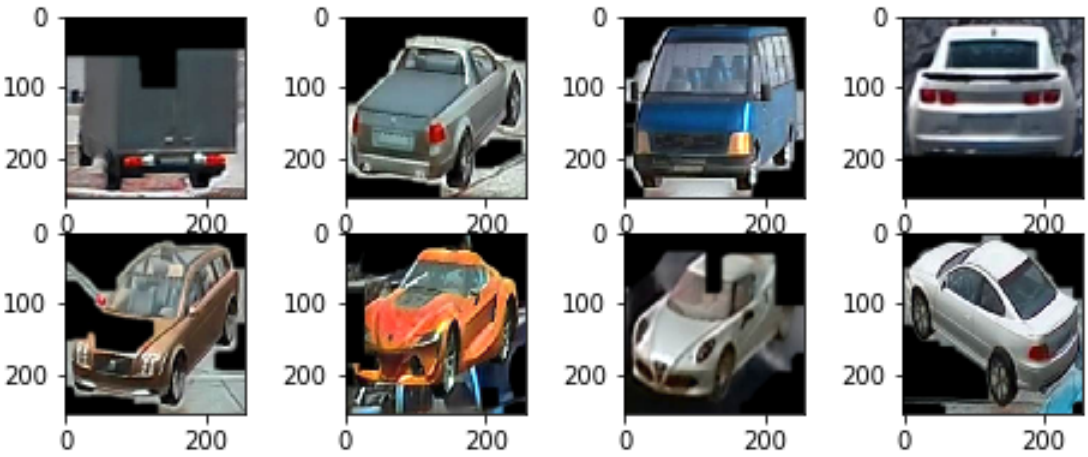

Figure 2.5 Images after Background Removing

3. Methodology

My neural network model can be separated into two main parts, the convolution part and fully connected part. The images will first be processed by the convolution part which is adopted from AlexNet, and the fully connected part is adopted from [1,7] with some modifications. Key hyper parameters of the neural network are as follows:

- Loss function: cross entropy
- Optimizer: adam
- Lerning rate: 0.001
- Num of epochs: dynamic, basically set a large number and stop when overfitting

- **Convolution part**

The convolution part of my model has five 2d convolution layers. Relu activation function is implemented after each convolution layer. Max pooling is implemented after each relu except the fourth convolution layer. The input is vehicle images whose shape is 256*256 with RGB channel. Detailed structure and hyperparameter values are shown in fig3.1.

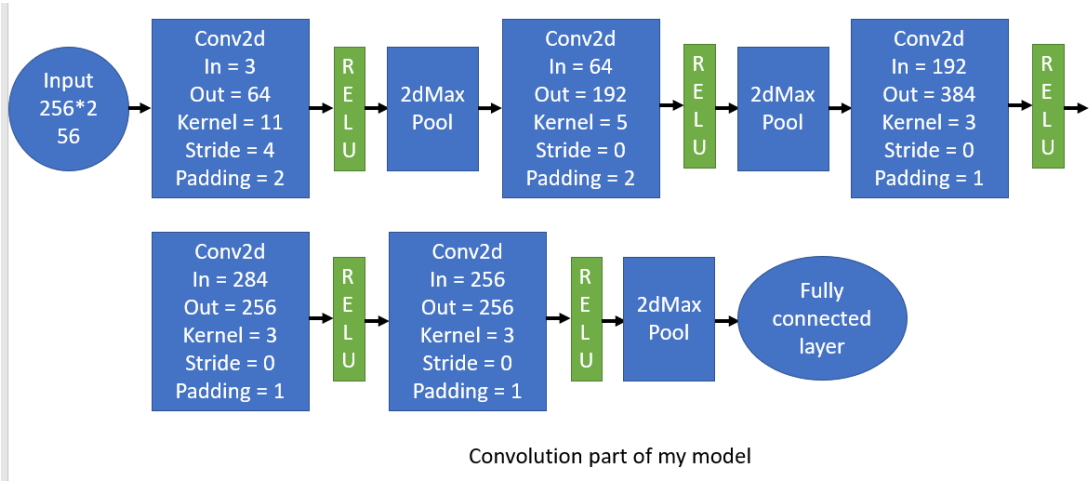

Figure 3.1 AlexNet Convolution Layers (all max pooling are set kernel_size = 3, and stride = 2)

