VNUHCM Journal of Engineering and Technology 2026, 9(2):2992-3001
Open Access Full Text Article

Research Article

Application of drone in detecting drain cover damages on road

Nguyen Dinh Hoang[1], Ngo Khanh Hieu[2,*]

ABSTRACT

Vietnam is a country where motorbikes are the primary mode of transportation, with millions of people commuting daily on two wheels. However, the safety of these road users is significantly threatened by damaged or missing drain covers. During rainy seasons, wet and slippery road conditions combined with height differences between the drain cover and the road surface often lead to serious accidents. Pedestrians, cyclists, and motorcyclists near the roadside are especially vulnerable to such hazards. In recent years, Ho Chi Minh City and many other regions in Vietnam have recorded numerous injuries and fatalities resulting from these damaged infrastructures. Conventional inspection methods for detecting drain cover failures are mainly manual, requiring considerable time, manpower, and cost. These methods are not efficient enough to ensure timely detection and repair. Therefore, developing an automated and intelligent system for early identification of damaged drain covers is essential to improving urban road safety. This study proposes an enhanced detection approach based on computer vision and unmanned aerial vehicle technology. Specifically, the YOLOv8 deep learning model is employed to identify and classify damaged drain covers from aerial images. A new dataset is constructed and annotated to train and evaluate the model under various real-world conditions. The proposed system enables efficient large-scale monitoring, minimizes human effort, and supports authorities in taking prompt maintenance actions. The results of this research are expected to contribute to safer transportation environments, reduce accident risks, and demonstrate the potential of integrating artificial intelligence and unmanned aerial vehicles in urban infrastructure management. In the future, this project will be further developed by expanding the dataset, improving detection accuracy in diverse lighting and weather conditions, and integrating real-time processing to support continuous monitoring and smart city applications.

Key words: Terms—drain cover, drone, YOLOv8

[1]*Terrapinn Pte Ltd, Singapore*

[2]*Key Lab. for Internal Combustion Engine, Ho Chi Minh City University of Technology, VNU-HCM, Vietnam*

Correspondence

Ngo Khanh Hieu, Key Lab. for Internal Combustion Engine, Ho Chi Minh City University of Technology, VNU-HCM, Vietnam

Email: ngokhanhhieu@hcmut.edu.vn

History

- Received: 13-02-2025
- Revised: 24-3-2025
- Accepted: 07-11-2025
- Published Online: 10-07-2026

DOI : https://doi.org/10.32508/vnuhcmj-et.v9i3.1488

Check for updates

INTRODUCTION

Adamaged drain cover (DDC) is often a drain cover with a cracked cover caused by traffic conditions or weather. This type of damage can pose risks to road users (especially cyclists and motorcyclists), even resulting in death, so it needs to be detected and rectified promptly. However, the maintenance process typically begins with inspecting and receiving reports of damage from the public or through periodic inspections, followed by planning for rectification. This process is both time-consuming and inefficient. Moreover, inspectors may also risk injury from DDC that are difficult to detect. Therefore, to shorten the process and protect inspectors, the authors propose using unmanned aerial vehicles (UAV) to automatically detect DDC.

This research is the first step of a project aimed at detecting DDC, one of the most dangerous defects from aerial imagery. The authors propose using UAVs to collect high-resolution images of roads with DDCs of different shapes, in addition to creating hypothetical scenes of common cases of DDCs that are up to 90% similar in reality. Then create a custom dataset to train the YOLOv8 model to detect DDC or use image processing methods in Matlab to quickly identify drain cover cracks. The images obtained and the trained model can be used in future studies. Additionally, we also developed a simple tool in which information about DDCs detected from images such as 2D size and location are extracted, summarized on a table, and presented on a map.

This study proposes a method for detecting damaged drain covers on roadways using the YOLOv8 model combined with image processing in Matlab. The novelty of the research includes:

- *Application of YOLOv8 in Damaged Drain Cover Detection*: YOLOv8 is one of the most advanced object detection models, but there has been limited research on its application to detecting and assessing the condition of drain covers on roadways. This study leverages YOLOv8 to automatically detect damages of drain covers in real-world images, facilitating faster and

Cite this article : Dinh Hoang N, Khanh Hieu N. **Application of drone in detecting drain cover damages on road**. *VNUHCM J. Eng. Technol.* 2026; 9(3):2992-3001.

more accurate infrastructure inspection and maintenance.

