Advancing UAV-Based Inspection System: The USSA-Net Segmentation Approach to Crack Quantification

Kwai-Wa Tse, Rendong Pi, *Student Member, IEEE*, Wenyu Yang, Xiang Yu, and Chih-Yung Wen

***Abstract*— In the realm of crack inspection for complex infrastructures, traditional methods have primarily relied on expensive structural health monitoring instruments and labor-intensive procedures. The emergence of unmanned aerial vehicle (UAV) technology brings about effective and innovative solutions for bridge inspection. To advance the technology, this study presents a novel crack inspection system that employs light detection and ranging (LiDAR) scanning to construct a 3-D model of the target structure. A path planner is then developed to ensure complete coverage of all crack points on the structure being inspected. Through extensive testing, the proposed system demonstrates successful detection and localization of various types of cracks. Furthermore, our improved deep crack segmentation model, U-Net with spectral block and self-attention module, surpasses the performance of the original U-Net model, exhibiting a 3.2% higher Dice coefficient and a 3.3% higher mean intersection over union (mIoU) evaluation metric on our self-established crack dataset. In the case of the Crack500 public dataset, our model outperforms the original U-Net model by 10% in Dice coefficient and 14% in mIoU. Moreover, our U-Net with spectral block and self-attention module (USSA-Net) outperforms other latest state-of-the-art (SOTA) models on the DeepCrack500 dataset, surpassing the progressive and adaptive fusion (PAF)-Net and progressive and hierarchical context fusion (PHCF)-Net by approximately 5% in Dice coefficient and 2.7% in mIoU. For crack size estimations, our proposed system accurately estimates the horizontal and vertical dimensions of cracks, achieving a root-mean-square error (RMSE) of 9.9 and 6.2 mm, respectively. Overall, the system achieves millimeter-level crack size estimation accuracy. Moreover, our system is characterized by its low-cost nature and lightweight design. Experimental results showcase the system's robustness and effectiveness in executing real-world crack inspection tasks, even within complex environments.**

***Index Terms*— Attention module, autonomous inspection system, crack detection, crack quantification, crack segmentation, unmanned aircraft systems (UAS), U-shape network (UNET), unmanned aerial vehicle (UAV).**

Manuscript received 14 January 2024; revised 17 May 2024; accepted 24 May 2024. Date of publication 24 June 2024; date of current version 3 July 2024. This work was supported by the Research Grants Council of the Hong Kong Special Administrative Region, China, under Grant T22-501/23-R. The Associate Editor coordinating the review process was Dr. Benkuan Wang. *(Corresponding author: Chih-Yung Wen.)*

Kwai-Wa Tse, Wenyu Yang, and Chih-Yung Wen are with the Department of Aeronautical and Aviation Engineering, The Hong Kong Polytechnic University, Kowloon, Hong Kong, SAR, China (e-mail: kwai-wa.tse@connect.polyu.hk; allen.yang@connect.polyu.hk; chihyung.wen@polyu.edu.hk).

Rendong Pi and Xiang Yu are with the Department of Mechanical Engineering, The Hong Kong Polytechnic University, Kowloon, Hong Kong, SAR, China (e-mail: devin.pi@connect.polyu.hk; lucien.yu@polyu.edu.hk).

Digital Object Identifier 10.1109/TIM.2024.3418073

I. Introduction

TRADITIONAL inspection methods for infrastructures, including bridges, power plants, and buildings, have historically faced challenges due to their high costs and risks to inspectors. However, the demand of rapid, accurate, and cost-effective infrastructure health inspection is consistently growing, especially in densely populated areas, such as Hong Kong. With the advancements in flight controllers and airborne sensors, unmanned aerial vehicles (UAVs) have opened new opportunities for revolutionizing infrastructure inspection technologies, overcoming the limitations of traditional inspection methods and enhancing their performance and efficiency.

In recent years, extensive research has been conducted to design autonomous UAV systems that incorporate various payload sensors for inspection tasks [1]. A comprehensive and automated bridge inspection typically involves three key steps: coverage path planning (CPP) of the structure, defect detection, and defect size estimation. Once the inspection trajectory is planned, the UAV autonomously follows the designated path, ensuring visibility of all crack points on the inspection target [2], [3]. With the advancement of lightweight light detection and ranging (LiDAR) technology, embedding a lightweight LiDAR on the aerial vehicle enables the acquisition of a point cloud of the inspection target. This facilitates the optimization of the inspection path, covering the entire inspection target while considering the energy consumption of the aerial vehicle.

Moreover, progress made in the edge computing technology facilitates an accurate identification and classification of cracks in real time using aerial images through artificial intelligence (AI)-assisted crack detection [4], [5]. By detecting and localizing cracks, a crack map can be created, enabling inspectors to conduct postinspection health analysis based on the crack images. However, the existing AI-powered crack detection models have their limitations. Some models are computationally demanding, making them unsuitable for onboard deployment on UAVs. Other models may lack the required robustness, leading to unsatisfied detection accuracy. Hence, there is ample room for the development of AI-based crack detectors.

Lastly, the accurate estimation of crack size is crucial for maintenance and decision-making purposes. Image semantic segmentation [6], [7] provides an innovative approach that eliminates the requirement for additional specialized

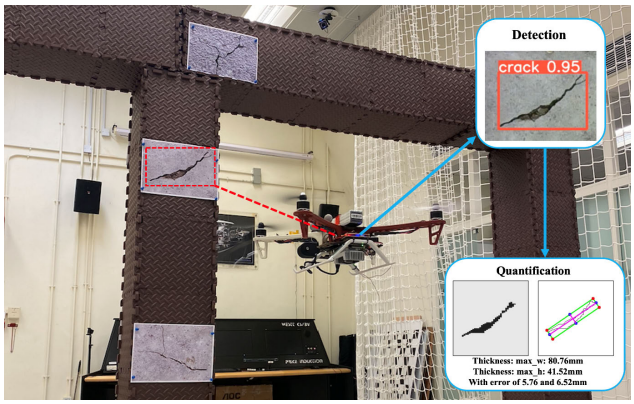


Fig. 1. UAV flight experiment in laboratory environment.

instruments. Many researchers have explored segmentation techniques to obtain accurate segmented crack images based on red-green-blue (RGB) images captured by stereo cameras on the UAV. Subsequently, integrating the segmented crack contours with depth images of the crack assists in precise crack size estimation.

The objectives of this research are to improve the generation of complete coverage inspection paths, enhance crack segmentation models, advance crack detection capabilities, and refine crack size estimation techniques. These objectives collectively contribute to the establishment of a more efficient and reliable bridge inspection system utilizing UAVs, as illustrated in Fig. 1. The main contributions of this article are summarized as follows.

1) *USSA-Net for crack segmentation:* We have introduced U-Net with spectral block and self-attention module (USSA-Net), a novel crack segmentation model customized for crack quantification. This model significantly enhances segmentation accuracy by incorporating spectral and self-attention mechanisms, which effectively differentiate crack features from complex backgrounds. USSA-Net incorporates two key spectral blocks, SPB1 and SPB2, strategically embedded to process image data at varying depths, thereby enhancing feature detection. Complementing these blocks, an advanced self-attention module focuses on distinguishing relevant crack features from complex backgrounds, significantly boosting segmentation precision. This integrated approach achieves a Dice coefficient of 0.968 and a mean intersection over union (mIoU) of 0.939, demonstrating a substantial improvement over the original U-Net model. Notably, it outperforms the latest state-of-the-art (SOTA) models, such as progressive and adaptive fusion (PAF)-Net [8] and progressive and hierarchical context fusion (PHCF)-Net [9] on the DeepCrack500 dataset.
2) *Quantitative crack measurement techniques:* Advanced techniques, including the sliding window and minimum area rectangle methods, are used with RGB and depth imagery to measure crack dimensions accurately. These techniques achieve a root-mean-square error (RMSE) of 9.9 mm for horizontal and 6.2 mm for vertical crack thickness, crucial for structural health assessments.
3) *Validation through UAV flight tests:* The efficacy of USSA-Net was demonstrated through UAV flight experiments focused on structural inspections, confirming its robustness and reliability in real-world conditions and establishing its utility for structural health monitoring.

II. Related Works

A. CPP for Inspection

The objective of CPP is to determine the most efficient path while ensuring complete coverage of the inspection target, taking into account the specifications of the UAV and its onboard sensors [10]. Airborne cameras and inertial measurement units (IMUs) are lightweight and easily deployable on UAVs. Various localization and mapping methods, such as ORB-SLAM [11], Open visual inertial navigation system (VINS) [12], and feedback loop based visual inertial SLAM (FLVIS) [13], have been widely employed in UAV applications. However, these methods face challenges in outdoor environments with significant illumination changes and limited feature points, especially in complex structures, such as cross-sea bridges. While GPS can be used for positioning in open areas during terrain inspection, it becomes unreliable in dense building clusters or underneath bridges. In such cases, UAVs rely on onboard sensors for positioning. Commonly used onboard sensors for UAV navigation include cameras and IMUs. Recently, LiDAR, previously used in autonomous driving, has emerged as a viable option for UAVs [14].

LiDAR-based SLAM systems offer greater robustness in various lighting conditions. Initially, LiDAR was unsuitable for UAV onboard systems due to its large size and high power consumption. However, recent technological advancements have led to the availability of lightweight LiDAR systems, making onboard LiDAR increasingly popular. Classic LiDAR odometry and mapping algorithms, such as LOAM [15], have been developed. LiDAR can also be combined with IMU sensors to enhance localization accuracy, as demonstrated by works, such as Lio-Sam [16] and Fast-Lio2 [17].