- **Fully connected part**

The fully connected part is adopted from [1,7] and modified to suit the data better. As shown in figure 3.2, batch normalization is added to each hidden layer and RELU activation function is removed after final output layer as described in [7]. I also remove the RELU function for the first hidden layer because I find that the first hidden layer is not very informative and if I apply RELU to the first hidden layer, there might be information loss. I used the normal neural network structure as a baseline and compared their performance. The normal neural network structure is also shown in fig 3.2 for comparison. The input is flattened results from convolution part of my neural network which is a 1-d array. The output of the fully connected layers is the predicted class of vehicle type/ID. The number of hidden neurons is carefully chosen to balance the 12,544 inputs and 1,362 outputs. 3,000 is a reasonable number in between which I think will preserve most information and produce good result.

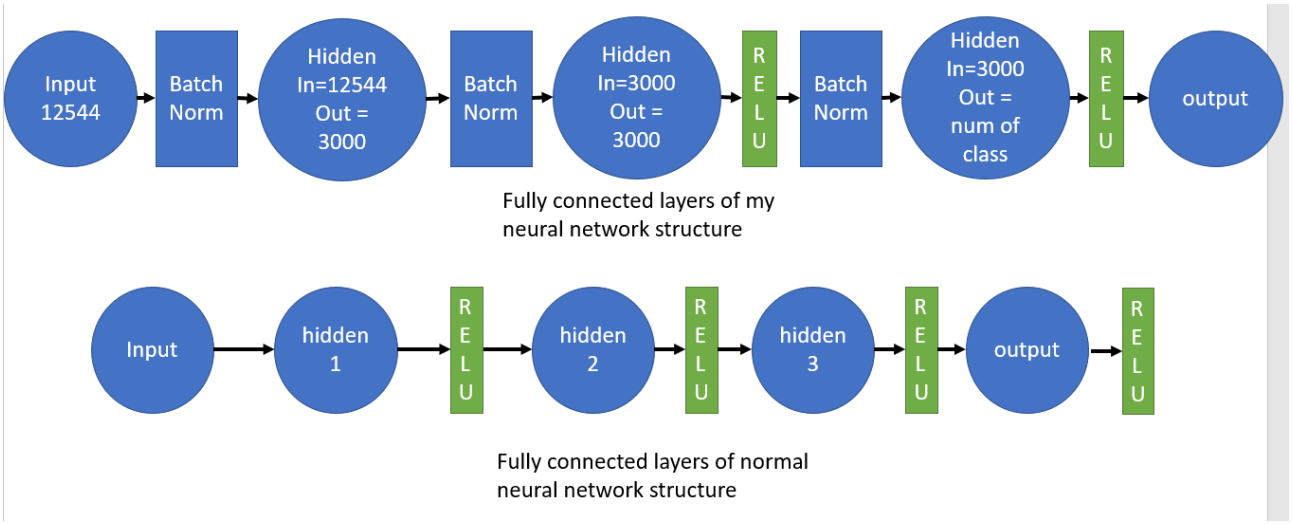

Figure 3.2 Comparison of Fully Connected Layers in my Neural Network & Normal Neural Network

I used features extracted from residual network to compare the performance of my fully connected layers with the normally fully connected layers. The detailed discussion of extracted features and data can be found in my previous paper [12]. With the modified fully connected part, the test accuracy for vehicle id prediction is increased from 0.04% to 19.32%. This indicates that the modified fully connected layer is quite suitable for this specific task.

- **Pruning**

I performed pruning on the fully connected part of my neural network. The reason for pruning is that the number of hidden neurons is a fixed hyperparameter. Adjusting it accurately is hard which needs to guarantee that each neuron has different functionalities. That means there are accurate number of neurons which does not contain redundant neurons but also preserves all functionality of the network.

