GPU acceleration design method for driver's seatbelt detection

Jing Yongquan[1],Wu Tianshu[12],Li Jin[3],Zhang Zhijia[1],Gao Chao[3]
1.School of Information Science and Engineering,Shenyang University of Technology
2.Opto-Electronic Information Technology Department,
Shenyang Institute of Automation Chinese Academy of Sciences
3.Technology Development Department Liaoning Aerospace Linghe Automobile Co., Ltd.
Shenyang, China
Phone:15566545830, Email:1052465060@qq.com

Abstract –*With the development and maturity of deep learning algorithms, CNN have emerged in the field of computer vision.Image recognition is one of the important research directions in the field of computer vision.The traditional image recognition method is to extract features by constructing feature descriptors and then classify them by classifiers, such as gradient direction histogram and support vector machine.These methods generally have the problems of poor robustness and insufficient ability to extract features in complex application scenarios.At the same time, convolutional neural network has not been well applied in image recognition due to its large amount of computation and slow speed.With the development of GPU, the parallel computing capability has been greatly improved.This paper designs a GPU acceleration method for the driver's seatbelt detection system based on CNN. The system is based on the Deconv-SSD target detection algorithm for vehicle detection, the Squeeze-YOLO algorithm for vehicle front windshield location, and the semantic segmentation for seat belt detection. Based on the characteristics of GPU, through the off-line merging bath normlization and convolution layer, Tensorrt model conversion technology to realize the GPU optimization speed. The results show that the proposed acceleration method can effectively improve the detection efficiency.*

Keywords *– GPU acceleration,seatbelt detection,SSD,YOLO*

I. INTRODUCTION

With the increasing number of motor vehicles, the traffic safety problem is becoming more and more serious. When a traffic accident occurs, wearing a seat belt can protect the driver's life as much as possible. At present, the main way to supervise drivers to wear safety belts is to detect traffic monitoring images manually. With the development of artificial intelligence, the automatic seat belt detection method based on image recognition has effectively solved the problems of low efficiency and the high cost of manual detection.

The excellent performance of in-depth learning in the field of image recognition makes CNN widely used in various fields of computer vision.In the fields of image classification, target detection and semantics segmentation, CNN has become the most accurate algorithm[1].

But the convolution neural network relies on a lot of convolution computation, and the algorithm is slow.With the continuous development of computer technology, especially the development of GPU (Graphics Processing Unit),parallel computing technology is becoming more and more popular. Parallel computing can effectively solve the problem of slow computing speed of cnn.

The widely used NVIDIA GPU(Graphics Processing Unit) is a general-purpose GPU, that is, GPGPU. General GPU has SIMD(Single Instruction Multiple Data) instruction sets, which can carry out the parallel computation of multiple data streams with one instruction at the same time, so it is suitable for parallel computing of parallel data such as CNN and image[2]. However, the parallel architecture of GPGPU is not specially customized for user algorithms, which may lead to the problem of reading data delay in the SIMD instruction set of general GPU. In addition, when the parallelism degree of GPU is insufficient, serial computation will be carried out in each thread, which results in the delay of computation.

978-1-7281-0510-9/19/$31.00 ©2019 IEEE

In this paper, a GPU method is designed for the driver seat belt detection system based on CNN, which realizes algorithm optimization and acceleration and makes the system better applied to the automatic detection of the driver wearing the seat belt. It has important practical significance for improving the drive's safety driving awareness and traffic safety.

II. ALGORITHM OF DRIVER SEAT BELT DETECTION

In order to realize the detection of the driver's seat belt, the vehicle detection must be completed by Deconv-SSD target detection algorithm from traffic surveillance video. Then the Squeeze-YOLO algorithm is used to locate the front windshield of each vehicle to locate the driver area, and then the semantic segmentation algorithm is used to detect the driver area for seat belt detection[3].

