Acceleration of Pedestrian Detection in High Resolution Image

Haodi ZHANG, Alexandrina ROGOZAN, Abdelaziz BENSRHAIR

Laboratoire d'Informatique, de Traitement de l'Information et des Systèmes
685 Av. de l'Université, 76800, Saint-Étienne-du-Rouvray, France
haodi.zhang@insa-rouen.fr, alexandrina.rogozan@insa-rouen.fr
abdelaziz.bensrhair@insa-rouen.fr

Résumé – Les images à haute résolution (HR) fournissent des informations plus détaillées tout en créant une lourde charge de calcul, ce qui entraîne une vitesse de détection plus lente. Nous proposons un nouvel algorithme d'accélération de la détection des piétons appelé NPAE (Non-Pedestrian Area Estimation). L'algorithme NPAE estime et supprime les zones non piétonnes de l'image, avant de procéder à la détection des piétons sur l'image de sortie NPAE. L'algorithme NPAE proposé est un module plug-and-play qui peut être utilisé sur des plateformes GPU et CPU. Notre algorithme augmente la vitesse de détection de 42% sur la plate-forme GPU et de 52% sur la plate-forme CPU, pour les images de test d'une résolution de 1920×1080 pixels. En termes de précision de détection, NPAE améliore jusqu'à 37,94% par rapport aux méthodes classiques d'accélération de l'optimisation des données.

Abstract – High resolution (HR) images provide more detail information while creating a heavy computational load, resulting in slower detection speed. We propose a novel pedestrian detection acceleration algorithm called Non-Pedestrian Area Estimation (NPAE). The NPAE algorithm estimates and removes non-pedestrian areas of the image, followed by pedestrian detection of the NPAE output image. The proposed NPAE algorithm is a plug-and-play module that can be used on both GPU and CPU platforms. Our algorithm increases the detection speed with 42% on the GPU platform and with 52% speed increase on the CPU platform, for the test images with a resolution of 1920×1080 pixel. In terms of detection accuracy, NPAE improves by up to 37.94% compared to classical data optimization acceleration methods.

1 Introduction

Pedestrian detection technology has made great progress in the past decade [1] due to deep neural networks. As more and more layers are adopted, deep neural networks usually have higher detection accuracy. The increased complexity requires more computing resources and usually consumes additional computing time.

High-resolution (HR) image is required to detect pedestrian targets at long distances. HR image has many advantages, such as improved detection results [2]. It is also a basic need for autonomous driving [3, 4]. The rich detail information provided by HR images improves the accuracy of long-distance detection. However, it increases the amount of computational consumption and time. Therefore, Pedestrian detection needs to be accelerated and kept accurate.

The NPAE algorithm is proposed for acceleration pedestrian detection. It works by estimating and cropping the Non-Pedestrian Area of the image. Due to the reduction of image size, the pedestrian detector can process faster. Moreover, the accuracy of the pedestrian detector can be maintained because of the pedestrian area of the image is not modified. The NPAE algorithm has several highlights : 1) The NPAE algorithm can calculate and crop the Non-Pedestrian Area uniformly for the images acquired under the same capture conditions. 2) The NPAE algorithm does not require training and is capable of accelerating a variety of state-of-the-art pedestrian detectors. 3) The proposed algorithm is an image data optimization method that can also be used in a variety of applications such as image segmentation, object tracking.

2 Related Work

2.1 Pedestrian Detectors

In recent years, machine learning methods such as SVM, ELM, ANN, CNN have gradually replaced manual design rules [5]. In the field of pedestrian detection, deep learning has been adopted by most algorithms with excellent performances, such as the R-CNN [6] and the YOLO [7]. In [8], the RetinaNet introduces a novel loss function and fuses different levels of features through lateral connections. These designs ensure that RetinaNet is an efficient and accurate detector. Therefore, we chose RetinaNet as the reference of the pedestrian detector to test the performance of the proposed NPAE algorithm.