The basic idea for pruning is to remove neurons with similar functionality or opposite functionality. During backward passing in training, weight of each neuron will be updated according to the gradient and step size. The measurement for functionality between two neurons is the angle of two weight vectors [2]. If the angle between two vectors is 0, then it means that these two vectors are pointing to the exact same direction. The two neurons have exact same functionality and one of them can be removed. For angle 180, the reason is similar, which means that the two neurons have exact opposite functionality and both can be removed. In my implementation, normalization of the weight matrix is performed before angle calculation. If the angle is less than 15 degrees, the neurons are determined as similar functionality; and if the angle is larger than 165 degrees, the neurons are determined as opposite functionality [2]. For neurons with similar functionality, one neuron can be removed and the weights can be added to the other neuron. For neurons with opposite functionality, both neurons should be removed as it is causing conflict. The purpose of neuron pruning is to remove the redundant neurons in order to simplify the neural network so that the training and testing speed can be improved. Besides, neuron pruning in [2] is performed in each training epoch. However, as my neural network has larger size of hidden layers, the calculation between any two neurons' weights was quite slow. Therefore, I performed neuron pruning in every 3 epochs instead of every epoch.

4. Result

As I mentioned before, there are two tasks, one for predicting vehicle type and the other for predicting vehicle ID. The result of my modified AlexNet, residual network, and normal neural network are shown in the below tables (table 4.1, 4.2)

Table 4.1 Vehicle Type Prediction using Different Neural Network Structures

	Alex network Original data	Alex network Processed data	residual network original data	Normal network original data
Train accuracy	71%	81%	66.1%	99%
Validation accuracy	73.9%	84.8%	41.1%	-
Test accuracy	63.48%	75.5%	43%	28.2%
pruning	966	298	19	

Table 4.2 Vehicle ID prediction using Different Neural Network Structures

	Alex network Original data	Alex network Processed data	residual network original data	Normal network original data
Train accuracy	96%	89.2 %	100%	0.02%
Validation accuracy	95.3%	79.4%	17.9%	-
Test accuracy	62.6%	39.7%	18.0%	0.02%

The detailed discussion about the results on residual network compared to normal network can be found in my previous paper [12].

There are several interesting points in the results, detailed discussion of these results can be found in next section:

1. AlexNet performs much better (higher accuracy) than residual network and normal neural network. The ID prediction performed by AlexNet has even better results than that described in [1] which is around 35.33%

2. In AlexNet, the processed data performs better than original data in vehicle type prediction while the original data performs better than processed data in vehicle ID prediction.

3. Focus on the first column in table 4.2, we can find that the train and validation accuracy are higher than test accuracy. Similar results among different data and tasks in AlexNet. But this does not happen on the residual network which has similar validation and test accuracy.

4. Pruning is implemented in both prediction tasks. I did not run pruning on ID prediction because I do not have a powerful computation resource and it adds too much time on training when performing pruning. However, similar situation happens as I described in my previous paper [12]. There were no neurons to be pruned when setting the pruning range lower than 15 degrees and higher than 165 degrees, unless I increase the range to lower than 30 degrees and higher than 150 degrees. This happens in AlexNet as well as residual network even after performing normalization on weight matrix.

5. Discussion

- **AlexNet performs better than residual network**

Residual network and AlexNet have entirely different structures. Residual net adds extra connections between neurons in different layers to solve the vanishing gradient problem while AlexNet avoids this problem in the structure. Residual network is designed for solving the vanishing gradient problem. Compared to the residual network, AlexNet is designed and tuned carefully for image recognition tasks. Therefore, I think the convolution layer of AlexNet is able to extract most valuable information from the images. Besides, under same size of network, AlexNet has more variables for training and less training time, because AlexNet can be split into multiple GPUs during training [13].

- **Processed data performs better in type prediction while worse in ID prediction than original data**

Theoretically, if the background removing algorithm is sophisticated enough to remove all the background of the image, the performance should be better than the original data. However, my background removing method does not generate perfect results. There are some parts of vehicles being treated as background and removed by the algorithm; and there

are also parts of background being treated as vehicle and not removed by the algorithm. The purpose of background removing is to reduce noise in the dataset, but the result is not very satisfactory and may add noise into the data set. For vehicle ID prediction, preprocessing may remove key features and these features matters in the prediction task. However, for vehicle type prediction, I think that the key features may not affect too much on the prediction result and removing background is quite helpful for producing better results although the background removing is not very precise.

