Computer Engineering
ISSN 1000-3428,CN 31-1289/TP

NCS2

DOI 10.19678/j.issn.1000-3428.0056214
2020-03-26

NCS2
. https://doi.org/10.19678/j.issn.1000-3428.0056214

ISSN 2096-4188 CN 11-6037/Z

网络首发时间：2020-03-26 16:50:55
网络首发地址：http://kns.cnki.net/kcms/detail/31.1289.tp.20200326.1307.004.html

基于 NCS2 神经计算棒的车辆检测方法

610500

：PC GPU 3b+ NCS2 Tiny-YOLO Tiny-YOLO NCS2 3b+ 12 MS COCO VOC2007 1.12% 0.23% Tiny-YOLO

(OSID):

Vehicle detection method based on NCS2 neural compute stick

Jiang Xiaoyu, Li Zhongbing, Zhang Junhao, Peng Jiao, Wen Ting.

(School of Electrical Engineering and Information, Southwest Petroleum University, Chengdu 610500, China)

Abstract At present, most methods of deep learning to detect vehicles have high accuracy, but the real-time performance depends on PC or GPU devices with excellent performance, which limits its practical applications in embedded devices with relatively low performance. Therefore, an embedded vehicle detection system based on raspberry PI 3b+ is designed in this paper, which uses NCS2 neural computing rod (a miniature deep learning hardware driver) to provide deep learning acceleration for low-performance devices. A Tiny-YOLO network model based on depth separable convolution is proposed for real-time vehicle detection to improve the accuracy of target detection of embedded devices. Real-time experiments show that the real-time performance of the proposed algorithm is twice that of the original Tiny-YOLO algorithm. After the retrained network model is deployed to the raspberry PI 3b+ with NCS2 neural stick, the processing speed can reach 12 frames per second. At the same time, the improved algorithm improves the average detection accuracy by 1.12% and 0.23% respectively in MS COCO and VOC207 vehicle data sets.

Key words Vehicle detection; Depth separable convolution; Deep learning; Tiny-YOLO; Embedded device;

DOI:10.19678/j.issn.1000-3428.0056214.

1

HOG [1] SIFT [2] SVM [3] Adaboost[4] Gradient Boosting[5]

Back Propagation [6] one stage two stage two stage Region Proposal [7-8] R-CNN[9] Fast R-CNN[10] Faster R-CNN[7] one stage YOLO[11-12] SSD[13]

201810615094 201801006095

(1996—) E-mail:ceroo1005@gmail.com

Joseph Redmon[14] YOLO

Tiny-Yolo

[15] Tiny-YOLO

depthwise separable

convolutions [16] NCS2

12

2 YOLOV3

YOLOV3 Darknet53

[17] Darknet53 53

[18]

416 × 416 Drknet53

13 × 13 26 × 26 52 × 52

1×1×(B×(5+C 1 × 1 B

5+C

t_x, t_y, t_w, t_h objectness C

YOLOV3

K-means [19] 3

3 9

P_x, P_y P_w, P_h

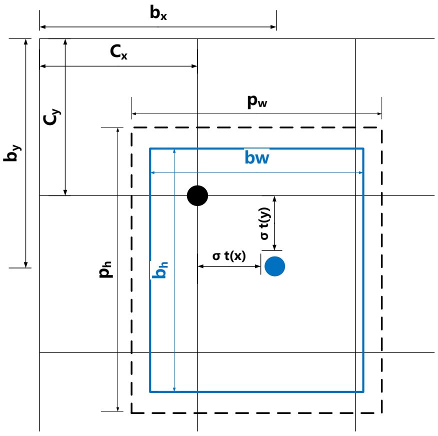
G_x, G_y, G_w, G_h ground truth

4 t_x, t_y, t_w, t_h

$$\begin{aligned} t_x &= G_x - P_x \\ t_y &= G_y - P_y \\ t_w &= \log(G_w / P_w) \\ t_h &= \log(G_h / P_h) \end{aligned} \quad (1)$$

ground truth

1



1

Fig.1 The offset between prior box and prediction box

$(C_x \quad C_y)$ $(C_x \quad C_y)$ P_x, P_y

b_x, b_y, b_w, b_h 1

t_x, t_y, t_w, t_h

$$\begin{aligned} b_x &= \sigma(t_x) + C_x \\ b_y &= \sigma(t_y) + C_y \\ b_w &= P_w e^{t_w} \\ b_h &= P_h e^{t_h} \end{aligned} \quad (2)$$