- *Optimization of Image Processing in Matlab*: In addition to using YOLOv8, the study incorporates various image processing techniques in Matlab to enhance input data quality, reduce noise, and improve the accuracy of detecting drain cover conditions. The combination of deep learning-based object detection and traditional image processing enhances the system's effectiveness.

- *High Performance and Practical Applicability*: The proposed method is evaluated in terms of accuracy and processing speed, demonstrating its potential for real-world applications, especially in automating infrastructure inspection and maintenance. Compared to traditional methods, this approach reduces time, labor, and improves the efficiency of fault detection.

- *Look up faster on map*: Users can look up the surveyed drain covers quickly and accurately on map to facilitate searching and repairing.

Although data is collected using UAV, the primary contribution of this research lies in the development and optimization of the data analysis process. Therefore, the study focuses on applying YOLOv8 and image processing in Matlab to accurately and efficiently detect damaged drain covers, also showing all the surveyed drain covers on map.

METHODOLOGY

Workflow overview

There are 3 main steps in this process. It includes training to identify and analyze the results – which is based on the dataset we have collected by drone. There is also an additional step to perform Image Processing in Matlab to identify cracks more quickly and accurately. The last step is to locate all the surveyed drain covers on both created map and satellite map. This workflow is straightforward, and the people can widely implement it with the necessary tools.

Analyze camera angle and altitude to obtain clear images

Currently on the internet there is no dataset taken by drone about damaged drain cover. So, we have to make our own dataset. We selected the DJI Mini 2 drone (**Figure 1**) for this task due to its compact size and high flexibility. And dataset is an extremely important part of deep learning, so we follow these requirements:

- *Camera angle*: 90 degrees, take 4 pictures/drain cover at 4 positions, which means that the 4 edges of the drain cover (also gully grating) are parallel to the 4 edges of the image. No vehicles stepping on drain covers when taking photos.

Table 1: PARAMETERS OF DJI MINI 2

Technical specifications	Values
Take-off weight	< 249g
Max flight time	31 minutes (measured while flying at 4.7 m/s in windless condition)
Camera sensor	CMOS: 1/2.3 inch (6.26 x 4.70 mm
Effective pixels	12 MP (4000 x 3000 pixels)
Focal length	4 mm
Max flight distance	15.7 km
Operating temperature	-10 ^{o}C to 40 ^{o}C

Figure 1: DJI Mini 2 drone

- *Altitude*: Altitude calculation relies on camera parameters, the minimum detectable pixel size, and criteria for evaluating drain cover damage.

The specifications of the DJI Mini 2 drone are provided in Table 1 by the DJI company [1]. With flight altitude H, the focal length f, the sensor width Sw (the value of it is 6.26 mm in Table 1) and the image width Imw (the value of it is 4000 pixels in Table 1), we have the formula used to calculate required altitude is ground sampling distance GSD:

$$GSD = \frac{H\ (cm)}{f\ (mm)} \times \frac{Sw\ (mm)\ (or\ Sensor\ height\ (mm))}{IMw\ (\ pixel)\ (or\ Image\ height\ (pixel)} \quad (1)$$

Ho Chi Minh City Urban Drainage Company Ltd (UDC) provided that if the length and width of the crack on a drain cover are both $\geq$ 1 cm (or the crack area $\geq$ 1 cm2), that drain cover is seriously damaged and requiring replacement.

Referring to the calculation of the required height when flying a drone, we referred to the calculation of

L. N. K. Ngan in the topic of identifying potholes on the road [2]. Based on the drain cover damage risk assessment of UDC above and we have the minimum object size that can be detected is 2 pixels, the required altitude that 1 cm of crack can be detected is 13 meters. But the general rule for image quality is that at the same image resolution, the lower the drone hovers, the more detailed the image is. So based on the maximum height of vehicles running on the road, we chose altitude capturing images from 5 meters - 6 meters.

Dataset and Training model

Gully-gratings are often next to drain covers (especially on sidewalks) and are different in shape and colour from drain covers. And their damage (if any) can be a cause of traffic accidents. However, there are no reports of gully-grating's damage from the public. Therefore, this study will focus on damaged drain covers (specifically the cracks), and in the future, we will develop this dataset to identify missing drain covers and missing gully-grating cases.