Once the 3-D model of the structure is obtained, viewpoint planning (VPP) aims to generate a set of viewpoints (VPs) that cover the building according to specific requirements. Sampling methods, often based on the art gallery problem (AGP) [18], are commonly employed to solve the VPP problem. It has been observed that viewpoint generation methods that closely consider the target's geometry are better suited for crack inspection applications.

After generating the required viewpoints, the algorithm needs to determine the optimal sequence for visiting them. Many approaches formulate this as a traveling salesman problem (TSP) [19], [20]. In view of a tradeoff between computational speed and finding the optimal solution, we employ a two-step approach using VPP and TSP in sequence to generate a complete coverage inspection path planning in our system.

B. Crack Detection, Segmentation, and Quantification

Park et al. [21] introduced a scheme to detect surface cracks and quantify their thickness and length by employing airborne

structured light. However, this method's payload is costly, rendering it a nonlightweight solution. Other approaches have been developed to measure crack widths by enhancing the resolution of crack images [22]. From segmentation perspective, Weng et al. [23] introduced a segment-based method for the quantification of pavement cracks. With an average accuracy of 93.7% for pixelwise crack width, this method robustly quantifies various types of cracks. These findings demonstrate the feasibility of achieving crack quantification in millimeter-scale precision if precise segmented crack images can be obtained from UAV inspection tasks.

To obtain accurate evaluations of crack conditions, a range of innovative deep learning methods have been employed for crack detection and segmentation [24]. Liu et al. [25] were among the pioneers to adopt the U-Net architecture for concrete crack segmentation. Similarly, Huyan et al. [26] approached crack segmentation as a pixel-level detection task and proposed a U-shaped neural network specifically designed for crack detection in pavements. Building upon this foundation, Han et al. [27] introduced a novel model called CrackW-Net, which incorporated a ski-level round-trip sampling module to enhance the neural network's ability to extract crack pixel-level features. To further enhance crack segmentation performance, the attention mechanism was introduced. Chu et al. [28] tackled pavement crack segmentation by incorporating channel and spatial attention into the residual network. Additionally, researchers have extended their focus to crack segmentation in UAV aerial images. Hong et al. [29] and Sun et al. [30] developed deep learning models with attention modules specifically tailored for crack segmentation in UAV aerial images.

In summary, the works reviewed in this chapter lay a solid foundation for this study. However, a fully autonomous pipeline for crack inspection on UAVs from a lightweight perspective is currently lacking. Therefore, the primary objectives of this work are to develop a fully autonomous building inspection solution, encompassing three major areas: 1) coverage inspection path planning; 2) crack detection and crack segmentation on aerial image data; and 3) a precise crack size estimation based on the segmented crack images.

III. Methodology

A. CPP for Inspection

To address the inspection CPP problem, we adopt a two-stage approach, as explained in Section I. Specifically, the VPP algorithm [31] first generates a set of viewpoints that completely cover the inspection target structure and the TSP solver determines the order to visit all the generated viewpoints.

In simulation and flight tests in the laboratory environment, we preset the orientation of UAV's pose as level or horizontal. This indicates that the UAV is maintaining a stable and flat flight position without any tilting or rotation along its lateral (roll) or longitudinal (pitch) axes, namely, both the roll and pitch angles of the UAV are assumed to be equal to zero. Under this configuration, the state of the UAV can be represented by $\xi = (x, y, z, \psi)^T$, where (x, y, z) is the vehicle position and ψ is the yaw angle. For different inspection tasks, the distance between the building surface and the UAV may vary. With a minimum viewing distance $D_{\min}$ and the camera intrinsic matrix $\mathcal{K}$, the coverage area of the camera at one viewpoint can be determined. The viewpoints are sampled under the constraints of the building 3-D model $\mathcal{M}$, the minimum viewing distance $D_{\min}$, and the camera intrinsic matrix $\mathcal{K}$. To combine with the LiDAR odometry and mapping module, the building 3-D model in this article is of the point cloud format. Algorithm 1 shows the viewpoint generation procedure.

Algorithm 1 Viewpoint Planning

Input: 3-D model of the structure $\mathcal{M}$, minimum inspection distance D_{min}, camera intrinsic $\mathcal{K}$
Output: The coverage viewpoints $\mathcal{V}$

```
Initialization of viewpoints V ← ∅
VOXEL = Voxelize(M, D_min, K)
for i = 1 to VOXEL.size() do
    Compute surface normal vector n_i
end for
V ← downsample (n_i)
check uncovered surface patch S
return viewpoint V
```

In Algorithm 1, the viewpoint sampling procedure encompasses several key steps: viewpoint initialization, viewpoint optimization, updating resampled viewpoints, and obtaining the essential viewpoints. Specifically, to achieve viewpoint downsampling, we remove adjacent viewpoints based on the inspection distance and camera intrinsic parameters provided as an input. Our downsampling method involves removing adjacent viewpoints that are within a threshold value, typically 1 m. Additionally, we set the viewpoint visibility overlap area ratio to 0.1, ensuring that two image planes from two viewpoints do not share more than 10% of the voxels within the rectangle of camera visibility. By following this viewpoint sampling method, we obtain the minimum essential viewpoints for the inspection. The viewpoint sampling results will be further discussed in Section V-A.

Once the essential viewpoints have been obtained, they are organized and arranged in a specific order using the TSP algorithm [2] to determine the minimum traveling distance. To mitigate the risks associated with steering and potential collisions, the UAV's yaw angle is carefully controlled and restricted. This ensures that the UAV follows a path that minimizes the likelihood of encountering obstacles or hazardous situations while crossing the bridge piers. By incorporating these measures, we aim to enhance the safety and efficiency of the UAV's flight during the inspection process, reducing the potential risks and ensuring smooth navigation along the optimal access path. The modified TSP with n viewpoints can be expressed as

$$\begin{aligned} &\min \sum_{i=1}^{n} \sum_{i \neq j, j=1}^{n} d_{ij} \times y_{ij} \\ &\text{s.t.} \sum_{i=1, i \neq j}^{n} y_{ij} = 1, \quad j = 1, \ldots, n; \\ &\sum_{j=1, j \neq i}^{n} y_{ij} = 1, i = 1, \ldots, n, \quad \text{where } y_{ij} \in \{0, 1\} \end{aligned} \tag{1}$$

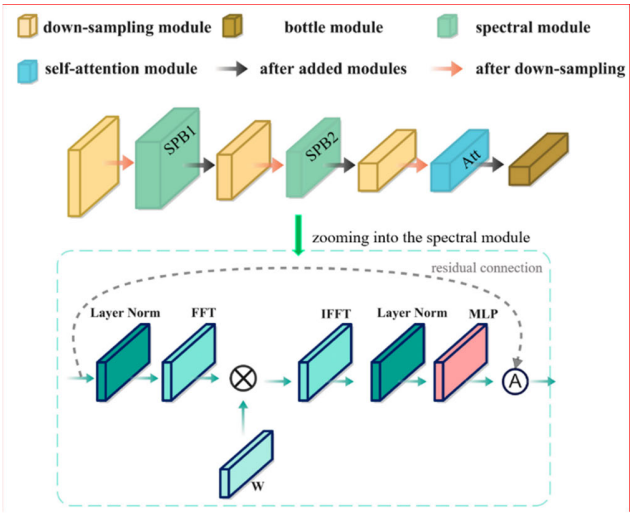


Fig. 2. Architecture of our improved modules in USSA-Net.

where y_{ij} is the logical variable that represents the path that travels from viewpoint i to j, and d_{ij} represents the Euclidean distance between viewpoints i and j. In our TSP solution, there should be only one designated starting point and one designated destination point.

B. U-Net With Spectral Block and Self-Attention Module

The main objective of crack segmentation is to accurately identify and segment the small cracks on the bridge, as mentioned previously. Like previous studies in crack segmentation, our proposed system builds upon the U-Net architecture [32] as a foundation. To enhance the feature extraction and refinement capabilities of the neural network without overwhelming computational resources, we focus our enhancements solely on the encoder portion of the original U-Net.

To extract both high-level and low-level information from the crack images, we introduce two improvements to the original U-Net model. First, we propose a novel attention module called the spectral module, which operates in the frequency domain and utilizes the feature fusion pyramid network (FFPN) to capture the profile and detailed information of the cracks. The second modification involves adding a self-attention module to the last layer of the encoder, further enhancing the neural network's ability to extract high-level semantic information. In summary, our proposed model, named USSA-Net, is based on the U-Net framework and incorporates the spectral module and self-attention module. The architecture of our proposed crack segmentation model is illustrated in Fig. 2. The model takes three-channel RGB crack images as the input and produces the predicted crack segmentation mask as the output.

Due to their small size, the tiny cracks on the bridge pose a challenge for accurate segmentation by the neural network. It is difficult to distinguish between the cracks and the background in the spatial domain. Inspired by [33] and [34], we leverage the spectral module to extract crack information in the frequency domain. This enables the neural network to separate the foreground and background based on the frequency differences in the image components. The structure of the spectral module is depicted in Fig. 2.

The spectral module can convert the image information from the Euclidean space to the Fourier space by the fast Fourier transform (FFT). Before feeding into the Fourier space, layer normalization is used to normalize each channel. Given a feature map $x \in \mathbb{R}^{H\times W\times C}$, H denotes the height, W denotes the width, and C denotes the channel of the input image; then, the operation in Fig. 2 can be described as

$$\mathrm{x}' = \mathcal{F}(\mathrm{LN}(\mathrm{x})) \in \mathbb{C}^{H\times W\times C} \tag{2}$$

where LN means layer normalization, $\mathcal{F}$ means 2-D FFT operation, and x' denotes the output feature map after layer normalization and FFT operation.