- **Train and validation accuracy are higher than test accuracy**

This only happens on the AlexNet. As the train and validation accuracy are similar, this means that the model is not overfitting. I think reasons for the lower test accuracy are that the test data is not generated under the same circumstance as the train/validation data; and there are more vehicles in test dataset blocked by advertisement board or road signs than in train/validation data. This significantly reduced the test accuracy. However, this problem does not happen in the residual network. In residual network, the train accuracy is extremely high, but the validation and test accuracy are similar and extremely low. I think the reason is that the performance of residual network is very disappointing on this dataset and different test datasets may not even contribute to the performance. In other words, under extreme circumstances, the validation and test accuracy are close to zero; then, the result will not indicate whether the test data set and the validation set are under the same standard.

- **Pruning: neurons all have different functionalities.**

As I discussed in my previous paper [12], pruning will not affect the accuracy of the model. I got similar result perform pruning on residual network and on AlexNet. Even after normalization, there were no neurons to be pruned, I had to increase the number of hidden neurons in fully connected layer in order to create some redundant neurons to be pruned. For the pruning technique performed during training, I consider it as less useful on the training process; because it costed extra time computing the angles and there were not many neurons with similar/opposite functionality. I think as long as the number of neurons for each layer are chosen carefully and does not have redundant neurons, the pruning procedure is not needed. I also believe that in order to achieve required accuracy, there is a range for the number of neurons. At the minimum number, the accuracy will be guaranteed; while exceeding the maximum number, pruning is necessary. In other words, I think neuron pruning can be used to determine if there are too many numbers of neurons in each layer and whether it is necessary to decrease the number of neurons. This would be more helpful for hyperparameter tuning rather than increasing training speed.

6. Conclusion & Future Work

AlexNet is a state-of-the-art neural network structure and produces surprisingly good result on the vehicle dataset. The accuracy is better than [1]. The convolution layers of AlexNet can extract valuable features from images and perform good results on image recognition.

Background removing algorithm needs to be improved in order to remove the background accurately. I firmly believe that removing background can significantly increase the accuracy. I also plan to try the method from [6] to crop the vehicle out of an image which can also reduce the noise around the edge of the vehicle images.

Pruning is more suitable to be applied in hyperparameter tuning, which can help determine the number of hidden neurons in fully connected layer. Using pruning in training process will significantly increase computation time because it needs to calculate angles between every vector pairs.

7. Reference

1. Yao,Y., Zheng,L., Yang, X., Naphada, M., Gedeon, T.: Simulating content consistent vehicle datasets with attribute descent. In: ECCV (2020)

2. Gedeon,T.D.: Indicators of hidden neuron functionality: the weight matrix versus neuron behaviour. In: IEEE (1995)

3. Bak,S., Carr,P., Lalonde,J.F.: Domain adaptation through synthesis for unsupervised person re-identification. In: ECCV (2018)

4. Hermans,A., Beyer,L., Leide,B.: In defense of the triplet loss for person re-identification. In: arXiv (2017)

5. Liu,H., Tian,Y., Yang,Y., Pang,L., Huang,T.: Deep relative distance learning: Tell the difference between similar vehicles. In: CVPR (2016)

6. Liu,X., Liu,W., Ma,H., Fu,H.: Large-scale vehicle re-identification in urban surveillance videos. In: ICME (2016)

7. Luo,H., Gu,Y., Liao,X., Lai,S., Jiang,W.: Bag of tricks and a strong baseline for deep person re-identification. In: CVPR Workshops(2019)

8. Tang,Z., Naphade,M., Birchfield,S., Tremblay,J., Hodge,W., Kumar,R., Wang,S., Yang,X.: Pamtri: Pose-aware multi-task learning for vehicle -re-identification using highly randomized synthetic data. In: ICCV (2019)

9. Zheng,L., Bie,Z., Sun,Y., Wang,J., Su,C., Wang,S., Tian,Q.: Mars: A video benchmark for large-scale person re-identification. In: ECCV (2016)

10. Ahmed,E., Jones,M., Marks,T.: An improved deep learning architecture for person re-identification. In: CVPR (2015)

11. nathan: automatic_brightness_and_contrast, (2020)

< https://stackoverflow.com/questions/56905592/automatic-contrast-and-brightness-adjustment-of-a-color-photo-of-a-sheet-of-pape>

12. Pan,C.: Vehicle re-identification using neural network and neuron pruning on VehicleX dataset. (2021)

13. Wei,J,: AlexNet: The Architecture that Challenged CNNs (2019)