Figure 2: An image of drain cover taken by drone

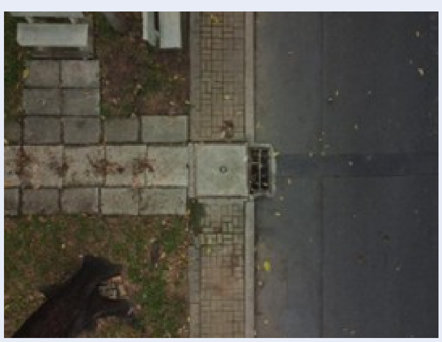

Figure 3: Another image of drain cover with a gully-grating next to it

After capturing drain covers with a drone (such as Figure 2 and Figure 3 above), we label them like in Figure 4. This turns the captured images into data stored in a dataset.

Figure 4: Example of labelling imag

We referred to article [3] to apply the latest version of YOLOv8 at that time to apply object detection, and we also referred to the steps to apply it most effectively in article [4].

Total images in the dataset: 516 images, in which containing "drain covers" class, "gully grating" class, and "cracks" class (all are cracks of drain covers).

Of those 516 images, the number of images for training is 78%, the number of images for validation is 19% and number of images for testing is 3%. We also used the augmentation (Flip Horizontal) to create new data, so that the model can learn more, thereby increasing the model's performance.

We chose the strongest model of YOLO, YOLOv8. YOLOv8 has five types of models in total, including: nano (n), small (s), medium (m), large (l), and extra-large (x). The small version YOLOv8s takes less time while the biggest version YOLOv8x takes the longest time but is the most accurate. We decided to use the strongest model YOLOv8x with epoch = 100, batch size = 16, image size = 640.

RESULTS AND DISCUSSION

After 1.613 hours, the training reached the optimal value as shown below:

Table 2: PARAMETERS AFTER TRAINING

Parameters	Values
Box – precision	0.97
Box – recall	0.975
Box – mAP@50	0.986
Box – mAP@50-95	0.914

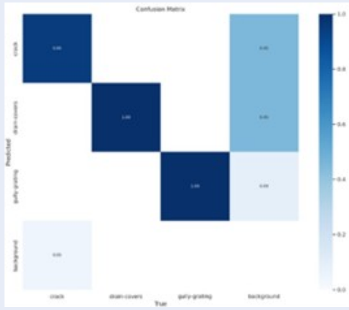

Figure 5: Confusion matrix

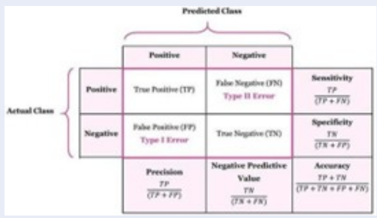

Figure 6: The meanings in confusion matrix

Figure 6 shows all the meanings in the confusion matrix. As shown in Figure 5, the model can detect 100% of real drain covers, 100% real gully-gratings and 95% of real crack, which means that the model almost accurately recognizes drain covers, gully-gratings and cracks. Since YOLOv8 is a one-stage object detection model, it identifies drain covers, gully-gratings, and cracks simultaneously in a single inference pass. The model processes the entire image at once and detects multiple objects at the same time, rather than handling each task separately. The model primarily identifies drain covers and gully-gratings based on their distinctive shape and color, while cracks are detected based on their shape, colour and contrast with the drain cover's surface. However, due to the lack of diversity in the training data, the background (such as pavement stone) may sometimes be mistaken for a drain cover, gully- grating; the image noise due to tree shadows is the main cause of being mistaken for cracks. In Table 2, the training results indicated that the model achieved precision, recall, and F1 scores of 0.97, 0.975, and 0.97, respectively. These metrics are satisfactory, demonstrating that the model is reasonably effective at detecting drain covers, their cracks and gully-gratings. Additionally, the mAP@50 and mAP@50-95 scores are 0.986 and 0.914, respectively, which are both high and acceptable. Therefore, users can rely on this model with confidence.

TESTING THE MODEL

Testing dataset

After training, the purpose of this part is to check if the model correctly predicts drain covers, their cracks and gully-grating. Test data must be unseen data. There was a problem here: we could not know where there was a damaged drain cover so we could collect data for testing the model. And if we went to places having damaged drain covers that people reported to the authorities on the internet, it was extremely impossible because we did not know whether these drain covers had been fixed or not.