Within our enhanced USSA-Net, we incorporate two spectral blocks: SPB1 and SPB2. Although SPB1 and SPB2 utilize the same mathematical transformations as depicted in (2), they are strategically placed in different locations within the network architecture, as shown in Fig. 2. SPB1 is positioned following the first downsampling module and is designed to enhance initial crack pattern detection by applying the transformations in (2). SPB2, on the other hand, is located in a later downsampling module. It processes a broader range of spectral features, reducing the spatial dimensions of the input image while increasing the feature channel count. The strategic use of SPB1 and SPB2 collectively improves the overall performance of our crack segmentation model, enabling it to effectively distinguish between crack features and similar textural patterns in the background, thereby enhancing the robustness of crack detection under varied imaging conditions.

Then, a learnable weight parameter $\mathcal{W}$ is employed to help the spectral module to capture the critical frequency parts, which is beneficial for the crack segmentation process, and the introduction of $\mathcal{W}$ can be described as

$$\mathrm{x}'' = \mathcal{W} \otimes x'. \tag{3}$$

To be noted, the learnable parameter in the spectral module contains gradient information, allowing for updates during the training process through backward propagation. This feature ensures learning capability of the crack segmentation module.

In addition, the inverse FFT (IFFT) is utilized to restore the image information from the Fourier space to the Euclidean space. Following the IFFT operation, a layer normalization technique and a multilayer perceptron (MLP) are employed to extract and refine the high-level semantic information related to cracks. To enhance the crack information flow, a residual connection is introduced, enabling shortcut connections that directly transmit crack information to deeper layers. The formulation of this process can be expressed in the following equation, where $\tilde{x}$ denotes the feature map obtained after spectral module:

$$\tilde{x} = x + \left(\mathrm{MLP}\left(\mathrm{LN}\left(\mathcal{F}^{-1}\left(\mathrm{x}''\right)\right)\right)\right). \tag{4}$$

The self-attention module considers the characteristics observed in bridge crack images, where cracks typically exhibit a long and narrow shape. However, due to the limited receptive field range of convolutional neural networks, these cracks are prone to being segmented into smaller, disjointed

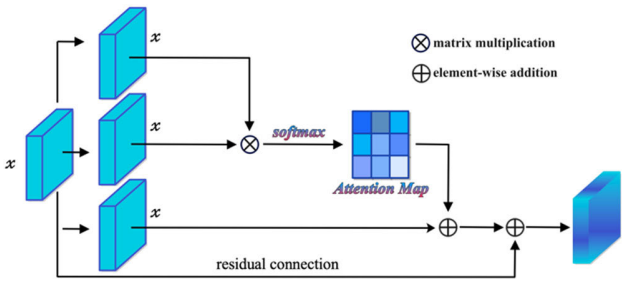


Fig. 3. Structure of the self-attention module.

fragments, resulting in a lack of continuity across the entire crack. To enhance the ability of the model to capture the crack-continuity among pixels in the images, the self-attention module is introduced in the encoder following the last layer. Fig. 3 illustrates the structure of the self-attention module. As the common self-attention module does [35], we obtain the attention map by conducting matrix multiplication for crack feature map itself. Then, we employ the softmax operation to normalize the output as the attention map. Thus, the important information in the crack images can be obtained by computing the weighted sum of the crack feature maps. Furthermore, a residual connection is employed to take the original feature maps as a supplementation to the self-attention computing results. Overall, the self-attention module can be described in the following equation, where x denotes the crack feature map and $\otimes$ denotes the matrix multiplication:

$$A = x + x \otimes \text{softmax}(x \otimes x). \tag{5}$$

The loss function, also known as the objective function, plays a crucial role as a fundamental mathematical component in neural networks. It serves as a guide for the model optimization process, providing the direction for minimizing errors and improving performance. The commonly adopted loss functions in the field of image segmentation are dice loss function, cross-entropy loss function, and intersection over union (IoU) loss function. In our study, the cross-entropy loss function is employed, and it can be expressed as

$$\text{Loss} = -\frac{1}{N}\sum_{n=1}^{N} y_n \log(p_n) + (1 - y_n)\log(1 - p_n) \tag{6}$$

where y_n denotes the ground truth label (0 or 1) and p_n denotes the predicted results (between 0 and 1). The loss is computed for each sample in the dataset, and the goal is to minimize this loss value during the training process. Minimizing the cross-entropy loss helps the model learn to make accurate predictions in crack classification tasks.

C. Crack Detection and Localization

Our system employs a pretrained YOLOv4 with channel-based attention modules squeeze-and-excitation networks (SENets) [36] as crack detector to identify cracks during UAV online inspection. Specifically, in the backbone network of YOLOv4, we integrate the channel-based attention modules [37] into the origin YOLOv4 to enhance the performance

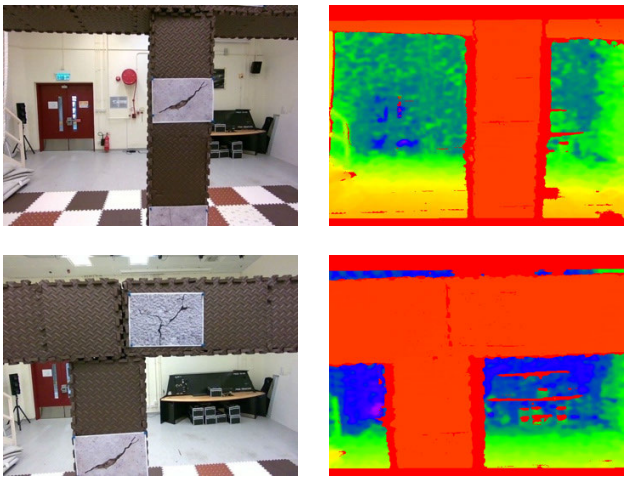


Fig. 4. RGB and depth image pairs of cracks.

of crack detection. First, the global spatial information is collected in the squeeze module by global average pooling. Then, the excitation module captures channelwise relationships and outputs an attention vector using fully connected and nonlinear layers. Finally, each channel of the input feature is scaled by multiplying the corresponding element in the attention vector.

In our proposed crack inspection system, one of the primary objectives is to provide the precise location of the 3-D defect position, while also ensuring a collision-free environment around the defect region. To achieve this, we leverage streaming depth images, as shown in Fig. 4. Once we have detected crack images, we can transform the pixel coordinates into metric units (meters) for the 3-D point clouds using camera intrinsic parameters. The intrinsic parameters describe the properties of the imaging system used to capture the depth frame. In (7), the variables α and β denote the focal length of the camera along the x- and y-axes, respectively. Additionally, c_x and c_y denote the principal points of the camera along the x- and y-axes, accounting for scaling and transition between pixel coordinates and the camera coordinates. The camera intrinsic matrix $\mathcal{K}$ is denoted as

$$\mathcal{K} = \begin{bmatrix} \alpha & 0 & c_x \\ 0 & \beta & c_y \\ 0 & 0 & 1 \end{bmatrix}. \tag{7}$$

The conversion from pixel coordinates (u, v) to 3-D point clouds ($\widehat{X}_c, \widehat{Y}_c, \widehat{Z}_c$) in camera coordinates can be mathematically described in the following equations:

$$u = \frac{\alpha \widehat{X}_c}{\widehat{Z}_c} + c_x \tag{8}$$

$$v = \frac{\beta \widehat{Y}_c}{\widehat{Z}_c} + c_y. \tag{9}$$

Moreover, to convert the coordinates from the pixel frame to the world frame, extrinsic parameters are introduced [37]. r_{ij} serves as a rotation matrix, $\widehat{Z}_c$ denotes the depth value of the crack point, and t_x, t_y, and t_z serve as translation parameters. These extrinsic parameters facilitate the transformation of the pixel coordinates to the world coordinates ($\widehat{X}_W, \widehat{Y}_W$, and $\widehat{Z}_W$), providing a spatial representation of the

crack in a world frame reference system. This conversion can be formulated as

$$\widehat{Z}_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathcal{K} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \begin{bmatrix} \widehat{X}_W \\ \widehat{Y}_W \\ \widehat{Z}_W \\ 1 \end{bmatrix}. \tag{10}$$

Equation (10) represents the mathematical transformation from pixel coordinates to 3-D point clouds. By applying this conversion to the pixel coordinates of interest in the depth map, our system retrieves the metric values in meters for the corresponding points in the point clouds. This enables the crack size estimation with computer vision technique described in Section III-D.

D. Crack Size Quantification

After training a crack segmentation model described in Section III-B, we now can obtain accurate pixel-level segmentation contours from the aerial image with crack. Based on the segmented crack contours, we employ the sliding windows method to estimate the size of cracks in both horizontal and vertical directions. Leveraging computer vision techniques, we scan each individual crack contour in the segmented image in both horizontal and vertical directions. By iterating over the contours of the crack image, we can identify the maximum horizontal thickness (max_w) and maximum vertical thickness (max_h) of the crack, as illustrated in Fig. 5(a). When the maximum width values are obtained, the starting and ending points of the edge can be determined. The starting pixel coordinate in the horizontal direction is recorded as start_w, and the ending pixel coordinate is recorded as end_w. Similarly, the starting pixel coordinate in the vertical direction is recorded as start_v, and the ending pixel coordinate is recorded as end_v. Subsequently, we can combine this information with the depth frame obtained during the inspection to convert the pixel coordinates in the depth map to a 3-D point cloud world coordinate. Finally, by subtracting the transformed starting and ending world coordinates, we can derive the desired dimensions of the crack in the world frame.

