

RESEARCH

Open Access



# Improvements for pedestrian safety application P-Minder

Chang Sun\*

\*Correspondence:  
sunch@cnu.edu.cn

First Research Institute  
of the Ministry of Public Security  
of PRC, Beijing, China

## Abstract

P-Minder is a lightweight sidewalk segmentation and obstacle detection-based approach designed for phubbers' safety, which can help phubbers to avoid obstacles and other dangers during phubbing walking. But two issues have been found in previous experiments: P-Minder undifferentiated scores the recognition results and costs high resource occupation during background operation. Therefore, this paper proposes the sector correlation model to classify the recognition and segmentation results for an accurate judgment. Then, the new application architecture design and walking detection model decrease the resource occupation. Besides, the datasets are expanded, and the models are retrained to adapt to the improved architecture. In the same datasets, P-Minder achieves a 1.7 percentage point accuracy improvement during the experiment. The experimental result proves that the segmentation model can reach 81.2% mean intersection over union. Besides, the method finally achieves 75.52% detection accuracy in the experiment with a low computing resources cost.

**Keywords:** Sidewalk dataset, Sidewalk segmentation, Obstacles detection, Phubber safety, Reliable inspection

## 1 Introduction

Phubbers [1] means a group of people who are over-reliance on mobile phones. Phubbing walking means keeping their head down and looking at screens while walking. According to the statistical results, phubbing walking is more and more common, and it leads to a rapid increase in sidewalk accidents [2, 3]. Obstacle collision and driveway phubbing walking are the primary sources of danger [4], because when people concentrate on their cell phone screens, they usually ignore the obstacles or enter the dangerous driveway.

Smartphone cameras and computing power have evolved, which makes it possible to develop vision-based sidewalk segmentation or obstacle detection applications. Mobile devices with these applications could be used in numerous applications: pedestrian safety applications, blind assistance, automatic robot driving, or communal facilities management and maintenance. However, the resources are over-occupied, and poor reliability problems still exist. Existing pedestrian safety methods based on convolutional neural networks (CNN) are usually accompanied by complex detection algorithms and massive resource consumption.

This paper aims to design a low consumption and unfailingly sidewalk segmentation-based phubber safety approach on the smartphone. Existing methods, such as Terra-Firma [5], can detect street entrance by material classification and texture recognition. The method is desirable and widely applicable, but it includes complex preprocessing and can only work on lower frame capture rates. The method reckons without the question of computing resource occupation. Inspector [6] uses the back camera of smartphones to detect a kind of particular boundary between sidewalk and road, then alerting the phubbing walking pedestrians. Besides, the researcher provides a result-checking method to improve reliability. However, this kind of tactile paving is unusual, so this method can only apply to a specific region and is difficult to extend. Besides, this method needs the image processing-based feature extraction process, which will cost computing time and influence the accuracy.

Generally, the computing resources provided by mobile devices are limited. It is necessary to lighten the detection model and simplify the detection process to reduce computing resources usage. And to improve the detection reliability, it is essential to design a result checker. This paper proposes a sidewalk segmentation and obstacle detection-based phubber safety approach named P-Minder to reduce the risk caused by driveway phubbing walking. The main idea is to detect the range of driveway and sidewalk and then alert the phubbers before they enter the street. A lightweight segment model and a walking detector are used to reduce the consumption and design an adjacent detection algorithm to improve the reliability.

Specifically, the critical contributions of this paper are as follows:

- Designing and implementing the P-Minder phubber safety approach on the smartphone, proving the feasibility of the phubber safety method on mobile devices.
- Expanding and improving the sidewalk dataset for segment model training. It contains different types of surfaces of sidewalks and common obstacles on the sidewalk.
- Providing a walking detector method for saving the computing resources and energy costs for the phubber safety method.
- Proposing an adjacent detection algorithm and a sector correlation model to post-processing identification results to improve reliability.

It is worth noting that this paper is an extended version of a conference paper [7]. A correlation model is added in this version, and the application architecture is designed, discussed, and implemented. Besides, improved experiments are implemented and discussed.

The rest of this paper is organized as follows. Section 1 describes the application directions of the segmentation-based pedestrian safety approach. Section 2 discusses the approach design and algorithms. Section 3 shows the experimental results of model testing and application testing. Section 4 provides the conclusion and future work of this paper.

## 2 Proposals

### 2.1 Approach overview

The main target of P-Minder is detecting the driveway by a smartphone back camera and reminding the phubbers before their arrival. Figure 1 shows the overall framework of the approach, in which the first step is to establish a sidewalk dataset that is used

to train a CNN model for the detection process. After that, the application contains 3 controllers for post-processing the detection results: walking detector, adjacent detector, and correlation model.

The walking detector controls the detection process to decrease the electric quantity and computing resources costs. The adjacent detector is designed for reliability improvement. The correlation model determines the weight of every detection result.