σ sigmoid 1 × 1

sigmoid t_x, t_y 0 1

t_w, t_h

G_w / P_w G_h / P_h

P_w P_h

objectness

$$C_{Object} = P_{Object} \times IOU_{pred}^{truth} \quad (3)$$

P_{Object}=1 0 IOU

ground truth

YOLOV3 B 3 C 80

3 80

3 Tiny-YOLO

Tiny YOLO

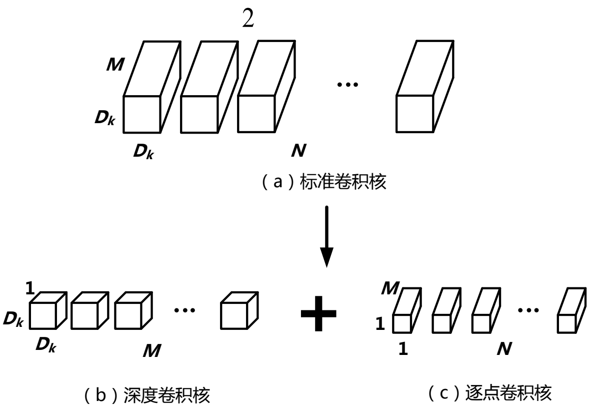
13 × 13 26 × 26

MobileNet[20]

3.1

(depthwise convolution) (pointwise convolution)

2



2

Fig.2 The structure of deep separable convolution

$F \in R^{D_f \times D_f \times M}$ $D_f \times D_f$

M

$K \in R^{D_k \times D_k \times M \times N}$ $D_k \times D_k$

M N

$G \in R^{D_g \times D_g \times N}$ $D_g \times D_g$

N

$$\mathbf{G}_{k,l,n} = \sum_{i,j,m} \mathbf{K}_{i,j,m,n} \cdot \mathbf{F}_{k+i-1,l+j-1,m} \qquad (4)$$

$$O_{(conv)} = D_K \cdot D_K \cdot M \cdot N \cdot D_f \cdot D_f$$

$F \in R^{D_f \times D_f \times M}$

$\hat{K} \in R^{D_k \times D_k \times 1 \times M}$

$$\hat{\mathbf{G}}_{k,l,m} = \sum_{i,j} \hat{\mathbf{K}}_{i,j,m} \cdot \mathbf{F}_{k+i-1,l+j-1,m} \qquad (5)$$

$$D_K \cdot D_K \cdot M \cdot D_f \cdot D_f$$

1x1

$$O'_{(conv)} = D_K \cdot D_K \cdot M \cdot D_f \cdot D_f + M \cdot N \cdot D_f \cdot D_f$$

$$\frac{O'_{(conv)}}{O_{(conv)}} = \frac{D_K \cdot D_K \cdot M \cdot D_f \cdot D_f + M \cdot N \cdot D_f \cdot D_f}{D_K \cdot D_K \cdot M \cdot N \cdot D_f \cdot D_f} = \frac{1}{N} + \frac{1}{D_K^2} \qquad (6)$$