Some cracks may possess microscopic inner thickness, but the expansion of the crack can significantly impact the structural health, as illustrated in Fig. 5. In this case, toward giving another interpretation for the size of the crack, bounding box representation is also used to describe the size for a whole crack pattern. It is remarkable that the yellow area in Fig. 5(a), which represents the ground truth segmented crack area, serves as the most fitting region for crack size estimation. In contrast, the pink bounding box area is inferred by our deep learning crack detector, an improved YOLOv4 model. Our detector is robust on crack detection and localization but is not ideal for accurately compute the width and height of the crack size. This is because the width of bounding box (box_w) and the height of bounding box (box_h) often overestimate the actual size of crack width and length, as depicted in Fig. 5(b). box_w and box_h are much greater than horizontal thickness (max_w) and vertical thickness (max_h) in Fig. 4(a), respectively. To derive another precise interpretation of crack size, we utilize the minimum area rectangle method, which introduces the green

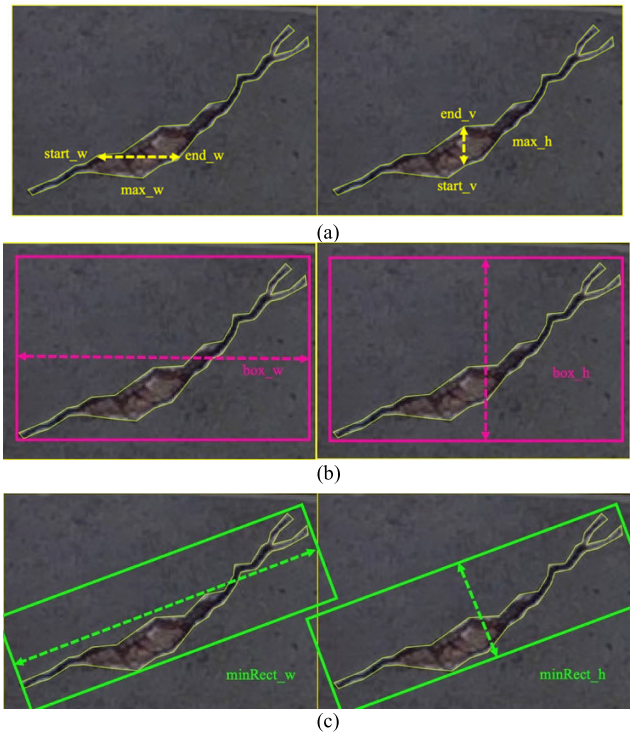


Fig. 5. Comparison between bounding box method and minimum area rectangle method. (a) Segmented crack area. (b) Bounding box method. (c) Minimum area rectangle method.

bounding box corresponding to the minimum area rectangle of the crack defect, as shown in Fig. 5(c). By calculating the difference between the maximum and minimum values of the sides of a rectangle, we can determine the width (minRect_w) and height (minRect_h) of the minimum area rectangle. It is important to emphasize that this method of obtaining the minimum area rectangle relies on having a precisely segmented image that enables the identification of four corner points for each crack contour. Consequently, until robust crack segmentation models are developed, relying solely on the bounding box method from object detectors for accurate crack size estimation has been proven to be inadequate.

Moreover, through our analysis of various types of cracks, the area of the crack is equally important as the thickness, whereas we find that the interpretation of crack criticality can be implicitly presented by their area, while the cracks are in irregular shapes, as exemplified in Fig. 6.

In our proposed segmentation method, the segmented image, as presented in Fig. 7, provides implicit parameters for calculating the crack area. Specifically, Fig. 7 showcases the contours of the segmented crack area, which encompass the total number of image pixels within the crack area. Additionally, by combining depth images, as discussed in Section III-C, we have the capability to convert pixel coordinates to world coordinates for each pixel point, enabling the aggregation of all pixels within the crack area. This allows for a comprehensive understanding of the crack's size and spatial coverage.

In summary, our proposed USSA-Net segmentation method represents a significant advancement in providing precise and valuable information, including crack area information, surpassing the limitations of traditional bounding box methods.

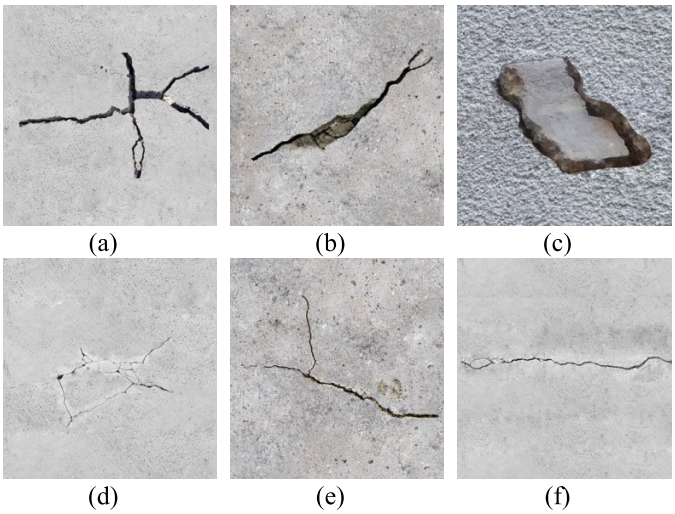


Fig. 6. Different types of cracks. (a) Expansion concrete crack. (b) Heaving crack. (c) Settling concrete crack. (d) Concrete crack caused by premature drying. (e) Shrinkage crack. (f) Long and flattening crack.

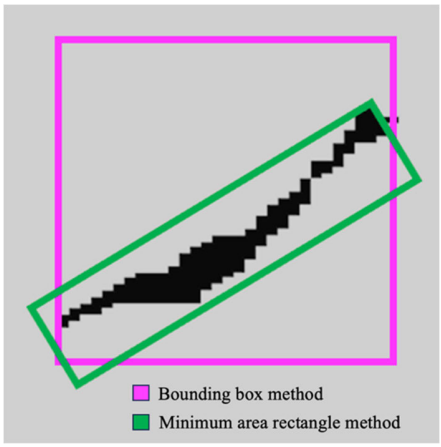


Fig. 7. Comparison between traditional bounding box method (the pink box) and the proposed minimum rectangle area method (the green rectangle).

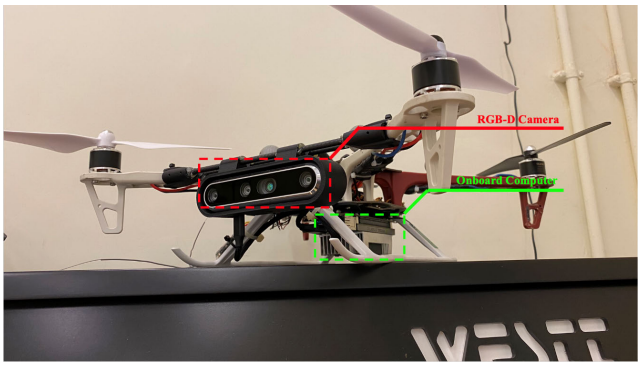


Fig. 8. UAV with RGB-D camera.

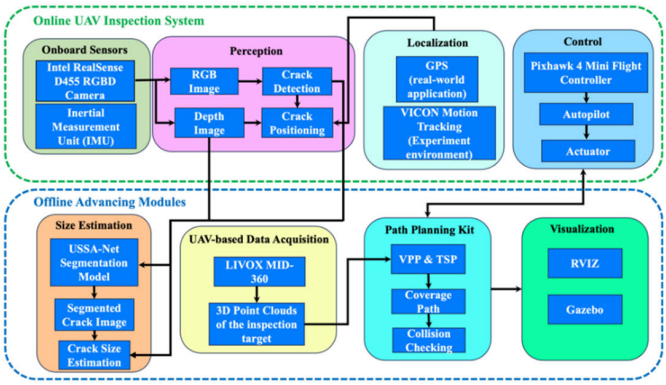


Fig. 9. Software architecture of the UAV inspection system.

Fig. 10. UAV hardware comparison. (a) ICGT inspection UAV [1]. (b) DJI M300 inspection UAV. (c) Our proposed lightweight inspection UAV.

IV. Experimental Implementation

A. System Overview

The aircraft utilized in this study is equipped with an Nvidia Jetson TX2 onboard computer, shown in Fig. 8, which efficiently handles the online modules of the UAV inspection system. These modules consist of a perception module, a localization module, and a control module, as illustrated in Fig. 9. To perceive the environment, the proposed system relies solely on the Intel RealSense D455 Depth camera. Within the perception module, the AI detector uses compressed depth aligned RGB images from the D455 camera to identify cracks. Each pixel is assigned a depth value calculated by the camera, and a depth-to-color-align frame, referred to as the depth frame, is generated. This depth frame is subsequently used for real-time crack position calculation as well as offline crack size estimation.

Furthermore, when comparing our overall UAV solution to systems, such as Da Jiang (DJI) M300 and image capturing and geo-tagging (ICGT) aircraft [1], our proposed system stands out as much lighter, as depicted in Fig. 10. These solutions also incorporate LiDAR and RGB cameras, but our system offers a significant weight advantage from both hardware and the neural network model perspective. Additionally, the computational efficiency analysis will be discussed in detail in Section V-D.

It is important to note that the crack localization component is primarily influenced by the camera and aircraft pose, independent of the UAV inspection trajectory. Extensive verification of the proposed system has been conducted using diverse trajectories, yielding promising crack localization results as reported in our previous work [37]. Consequently, the system is not constrained by predetermined paths and demonstrates robustness in various applications. In the experiment environment, the VICON motion tracking system provides the aircraft pose, which enables the transformation of crack coordinates from the camera frame to the world frame. Additionally, the aircraft pose serves as critical information for the path planning kit. Furthermore, the proposed system incorporates a collision-checking technique. Leveraging the long-range capabilities of the onboard D455 camera, which

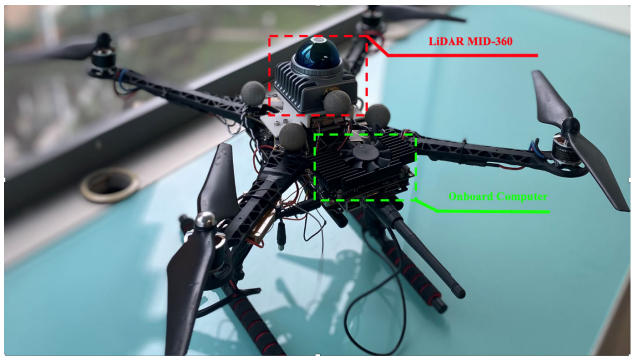


Fig. 11. UAV with a LiDAR MID-360.

