Comparative effectiveness of different neural network models in fine-grained vehicle re-identification task

Ruiqi Chen

Australian National University

Abstract. This paper using a deep learning approach to do a vehicle re-identification task on a synthetic data vehicle-X. Different approaches are implemented to get the best performance in a 100 Vehicle re-id classification task. Three different neural network models are implemented. They are a two layer neural network model with the input of 2048 dimension of image features extracted from ResNet, a specific CNN model built and tuned by myself with the input of images and a DenseNet model with the input of images. Compared with the simple neural network. Two deep learning models using the image directly show a significantly improvement in classification ability. Different architectures and hyper parameters are tested with the criterion of the accuracy. The simple neural network model with image features shows around 26% accuracy while my own CNN and DenseNet shows around 69% and 62% accuracy respectively.

Keywords: Deep learning · Neural Network · Synthesis dataset· vehicle Re-identification

1 Introduction

Fine-grained classification has attracted intensive attention recently. However, it is a challenging job since different classes can only be distinguished by subtle distinction parts. Objects rotation, scales or other morphology perspectives in different images increases the difficult [6]. The task of vehicle re-identification is given a vehicle image and predicts which vehicle is in the whole data set [3]. Nerual netwok classifier has been proved to working effectively in classification task. In this paper, Three models are tested and compared. A simple neural network named SimpleNN is implemented based on the feature extracted from ResNet pretrained on the imageNet. A specific CNN model named SimpleCNN is built with the test and tune. And a popular deep learning model DenseNet[1] are also implemented for comparison purpose. The final accuracy of three model are 26%, 70% and 65% respectively in total 100 classes which shows that this neural network can effectively been used to do the vehicle re-identification task in 100 different ids.

2 Data preparation

The data is collected from Vehicle-X dataset [7]. This data-set contains 1362 vehicle classes with totally 45438 images in training set, 14936 in validation set and 15142 in testing set. This project extracted part of them. 3866 samples from training set with 100 vehicle classes are selected to be training set. 1299 samples with the same labels in validation set are selected as the validation set and testing set contains 1340 samples. There are two forms of input in this project. One is the extracted feature and the other is the image with the size of 256*256(RGB channels) The extracted features for all the samples are extracted by the ResNet which is pretrianed on the ImageNet. 2048 dimensions of features are extracted in this process. The image format are the orgin image.

3 Data pre-processing

3.1 Extracted Feature preprocessing

Although the raw data can be directly used as the input of the model. The raw data are all the feature extracted from Resnet range from 0-1. Different normalization and and pre-processing step are still highly recommended according to the different performance on the model. A squared root normalization followed by a $l2$ normalization are applied on the 2048 dimension extracted features [2]. The formulas are shown below.

$$y = \sqrt{x} \tag{1}$$

$$z = y/||y||_2 \tag{2}$$

where x is the raw vector of 2048 dimension vector. After the normalization approach. All the vector in data are range from 0 to 1. Similar normalization method are applied on the validation set and testing set.

3.2 Image preprocessing

The size of each image is 256 * 256 . The range of pixel in each RGB channel is 0-255. Firstly, I convert the range from 0-255 to 0-1 and a normalization method are applied to further convert the pixel in the range of (-1)-(+1). Those two prepossessing steps can help the model converge more quickly. The formula are shown below

$$x = (x - \mu)/\sigma \tag{3}$$

where μ represent the average of the all channel and σ is the standard deviation.

4 Network build and improvement

In this section. Several different trails are applied with the aiming of finding the best accuracy performance on testing set. The accuracy is calculated by the percentage of the samples in testing set are classified into the right vehicle-id.

4.1 Baseline model

There is no baseline in this model. Baseline model is built according to the rule of thumb with the help of neural network [4] . The baseline architecture is a three layer network.This baseline model includes 2048 inputs nodes with the input of extracted features. The first hidden layer contains 512 nodes and the second hidden layer contains 256 hidden nodes. The activation function are all sigmoid activation function. The model use standard back propagation approach with cross-entropy loss as the loss function. The model is trained with 500 epoch times. The best performance for a three layer architecture is achieved as the hyper-parameter setting which is proved by my own test. The accuracy in training set achieve 90%. However, The performance on testing set is only achieved 13.68% which is an unsatisfied result. Although it has achieve above 1 % (randomly guess) accuracy a lot.

4.2 SimpleNN

The best model this paper achieved is a two layer network with 128 hidden nodes. According to the experiment.Tanh activation function and cross entropy loss are selected. Adam optimizer are selected as the optimization function. Early stopping with stopping time evaluated by the validation set with 100 patient time has proved to be the most effective approach for this problem. The whole architecture are shown in Fig. 1

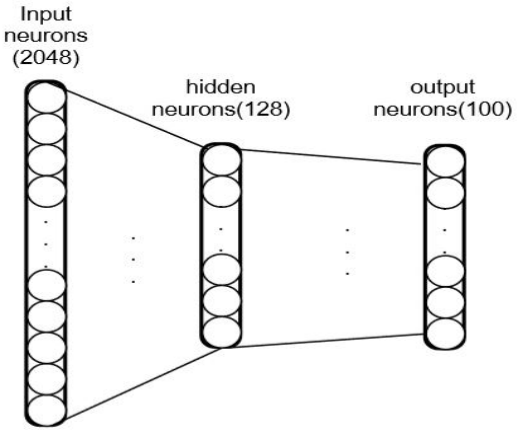

Fig. 1. The network architecture

Final accuracy shows that the accuracy in test set shows 26.38 % in average 10 times running.