A. Vehicle Detection based on Deconv-SSD

By improving the algorithm of SSD[4] target detection, the ability of small target detection is improved. At the same time, the Deconv-SSD vehicle detection algorithm is proposed for the lightweight design of the front-end feature extraction model. The Deconv-SSD algorithm uses transposed convolution to achieve multi-scale feature fusion on feature graphs and designs a lightweight feature extraction minimum unit using depth-separable convolution and channel rearrangement,lightweight design for feature extraction.

By comparing the PASCAL VOC data set with KITTI data set, the average accuracy of the original SSD algorithm is improved from 77.2% to 79.6% in the case of the same feature extraction model and the same resolution input image.

B. Driver region location based on Squeeze-YOLO

Squeeze-YOLO is a driver region location algorithm based on YOLO[5] V1 target detection algorithm and Squeezenet lightweight network structure.The front windshield of vehicle has obvious characteristics, which is only a two-class task. The difficulty of target detection is low and multi-scale features are not needed.The algorithm is used to locate the front windshield of the vehicle, and the right half of the front windshield of the vehicle is taken to be the driving area of the driver. Squeeze-YOLO can achieve the balance of speed and average accuracy by adding or subtracting the minimum module of feature extraction according to the actual situation in the front-end feature extraction part, and use YOLO V1 loss function to design target detection algorithm to quickly complete driver area location.

Because of the obvious features of the front windshield, high precision can be obtained when using Squeeze-YOLO target detection algorithm. When the precision is 99.96%, the speed can reach 73 frames / s.

C. Seat belt semantic segmentation algorithm

FCN[6] semantic segmentation algorithm and full convolution network pruning are used to determine whether the driver is wearing a seat belt or not. In the semantic segmentation network structure, 32 times up sampling is used in the decoding part, and the edge precision is ignored. On the basis of determining the area of the maximum connected domain after the semantic segmentation by the threshold value to realize the seatbelt detection, the semantic segmentation algorithm is pruned in the coding part of the channel, and the algorithm is accelerated by removing the redundant convolution kernel. The comparison experiment on the self-made seatbelt detection data set shows that the speed of the proposed algorithm can reach 305 FPS when the accuracy of the algorithm is 94%.

III. GPU ACCELERATED DESIGN

In this chapter, the time-consuming algorithm in the system is optimized, through the off-line merging bath normlization and convolution layer, to achieve the effect of acceleration. According to the GPU hardware characteristics, the Tensorrt acceleration library is used to reduce the data interaction between CPU and GPU, so as to further achieve the acceleration effect.

A. Batch normlization layer merge

Convolutional neural network algorithm can be developed without changing the existing open source framework of deep learning and the structure of convolutional neural network algorithm from training in R&D design stage to reasoning deployment in the system.In the design stage, the main consideration is to improve the accuracy and speed of the algorithm by improving the structure design of the algorithm.In

deployment system, the network speed can be improved without loss of accuracy by optimizing the calculation method of convolution neural network.

Because of the universal design of deep learning open source framework, if no engineering optimization is done, the training algorithm is directly deployed into the system. Although it can still run, it will not get the optimal computing speed.At present, the main application scenario of deep learning open source framework is design and training algorithm. Its goal is to reduce the development difficulty of R&D personnel and focus on the improvement of the algorithm itself.

Batch normlization can effectively prevent the network appeared gradient explosion and gradient dispersion phenomenon raise the fitting ability of the network, so widely used in the convolution layers of CNN, the calculation method is divided into two steps: the first step is to find the mean and variance of the current batch of eigenvectors; the second step is to adjust the mean and variance through two coefficients.