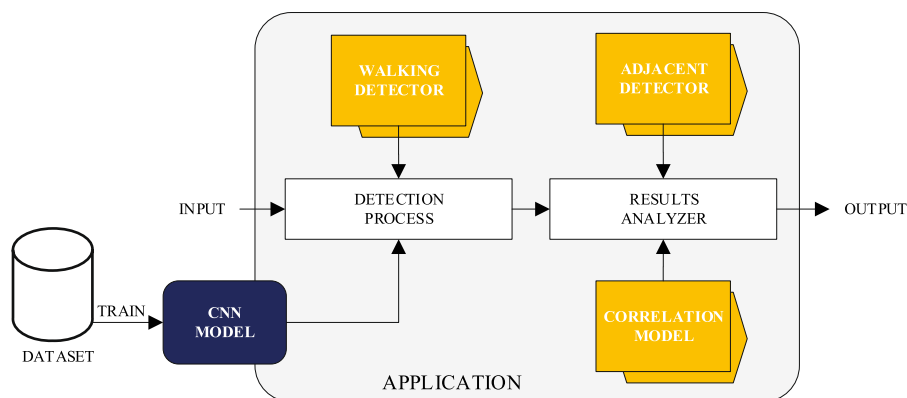
## 2.2 Segmentation model building

### 2.2.1 Sidewalk dataset

An appropriate sidewalk dataset is established [8] to train a detection model that can be used to segment the sidewalk and driveway. The sidewalk dataset fully simulates the walking environment of phubbers, and it was collected on actual sidewalks from different districts and cities which includes Beijing, Tianjin, Shanghai. The dataset was gathered by different devices and volunteers, with their peculiar phubbing walking postures to increase the robustness. Meanwhile, the influences of light or weather are ignored. Captured videos are required 1920×1080 resolution at least, and all the videos are cut into images with only two frames per second retained. Blurred images are deleted, and Labelme [9] tool is used to generate the ground truths. The dataset contains over 20 h of video sequences of raw data. Figure 2 shows the typical 8 kinds of walkways with different textures. Besides, there are 4 kinds of common sidewalk targets, including car barriers, manhole covers, stairs, and blind roads.

### 2.2.2 Model training

The model size is strictly restricted to make the model finally run on mobile devices. Selecting an appropriate network architecture to train the model is quite tricky. The lightweight network architectures gradually appeared recently, admittedly, but most of them are accompanied by the sacrifice of their accuracy. Therefore, the selection process is a trade-off between accuracy and lightness. Accordingly, 4 kinds of light network architectures are tested on the sidewalk dataset, and the results are shown in Table 1. Finally, considering MIOU, parameter quantity, operation number, and model size comprehensively, the lightest MobileNetV2-dm05+DeepLabv3 [10] [11] was selected to train the model.



**Fig. 1** Approach overview



**Fig. 2** The 8 typical kinds of textures of sidewalk in dataset

The models are trained on TensorFlow [12]. The MobileNetV2 is used as a feature extractor. The setup of the training environment is according to the experiment process of MobileNetV2 [10]. The output stride is set to 16, and the crop size is 513. The deeplabv3\_mnv2\_dm05\_pascal\_traina [13] is used as the initial checkpoint and trains the models on an 8 GPU DGX-1 deep learning server [14].

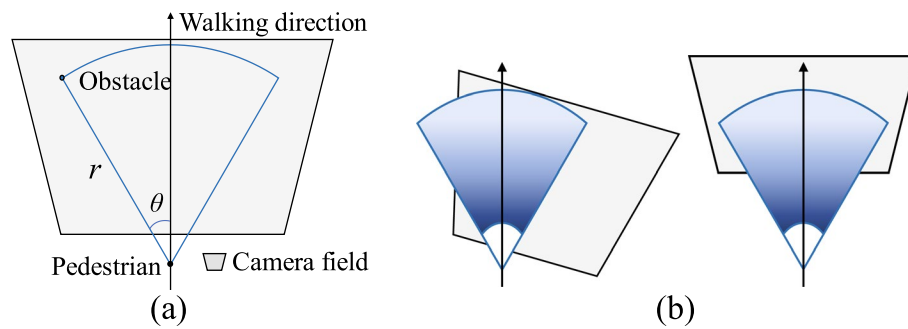
### 2.3 Sector correlation model

For P-Minder, not all areas in the camera field must be detected. It is easy to understand, for example, that obstacles which are not in the direction of pedestrian movement or are almost invisible in the camera field are unimportant. At the same time, the obstacles with different locations mean different importance. On the other hand, higher detection accuracy is required for the obstacles with a closer distance in the forward direction, while there is no need for accurate detection for far obstacles. At the same time, in the practical application process, the camera should be able to photograph obstacles with high correlation as far as possible. Therefore, a sector correlation model is defined to describe the correlation of each obstacle in the camera's field of vision.

