Optimization of Lightweight Convolutional Neural Networks for Pedestrian Recognition

Kaifu Dai

School of Computer and Information Technology, Hohai University, Nanjing, Jiangsu, 211100, P.R.China

ABSTRACT

Based on the wide application of pedestrian recognition in the fields of smart city and automatic driving, a suitable deep neural network model is selected for the devices that are in embedded systems and with poor computing power. In this paper, by establishing a pedestrian behavior data set, the MobileNet series depth separable convolution structure is optimized and improved. MobileNetV3 is pruned and optimized in the course of the experiment, and suitable hyper parameters are selected. The experimental results show that under the condition of training with Nadam optimizer, when the width multiplier coefficient α is 0.6 and the resolution multiplier coefficient ρ is 0.9, the results of the MobileNetV3 Small model are optimal and the accuracy can reach 94.4%.This model has fewer parameters and computational effort than the traditional neural network model, and is more suitable for deployment on embedded systems and devices with poor computing power.

KEYWORDS

Lightweight; Pedestrian recognition; Deep neural network

DOI: 10.47297/taposatWSP2633-456911.20230402

1 Introduction

With the explosive growth of data and information in today's information age and the rapid development of computer hardware and software technology, it makes deep learning based artificial intelligence technology high-quality development. Lightweight recognition models in intelligent identification, smart cities, human identification, pedestrian detection and other aspects have become a popular research direction in the field of computer. However, due to the wide application of convolutional neural networks, the performance requirements are getting higher and higher, however, the increase in performance is accompanied by the problem of efficiency and hardware requirements, which is struggling in embedded platforms such as cell phones and automatic driving platforms. And pedestrian recognition has algorithms based on image color, texture and other HOG features; Boosting algorithm of traditional machine learning; two-step method of selecting regions before classification and regression in deep learning and YOLO and other algorithms of direct classification and regression.

The efficiency and recognition accuracy of pedestrian detection algorithms are highly demanded in practical scenarios, but there is a problem that the two cannot be taken into account at present. Therefore, designing an efficient and lightweight network structure that can be applied to embedded or mobile devices, and guaranteeing high-precision recognition on this basis, is also a problem that needs to be solved. The goal of the research is to streamline and optimize the network model based on the deep learning approach, so that it can be operated on embedded devices with limited

hardware and have better performance.

2 Model Selection

In the experiments of this thesis, the authors chose to use the lightweight neural network model, the lightweight neural network model compared to the general neural network model has a better application effect in the mobile terminal and edge terminal. The number of parameters and computational complexity is greatly reduced compared with the general neural network model, the number of parameters is about 1/50, which is better adapted to the mobile terminal. At the same time, the energy consumption and carbon emission are much reduced, which alleviates the heat dissipation problem of the processor and ensures the performance of the chip.

Lightweight networks can be obtained through two main efforts, one is to compress common models, i.e., to reduce the number of parameters and computational complexity of a model by optimizing an existing mature neural network through model compression techniques such as knowledge distillation, pruning, etc., including the use of pruning techniques to remove redundant connections in the model, or the transfer of knowledge from a large scale model to a lightweight model through distillation. The second is the use of pre trained models, in which pre-trained models in publicly available datasets are used as a starting point and fine-tuned to adapt them to the scenario of the current application. The author uses the MobileNetV3 [1] model in this article.

3 Preparation of the Study

For the preparation of the dataset, the author used the KITTI dataset provided by the Karlsruhe Institute of Technology (KITTI), Germany, which is an open dataset for visual perception and autonomous driving research. The author chose to use data from the KITTI dataset that has been corrected and undistorted. The author obtained a total of 5238 images, the image size is 1224*370, each image contains at most one pedestrian, and the pedestrian's behavior is divided into two categories, namely walking and standing. The pictures of each type of action are extracted, and 800 pictures of each type of action are obtained, and then they are mirrored and flipped, and 3200 pictures are obtained, together with 800 pictures of blank scenes without pedestrians, to get a total of 4000 pictures in the training dataset, and the example pictures are as follows.

Fig.1 Different action behaviors of pedestrians

Considering the image data in the dataset, the author decided to use Binary Cross entropy as a loss function, which is used to measure the difference between the model output and the real labels in

order to gradually adjust the parameters of the model during the training process, enabling the model to fit the data better. Two output layers are added to the top of the base network, one for the binary classification task of recognizing the presence or absence of pedestrians and the other for the binary classification task of recognizing the pedestrian's state (walking or standing).