N 1 D_k

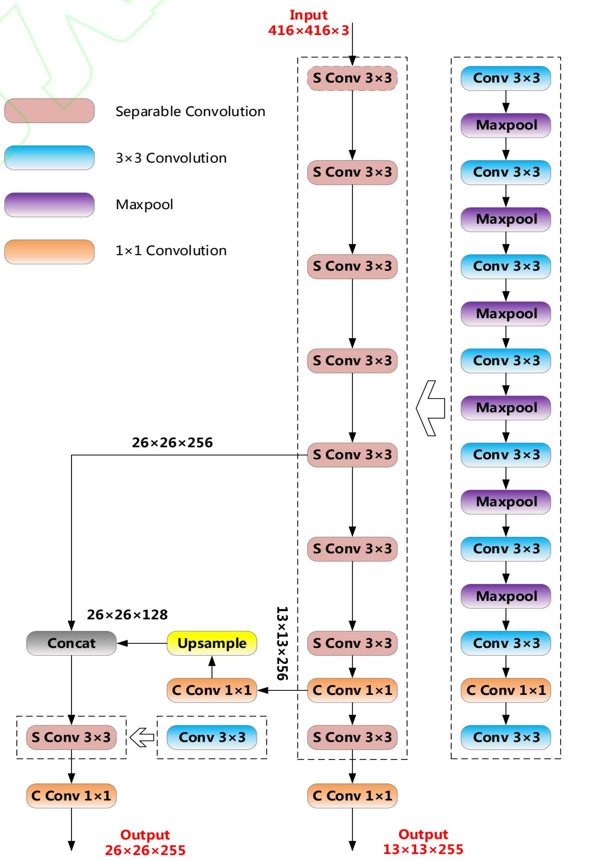
3.2 Tiny-YOLO

Tiny-YOLO 3 " S Conv 3 × 3 " Tiny-YOLO 3 × 3 3 " Conv 3 × 3 " 9 1 Tiny-YOLO 3 " Maxpool " Tiny-YOLO 3

1 3x3

Tab.1 3 x 3 convolution information in the original network

1	3x3	16	416x416x3	416x416x16
2	3x3	32	208x208x16	208x208x32
3	3x3	64	104x104x32	104x104x64
4	3x3	128	52x52x64	52x52x128
5	3x3	256	26x26x128	26x26x256
6	3x3	512	13x13x256	13x13x512
7	3x3	1024	13x13x512	13x13x1024
8	3x3	512	13x13x256	13x13x512
9	3x3	256	13x13x384	13x13x256



3 Tiny-YOLO

Fig.3 Original and improved Tiny-YOLO network structure

4

MS COCO

821

400

700 121

VOC

1 ubuntu 16.04 2

tensorflow 3

raspberry 3b+ NCS2 4 E5 2680 + GTX1066

4.1

tensorflow

OpenVINO tensorflow NCS2

IR NCS2

4

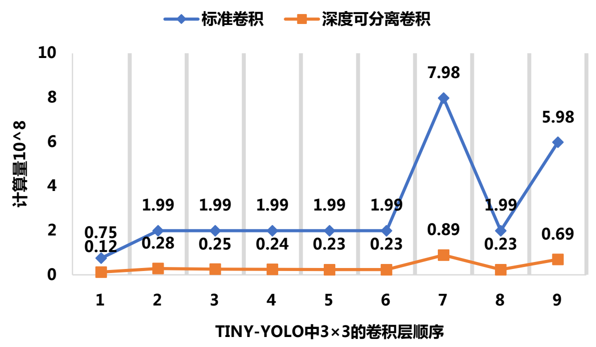
4

Fig.4 Development flow chart

4.2

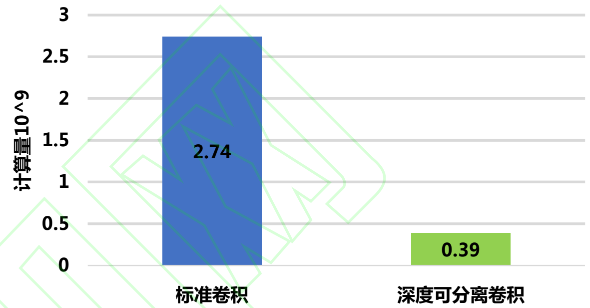
Tiny-YOLO 9

5



(a)