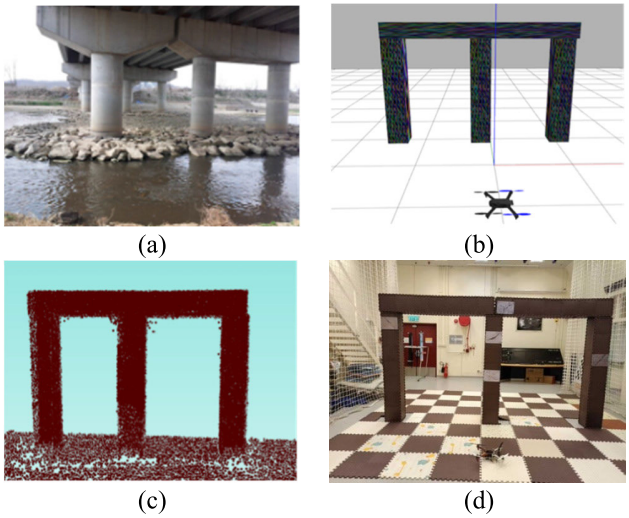


Fig. 12. Bridge inspection environment setup. (a) Real double-gate shape bridge. (b) Simulation environment in Gazebo. (c) Visualization of point cloud map. (d) Real-world scenario setup in our laboratory.

operates effectively within a range of 0.6–6 m, depth information is derived and utilized for collision checking during inspection scenarios.

In this proposed crack inspection system, three additional innovative modules are introduced: a UAV-based structural data acquisition module, an inspection path-planning module, and a deep learning-based crack size estimation module. Our system employs a LIVOX MID-360 LiDAR mounted on a UAV, as shown in Fig. 11, to acquire point cloud data of the inspection target, enabling the creation of a 3-D model.

To employ LiDAR technology for acquiring point cloud data and constructing the 3-D model of the inspection target structure, three main steps are involved: point-cloud data acquisition, point-cloud data processing, and 3-D model construction. Using the fast LiDAR-inertial odometry (FAST-LIO) system [17], our UAV flies around the experimental scenes, as shown in Fig. 12, capturing point cloud data of the inspection model from desired locations. The data acquisition process starts by triggering the LiDAR scanner while the UAV flies around the target area, capturing different perspectives and angles to ensure comprehensive coverage of the experimental bridge pier scene. This process is iterated until we have collected sufficient point cloud data to represent the desired object area.

The constructed 3-D model is then fed into the path planning kit to generate a coverage inspection path that incorporates collision avoidance techniques. Finally, a deep learning model with U-shape network (UNET) incorporating a spectral block and self-attention module, named USSA-Net, is applied for crack segmentation. The detected aerial crack image serves as the input to the pretrained USSA-Net, resulting in a segmented crack image with only crack contours. By combining the depth image of the detected crack, we can estimate the physical size of the crack in the world coordinate frame.

B. Experimental Scene Setup

The entire system, as illustrated in Fig. 9, is initially tested and validated in the robot operating system (ROS) and Gazebo simulation environment before conducting real flight experiments. We develop a simulation platform within Gazebo specifically designed to replicate the inspection of a double-gate-shaped structure, which serves as a representative model for a bridge pier—a common shape encountered in bridge structures. In our simulation, we utilize the 3DR IRIS drone model and the PX4 firmware as the underlying controller. To ensure stable and maneuverable flight, we implement a cascade proportional integral derivative (PID) controller for trajectory tracking control.

In addition to the simulation environment, we conduct real-world experiments using a custom-built DJI F450 airframe with an Nvidia TX2 serving as the onboard computer. The real-world setup, along with the simulation environment and the point cloud map of the real-world setup, is illustrated in Fig. 12. These experiments involve the coverage of nine crack points on the inspection target structure.

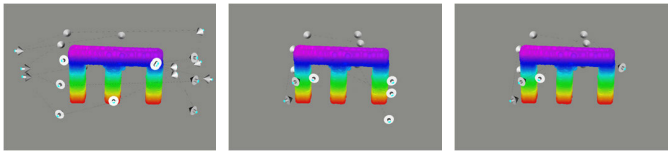
C. Dataset Preparation and Augmentation

Based on the experimental settings and aircrafts described above, we collect 201 raw crack images. The resolution of each image is 1280 $\times$ 720 pixels. Subsequently, we perform data augmentation on the raw crack images. Data augmentation has been proved to be a simple yet effective method to enhance the generalization ability of the neural network by increasing the amount of the training data. There are many ways to augment data, such as random rotation, vertical or horizontal clip, and random mirroring. In this study, we employ image scaling with scaling factor is between 0.5 and 1.5, random rotation with rotation angle between -10° and $+10^\circ$, and random mirror to perform the augmentation during the training process. After that, we obtain 764 crack images in total, where 688, 38, and 38 images are allocated for training, validation, and testing, respectively.

Apart from the self-established dataset above, we also employ Crack500 dataset to evaluate the performance of our proposed crack segmentation model. This dataset is collected at the Temple University using a smartphone. It has 500 pavement crack images with the resolution 2000 $\times$ 1500 pixels. Since the image resolution is high, each crack image is cropped into subimages with nonoverlapped regions to facilitate the efficient model training. Consequently, there are 3368 images from Crack500 dataset in total; therein, 1896,

TABLE I
CPP EVALUATION

Inspection Distance	Number of VP	Coverage Rate
1.0 (m)	25	97.8 %
1.5 (m)	**12**	**96.4 %**
2.0 (m)	10	93.5 %



(a) (b) (c)

Fig. 13. Viewpoint generation with different inspection distance settings. (a) Result of 1-m inspection distance. (b) Result of 1.5-m inspection distance. (c) Result of 2-m inspection distance.

348, and 1124 images are allocated to training, validation, and test, respectively.

D. Implementation of Our USSA-Net Segmentation Model

The segmentation model USSA proposed in this study is trained from scratch without any pretrained weights on a workstation with a single graphics processing unit (GPU), NVIDIA RTX 3090. Deep learning framework PyTorch Lightning is used to simplify the process of training and organizing the model. Each benchmarking model is trained for 500 epochs, which is sufficient for the model to converge. Besides, the Adam optimizer is selected to speed up the convergence. The initial learning rate is set to 0.0001. Notably, the batch size for training and validation is set to 12, while the batch size for test is set to 1 to replicate our work.

V. RESULTS AND DISCUSSION

A. CPP for Inspection

In the context of UAV building inspection, the settings can be adjusted to meet specific mission requirements, as outlined in Section II-A. The variance of UAV inspection distance between the UAV's camera and the inspection target plays a crucial role. This adjustment is primarily influenced by two main considerations: 1) the resolution of the employed RGB-D camera and 2) the size of the crack to be detected on the inspection target. To accurately compare the results of coverage path generation for different crack sizes, it is necessary to vary the minimum inspection distance and evaluate the performance of the inspection system. This variation in inspection distance allows for a comprehensive coverage evaluation. Table I presents the parameters and their corresponding inspection coverage rates, with VP denoting the viewpoint. Fig. 13 displays the results of viewpoint generation for a variety of inspection distance settings, where the arrows denote the camera viewpoints.

Given that the 3-D model of the structure and the camera intrinsic matrix remain fixed, the inspection distance serves as the sole variable in the CPP module of our proposed system. As the minimum inspection distance increases, two notable observations arise: 1) the number of viewpoints tends to decrease, while the inspection distance increases and 2) the coverage rate changes slightly, while the inspection distance increases. These observations demonstrate the reliability of the CPP module incorporated in our proposed inspection system.

Various flight tests are conducted in this study. In the real-world experiment, we build the bridge structure with obstacles in the laboratory, as shown in Fig. 12(d). We can completely collect the point clouds of the bridge model incorporating Fast-Lio2 [17], and then, the point clouds data further serve as the input for CPP module. Subsequently, the UAV executes the flight mission by following the generated coverage trajectory. During the flight tests, we successfully detected all cracks situate on the bridge surface, and the detected crack aerial images are further analyzed in both the crack segmentation module and crack size estimation module. The corresponding results will be presented in Section V-E.

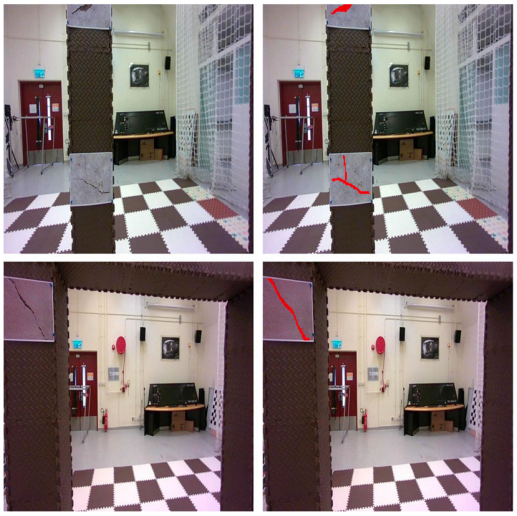


Fig. 14. Crack segmentation generated by our proposed USSA-Net model.