In reasoning deployments, weights and offsets act on batch normlization adjusted feature vectors, as shown in the formula (1) w and b for the weights and bias, as the feature vector, x is the eigenvector, γ and β is batch of normlization coefficient.

$$w(\gamma \frac{x-\mathrm{E}(x)}{\sqrt{\mathrm{var}(x)}}+\beta)+b \tag{1}$$

Looking at the formula above, it can be found that part of the calculation can be stored in memory in advance in the process of off-line calculation, so as to avoid re-calculating in the process of execution of the system. If the trained weights are directly deployed to the system, it will waste computing resources. In the reasoning deployment of the trained model, mean value E(x) and variance var(x) are derived from the values obtained in the training stage, so the weight and bias can be integrated. The integrated weight and bias are shown in equations (2) and (3).

$$w_{merged}=w\times\frac{\gamma}{\sqrt{\mathrm{var}(x)}} \tag{2}$$

$$b_{merged}=(b-\mathrm{E}(x))\times\frac{\gamma}{\sqrt{\mathrm{var}(x)}}+\beta \tag{3}$$

Where wmerged and bmerged represent the new weight and bias after integration, and the integrated weight and bias are stored as the new weight and bias, thus avoiding online calculation in the reasoning process. Because batch normlization is a combination of coefficients and weights, it does not result in any reduction in accuracy.

When using the GPU for CNN calculation, the host CPU needs to interact with the data of the GPU device. Host CPU completes data preparation, copies data to GPU, and opens GPU calculation function. When the GPU calculation is completed, the computing structure is copied back to the host side to complete a parallel acceleration of heterogeneous computing. GPU heterogeneous computing is shown in Figure Fig.1.

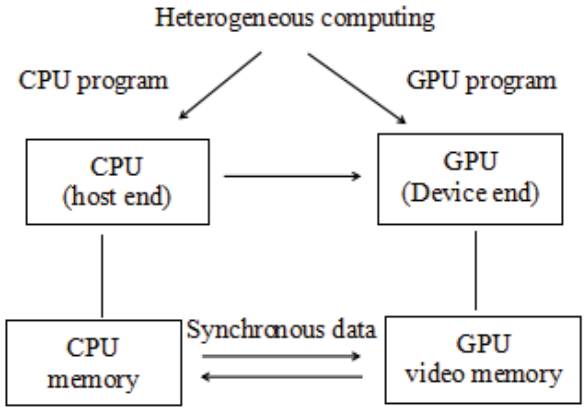

Fig. 1 GPU heterogeneous computing structure diagram

In the development and training phase, in order to consider generality, the in-depth learning development framework will transmit the data back to the memory of the CPU when the GPU completes parallel computing. Wait until the next layer of computing before copying data from CPU memory to GPU memory for parallel computing again.

Due to the limitations of current technologies, data interaction between CPU and GPU mainly USES the PCI-E interface. Frequent data interaction will bring about the delay,for example, the image of 512 x 512 is convoluted with a convolution template of size 5 x 5. The calculation time of GPU is 0.08ms, but the two data transmission time of CPU and GPU is 0.5ms.When deploying the trained model into the system, consideration should be given to merging the calculation

of GPU terminal to avoid redundant data transmission and reduce delay.

B. GPU acceleration based on Tensorrt

Tensorrt is an accelerated computing library developed by NVIDIA, it can automatically analyze the trained model, merge the computing layer without CPU and GPU interaction, no longer send data back to CPU and memory, and complete the calculation in GPU and display memory. while supporting the storage of the merged model as a Tensorrt format to facilitate off-line calls. Tensorrt supports the conversion of weights of various mainstream deep learning development frameworks, which avoids the redundancy of deploying multiple deep learning development frameworks in system integration and achieves unification on the system deployment side.

Tensorrt can also be used for off-line quantization, using float 16-bit semi-precision floating-point calculations instead of float 32-bit single-precision floating-point calculations. Semi-precision floating-point Numbers have less bit width than single-precision floating-point Numbers, and can theoretically perform the same calculations in half the clock cycle as float 32-bit full-precision floating-point Numbers. However, in practical use, since the NVIDIA GPU architecture is different, the semi-precision floating-point SIMD computing power is different, and the final acceleration effect is dependent on the hardware itself architecture. Because of the redundancy and the robustness of the CNN itself, the semi-precision floating-point number cannot be reduced by the method of off-line quantization.