Figure 3a shows the image description of the model. Where the gray trapezoid is the projection of the camera field on the ground, the obstacle and pedestrian are abstracted as these central points to simplify operations. The type, size, and shape of an obstacle will impact its relevance, but only the effect of its location on correlation is discussed, so it is abstracted as a central point. The  $r$  is the distance between pedestrian and obstacle, and the  $\theta$  is the angle between the direction from the pedestrian to the obstacle and the walking direction of the pedestrian. The correlation  $C$  can be defined by (1).

$$C = \frac{1}{(\lambda \cdot \theta)^2 + r^2} \quad \theta \in \left[-\frac{\pi}{3}, \frac{\pi}{3}\right] \quad r \in [1, 4] \quad (1)$$

Where the  $\lambda$  is a weight parameter, the purpose of setting it is to magnify the effect of angle on correlation. If the  $\lambda$  is set to 3, then  $C$  ranges from 4% to 100%. In the practical application of P-Minder, assuming that  $\theta$  in Fig. 3a is  $\pi/3$ , this should be an ideal case for



**Fig. 3** Sector correlation model. **a** The sector correlation region. **b** The impact of different posture changes

the distribution of camera field and sector-related areas, because almost all pedestrian-related obstacles are currently included in the camera's field.

Figure 3b shows the impact of posture changes on handheld mobile phones, the left part indicates that the direction of the mobile phone deviates from the direction of walking, and the right amount shows the angle change between the mobile phone and the ground. The shade of blue in the sectors represents the degree of correlation defined by (1). Through actual testing and calculation, for the first condition, although the camera field deviates from the direction of travel, most of the regions with high correlation are still included. Of course, this kind of deviation refers to the slight and natural variation in the actual use process, rather than deliberate and large-scale action change. However, this kind of change makes it easy for the region with the highest correlation to be outside the camera field for the second condition. Therefore, every time the P-Minder is used; the user will be helped to adjust the appropriate posture by detecting the matching degree of the camera field and the correlation sector. Once the camera field deviates from the correlation sector, the user will be alerted in the course of use.

In the current version of P-Minder, the mobile phone's posture and the pedestrian's walking direction are determined by calling the results of the gyroscope and GPS sensor in the walking detector. The distance between the camera and the obstacle is calculated by measuring the reference object in advance and the user's height of the mobile phone input. After improving the accuracy of the algorithm, the model will be applied to the auxiliary detection process, such as adjusting the classification accuracy requirements of different correlation regions, rather than just helping users adjust their postures.

## 2.4 Walking detector and adjacent detection

### 2.4.1 Walking detector

As a kind of application run on smartphones, practicality is a significant feature of P-Minder. As shown in Fig. 1, P-Minder is a continuously capturing and detecting process. It consumes smartphone power and takes up memory and CPU resources until the application is shutdown. So there is a lot of unnecessary consumption. In the actual usage scenario, it is meaningful to work for P-Minder before users arrive at the bus station and stop walking. Generally, P-Minder only needs to work on a walking status, and the walking detector is essential for it to become practical.

The deep learning or multi-sensors-based detection method is not adopted because lightweight and practical are still the first considerations in the design of the walking detector. During a phubbing walk, the return values X, Y, Z of the gravity sensor (G-sensor) have regular changes. Hence, P-Minder finally employs a G-sensor-based walking detection algorithm. The process includes wave filtering, peak detection, and threshold modification. At the beginning of P-Minder, the walking status is checked firstly; only if the walking detector returns true, the following steps will be executed.

---

**Algorithm 1:** AdjacentDetection
 

---

**Input:** R1: Segment Result of Image 1  
 R2: Segment Result of Image 2  
 T: Threshold of MIOU  
 n: Number of iterations  
**Output:** MIOU: MIOU between R1 and R2

```

1 if  $MIOU(R1, R2) \geq T$  then
2   T = Mean T (T, MIOU)
3   return MIOU = MIOU (R1, R2)
4 else if  $n \geq 3$  then
5   return MIOU = -1
6 else
7   if isPostureChangeOfMobilePhone() then
8     return
9   else
10    R1 = R2
11    R2 = Capture Image()
12    return adjacentDetection(R1, R2, T, ++n)
13  end
14 end
  
```

---

### 2.4.2 Adjacent detector

The reliability of the detection results is the most important thing for a safe approach. It is difficult to verify the reliability of the detection process because of the lack of standards and comparisons. However, in walking, due to jitter, images captured by smartphone cameras are often blurred, and blurred images can easily lead to detection errors. Consequently, this paper proposes an adjacent detection method for result evaluation and error correction. An adjacent detection algorithm can effectively avoid the anomaly detections caused by image distortion. Adjacent detection means two consecutive photographs will be taken during the image capture. Due to the interval being about one-thirtieth of a second and the walking speed of phubbers being about 1.2 ms per second, the two adjacently captured images are almost the same. Our adjacent detection method regards the segmentation results of the two images as ground truth and a segmentation result, respectively, and then computes the MIOU for the classes: sidewalk, blind road, and driveway. If the MIOU reaches the threshold, the segmentation results are considered valid. Conversely, the algorithm will first determine whether a significant change in mobile phone posture is the cause. If not, the segmentation results are considered invalid. The above processes will be repeated until the next preset capturing time. The threshold is the average of all MIOU. It will be updated after each computing. Besides, it will be saved when the application is shut down as an initial value for the next run time.