The binary cross-entropy loss function is a commonly used machine learning loss function, which can be used to measure the prediction accuracy of the trained model. It measures the predictive accuracy of the model by computing the difference between the true and predicted labels. In order to calculate the binary cross-entropy loss function, we need to first calculate the probabilities of the two labels, and then calculate the cross-entropy loss function: L = -(y*log(p) + (1-y)*log(1-p)). Furthermore, the binary cross-entropy loss function can only handle binary classification problems, and it cannot handle multi-class classification problems. The advantage is that it can measure the predictive accuracy of the model, allow the model to converge faster, and can be updated online without retraining the entire model.

$$BCELoss = -\frac{1}{n}\sum_{i=1}^{n}[y_i \cdot \log p(y_i) + (1 - y_i) \cdot \log(1 - p(y_i))]$$

Fig.2 Calculation formula for binary cross entropy

MobileNetV3 is an efficient deep convolutional neural network architecture proposed by Google, specifically designed for image classification and computer vision tasks on mobile devices and embedded systems. It is the third generation of the MobileNet series. Compared with previous versions, MobileNetV3 has significantly improved performance and computational efficiency, performs better, has higher accuracy and faster inference speed, and contains more lightweight modules. It is a very useful deep learning model, especially suitable for real-time image processing and computer vision tasks on mobile devices and embedded systems. Due to its excellent performance and flexibility, MobileNetV3 has broad application prospects in the fields of image recognition and computer vision.

Deep convolution of 5X5 size is introduced in MobileNetV3 instead of partial 3X3 deep convolution. Squeeze-and-excitation (SE) module and h-swish (HS) activation function are introduced to improve the model accuracy. Compared to the previous MobileNetV2, partial optimization is done at the end by removing the convolutional layers such as 3x3 Dconv, 1x1Conv, etc., but there is no big loss of model accuracy

MobileNetV3, compared with AlexNet and other classic neural network models in the field of computer vision, has a depth-separable convolution to reduce the amount of computation and the number of parameters, the depth-separable convolution actually consists of a standard convolution layer and three point-by-point convolution, which divides the original standard convolution process into two steps. First, the channel-by- channel convolution first decomposes the input feature map into n and then performs the convolution operation on each channel separately, that is, group convolution, to obtain the output results of individual features. Next, the output of the channel-by- channel convolution is used as the input of the point-by-point convolution, for which a 1 X 1 convolution kernel is computed to integrate the resulting output channels. If a 3x3 convolution kernel is used in constructing the network structure, the computational effort for deep separable convolution in MobileNetV3 will be reduced by 8-9 times compared to standard convolution. [2]

4 Experimental Component

(1) Experimental environment

The hardware environment for this experiment was an Intel(R)Core(TM)i7-10750H CPU 2.60GHz processor with 16GB of RAM and an NVIDIA GeForce RTX 2060 graphics card, and the operating system was a Microsoft Windows 64-bit operating system.

In the experiments, the dataset that underwent processing was divided into a training set and a validation set according to a ratio of 4:1.The training was performed using the lightweight neural network MobileNetV3, with the training network parameters set to a learning rate of 0.01,100 iterations, and 50 photographs in each batch.

(2) Experimental results and optimization

The MobileNetV3 network achieved an accuracy of 92.3%, and the loss of the model is very low, between 0.01-0.002, this data favorably indicates that this model has a good fitting effect as well as a high accuracy. After that, I used the MobileNetV3 Small model, this model compared to MobileNetV3, the number of bneck and the number of channels have a certain amount of reduction. small version has 12 bottleneck layers, a standard convolutional layer, two point-by-point convolutional layer, which is now mainly used in the computational resources are limited, the older mobile devices and embedded systems. devices and embedded systems with limited computational resources. I have achieved similar accuracy with MobileNetV3 Small, and I believe that the duplicate depth layers do not have the ability to provide image feature information when dealing with smaller sized and fewer pedestrian images. [3]

In further experiments, in order to be able to improve the model's effectiveness in classifying pedestrian behavior in images under complex environmental disturbances, the author chose to use the Adam and Nadam optimizers to train the MobileNetV3 and MobileNetV3 Small models. It was concluded that both models showed some improvement after optimization with the optimizers, with the MobileNetV3 Small model showing the largest improvement of 3.2% after optimization with the Nadam optimizer. While the MobileNetV3 model was optimized by both optimizers with an improvement of around 1%.