IV. SYSTEM IMPLEMENTATION AND RESULT ANALYSIS

A. system implementation

This safety belt detection algorithm is completed in the framework of Caffe in-depth learning and development. At the same time, the CUDA environment of NVIDIA in parallel computing is the mainstream development environment at present, so this paper uses GPU of NVIDIA to implement driver's seat belt detection system.Since Caffe framework is not mature in supporting GPU parallel computing under Windows, this system is developed under Ubuntu[7]. In addition to depth learning algorithms, the seat belt detection system also includes basic image operations such as reading and writing images, clipping images and zooming images. The image basic operation part is developed by using OpenCV, an open source computer vision computing library. Making user man-machine interaction interface by using the QT graphical interface library in Ubuntu, and using the Qt Creator IDE to complete the system development.

The system calculation time is mainly focused on vehicle detection and vehicle windshield detection. Because of the large resolution of the input images, the proposed method mainly optimizes the vehicle detection and windshield detection system.

B. Results analysis

The experimental hardware environment GPU is NVIDIA 1070 mobile, the processor is i7 7700 HQ, and the operating system environment is Ubuntu 14.04.Merging batch normalization layer and using Tensorrt 4.0 software to accelerate the calculation time are shown in Table Ⅰ.

Table I. Comparison of Optimization effects of GPU

Optimization algorithm	Not optimized time / s	Merge batch normlization layers / s	Merge batch normlization layer and Tensorrt acceleration/s
Vehicle detection	0.14	0.10	0.06
Windshield detection	0.10	0.07	0.03

Through the comparison experiment, it can be found that the combination of batch normlization layer and the use of Tensorrt acceleration can optimize the acceleration of the original algorithm. The total time saved by vehicle detection and windscreen positioning is about 0.15s.

In short, merger batch normlization layer and using Tensorrt for GPU acceleration method has a lot to improve image processings.

V. CONCLUSION

In this paper, the design and implementation of GPU

acceleration method for driver seat belt detection system are presented. Based on GPU features,in the security band detection system, the computation time is shortened by offline merging batch normalization and convolution layer, and the data interaction between CPU and GPU is reduced by Tensorrt model transformation technology.The experimental results show that the accelerated GPU model is effective.

ACKNOWLEDGEMENT

This work was financially supported by Key Projects of the Joint Fund of the Chinese Academy of Sciences(Y8K4160401),Shenyang Artificial Intelligence Key Laboratory Fund and National Natural Science Foundation of China(61540069).

REFERENCES

[1] ZHOU F Y, JIN L P, DONG J. A Survey of convolution Neural Networks[J]. Chinese Journal of Computers, 2017, 40(6): 1229-1251.

[2] HO T Y, LAM P M, LEUNG C S. Parallelization of cellular neural networks on GPU[J]. Pattern Recognition, 2008, 41(8):2684-2692.

[3] HE K, ZHANG X, REN S, et al. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification[C].Proceedings of the IEEE international conference on computer vision，2015: 1026-1034.

[4] WU T SH, ZHANG ZH J, LIU Y P，et al. Dirver Seat Belt Detection Based on YOLO Detection and Semantic Segmentation[J]. Jisuanji Fuzhu Sheji yu Tuxingxue Xuebao,2019(01).

[5] LIU W, ANGUELOV D, ERHAN D, et al. Ssd: Single shot multibox detector[C].European conference on computer vision. Springer, Cham, 2016: 21-37.

[6] REDMON J, DIVVALA S, GIRSHICK R, et al. You only look once: Unified, real-time object detection[C].Proceedings of the IEEE conference on computer vision and pattern recognition,2016: 779-788.

[7] LONG J, SHELHAMER E, DARRELL T. Fully convolutional networks for semantic segmentation[C].Proceedings of the IEEE conference on computer vision and pattern recognition,2015: 3431-3440.

[8] JIA Y, SHELHAMER E, DONAHUE J, et al. Caffe: Convolutional architecture for fast feature embedding[C].Proceedings of the 22nd ACM international conference on Multimedia,ACM,2014: 675-678.