Therefore, the most optimal way is that we chose any drain cover and made hypothetical cases of serious cracks 90% similar to real life using a black marker (or black cloth). And we managed all the sizes of these cracks to compare with the results processed from the model in the following section.

After consulting on the internet about damaged drain covers that people reported to the authorities and in the data I had collected, I saw that there were 3 common cases of crack of drain covers in real life:

Figure 7: A drain cover is cracked at the corner in real life

Figure 8: A drain cover has a crack in the middle of it in real life

Figure 9: A drain cover has a line crack in our campus in Thu Duc city, Vietnam

From there, we created the corresponding hypothetical cases based on Figure 7, **Figure 8** and Figure 9 above:

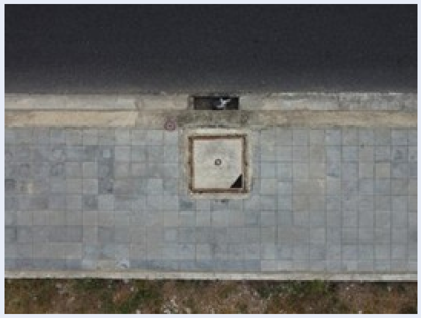

Figure 10: An example of hypothetical case of a drain cover being cracked at the corner

Figure 11: An example of hypothetical case of a drain cover being cracked in the middle of it

Figure 12: An example of hypothetical case of drain cover having a line crack

Detection result images

We used the hypothetical cases Figure 10, Figure 11 and Figure 12 above to test the model. Then we have the results as shown below:

Figure 13: Detection result 1

Figure 14: Detection result 2

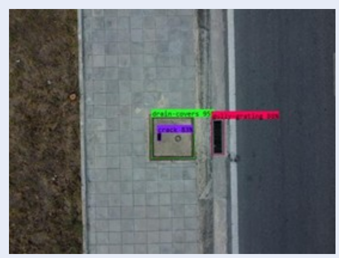

Figure 15: Detection result 3

Figure 16: Detection result 4

From Figure 13, **Figure 14, Figure 15** and Figure 16, we see that the model detects drain covers, their cracks and gully-gratings accurately with high confident score (more than 70%). The 4 edges of the bounding box almost coincide with the 4 edges of the drain covers and the gully- grating.

Detection result analysis

We have referred to the paper[5] on how to evaluate the accuracy between the calculated size and their actual size. However, the error is quite high because the shooting angles of smartphones are not direct (means not 90 degrees). Calculating the sizes based on bounding boxes is the optimal solution because when recognizing, the 4 edges of the bounding boxes almost cover the drain covers, gully-gratings and cracks. From there, the error will be significantly reduced.

Take the Figure 14 for example. After detecting it, the model will export a txt file of it. The file has YOLO format:

```
0 0.501667 0.489778 0.028667 0.032 0.916237
1 0.501667 0.495111 0.127333 0.170667 0.960410
```

Figure 17: The txt file of Figure 1

On each line, the values are respectively:

[Object – class – ID] [X_center] [Y_center] [BB_width] [BB_height] [Confidence_score]

Figure 18: The YOLO format in txt file

In which:
- *BB* means Bounding box.
- *[Object – class – ID]* is the name of label class, formatted as an integer (starting from 0). Each class of the object corresponds to a unique integer. In this project, 0 for "crack" class; 1 for "drain cover" class and 2 for "gully – grating" class.
- *[X_center]* is the center coordinate of the bounding box on the x axis.
- *[Y_center]* is the center coordinate of the bounding box on the y axis.
- *[BB_width]* is the width of the bounding box.
- [*BB_height*] is the height of the bounding box.
- [*Confidence_score*] is the confidence score of the bounding box.