B. Performance of Our USSA-Net Segmentation Model

The evaluation metrics used in this study are the same as many previous segmentation-related studies; Dice coefficient (also known as Dice) and mIoU are used to evaluate the performance of our proposed model. $|X|$ represents the number of the pixels belong to the ground truth, and $|Y|$ represents the number of pixels belong to the inference. Then, Dice efficient and mIoU evaluation metrics can be described as follows:

$$\text{Dice} = \frac{2|X \cap Y|}{|X| + |Y|} \tag{11}$$

$$\text{mIoU} = \frac{|X \cap Y|}{|X \cup Y|}. \tag{12}$$

Fig. 14 shows the result of crack segmentation based on our USSA-Net. The proposed model can accurately segment the cracks on the UAV aerial images. The segmented cracks are continuous and complete. Table II shows the comparison between the proposed method and some SOTA models on self-established dataset. Our model achieves a Dice coefficient 96.8% and a mIoU 93.9% on the self-established dataset.

Our USSA-Net outperforms the SOTA models both in Dice and mIoU metrics, as demonstrated in Fig. 15. Besides, the

TABLE II
PERFORMANCE COMPARISON ON SELF-ESTABLISHED DATASET

Baseline models	Performance Metric	
	Dice	***mIoU***
U-Net [32]	0.936	0.906
Segnet [38]	0.894	0.811
PSPNet [39]	0.879	0.789
USSA-Net (Ours)*	**0.968**	**0.939**

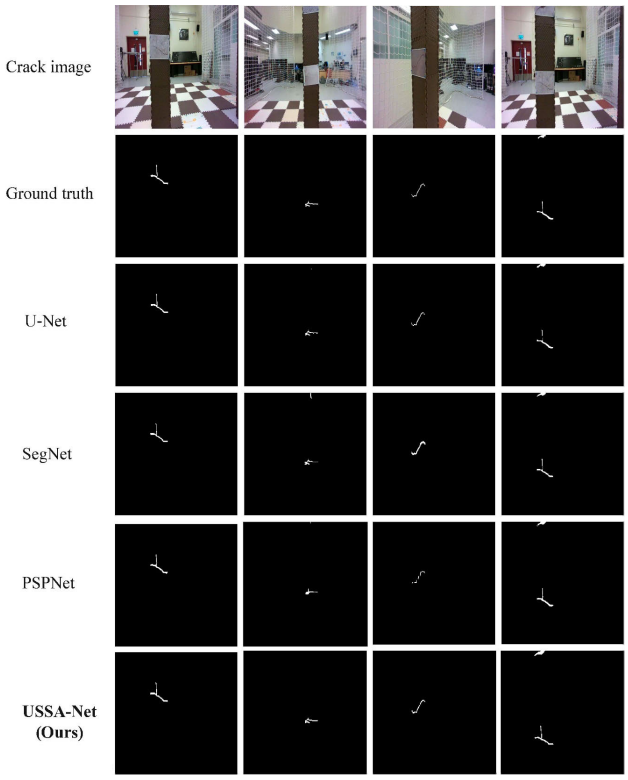


Fig. 15. Crack segmentation with various SOTA models.

performance of our proposed model is evaluated with the Crack500 dataset, as illustrated in Fig. 16. As depicted in Table III, our model still obtains the best performance than other SOTA models on Crack500 dataset. In comparison, our model outperforms the original U-Net model by 10% in Dice coefficient and 14% in mIoU on the Crack500 dataset.

In addition to the previously compared models, such as Segnet [38], pyramid scene parsing network (PSPNet) [39], Deeplab [40], and Deeplab with multi-scale attention network (DMA-Net) [30], we include the more recent SOTA crack segmentation models SegCrackNet [41], PAF-Net [8], and PHCF-Net [9], which were published in 2023 and 2024. These models are chosen as baseline models for performance evaluation. We conduct a fair comparison using and publicly available crack segmentation datasets, namely, DeepCrack500, as displayed in Fig. 17. The performance comparison results are presented in Tables III and IV. It surpasses the PAF-Net and PHCF-Net by approximately 5% in Dice coefficient and 2.7% in mIoU. These findings highlight the advancement of our proposed USSA-Net in the field of crack segmentation.

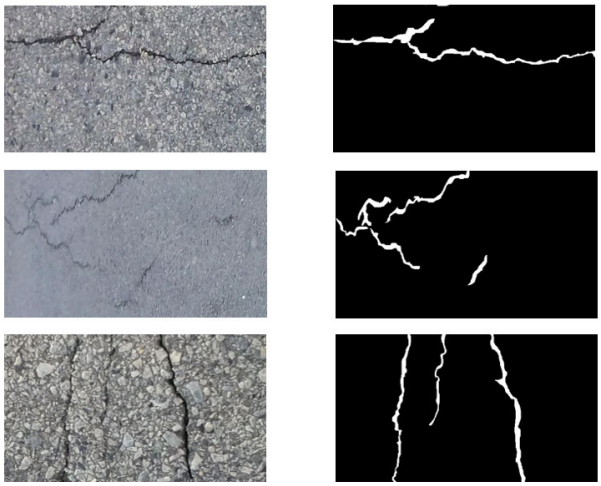


Fig. 16. Crack500 segmentation dataset.

TABLE III
PERFORMANCE COMPARISON ON CRACK500 DATASET

Baseline models	Performance Metric	
	Dice	***mIoU***
U-Net [32]	0.668	0.527
Deeplab v3+ [40]	0.686	0.542
PSPNet [39]	0.678	0.537
DMA-Net [30]	0.699	0.559
SegCrackNet [41]	N/A	0.496
USSA-Net (Ours)*	**0.776**	**0.670**

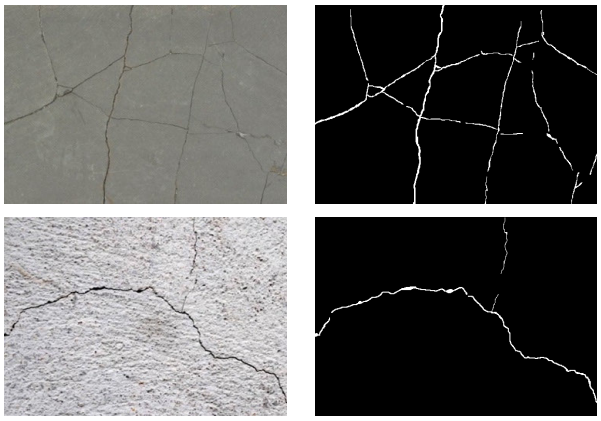


Fig. 17. DeepCrack500 segmentation dataset.

C. Ablation Experiment

We adopt the original U-Net model as the baseline in our ablation studies. Please note "SPB1" indicates that only the first spectral block has been integrated into the baseline model; "SPB2" denotes the integration of only the second spectral block; and "SPBs" refers to the addition of both spectral blocks. Additionally, "Baseline + Att" represents the scenario where only the self-attention module is added to the baseline model. The specific locations of these modules are illustrated in Fig. 2. As shown in Table V, the inclusion of the spectral blocks or the attention module individually and in combination significantly enhances the model's performance. However, the

TABLE IV
PERFORMANCE COMPARISON ON DEEPCRACK500 DATASET

Baseline models	Performance Metric	
	Dice	*mIoU*
Deeplab v3+ [40]	0.794	0.822
PSPNet [39]	0.815	0.841
PAF-Net [8]	0.915	0.902
PHCF-Net [9]	0.907	0.903
USSA-Net (Ours)*	**0.963**	**0.930**

TABLE V
ABLATION EXPERIMENT OF OUR PROPOSED USSA-NET ON DEEPCRACK500 PUBLIC DATASET

Ablation Configuration	Performance Metric	
	Dice	*mIoU*
Baseline	0.941	0.893
Baseline+SPB1	0.948	0.904
Baseline+SPB2	0.959	0.923
Baseline+SPBs	0.954	0.915
Baseline+Att	0.955	0.918
Baseline+SPBs+Att (Ours*)	**0.963**	**0.930**

TABLE VI
NETWORK COMPLEXITY ANALYSIS OF CRACK SEGMENTATION MODELS

Model	Performance Metrics	
	Parameters(M)	*Inference time (s)*
U-Net [32]	2.2	0.42
PSPNet [39]	4.9	0.64
Segnet [38]	2.9	0.63
USSA-Net (Ours)*	3.1	0.50

highest performance gains were observed when both spectral blocks and the attention module were combined, underscoring their collective impact on enhancing crack segmentation. Specifically, our model "Baseline + SPBs + Att" demonstrates a notable improvement, increasing the Dice coefficient by 2.2% and the mIoU by 3.7%. This comprehensive set of ablation studies illustrates that incorporating spectral blocks and self-attention modules to capture detailed profile information significantly enhances crack segmentation performance.

D. Computational Efficiency Analysis

Although the performance of the model is important, the model complexity is also a key factor to affect the segmentation computing efficiency on crack images. The complexity analysis among some crack segmentation models is shown in Table VI. From this table, it reveals that the addition of the proposed optimization blocks (spectral and attention blocks) has slightly increased the model complexity. However, it can still meet the requirements of the task for UAV crack segmentation.

In addition to the computational analysis on crack segmentation task, we evaluate the size and real-time detection speed of the proposed crack detection neural network. Table VII summarizes our findings on real-time crack detection speed, and these results confirm that our proposed system is lightweight in terms of neural network size, while still achieving real-time detection capabilities.

TABLE VII
CRACK DETECTION SPEED ANALYSIS

Real-time detection	Network input size	Inference frame per second (FPS)
Our method*	320 x 320	25
	416 x 416	17
	512 x 512	**13**
	608 x 608	9

Fig. 18. Crack detection tests with wooden boards background.

Fig. 19. Crack detection performance on asphalt pavement scene.