2.2 Pedestrian Detection Acceleration

Most of the advanced acceleration methods are summarized in [9], such as pruning, low-rank approximation, quantization, knowledge distillation and compact network design. All mentioned methods are based on optimizing deep neural networks for acceleration. These methods are incredible in thought and

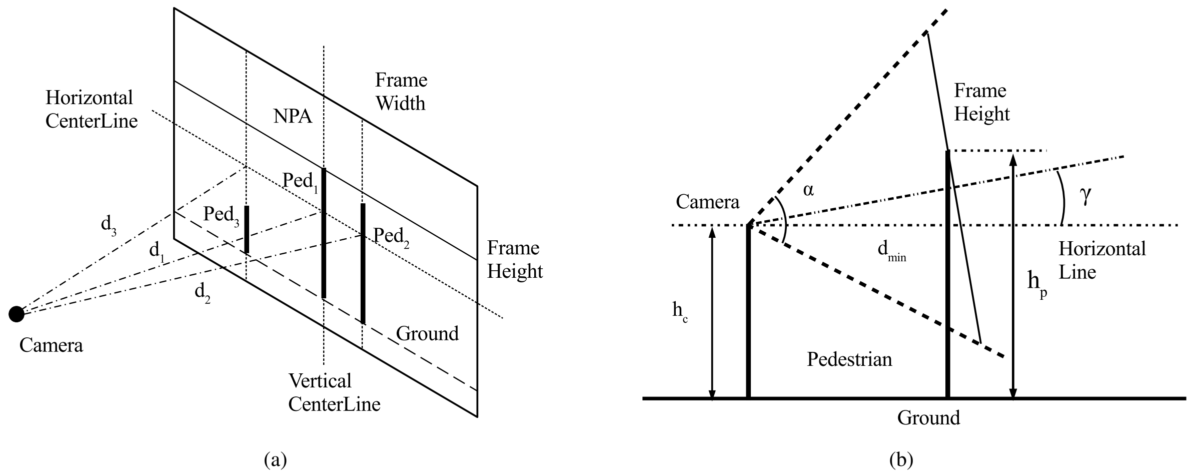

FIGURE 1 – Analyze and build NPAE algorithms for single images. (a) Analysis of non-pedestrian area at the top of the image containing multiple pedestrians. (b) Analysis of the environmental parameters involved in NPAE algorithm.

design, and have significant effects in practical.

However, we noticed that the input image is an under-optimized part of the detection process. There are two processing methods for the image input to the detector. One method is to compress the input image to a fixed size as in the [10, 11]. Another method crops part of the image area, such as [12]. Both of these image processing methods are very convenient. Nevertheless, the compress-image method results in a loss of detection accuracy. The crop-image method can only be used with an existing dataset and acquisition environment. Changes in camera model, spatial position, and scaling require researchers to redesign the system.

3 Algorithm Design

We observe that the HR images obtained by autonomous vehicles contain some areas without pedestrians. We named this area a Non-Pedestrian Area (NPA). We find that there is a relationship between the region size of NPA in the image and the environmental parameters. The environmental parameters contain camera height, camera pitch angle, camera vertical field of view (FOV), the distance between the pedestrian and the camera, pedestrian height and vertical resolution of the image. Therefore, we propose the Non-Pedestrian Area Estimation (NPAE) algorithm to describe the relationship between the two factors. Afterwards, the NPAE algorithm can estimate the NPA based on the environmental parameters. Once the NPA is acquired, we remove the non-pedestrian area from the HR image and then send the processed image to the subsequent pedestrian detector.

To accurately estimate the non-pedestrian area in a single image, it is necessary to analyze the highest position of the pedestrian bounding box in the image, as shown in Figure 1(a). Several environmental parameters are used : h_c is the camera height, γ is the camera pitch angle, α is the camera vertical field of view (FOV), d_{pc} is the distance between the pedestrian and the camera, h_p is the pedestrian height, px_v is the vertical resolution of the image, NPA is the Y-axis coordinate of the pedestrian bounding box top-left point according to the environmental parameters. Based on these parameters, the NPAE algorithm equation with γ is shown in equation (1) :

$$\text{NPA} = \frac{px_v}{2}\{1 - \frac{(h_p - h_c)\cos\gamma - d_{pc}\sin\gamma}{[d_{pc}\cos\gamma + (h_p - h_c)\sin\gamma]\tan\frac{\alpha}{2}}\} \quad (1)$$

Using NPAE to calculate the smallest non-pedestrian area in a dataset, it is possible to consistently accelerate for object detection.

4 Experiments and Results

4.1 Experiment preparation

The JAAD dataset [13] is adopted to verify the performance of proposed NPAE algorithm. It has various weather conditions such as sunny, cloud, rain and snow. JAAD dataset has 336 videos with a resolution of 1920×1080 pixels.