From the above values in Figure 17 and the meanings of them in Figure 18, we can calculate the actual sizes of that drain cover, its crack and evaluate the accuracy of the calculated area by using the formulas below and **formula (1)**:

$$width\_real = width\_image \times width\_BB \times GSD \quad (2)$$

$$height\_real = height\_image \times height\_BB \times GSD \quad (3)$$

$$area\_real = width\_real \times height\_real \quad (4)$$

We programmed Python to aggregate the txt files as shown in Figure 19 below. Then we have programmed to apply **formula (1), formula (2), formula (3)** and **formula (4)** to aggregate the results. We get:

Class ID	X Center	Y Center	BB Width	BB Height	Confidence Score	File Name
0	0.458667	0.517778	0.019667	0.032889	0.825735	DJI_0098.txt
1	0.501333	0.520889	0.130667	0.168889	0.954374	DJI_0098.txt
2	0.638667	0.517778	0.037333	0.144889	0.884295	DJI_0098.txt
0	0.468	0.480444	0.046	0.015111	0.744995	DJI_0123.txt
0	0.532333	0.465778	0.052667	0.021333	0.81711	DJI_0123.txt
1	0.502667	0.475556	0.130667	0.172444	0.961241	DJI_0123.txt
1	0.506333	0.506222	0.123333	0.164444	0.958319	DJI_0188.txt
0	0.588333	0.568778	0.03	0.048	0.905574	DJI_0171.txt
1	0.515	0.5168889	0.128667	0.173333	0.952114	DJI_0171.txt
0	0.501667	0.489778	0.028667	0.032	0.916237	DJI_0165.txt
1	0.501667	0.495111	0.127333	0.170667	0.96041	DJI_0165.txt
0	0.504667	0.544444	0.032	0.034667	0.85583	DJI_0212.txt
1	0.508667	0.496444	0.129667	0.168778	0.961587	DJI_0212.txt
0	0.458667	0.517778	0.019653	0.03288	0.825735	DJI_0153.txt
1	0.501333	0.520889	0.130637	0.168889	0.954374	DJI_0153.txt

Figure 19: Summary table of values in text files match with detected drain covers

File Name	Area of Drain Cover (cm^2)	Sum of Area of Crack (cm^2)	Classify & Comment
DJI_0098.txt	10134.381298060063	181.11015658506406	The crack is serious, this drain cover must be replaced with a new one
DJI_0123.txt	10347.703218884744	835.17858137688578	The crack is serious, this drain cover must be replaced with a new one
DJI_0188.txt	9313.888857665256	0.0	No crack, this drain cover is still good
DJI_0171.txt	10241.848579835433	881.29075	The crack is serious, this drain cover must be replaced with a new one
DJI_0165.txt	9879.768833328793	421.27158845	The crack is serious, this drain cover must be replaced with a new one
DJI_0212.txt	9875.858387729129	509.44388845069906	The crack is serious, this drain cover must be replaced with a new one
DJI_0153.txt	10130.854685095577	180.85467854762486	The crack is serious, this drain cover must be replaced with a new one

Figure 20: Summary table of area values of drain covers, their cracks, classification and comment

Following the detection process, YOLOv8 provided the text files for each detected drain cover. These text files contained the parameters of the bounding box. With the bounding box size information, the users could readily compute the drain covers area, their crack areas and from there classify the level of damage of drain covers based on UDC criteria.

From area values in Figure 20, upon comparison with real-life measurements by tape ruler in Table 3.

The findings indicate that the average error for drain cover area falls within the less than 5% range (the size in real life of the drain cover that we used to check this model is 90 cm x 90 cm). Meanwhile, the average error of crack is less than 30%. These values are acceptable. In which, most of the after detected areas have larger values than the real area. Especially the area of line crack after detection has extremely large

Table 3: THE REAL - LIFE MEASUREMENTS OF CRACKS

File Name	Area Values (cm2)
DJI_0098.txt	160
DJI_0123.txt	450
DJI_0188.txt	0
DJI_0171.txt	660
DJI_0165.txt	408
DJI_0212.txt	408
DJI_0153.txt	160

error (over 80%). This is because the bounding box is not close to the crack and has a residual. Except for DJI_0188.txt which is a case with no cracks in real life, all the remaining cracks in real life are seriously damaged (due to criteria of UDC) and the model has classified them accurately in Figure 20.

Image processing in Matlab

Since the drone data collected in this paper consists of images, we referred to previous studies to find another solution for identifying cracks in drain covers, namely the Image Processing application[6,7]. We chose to use Image Processing in Matlab because it provides the necessary tools to perform.