(a) comparison of calculation amount before and after replacing structure



(b)

(b) total calculation

5 a (b)

Fig.5 (a) is the calculation amount of the replacement structure part, and (b) is the total calculation amount of the algorithm

2.74 × 10^11

0.39 × 10^11 86%

4.3

Mean Average Precision MAP

$$m_{AP} = \int_0^1 P(R)\,dR \tag{7}$$

P R P(R)

Frames Per Second FPS

2

Tab.2 Comparison between this algorithm and the original algorithm

	MAP			
	VOC	COOC	MB	FPS
Tiny-YOLO	55.66%	33.10%	34.6	2
	55.89%	**34.22%**	22.4	4
NCS2	54.12%	**34.21%**	**13.2**	**12**

2

MS COCO VOC2007 Tiny-YOLO

1.12%

0.23% 2 OpenVINO

NCS2 12FPS

4.4

VOC

6 Tiny-YOLO

6

Fig.6 Effect contrast

Tiny-YOLO

5

STM32

[1] Dalal, N, Triggs, B. Histograms of Oriented Gradients for Human Detection. In 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05); IEEE, 2005, Vol. 1, 886–893.

[2] David G.Lowe. Object Recognition from Local Scale-Invariant Features. In Proceedings of the Seventh IEEE International Conference on Computer Vision; IEEE: 1999, vol. 2, 1150–1157.

[3] Sánchez A, V. David. Advanced Support Vector Machines and Kernel Methods. Neurocomputing, 2003, 55 (1–2), 5–20.

[4] Artue Ferreira, Mario Figueiredo. Boosting Algorithms: A Review of Methods, Theory, and Applications. In Ensemble Machine Learning, 2012, 35–85.

[5] Friedman, Jerome. Greedy Function Approximation: A Gradient Boosting Machine. The Annals of Statistics. 2000

[6] David E, Rumelhart, Geoffrey E, Hinton, Ronald J. Williams. Learning Representations by Back-Propagating Errors. 1986, 533–536

[7] Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. arXiv:1506.01497 [cs] 2015.

[8] Tao Kong, Anbang Yao, Yurong Chen, Fuchun Sun. HyperNet: Towards Accurate Region Proposal Generation and Joint Object Detection. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); IEEE, 2016, 845–853.

[9] Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. arXiv:1311.2524

[cs] 2013.

[10] Ross Girshick. Fast R-CNN. arXiv:1504.08083 [cs] 2015.

[11] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi. You Only Look Once Unified, Real-Time Object Dete. arXiv:1506.02640 [cs] 2015.

[12] Joseph Redmon, Ali Farhadi. YOLO9000 Better, Faster, Stronger. arXiv:1 1612.08242 [cs] 2016.

[13] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy. SSD: Single Shot MultiBox Detector. arXiv:1512.02325 [cs] 2016.

[14] Joseph Redmon, Ali Farhadi. YOLOv3: An Incremental Improvement. arXiv:1 1 1804.02767 [cs] 2018.

[15] ZHANG Yangshuo, MIAO Zhuang, WANG Jiabao, LI Yang. Pedestrian detection method based on Movidius neural computing stick. Journal of Computer Applications[j], 2019, 39 (08), 2230–2234. ;

; ; . Movidius

. 2019, 39 (08), 2230–2234.

[16] Francois Chollet. Xception: Deep Learning with Depthwise Separable Convolutions. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR);IEEE,2017 1800–1807.

[17] Jonathan Long, Evan Shelhamer, Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation; IEEE, 2016, 640–651

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren. Deep Residual Learning for Image Recognition. arXiv:1512.03385 [cs] 2015.

[19] Kiri Wagstaff, Claire Cardie, Seth Rogers, Stefan Schroedl. Constrained K-Means Clustering with Background Knowledge. Proceedings of 18th International Conference on Machine Learning[j], 2001, 577-584

[20] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. arXiv:1704.04861 [cs] 2017.