There are six environmental parameters need to be determined. According to [13], for JAAD with a resolution of 1920×1080 : Vertical resolution $px_v = 1080$ pixels. Camera vertical FOV $\alpha = 73°$. Camera pitch angle $\gamma = 0°$. Camera height $h_c = 1.6$ m. The pedestrian height $h_p = 1.8$ m, pedestrian-camera distance $d_{pc} = 1.2$ m.

Considering the car shudder, the Camera pitch angle γ is varied between $[-1°, 1°]$. The adjusted environmental parameters can be used to derive the corresponding NPA for each dataset according to equation (1). The NPA interval for JAAD 1920×1080 is between $\text{NPA}_{\gamma=-1}$ and $\text{NPA}_{\gamma=1}$. We take the minimum value, so NPA $= 405$ pixels. To facilitate the experiment, the value of NPA is modified using $\text{NPA}_{mod} = \text{NPA} - (\text{NPA} \bmod 10)$.

The RetinaNet is adopted to detect pedestrians. There are several versions of RetinaNet, depending on the backbone used.

TABLE 1 – The experimental results on JAAD 1920×1080 pixels test data. In combination with the proposed NPAE algorithm (*NPAE* in the table) the detection speed on the GPU platform is 8.36 fps improved by 46.92%. The detection speed on the CPU platform is 0.35 fps, which is a 52.17% improvement. *AP* mean average precision. *FPS* mean frame per second on gpu platform. *RT* mean runtime. Ped_{PC} mean partially cropped pedestrian. Ped_{CR} mean completely removed pedestrian. *G* mean GPU platform. *C* mean CPU platform.

Model	Resolution (pixels)	*AP* (%)	FPS_G	RT_G (ms)	FPS_C	RT_C (ms)	Ped_{PC}	Ped_{CR}
RetinaNet	1920×1080	84.60	5.69	175.7	0.23	4307.8	0	0
Compress	1920×680	61.75	8.36	119.6	0.35	2819.9	0	0
Crop-150	1920×930	82.94	6.04	165.4	0.26	3769.1	0	0
Crop-200	1920×880	87.78	6.38	156.7	0.27	3584.9	0	0
Crop-250	1920×830	82.59	7.12	140.4	0.29	3381.8	0	0
Crop-300	1920×780	86.74	7.58	131.8	0.31	3191.8	0	0
Crop-350	1920×730	83.01	7.91	126.4	0.33	2994.3	2	0
NPAE(Crop-400)	1920×680	**84.85**	**8.36**	**119.5**	**0.35**	**2813.2**	**33**	**0**
Crop-450	1920×630	84.03	9.14	109.4	0.38	2625.2	348	0
Crop-500	1920×530	81.50	9.50	105.2	0.41	2421.7	2657	3

We chose the resnet50 [14] as the backbone network. This network offers a better balance between detection performance and speed.

4.2 Experiments

In Table 1, RetinaNet is the baseline. The evaluation metric used in this paper is the Average Precision (AP) [15]. The detection result has 84.80% average precision (AP). It takes $RT_G = 175.7$ milliseconds to process an image on the GPU, which means 5.69 frames per second (FPS_G). On the CPU, the time to detect an image RT_C is 4307.8 milliseconds, which means 0.23 frames per second (FPS_C). The proposed NPAE algorithm reduces the detection time of RetinaNet to 119.5 ms and 2813.2 ms on two platforms. The detection speed is 8.36 fps on GPU and 0.35 fps on CPU. The detection speed is increased by 46.92% and 52.17% respectively. After NPAE acceleration, the AP of the detector RetinaNet remains at 84.85%.

Image compression method, as $Compress$ mentioned in Table 1, can also accelerate detection. However, there is a significant loss of accuracy. The experimental detection accuracy AP is only 61.75%. The results can be further improved to a certain extent by extensive multi-scale training. Nevertheless, this improvement is limited and leads to a more complex process.

NPAE is an accurate image cropping method. We have also labeled it *Crop-NPA*. In order to compare the performance of NPAE, seven experiments with different cropping values are designed in Table 1. These experiments are named *Crop-Value* and *Value* is the number of rows of cropped pixels. In addition to analyzing the detection accuracy *AP* and detection speed *FPS/RT*, the pedestrian targets affected by the cropping should also be considered. The partially cropped pedestrian target number is recorded as Ped_{PC}, and the completely removed pedestrian is labeled as Ped_{CR}.