The algorithm flow of adjacent detection is shown in Algorithm 1. It is an iterative function, and the termination condition is proper segmentation and maximum iterations. The maximum iteration time is 4, and it is decided according to the segment speed of the model. If the return value of the adjacent detection algorithm is not -1, it proves the segmentation result is valid and can be trusted in the follow-up steps. And for the accurate segmentation result, if the central axis of the image top-down intersects the driveway region and the sidewalk region, respectively, P-Minder will return a positive detection. On the contrary, a negative detection will be returned.

## 2.5 B-Minder for blind assist

Our sidewalk segmentation method can also be used for blind assistance, and the blind road segment effect is tested during the experiment. The application for testing is a modified version of P-Minder and named B-Minder (Blind Minder) for the sake of this discussion. In this task, the detection target of B-Minder is to confirm the blind user is walking on the blind road. In the beginning, by default, B-Minder approximately considers the bottom edge midpoint of the image captured as the user's location. Because for the walking postures of most people, the bottom edge midpoint of the image is precisely one of the foot or slightly ahead of it. However, different users' smartphone camera fields and hand postures prove the method is unworkable.

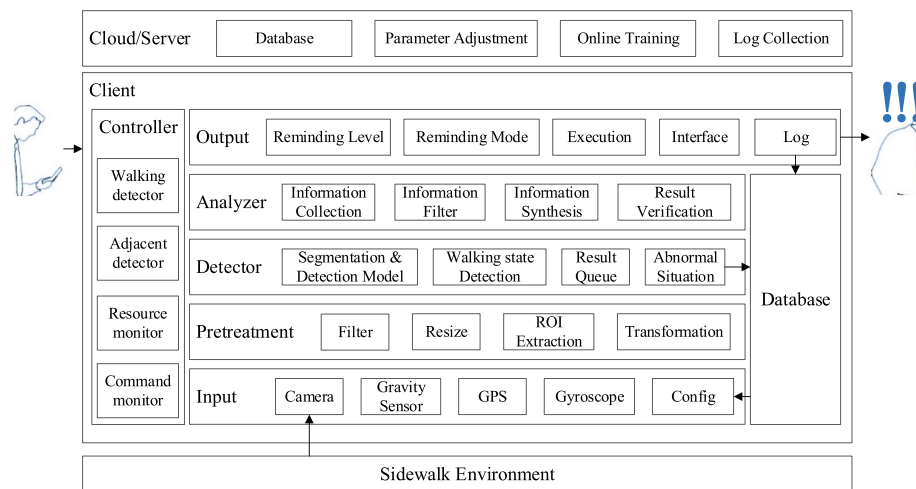
The direct method to locate the user is to recognize the user's foot. But in most cases, the foot cannot be included in the captured image. One solution is expanding the camera field through a fish-eye lens handset accessory. The images captured by the fish-eye lens ordinarily include one of the feet so that B-Minder can determine whether the blind is walking on the blind road.

Besides, there are other two functions of B-Minder. One is detecting the way ahead and warning the blind before he will leave the blind road or sidewalk. This algorithm is inherited from P-Minder. Another function is detecting the bends of the blind road and reminding in advance. B-Minder uses voice reminders to convey directives and prompts considering the practical requirements.

## 2.6 Application architecture

The application architecture of P-Minder is shown in Fig. 4. It is a typical Model-View-Controller (MVC) structure in the client. There are 5 main function models. The application gathers information from the sidewalk environment through the smartphone camera and the 3 sensors. The captured image is the central processing target. In the pretreatment model, the filter avoids the distortion error of the image caused by jitter. Besides, the image will be resized to the regular resolution for the detector, the region of interest (ROI) will be extracted, and at last, the image will be transformed into the frequency domain and grayscale.

The CNN model is used to process the image in the detection model. The information from sensors is used to detect the walking state, including the walking direction and walking speed. There are two queues in this model to store the detection results corresponding to that. The abnormal results from the detection process are stored in the database and submitted to the server for analysis along with the log information.



**Fig. 4** The application structure of P-Minder

The analyzer model collects the information, including the result queues and location. It aggregates this information to produce a comprehensive description of the environment and current user status. According to the analyzer's results, the output model determines the level and mode of the final reminder.

The controller contains the walking detector and adjacent detector, which are introduced in Sect. 2.4.2, the resource monitor, which assists in controlling the CPU and memory usage of the application, and the user's command monitor. The main work of the server is to train the abnormal results online and update the model.

### 3 Experiments and discussion

#### 3.1 Model testing

For P-Minder, the segment model determines the effect of algorithm execution. Besides, the size and parameter quantity decide whether the model can work on a smartphone. The model testing is a concernment process of method evaluation.