Table.1 Accuracy of Using Different Optimizers

	MobileNetV3	MobileNetV3 Small
None	92.3%	91.4%
Adam	92.9%	93.9%
Nadam	93.6%	94.6%

The author proceeded to control the performance and size of the model through the width multiplier and the resolution multiplier in his experiments. The accuracy of MobileNetV3 was compared to MobileNetV3 Small by adjusting the width multiplier α and resolution multiplier ρ under the condition of using the most effective Nadam optimizer. For the MobileNetV3 model, there is only a 0.6% decrease in accuracy when α is changed from 1.0 to 0.4, whereas ρ has a larger impact on accuracy when it is changed from 1.0 to 0.5, with a decrease in accuracy of about 4%.For the MobileNetV3 Small model, at α of 0.6 and ρ of 0.9, the model has optimal accuracy and higher accuracy than MobileNetV3 without optimizer training.

5 Conclusion

In this paper, we propose to use MobileNet family of models to recognize the pedestrian behavior dataset, and by comparing the accuracy of the MobileNetV3 model with that of the MobileNetV3 Small model, it is concluded that the MobileNetV3 Small model, which has a smaller number of parameters, can be used for recognition when dealing with images of smaller sizes. Then, under the comparison experiment of different optimizers, it can be concluded that the optimization of MobileNetV3 Small model using Nadam optimizer has the best improvement effect. Then, under the condition of training with Nadam optimizer, the MobileNetV3 Small model achieves the best result when the width multiplier coefficient α is 0.6 and the resolution multiplier coefficient ρ is 0.9. Therefore, the MobileNetV3 Small model can be used to recognize smaller images on embedded systems and terminals with weak computing power to speed up the computing speed while ensuring the accuracy.

There are some deficiencies in this experiment. First of all, the image data used in this experiment is relatively simple street view data, and the number of people in the picture does not exceed five people. Therefore, the accuracy of the final model is compared with that of complex environments. Experimental data may have large deviations. Secondly, this model may not perform as well as other existing models in some complex scenes, which requires further research. In addition, the experiment can further measure the comprehensive recognition effect of the model by adjusting the hyperparameters of the model and calculating the robustness of the model.

About the Author

Kaifu Dai is undergraduate of School of Computer and Information Technology, Hohai University, and his research field is Computer Science.

References

[1] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, Hartwig Adam. (2017) MobileNets:Efficient Convolutional Neural Networks for Mobile Vision Applications. https://www.researchgate.net/publication/316184205_MobileNets_Efficient_Convolutional_Neural_Networks_for_Mobile_Vision_Applications

[2] Lu Haiyan, Zhao Hongdong, Wang Tianmeng, etc. (2022) Infrared pedestrian behavior recognition based on lightweight convolutional neural network. Sensors and Microsystems, 2022, 41(09): 129-31.

[3] Du F J, Jiao S J. Improvement of lightweight convolutional neural network model based on YOLO algorithm and its research in pavement defect detection[J]. Sensors, 2022, 22(9): 3537.

[4] Nupoor Yawale. (2023) Design of a high-density bio-inspired feature analysis deep learning model for sub-classification of natural & synthetic imagery. https://www.researchgate.net/publication/372940300_Design_of_a_high-density_bio-inspired_feature_analysis_deep_learning_model_for_sub-classification_of_natural_synthetic_imagery?_sg=SRsbV9r_LO93c5UnPvxnqKHzoSABS3Axvq7nvY55OBHGFwszuBJR5rHV-NCj8cr_IvjSmPlJ0oRvYMM

[5] HERMOSILLA P,RITSCHEL T,VAZQUEZ P. (2018) Monte Carlo convolution for learning on non-uniformly sampled point clouds. ACM Transactions on Graphics(TOG),2018,37(6):1-12.

[6] Wang J, Li H, Yin S, et al. Research on improved pedestrian detection algorithm based on convolutional neural network[C]//2019 international conference on Internet of Things (iThings) and IEEE green computing and communications (GreenCom) and IEEE cyber, physical and social computing (CPSCom) and IEEE smart data (SmartData). IEEE, 2019: 254-58.