E. Crack Detection Under Various Scenarios

To assess the performance and generalization ability of our crack detection and segmentation models, we conduct tests under various crack scenarios, including concrete wall, wooden boards, asphalt pavement road, experimental bridge pier scene, and so on. In the following, we outline some scenarios and present the corresponding experimental results in Figs. 18 and 19.

F. Crack Size Quantification

In this study, we utilize our USSA-Net to obtain segmented crack images for crack size estimation using depth image pairs. The crack sizes are estimated through the sliding window method and minimum area rectangle method, as illustrated in Fig. 20. The ground truth values in metrics (mm) have been premeasured. Table VIII presents the ground truth values, estimations, and errors for max_*w*, max_*h*, minRect_*w*, and

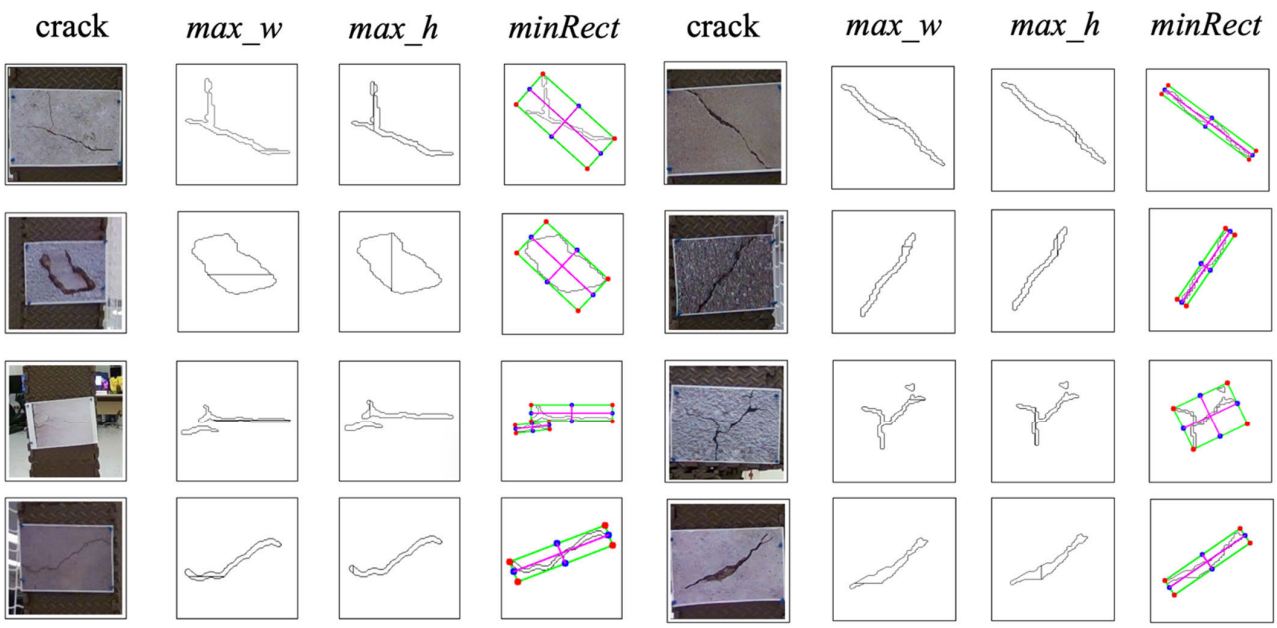


Fig. 20. Visualization of crack segmentation and quantification results.

TABLE VIII
CRACK SIZE QUANTIFICATION RESULTS

Metrics (mm)	*max_w*			*max_h*			*minRect_w*			*minRect_h*		
	(gt)	*(est)*	*(err)*	*(gt)*	*(est)*	*(err)*	*(gt)*	*(est)*	*(err)*	*(gt)*	*(est)*	*(err)*
Crack 1	62	62.72	**0.72**	105	105.09	**0.09**	236	230.61	-5.39	110	99.82	-10.18
Crack 2	75	80.76	5.76	35	41.52	6.52	270	266.51	-3.94	53	51.17	**-1.83**
Crack 3	50	44.38	-5.62	100	111.63	11.63	190	177.45	-12.55	160	135.83	-24.17
Crack 4	15	23.98	8.98	50	51.39	**1.39**	240	225.96	-14.04	30	31.90	**1.90**
Crack 5	30	38.66	8.66	20	22.03	2.03	275	261.95	-13.06	25	27.11	2.11
Crack 6	150	167.91	17.91	150	147.60	-2.40	225	225.66	**0.66**	120	116.10	-3.91
Crack 7	200	183.07	-16.93	35	36.14	**1.14**	200	196.79	-3.21	45	42.56	-2.44
Crack 8	105	112.69	7.69	100	112.38	12.38	250	248.24	**-1.76**	150	177.37	27.37
Crack 9	110	107.82	-2.18	30	29.23	**-0.77**	300	302.80	2.80	70	72.37	2.37
RMSE	9.96 (mm)			6.19 (mm)			8.08 (mm)			12.8 (mm)		

minRect_*h*, respectively. Here, *gt* refers to the ground truth data, *est* represents the estimated sizes, *err* denotes the estimation errors, and the bolded entries in Table VIII indicate the outstanding performance among the compared methods.

Overall, our crack size estimations closely align with the ground truth values. Notably, our approach achieves minimal errors in crack size estimation. Additionally, we evaluate the both the width (minRect_*w*) and height (minRect_*h*) of the minimum area rectangle method. RMSEs are employed to evaluate the crack size estimation performance of our model. Table VIII displays the RMSE errors of 9.9, 6.2, 8.1, and 12.8 mm for max_*w*, max_*h*, minRect_*w*, and minRect_*h* metrics, respectively.

VI. CONCLUSION

In this article, we have presented a novel technique for crack inspection based on UAVs and an autonomous path planner for complete inspection coverage, representing a significant advancement in the field of crack inspection. To further enhance the performance of our system, we have developed a robust deep learning crack segmentation model called USSA-Net, which has shown significant improvement in performance metrics, achieving a Dice coefficient of 0.968 and mIoU of 0.939. Comprehensive flight tests, both in simulation and real-world scenarios, have demonstrated the feasibility and effectiveness of our proposed system for accurately estimating and measuring horizontal and vertical dimensions of concrete crack achieving an RMSE of 9.9 and 6.2 mm, respectively, solely relying on an onboard RGB-D camera. The results highlight the system's potential to effectively handle structural inspection tasks in complex environments. The entire workflow of the proposed system, together with its confirmed robustness, has been meticulously detailed. This research could contribute to the broader development of crack inspection methodology, providing valuable insights to the field.

APPENDIX

A demonstration video of this study is available at https://youtu.be/Vho1Cx1tErQ. The crack segmentation dataset in this work is available at https://polyu.hk/iUqfJ,

and the source code of our proposed segmentation model and crack size estimation methods is available at https://github.com/everskyrube/uav-crack-segmentation.

ACKNOWLEDGMENT

The authors wholeheartedly thank their AAE research group, which is led by Prof. Chih-Yung Wen at The Hong Kong Polytechnic University, for providing great support on carrying out various experiments in this study.

REFERENCES

[1] M. R. Saleem, J.-W. Park, J.-H. Lee, H.-J. Jung, and M. Z. Sarwar, "Instant bridge visual inspection using an unmanned aerial vehicle by image capturing and geo-tagging system and deep convolutional neural network," *Struct. Health Monitor.*, vol. 20, no. 4, pp. 1760–1777, Jul. 2021, doi: 10.1177/1475921720932384.

[2] H. W. Tong, B. Li, H. Huang, and C. Wen, "UAV path planning for complete structural inspection using mixed viewpoint generation," in *Proc. 17th Int. Conf. Control, Autom., Robot. Vis. (ICARCV)*, Dec. 2022, pp. 727–732.

[3] K. Maeda, S. Doki, Y. Funabora, and K. Doki, "Flight path planning of multiple UAVs for robust localization near infrastructure facilities," in *Proc. 44th Annu. Conf. IEEE Ind. Electron. Soc.*, Oct. 2018, pp. 2522–2527.

[4] Q. Qiu and D. Lau, "Real-time detection of cracks in tiled sidewalks using YOLO-based method applied to unmanned aerial vehicle (UAV) images," *Autom. Construct.*, vol. 147, Mar. 2023, Art. no. 104745, doi: 10.1016/j.autcon.2023.104745.

[5] J. Xing, Y. Liu, and G.-Z. Zhang, "Improved YOLOV5-based UAV pavement crack detection," *IEEE Sensors J.*, vol. 23, no. 14, pp. 15901–15909, Jul. 2023, doi: 10.1109/JSEN.2023.3281585.

[6] S. L. H. Lau, E. K. P. Chong, X. Yang, and X. Wang, "Automated pavement crack segmentation using U-Net-based convolutional neural network," *IEEE Access*, vol. 8, pp. 114892–114899, 2020, doi: 10.1109/ACCESS.2020.3003638.

[7] N. H. T. Nguyen, S. Perry, D. Bone, H. T. Le, and T. T. Nguyen, "Two-stage convolutional neural network for road crack detection and segmentation," *Expert Syst. Appl.*, vol. 186, Dec. 2021, Art. no. 115718, doi: 10.1016/j.eswa.2021.115718.

[8] L. Yang, H. Huang, S. Kong, Y. Liu, and H. Yu, "PAF-Net: A progressive and adaptive fusion network for pavement crack segmentation," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 11, pp. 12686–12700, Nov. 2023.

[9] L. Yang, H. Huang, S. Kong, and Y. Liu, "A deep segmentation network for crack detection with progressive and hierarchical context fusion," *J. Building Eng.*, vol. 75, Sep. 2023, Art. no. 106886.