Another way to identify cracks of each drain cover is to use Image Processing in Matlab. For example, we have a drain cover that is in the best condition as Figure 21 below:

Figure 21: An example of a drain cover in the best condition

After putting this image into the YOLOv8 detection model, the model will output a text file including all parameters of the drain cover's bounding box (as we were saying in **Section 4.3**). We use that file to do the following steps:

- *Step 1:* The Matlab code only reads the drain cover's bounding box information from a text file and overlay the bounding box mask on the image:

Figure 22: The drain cover is overlayed the bounding box mask

- *Step 2*: From Figure 22 above, the Matlab code creates a binary mask using the bounding box coordinates:

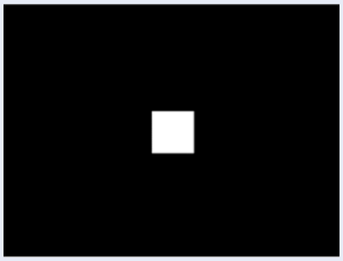

Figure 23: The binary mask of the drain cover

- *Step 3*: From Figure 23 above, the Matlab code applies the mask to the image to keep only the masked region:

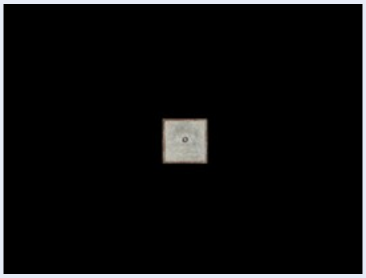

Figure 24: Keep only the drain cover on the image

- *Step 4:* Then the Matlab code converts the masked image (Figure 24) to grayscale by using **rgb2gray** function:

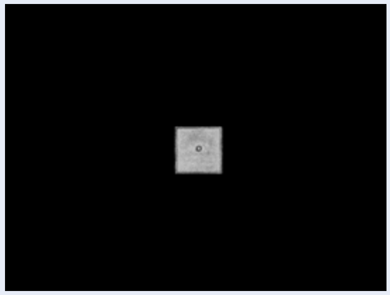

Figure 25: The grayscale masked image

- *Step 5*: Finally, from Figure 25, the Matlab code computes an automatic threshold value for the grayscale image by using **graythresh** function.

We have Figure 10, Figure 11 and Figure 12 above which are the hypothetical cases of damage to the drain cover after a period of time. Then we do the following steps:

Step 1, step 2, step 3, step 4: similar to above.

Step 5: The Matlab code converts the grayscale image to a binary image using the found threshold value. Finally, we will see all the drain cover irregularities, such as:

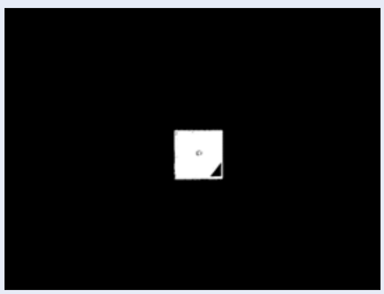

Figure 26: The result of the corner crack

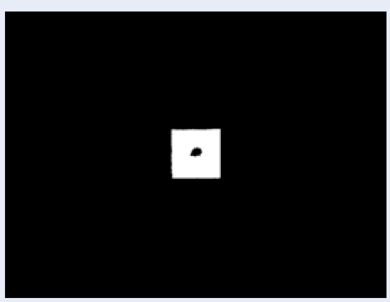

Figure 27: The result of the middle crack

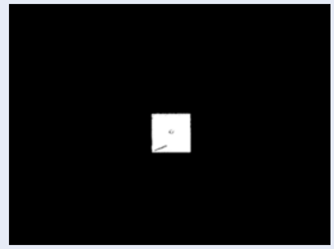

Figure 28: The result of the line crack

Cracks are clearly visible in Figure 26, Figure 27, Figure 28 and there is little noise in the data.

The idea of this method is that workers will collect all thresold values of each drain cover in the best condition and store them in the computer. And every 6 months, workers will collect data of those drain covers and use the corresponding thresold values to find abnormalities in the drain cover.

The strength of this method is that any abnormalities in the drain cover can be identified. Although there is a level of noise, the calculated crack area will be certainly most similar to reality that is, the error is very small when compared to the Machine Learning method in **Section 4.3**.

The weakness of this method up to now is that we have to manually input each image, not automatically multiple images at the same time as the Machine Learning method. We will definitely improve in the future.