The model training environment and procedure are described in Sect. 2.2.2. And the crop size is  $513 \times 513$ . Four types of network architectures are trained on the sidewalk dataset in the course of this testing, and the results are shown in Table 1. MobileNetV2-dm05 in the table means the depth multiplier is set to 0.5 while training. From Table 1, all the 4 kinds are common lightweight CNN structures, and it can be seen that the MobileNetV2-dm05 model is the lightest after considering all factors.

Processing speed is another critical index in model evaluation. Even real time (30 FPS) is not necessary for a phubber safety approach, however, it still requires a relatively high processing speed, because it usually takes only 1 to 2 s from the detection target appearing in the camera field to disappearing. And in P-Minder, due to the maximum iteration times of the adjacent detection method being 4, that means there are up to 10 ( $4 \times 2 + 2$ ) times detections in a second, so at least 10 FPS is required. Figure 5a shows the segment times of the 4 types of networks. The results are gathered from the same PC, equipped

with an AMD Ryzen-5 CPU and a GTX-1060 5 G GPU. And it can be seen that the MobileNetV2-dm05 has the fastest segment speed.

Beyond that, considering the pursuit of better detection results, a higher crop size is better. The effects of different crop sizes on the segment accuracy are tested, and the results are shown in Fig. 5b. From the results, it can be found that with the increase in crop size, the growth speed of MIOU is far less than the growth speed of segment time. Thus, it is unwise to raise accuracy by enlarging the crop size for a lightweight approach.

### 3.2 Approach testing

This section shows the results of the actual scene approaching testing. Before the experiment, based on the TensorFlow android example [17], P-Minder has been implemented on an Android smartphone HONOR Magic2 TNY-AL00. The CPU model of this smartphone is Huawei Kirin 980. It contains  $2 \times 2.6$  GHz,  $2 \times 1.92$  GHz, and  $4 \times 1.8$  GHz cores. The memory size is 8 GB, and the battery capacity is 3500 mAh. The tested smartphone can score an average of 12269 in the AI Benchmark application.

#### 3.2.1 Accuracy and speed

During the testing, 2 experimenters walked on different sidewalks 10 times in 5 different districts, and each walk lasted 10 min. The accuracy of P-Minder means the proportion of positive detections. The experiment results are shown in Fig. 6, where (a) is the detection accuracy and (b) is detection time, and the abscissa axis is the code of every test. It can be illustrated that the mean accuracy of P-Minder reaches 74.19% and the average detection time is about 0.08 s. Additionally, the negative detections are mainly caused by blurred images during the testing. And the mobile phone shaking which is caused by walking is the root cause.

Considering that the actual application will face different weather and light effects, the application was tested in the same place but in different weather. The results are shown in Fig. 7c. There are 2 places selected to conduct this experiment. In the first place, there are 5 times detection processes marked from 1-1 to 1-5. Empathy in the second place. The same experiment was carried out on both sunny and rainy days. The results show that the average accuracy slightly decreased on a rainy day. The effects of light, weather, and periods are complex and challenging to quantify. For example, different weather will lead to different illumination intensities, as shown in Fig. 7b. Similarly, different periods will lead to different illumination intensities. Also, illumination intensity is affected by other factors such as object occlusion and shooting angle. Moreover, the impact of different weather on the test results is not limited to the change of illumination intensity. For example, as shown in Fig. 7a, rainy days may lead to water accumulation, and snowy days may lead to snow accumulation, which will interfere with the judgment of sidewalks. Therefore, the most straightforward solution is to collect more data in the same place, including more cases as possible, similar to the approach of the Oxford RobotCar Dataset [18], eliminating the impact of different conditions through a robust dataset.

### 3.2.2 Computing resources and energy cost

The computing resources and energy cost are counted after every test time. Figure 8a and b shows the CPU and memory occupation of two versions P-Minder in a fully activated state (the walking detector always returns true) for 90 min. The upper half is the CPU and memory cost before output image reduction and model compression. The occupancy rate of computing resources is still relatively high, and particularly, the test machine is a comparatively high-spec smartphone in the current mobile phone market. During the experiments, about 40% of memory is used to display images on the screen. The detailed memory usage is shown in Fig. 8b. And for CPU, further reducing the number of operations of the model helps to reduce the CPU usage. The lower part of Fig. 8a shows the CPU and memory occupation of the optimized application, including a reduction of output image size and model compression. The mean CPU utilization and RAM footprint comparison of the two versions P-Minder are shown in Fig. 9a. The output picture is retained to get an intuitive result. In practice, it is unnecessary because P-Minder always runs in the background. In that case, the memory footprint of P-Minder will be further drastically reduced.