4.3 SimpleCNN model

The Convolution Neural Network has been proved to be the effectively model for image classification. This model has 4 layer. All the parameter are tested with the criterion of classification accuracy.

According to the personal test. The architecture are implemented with following parameters

Input	Operation	Output
3@256*256	Convolution layer 16, kernel size 7*7, padding 3, stride 2	16@128*128
16@128*128	Maxpooling 2*2	16@64*64
16@64*64	Convolution layer 32, kernel size 3*3, padding 3, stride 2	32@32*32
32@32*32	Maxpooling 2*2	32@16*16
32@16*16	Convolution layer 64, kernel size 3*3, padding 3, stride 2	64@8*8
64@8*8	Maxpooling 2*2	64@4*4
64@4*4	Convolution layer 128, kernel size 2*2, padding 0, stride 2	128@2*2
128@2*2	Fully connected layer 100	100
100	Fully connected layer 100	100

After each convolution layer, a batch normalization are implemented to overcome over- fitting followed by a ReLU activation function.

The model use stochastic gradient approach with the batch size of 16. Cross entropy are selected as loss function and Adam are chosen to be the optimizer. The model are training with 10 epoches.

The final accuracy on test set shows a 70% accuracy. However, the time cost of the convergence is much higher which shows around 10 minutes running time. The program environment is an Intel(R) Core™ i7-7700HQ with 2.80 GHz, 8.00 GB of RAM, Operating System 64-bit computer using python.

5 Result and discussion

In this project, two different models are built. This part will make a comparison for these two models. A DenseNet-121 and a baseline model are also implemented for comparsion purpose.

The criterion of comparison is accuracy. The table shows the final accuracy of 4 different models. Two of them use the extracted features from ResNet while other two use image with deep learning approaches. The final accuracy are shown in the table below. It can be seen from the result that the SimpleCNN has the highest accuracy among four models. While the DenseNet-121 also shows a satisfied result with 61.41 % accuray. Although SimpleNN and baseline model can not compare with the deep learning approach but they also show a effective classification ability.

Table 1. Classification ability in four different models.

Model Name	Accuracy
Baseline Model	13.68%
SimpleNN	26.38%
SimpleCNN	68.95%
DenseNet-121	61.41%

6 Result and conclusion

This paper using neural network model to distinguish 100 ids. Three different model are implemented with different input format. The best model is SimpleCNN which is built for this question accordingly. In this project, normalization method with a two layer network architecture with early stopping are recommended for extracted feature prepossessing while the image are prepossessed with normalization method which convert each pixel range from -1 to 1 are recommended. A four layer convolution neural network approach named SimpleCNN are proved to be the most effective way to do the vehicle re-id task which is around 69%.

7 Discussion and future work

In this research, for the Simple neural network(Baseline model and SimpleNN model) model. Different parameters are tested. The two fully connected layer model performs better than the deep layer model. And it points out that a simple neural network without too much technique may perform well in certain circumstance. This is also applied in CNN model (SimpleCNN and DenseNet-121). Different layer has been tested. 2-6 convolution layers are tested and it finds that the 4 convolution layers setting is the most suitable one in solving this problem. Besides, It is noticeable that Max pooling approach may not be necessary for all conditions. From my tested, the Max pooling layer in the final convolution layer shows a negative effect. The model with a max pooling layer shows around 5% accuracy lower than the model without max pooling layer. This also applied in the dropout layer. The dropout layer with 0.3 to 0.5 which is implemented in the fully connected layer does not help the model achieve a better performance. It brings some negative effects which decrease the re-id accuray by around 10%.

Several limitation shows that this task can be improved. First of all, this model can only classify 100 different vehicle labels. Due to the fact of limitation of computation units, The training process is a time-costing work. The convergence time of SimpleCNN is around 15 minutes while the DenseNet-121 trains the model almost an hour.

The further improvement should be focused on the classify much more vehicle labels and also increase the accuracy. More techniques should be implemented in further test with the help of the GPU server. I think may be an ensemble

approach can be used to classify the model which combines several different models.

References

1. Iandola, F., Moskewicz, M., Karayev, S., Girshick, R., Darrell, T., Keutzer, K. (2014). Densenet: Implementing efficient convnet descriptor pyramids. arXiv preprint arXiv:1404.1869.
2. Lin, T. Y., RoyChowdhury, A., Maji, S. (2015). Bilinear cnn models for fine-grained visual recognition. In Proceedings of the IEEE international conference on computer vision (pp. 1449-1457).
3. Liu, X., Liu, W., Mei, T., Ma, H. (2016, October). A deep learning-based approach to progressive vehicle re-identification for urban surveillance. In European conference on computer vision (pp. 869-884). Springer, Cham.
4. Milne, L. K., Gedeon, T. D., Skidmore, A. K. (1995). Classifying Dry Sclerophyll Forest from Augmented Satellite Data: Comparing Neural Network, Decision Tree Maximum Likelihood. training, 109(81), 0.
5. Prechelt, L. (1998). Early stopping-but when?. In Neural Networks: Tricks of the trade (pp. 55-69). Springer, Berlin, Heidelberg.
6. Xiao, T., Xu, Y., Yang, K., Zhang, J., Peng, Y., Zhang, Z. (2015). The application of two-level attention models in deep convolutional neural network for fine-grained image classification. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 842-850).
7. Yao, Y., Zheng, L., Yang, X., Naphade, M., Gedeon, T. (2019). Simulating content consistent vehicle datasets with attribute descent. arXiv preprint arXiv:1912.08855.