LOCATING SURVEYED DRAIN COVERS ON MAP

Thanks to the drone's GPS information, each photo taken from the drone has a specific latitude and longitude. We will separate the longitude and latitude of these images and summarize them in a file, then create our own map using Python programming like in Figure 29 or embed it in Google Earth software like in Figure 30. So, if any drain cover has a problem, we can find their location easily and accurately.

CONCLUSIONS

In summary, the research has built a model to detect drain covers, their cracks and gully-gratings from images taken by drones. This research helps initially support the process of storing drain cover maintenance records on a digital platform. With the help of drones, drain cover inspections can be conducted more effectively and quickly. Both methods of detecting damage to drain covers have advantages and disadvantages, which users can consider using accordingly. If there is any damage, users can find their location easily and

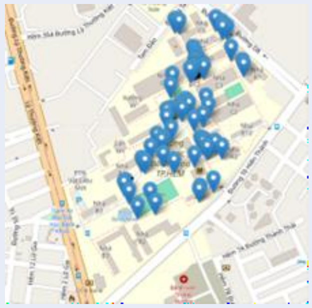

Figure 29: Presenting the drain covers location on created map

Figure 30: Presenting the drain covers location on satellite map

accurately. And we will fix the drawbacks in the future.

And it should be noted: When manually controlling the drone to collect drain cover data, humans should be close to handle it promptly because there will be drain covers under the shade of trees or electric poles. In addition, because drain covers are mostly concentrated in densely populated areas with many moving vehicles, human presence is necessary to control and prevent accidents. To sum up, in routine inspection, users need to stay close to the drones to control the drones, to keep the drones from being harmed, and to keep the drones from causing possible damage to other people or vehicles.

ACKNOWLEDGMENT

This research is funded by VNU-HCM under grant number C2022-20-07. We acknowledge the support of time and facilities from Ho Chi Minh City University of Technology (HCMUT), VNU- HCM for this study.

LIST OF ABBREVIATIONS

DC: nắp cống – drain cover

DDC: nắp cống hỏng – damaged drain cover

GSD: độ phân giải mặt đất – ground sampling distance

UAV: máy bay không người lái

YOLOv8: mô hình trí tuệ nhân tạo YOLO phiên bản thứ 8

CONFLICTS OF INTEREST STATEMENT

The authors certify that they have NO affiliations with or involvement in any organization or entity with financial interest, or non-financial interest in the subject matter or materials discussed in this manuscript.

AUTHOR CONTRIBUTION STATEMENT

Nguyen Dinh Hoang participated in data interpretation, data collection for detecting drain cover damages based on drones and manuscript writing.

Assoc.Prof.Dr Ngo Khanh Hieu proposed and analyzed the model detecting drain cover damages based on drones, and reviewed the article.

REFERENCES

1. D. Company, "DJI Mini 2," DJI, 2022.; 2022. Available from: https://www.dji.com/global/mini-2/specs.
2. Ngan LNK. "Autonomous drone-based imaging system for detection of potholes in rural road", in WCEAM, Ho Chi Minh city; 2023.
3. Rasheed, Fahad A, Zarkoosh M. Autonomous drone-based imaging system for detection of potholes in rural road. Journal of Real-Time Image Processing. 2024;22.
4. Pereira V, Tamura S, Hayamizu S, Fukai H. A deep learning-based approach for road pothole detection in timor leste. In: 2018 IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI). vol. 00; 2018. p. 279–284. Available from: https://doi.org/10.1109/SOLI.2018.8476795.
5. Allouch A, Kouba A, Abbes T, Ammar A. RoadSense: Smartphone Application to Estimate Road Conditions Using Accelerometer and Gyroscope. IEEE Sensors Journal. 2017;17(13):4231–8. Available from: https://doi.org/10.1109/JSEN.2017.2702739.
6. Sekeroglu B, Tuncal K. Image Processing in Unmanned Aerial Vehicles. Unmanned Aerial Vehicles in Smart Cities; 2020. p. 167–79.
7. Muhammad IM, Mohamed TH, Shahar FS, Nayak SY, Lukaszewicz A, Nowakowski M, et al. Image Processing Systems forUnmanned Aerial Vehicle: State-of-the-art. Preprintsorg. 2023;.