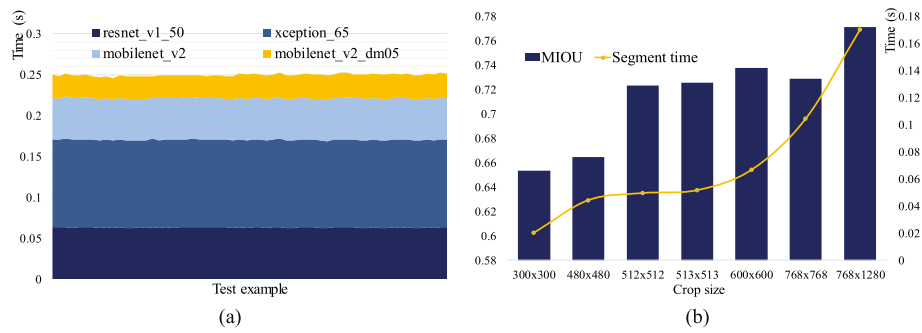
P-Minder usually runs in the background in the actual application scenarios, with another application running in the foreground. But in fact, the power consumption of P-Minder should not include the screen power consumption, and the power consumption speed of the application running and stopping states is compared and always keeps the screen lighting. The results are shown in Fig. 10. The testing time was 1.5 h, and the remaining capacity of the smartphone was recorded every minute. From the results, it can be concluded that in the actual application scenarios, the average electric consumption of P-Minder is about 451mAh with the 3.7V 3500mAh battery.

### 3.3 Comparison with inspector

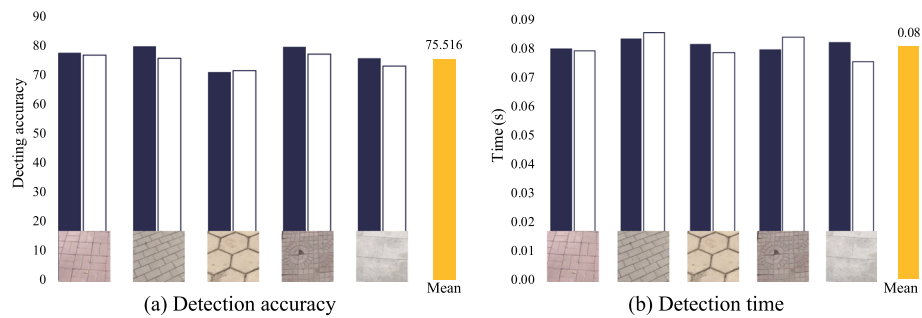
As described in Sect. 2, inspector is also a pedestrian safety application that aims to help walkers identify the sidewalk and driveway boundary. Similarly, the primary function of P-Minder is to remind pedestrians when they enter the motor lane. The accuracy shown in Fig. 6a is the mean accuracy which includes the obstacle detection results. To compare with inspector, only the results of sidewalk boundary detection of P-Minder are collected during the comparative tests. Inspector was evaluated in three real scenarios in Nanjing City, which contains two types of routes. The safe routes mean the pedestrians are walking on the sidewalk all the time. And the unsafe case mean the walkers step into the driveway from a sidewalk. Therefore, the same experimental environment was set up in the comparative experiment.

**Table 1** Models comparison

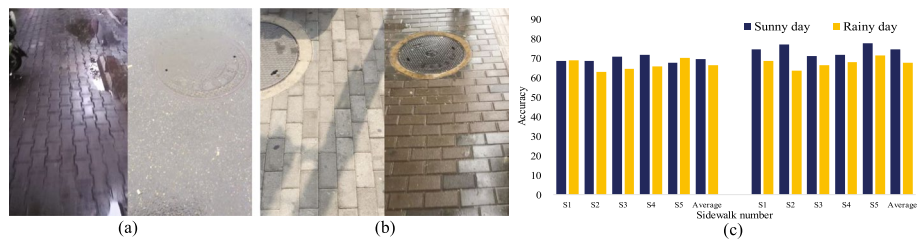
Network	Size	Params	MAdds
Xception_65 [15]	107408943	82313627	54.17B
ResnetV1_50 [16]	165615816	53523147	–
MobileNetV2 [10]	8772672	4260203	2.75B
MobileNetV2-dm05 [10]	3042951	1411339	0.88B



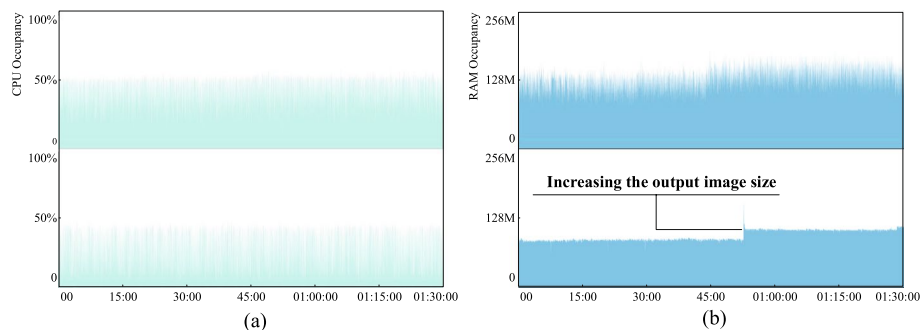
**Fig. 5** Model testing results. **a** The segment times comparison for different networks. **b** The MIOU and segment time comparisons for different crop sizes