[10] C. Papachristos et al., "Autonomous exploration and inspection path planning for aerial robots using the robot operating system," in *Robot Operating System (ROS)*, vol. 3. Cham, Switzerland: Springer, 2019, pp. 67–111.

[11] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "ORB-SLAM: A versatile and accurate monocular SLAM system," *IEEE Trans. Robot.*, vol. 31, no. 5, pp. 1147–1163, Oct. 2015.

[12] P. Geneva, K. Eckenhoff, W. Lee, Y. Yang, and G. Huang, "OpenVINS: A research platform for visual-inertial estimation," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2020, pp. 4666–4672.

[13] S. Chen, Y. Feng, C.-Y. Wen, Y. Zou, and W. Chen, "Stereo visual inertial pose estimation based on feedforward and feedbacks," *IEEE/ASME Trans. Mechatronics*, vol. 28, no. 6, pp. 3562–3572, Dec. 2023.

[14] T.-M. Nguyen, S. Yuan, M. Cao, Y. Lyu, T. H. Nguyen, and L. Xie, "NTU VIRAL: A visual-inertial-ranging-LiDAR dataset, from an aerial vehicle viewpoint," *Int. J. Robot. Res.*, vol. 41, no. 3, pp. 270–280, Mar. 2022.

[15] J. Zhang and S. Singh, "LOAM: LiDAR odometry and mapping in real-time," in *Proc. Robot., Sci. Syst. (RSS) Conf.*, Berkeley, CA, USA, Jul. 2014, pp. 1–9.

[16] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus, "LIO-SAM: Tightly-coupled LiDAR inertial odometry via smoothing and mapping," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2020, pp. 5135–5142.

[17] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "FAST-LIO_2: Fast direct LiDAR-inertial odometry," *IEEE Trans. Robot.*, vol. 38, no. 4, pp. 2053–2073, Aug. 2022.

[18] A. Savkin and H. Huang, "Proactive deployment of aerial drones for coverage over very uneven terrains: A version of the 3D art gallery problem," *Sensors*, vol. 19, no. 6, p. 1438, Mar. 2019.

[19] E. Galceran and M. Carreras, "A survey on coverage path planning for robotics," *Robot. Auton. Syst.*, vol. 61, no. 12, pp. 1258–1276, 2013.

[20] S. Chen, Y. Li, and N. M. Kwok, "Active vision in robotic systems: A survey of recent developments," *Int. J. Robot. Res.*, vol. 30, no. 11, pp. 1343–1377, Sep. 2011.

[21] S. E. Park, S. H. Eem, and H. Jeon, "Concrete crack detection and quantification using deep learning and structured light," *Construct. Building Mater.*, vol. 252, Aug. 2020, Art. no. 119096, doi: 10.1016/j.conbuildmat.2020.119096.

[22] J. Yoon, H. Shin, M. Song, H. Gil, and S. Lee, "A crack width measurement method of UAV images using high-resolution algorithms," *Sustainability*, vol. 15, no. 1, p. 478, Dec. 2022, doi: 10.3390/su15010478.

[23] X. Weng, Y. Huang, and W. Wang, "Segment-based pavement crack quantification," *Autom. Construct.*, vol. 105, Sep. 2019, Art. no. 102819, doi: 10.1016/j.autcon.2019.04.014.

[24] Y. Jiang, D. Pang, C. Li, Y. Yu, and Y. Cao, "Two-step deep learning approach for pavement crack damage detection and segmentation," *Int. J. Pavement Eng.*, vol. 24, no. 2, pp. 1–14, Apr. 2022, Art. no. 2065488, doi: 10.1080/10298436.2022.2065488.

[25] J. Liu et al., "Automated pavement crack detection and segmentation based on two-step convolutional neural network," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 35, no. 11, pp. 1291–1305, Nov. 2020, doi: 10.1111/mice.12622.

[26] J. Huyan, W. Li, S. Tighe, Z. Xu, and J. Zhai, "CrackU-Net: A novel deep convolutional neural network for pixelwise pavement crack detection," *Struct. Control Health Monitor.*, vol. 27, no. 8, p. e2551, Aug. 2020, doi: 10.1002/stc.2551.

[27] C. Han, T. Ma, J. Huyan, X. Huang, and Y. Zhang, "CrackW-Net: A novel pavement crack image segmentation convolutional neural network," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 11, pp. 22135–22144, Nov. 2022, doi: 10.1109/TITS.2021.3095507.

[28] H. Chu, W. Wang, and L. Deng, "Tiny-crack-Net: A multiscale feature fusion network with attention mechanisms for segmentation of tiny cracks," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 37, no. 14, pp. 1914–1931, Nov. 2022, doi: 10.1111/mice.12881.

[29] Z. Hong et al., "Highway crack segmentation from unmanned aerial vehicle images using deep learning," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022, doi: 10.1109/LGRS.2021.3129607.

[30] X. Sun, Y. Xie, L. Jiang, Y. Cao, and B. Liu, "DMA-Net: DeepLab with multi-scale attention for pavement crack segmentation," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 10, pp. 18392–18403, Oct. 2022, doi: 10.1109/TITS.2022.3158670.

[31] S. Jung, S. Song, P. Youn, and H. Myung, "Multi-layer coverage path planner for autonomous structural inspection of high-rise structures," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2018, pp. 1–9.

[32] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Med. Image Comput. Comput.-Assist. Intervent. (MICCAI)*, 2015, pp. 234–241, doi: 10.1007/978-3-319-24574-4_28.

[33] B. N. Patro, V. P. Namboodiri, and V. Srinivas Agneeswaran, "SpectFormer: Frequency and attention is what you need in a vision transformer," 2023, *arXiv:2304.06446*.

[34] J. Wu, W. Ji, H. Fu, M. Xu, Y. Jin, and Y. Xu, "MedSegDiff-v2: Diffusion based medical image segmentation with transformer," 2023, *arXiv:2301.11798*.

[35] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 30, 2017, pp. 5998–6008.

[36] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141, doi: 10.1109/CVPR.2018.00745.

[37] K.-W. Tse, R. Pi, Y. Sun, C.-Y. Wen, and Y. Feng, "A novel real-time autonomous crack inspection system based on unmanned aerial vehicles," *Sensors*, vol. 23, no. 7, p. 3418, Mar. 2023, doi: 10.3390/s23073418.

[38] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder–decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017, doi: 10.1109/TPAMI.2016.2644615.

[39] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6230–6239, doi: 10.1109/CVPR.2017.660.

[40] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder–decoder with Atrous separable convolution for semantic image segmentation," in *Proc. 15th Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 833–851, doi: 10.1007/978-3-030-01234-2_49.

[41] C. Guo, W. Gao, and D. Zhou, "Research on road surface crack detection based on SegNet network," *J. Eng. Appl. Sci.*, vol. 71, no. 1, p. 54, Dec. 2024.

Kwai-Wa Tse received the B.Eng. degree from The Chinese University of Hong Kong, Hong Kong, in 2013, and the M.Sc. degree (Hons.) from The Hong Kong Polytechnic University, Kowloon, Hong Kong, SAR, China, in 2018, where he is currently pursuing the Ph.D. degree with the Department of Aeronautical and Aviation Engineering, supervised by Prof. Chih-Yung Wen.

His research interests include object detection, semantic segmentation, computer vision, machine learning, and its applications on unmanned aerial vehicles.

Rendong Pi (Student Member, IEEE) received the bachelor's degree from the School of Transportation Science and Technology, Harbin Institute of Technology, Harbin, China, in 2018, and the master's degree from the School of Qilu Transportation, Shandong University, Jinan, China, in 2022. He is currently pursuing the Ph.D. degree with the Department of Mechanical Engineering, The Hong Kong Polytechnic University, Kowloon, Hong Kong, SAR, China.

His research interests include audio–visual localization, physics-informed neural networks, object detection, object tracking, and semantic segmentation.

Wenyu Yang received the B.Eng. degree from the Ocean University of China, Qingdao, China, in 2019, and the M.Eng. degree from Harbin Institute of Technology, Harbin, China, in 2021. He is currently pursuing the Ph.D. degree with the Department of Aeronautical and Aviation Engineering, The Hong Kong Polytechnic University, Kowloon, Hong Kong, SAR, China, supervised by Prof. Chih-Yung Wen and Dr. Boyang Li in MAV/UAV Laboratory.

Xiang Yu received the B.Eng. (Hons.) and Ph.D. degrees from the Department of Mechanical Engineering, The Hong Kong Polytechnic University, Kowloon, Hong Kong, SAR, China, in 2011 and 2015, respectively.

He is an Assistant Professor at the Department of Mechanical Engineering, The Hong Kong Polytechnic University. He has published over 50 articles in SCI journals. His research primarily focuses on the development of theories, numerical methods, advanced materials, and metamaterials related to the field of acoustics and vibrations.

Dr. Yu serves as an Assistant Editor for the *Journal of Sound and Vibration*.

Chih-Yung Wen received the B.S. degree from the Department of Mechanical Engineering, National Taiwan University, Taipei, Taiwan, in 1986, and the M.S. and Ph.D. degrees from the Department of Aeronautics, California Institute of Technology (Caltech), Pasadena, CA, USA, in 1989 and 1994, respectively.

In 2012, he joined the Department of Mechanical Engineering, The Hong Kong Polytechnic University, Kowloon, Hong Kong, SAR, China, as a Professor. In 2019, he became the Head of the Department of Aeronautical and Aviation Engineering, The Hong Kong Polytechnic University. His current research interests include modeling and control of tail-sitter unmanned aerial vehicles (UAVs), visual-inertial odometry systems for UAVs, and artificial intelligence (AI) object detection by UAVs.