VNUHCM Journal of Engineering and Technology 2026, 9(2):2992-3001

Open Access Full Text Article

Bài nghiên cứu

Ứng dụng thiết bị bay không người lái trong nhận diện hư hỏng nắp cống thoát nước trên đường giao thông

Nguyễn Đình Hoàng[1], Ngô Khánh Hiếu[2,*]

Use your smartphone to scan this QR code and download this article

[1]*Công ty Đa Quốc gia Terrapinn, Singapor*

[2]*Phòng thí nghiệm trọng điểm ĐHQG-HCM Động cơ đốt trong, Trường Đại học Bách Khoa, ĐHQG-HCM, Việt Nam*

Liên hệ

Ngô Khánh Hiếu, Phòng thí nghiệm trọng điểm ĐHQG-HCM Động cơ đốt trong, Trường Đại học Bách Khoa, ĐHQG-HCM, Việt Nam

Email: ngokhanhhieu@hcmut.edu.vn

Lịch sử

- Ngày nhận: 13-02-2025
- Ngày sửa đổi: 24-03-2025
- Ngày chấp nhận: 07-11-2025
- Ngày đăng: 10-07-2026

DOI : https://doi.org/10.32508/vnuhcmj-et.v9i3.1488

TÓM TẮT

Việt Nam là quốc gia mà xe máy là phương tiện giao thông chủ yếu, với hàng triệu người di chuyển hằng ngày bằng hai bánh. Tuy nhiên, sự an toàn của những người tham gia giao thông này đang bị đe dọa nghiêm trọng bởi tình trạng nắp cống bị hư hỏng hoặc mất. Trong mùa mưa, điều kiện đường trơn trượt cùng với sự chênh lệch độ cao giữa nắp cống và mặt đường thường dẫn đến nhiều vụ tai nạn nghiêm trọng. Người đi bộ, người đi xe đạp và người đi xe máy gần lề đường đặc biệt dễ gặp nguy hiểm trước những rủi ro này. Trong những năm gần đây, Thành phố Hồ Chí Minh và nhiều khu vực khác tại Việt Nam đã ghi nhận nhiều vụ tai nạn và thương vong xuất phát từ tình trạng cơ sở hạ tầng nắp cống bị hư hại. Các phương pháp kiểm tra truyền thống hiện nay chủ yếu dựa vào việc thực hiện thủ công, đòi hỏi nhiều thời gian, nhân lực và chi phí. Những phương pháp này không đủ hiệu quả để đảm bảo việc phát hiện và sửa chữa kịp thời. Do đó, việc phát triển một hệ thống tự động và thông minh nhằm nhận diện sớm các nắp cống bị hư hỏng là điều cần thiết để nâng cao an toàn giao thông đô thị. Nghiên cứu này đề xuất một phương pháp phát hiện cải tiến dựa trên thị giác máy tính và công nghệ thiết bị bay không người lái (UAV). Cụ thể, mô hình học sâu YOLOv8 được sử dụng để nhận dạng và phân loại các nắp cống bị hư hỏng từ ảnh chụp trên không. Một bộ dữ liệu mới được xây dựng và gán nhãn để huấn luyện và đánh giá mô hình trong nhiều điều kiện thực tế khác nhau. Hệ thống được đề xuất cho phép giám sát diện rộng một cách hiệu quả, giảm công sức kiểm tra thủ công và hỗ trợ cơ quan chức năng trong việc thực hiện các biện pháp bảo trì kịp thời. Kết quả nghiên cứu kỳ vọng sẽ góp phần tạo ra môi trường giao thông an toàn hơn, giảm thiểu rủi ro tai nạn và minh chứng cho tiềm năng của việc tích hợp trí tuệ nhân tạo và UAV trong quản lý cơ sở hạ tầng đô thị. Trong tương lai, đề tài sẽ được tiếp tục phát triển bằng cách mở rộng bộ dữ liệu, cải thiện độ chính xác nhận dạng trong các điều kiện ánh sáng và thời tiết khác nhau, cũng như tích hợp xử lý thời gian thực để hỗ trợ giám sát liên tục và hướng tới các ứng dụng thành phố thông minh.

Từ khoá: nắp cống hư hỏng, máy bay không người lái, YOLOv

Trích dẫn bài báo này: Đình Hoàng N, Khánh Hiếu N. **Ứng dụng thiết bị bay không người lái trong nhận diện hư hỏng nắp cống thoát nước trên đường giao thông**. *VNUHCM J. Eng. Technol.* 2026;9 (3):2992-3001.