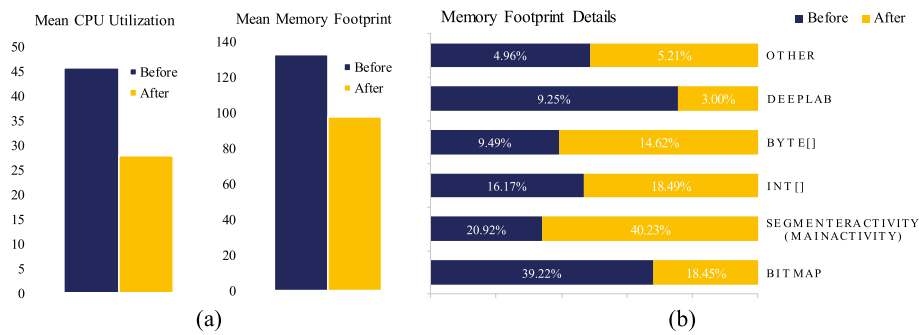
**Fig. 6** The results of the 10 times approach experiments in 5 districts. **a** The accuracy experiments results. **b** The detection speed experiments results



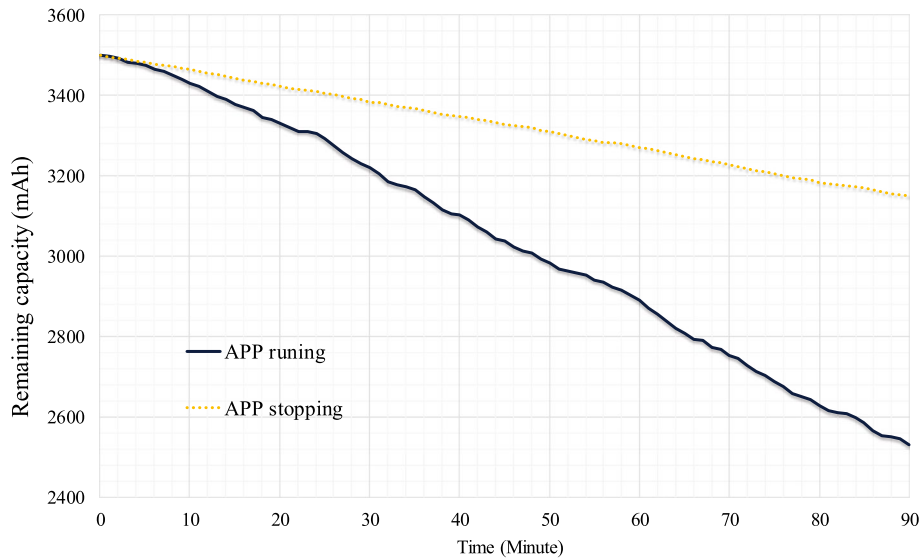
**Fig. 7** The accuracy tests in different weather. **a** The comparison of sidewalk and driveway in rainy days. **b** The comparison of sidewalk in rainy and sunny days. **c** The result of accuracy tests in different weather



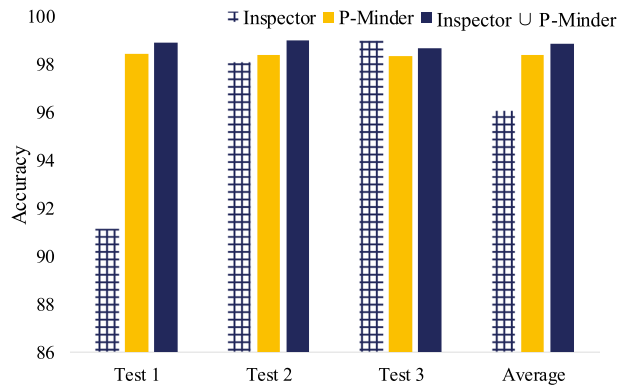
**Fig. 8** The CPU and memory occupation changes comparisons. **a** The CPU occupation comparisons. **b** The RAM occupation comparisons



**Fig. 9** The resources occupation analysis. **a** The mean CPU and memory occupation comparisons. **b** The detailed memory distribution comparisons



**Fig. 10** Tendency of remaining capacity of smartphone with time



**Fig. 11** Comparison of P-Minder and inspector

The main principle of inspector is to detect the tactile paving by a classifier, where P-Minder identifies the sidewalk and driveway directly by the CNN model and determines the boundary by segmenting two regions. In P-Minder, boundary detection also can be used according to Inspector. Thus, the performance of the cross-over method is discussed in comparative experiments.

The experimental results are shown in Fig. 11. Because only the ability to distinguish between sidewalks and motor lanes is tested, the accuracy of P-Minder improves obviously. This is because, in the sidewalk dataset, there are far more samples of sidewalks and driveways than obstacles. The experimental results show that the method of automatic feature extraction based on CNN is more stable and has higher average accuracy than the method of recognizing tactile paving. In particular, P-Minder behaves more reliable in scenes with uncertain color and texture, even in the absence of such paving.