In terms of acceleration, the speed increases as the degree of shearing increases. In terms of detection accuracy, it first remains constant and then decreases. The inflexion point occurs around the NPA value estimated by the NPAE algorithm. The Ped_{PC} and Ped_{CR} values also show an inflexion point around the NPA value. This pattern means that more and more pedestrian area information is lost as the cropping range exceeds the NPA value. This results in a decrease in detection accuracy.

5 Conclusion

In this paper, we present an algorithm for accelerating pedestrian detection with high resolution images, named Non-Pedestrian Area Estimation (NPAE). Based on six environmental parameters, the NPAE algorithm can accurately estimate the non-pedestrian areas in the dataset. Therefore, the proposed algorithm crops the image more accurately compared to other image data optimization methods. Because the image area is reduced and the target information is preserved, the pedestrian detector gains acceleration while maintaining accuracy.

The NPAE algorithm has four advantages : 1). It accelerates detection without loss of accuracy, 2). It can be used on a variety of computing platforms, 3). It is easily integrated with other acceleration and detection methods. 4). it does not require any training.

On 1920×1080 high resolution images, two different computing platforms, GPU and CPU, are used for the experiments. In the GPU platform, the detection speed is increased by 46.92%. In the CPU platform, the detection speed is increased by 52.17%. All experiments have shown that the proposed algorithm does not affect the detection accuracy. The NPAE algorithm performs better compared to traditional image data optimization methods. Image compression usually results in a loss of detection accuracy. Compared to NPAE algorithm, image compression loses 23.10% of average precision on 1920×1080 resolution. Image cropping cannot be calculated to obtain the optimal size of the cropping area, so it is difficult to accurately balance detection accuracy and detection speed.

In future work, we will explore the use of NPAE algorithm for pedestrian image position estimation by introducing depth information. At the same time, the NPAE algorithm can also be used to accelerate pedestrian detection of multi-modal fusions.

Acknowledgement

This research was supported by the China Scholarship Council. The authors would like to thank the Centre Régional Informatiqueet d'Applications Numériques de Normandie for providing computational resources.

Références

[1] Antonio Brunetti, Domenico Buongiorno, Gianpaolo Francesco Trotta, and Vitoantonio Bevilacqua. Computer vision and deep learning techniques for pedestrian detection and tracking : A survey. *Neurocomputing*, 300 :17–33, 2018.

[2] Brais Bosquet, Manuel Mucientes, and Víctor M Brea. Stdnet : Exploiting high resolution feature maps for small object detection. *Engineering Applications of Artificial Intelligence*, 91 :103615, 2020.

[3] Xinyu Huang, Xinjing Cheng, Qichuan Geng, Binbin Cao, Dingfu Zhou, Peng Wang, Yuanqing Lin, and Ruigang Yang. The apolloscape dataset for autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 954–960, 2018.

[4] Haodi ZHANG, Alexandrina ROGOZAN, and Abdelaziz BENSRHAIR. Cross-modal verification for 3d object detection. In *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, pages 195–200, 2021.

[5] Wen-jing Niu and Zhong-kai Feng. Evaluating the performances of several artificial intelligence methods in forecasting daily streamflow time series for sustainable water resources management. *Sustainable Cities and Society*, 64 :102562.

[6] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[7] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once : Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[8] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

[9] Haodi Zhang, Dănuţ Ovidiu Pop, Alexandrina Rogozan, and Abdelaziz Bensrhair. Accelerate high resolution image pedestrian detection with non-pedestrian area estimation. *IEEE Access*, 9 :8625–8636, 2021.

[10] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd : Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[11] Joseph Redmon and Ali Farhadi. Yolo9000 : better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271, 2017.

[12] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics : The kitti dataset. *The International Journal of Robotics Research*, 32(11) :1231–1237, 2013.

[13] Amir Rasouli, Iuliia Kotseruba, and John K Tsotsos. Agreeing to cross : How drivers and pedestrians communicate. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 264–269, 2017.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[15] Mu Zhu. Recall, precision and average precision. *Department of Statistics and Actuarial Science, University of Waterloo, Waterloo*, 2 :30, 2004.