## 4 Conclusions

In this paper, a sidewalk detection approach, which is called P-Minder, is proposed to ensure phubbers safety during their phubbing walking. P-Minder involves a precise and lightweight sidewalk segment model, which supports the function of sidewalk and driveway boundary detection. A unique sidewalk dataset is established for model training. Besides, this paper designs a detection procedure that includes a walking detector to save energy and computing resources cost and an adjacent detection method to do result evaluation. Extensive experiments prove that P-Minder achieves 75.52% detection accuracy when used in practice, and it can run smoothly on the test mobile phone.

To further improve the accuracy and applicability of the segmentation model, it is no doubt that the scale of sidewalk dataset should be further expanded. Additionally, through the experiments, it is difficult to determine the location and size of the obstacles because the edges of obstacle areas in segmentation results are usually incomplete. Exploring the solution to this problem is a significant direction for the future research.

Additionally, for the P-Minder application, if it is wanted to be applied to the safety field, the accuracy of detection must be improved as much as possible. Therefore, in the future work, it is necessary to design an algorithm that can comprehensively analyze the available information, including location information, sound information, the user's information captured by the front-end camera, and the model test results, so that P-Minder can provide more accurate, more diverse, and timely dangerous reminders for pedestrians.

### Author Contributions

The central ideas were formulated by Chang Sun. Chang Sun is the major contributor to writing the manuscript. All authors read and approved the final manuscript.

### Availability of data and materials

All data used to support the findings of this study are included within the article. The code and models used to support the findings of this study are available from the corresponding author upon request.

### Declarations

#### Competing interests

The authors declare that they have no competing interests.

Received: 15 September 2021 Accepted: 17 October 2022

Published online: 28 October 2022

## References

1. Urban Dictionary: phubber (2022). <https://www.urbandictionary.com/define.php?term=phubber>. Accessed 12 Apr 2022
2. NHTSA: Pedestrians (2022). <https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/812493>. Accessed 12 Apr 2022
3. GHSA: Pedestrian traffic fatalities by state 2017 preliminary data (2022). <https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/812493>. Accessed 12 Apr 2022
4. S. Yoshiki, H. Tatsumi, K. Tsutsumi, T. Miyazaki, T. Fujiki, Effects of smartphone use on behavior while walking. *Urban Reg. Plan. Rev.* **4**, 138–150 (2017)
5. S. Jain, M. Gruteser, Recognizing textures with mobile cameras for pedestrian safety applications. *IEEE Trans. Mob. Comput.* **18**(8), 1911–1923 (2019)
6. M. Tang, C.T. Nguyen, X. Wang, S. Lu, An efficient walking safety service for distracted mobile users. In: 2016 IEEE 13th International Conference on Mobile Ad Hoc and Sensor Systems (MASS) (2017)
7. C. Sun, J. Su, Z. Shi, Y. Guan, P-minder: A cnn based sidewalk segmentation approach for phubber safety applications. In: 2019 IEEE International Conference on Image Processing (ICIP) (2019)
8. C. Sun, J. Su, W. Ren, Y. Guan, L. Wu, Wide-view sidewalk dataset based pedestrian safety application. *IEEE Access* **PP**(99), 1 (2019)
9. LabelMe: Image polygonal annotation with python (2022). <https://github.com/wkentaro/labelme>. Accessed 12 Apr 2022
10. M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L.C. Chen, Mobilenetv2: inverted residuals and linear bottlenecks. *IEEE* (2018)
11. L.C. Chen, G. Papandreou, F. Schroff, H. Adam, Rethinking atrous convolution for semantic image segmentation (2017)
12. M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G.S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, X. Zheng, TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. Software available from tensorflow.org (2015). <https://www.tensorflow.org/>
13. TensorFlow DeepLab-Model Zoo: MobileNetV2-dm05-coco-voc-trainaug (2022). <http://download.tensorflow.org/models/>. Accessed 12 Apr 2022
14. NVIDIA DGX-1: Deep learning server for AI research (2022). <https://www.nvidia.com/en-us/data-center/dgx-1/>. Accessed 12 Apr 2022
15. F. Chollet, Xception: deep learning with depthwise separable convolutions. *IEEE* (2017)
16. K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition. *IEEE* (2016)
17. TensorFlow: Tensorflow android camera demo (2022). <https://github.com/tensorflow/tensorflow/tree/master/tensorflow/examples/android>. Accessed 12 Apr 2022
18. W. Maddern, G. Pascoe, C. Linegar, P. Newman, 1 year, 1000 km: the oxford robotcar dataset. *Int. J. Robot. Res.* 0278364916679498 (2016)

## Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Submit your manuscript to a SpringerOpen<sup>®</sup> journal and benefit from:**

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

---

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)