Shelled Unmanned Aerial System for Bridge Structural Health Monitoring

Margie Arda, Jeanette Pao, *Member, IAENG,* Carl John Salaan, *Member, IAENG,* Charles Alver Banglos, Steve Clar, Lester Librado, and Jonathan Maglasang, *Member, IAENG*

***Abstract*—Bridge inspections are critical for maintaining structural integrity and must be inspected on a regular basis to ensure their dependability. Aerial vehicles provide a safer, cost-efficient, and time-saving method for inspection. However, UAVS (Unmanned Aerial Vehicles) are limited from flying near the bridge structure due to their exposed propellers. Using shelled UAVs addressed the collision problem while giving a good visual condition and increasing motion effectiveness.**

In this study, a shelled UAV system that structural inspectors can use to perform a close visual inspection of the bridge was developed. The newly developed shelled UAV features a passive rotating shell with a two-axis gimbal. This shelled UAV is waterproof, modular, capable of water takeoff and landing, and integrated with a crack detection system. A functional shelled UAV was fabricated, and test flights were successfully conducted.

The results of the computational simulations and actual flight tests showed that the shelled UAV is overall safe and effective in terms of its strength-to-weight consideration, drag force, and stability performance. Moreover, crack detection systems and software applications were developed. A curated dataset was also produced for the purpose of training the crack detection system. The U-Net architecture was used as the segmentation model trained on the dataset. The trained model was effective and could predict and segment cracks in the gathered dataset images. The functionality of the Data Acquisition App, Damage Detection App, and Inspection Details App was tested and verified.

***Index Terms*—bridge inspection, crack detection, drag force, shelled UAV, semantic segmentation, stability, structural health monitoring, unmanned aerial vehicle**

I. Introduction

BRIDGE inspections play a vital role in maintaining the structural integrity and safety of the infrastructure. These inspections help engineers detect any damage or defects and determine areas in the bridge that may pose potential problems. Early detection of potential risks is important before they develop into serious issues that may cause catastrophic incidents, such as bridge failure or total collapse. Thus, national agencies responsible for monitoring infrastructures are mandated by the government to perform bridge inspections [1].

Fig. 1. The shelled Unmanned Aerial System used for bridge assessment.

In the Philippines, the Department of Public Works and Highways (DPWH) is responsible for maintaining and monitoring the condition of all national bridges [2].

Traditional bridge access methods use specialized equipment such as scaffoldings, man-lifts, bucket trucks, scissor lifts, or boats to allow proximity inspection of the structure. Inspectors assess the bridge's condition through visual inspection and manually obtain photographic records of the defects [3], [5]. The use of specialized equipment like under-bridge inspection trucks is expensive, has high maintenance costs, and is difficult to schedule because of its availability and the traffic concerns it poses to the public. In addition, this bridge inspection method is risky to the inspectors [5].

With the number of bridges to be monitored and the frequency of undertaking bridge inspections, the traditional method eventually leads to a backlog of maintenance activities which might hinder the early detection of defects and other structural issues. Recent studies adopted unmanned aerial vehicles (UAVs) for structural health monitoring to provide a safer, cost-efficient, and time-saving method.

Numerous studies on the use of UAVs for automated bridge inspection have been conducted in recent years [4], [5], [6]. One major challenge in adopting UAVs for structural health monitoring is their limitation from flying near the

Manuscript received April 17, 2023; revised August 21, 2023.
This research is funded by Department of Science and Technology (DOST) and monitored by the Department of Science and Technology - Philippine Council for Industry, Energy and Emerging Technology Research and Development (DOST- PCIEERD).

M. Arda is a researcher of Mindanao State University – Iligan Institute of Technology, Iligan City, Philippines (e-mail: margie.arda@g.msuiit.edu.ph).
J. Pao is a researcher of Mindanao State University – Iligan Institute of Technology, Iligan City, Philippines (e-mail: jeanette.pao@g.msuiit.edu.ph).
C. J. Salaan is a professor of Electrical Engineering and Technology Department, Mindanao State University – Iligan Institute of Technology, Iligan City, Philippines (e-mail: carljohn.salaan@g.msuiit.edu.ph).
C. A. Banglos is a researcher of Mindanao State University – Iligan Institute of Technology, Iligan City, Philippines (e-mail: charlesalver.banglos@g.msuiit.edu.ph).
S. Clar is a researcher of Mindanao State University – Iligan Institute of Technology, Iligan City, Philippines (e-mail: steveel.clar@g.msuiit.edu.ph).
L. Librado is an instructor of Mindanao State University – Iligan Institute of Technology, Iligan City, Philippines (e-mail: lester.librado@g.msuiit.edu.ph).
J. Maglasang is a director and professor of Mechanical and Aerospace Engineering at Cebu Technological University, Cebu City, Philippines (e-mail: jonathan.maglasang@ctu.edu.ph).

Fig. 2. Shelled UAV in Salaan et al. (2018) designed for bridge inspection provides close images of possible damage.

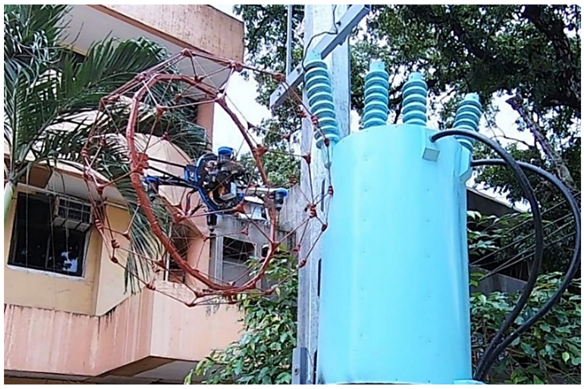


Fig. 3. Close proximal visual inspection of distribution lines using a shelled drone with meshed net in Librado et al. (2022).

bridge structure due to their exposed propellers. With the complexity of bridge structures, flying a UAV under the bridge makes it prone to collisions. To address this problem, several studies on using UAVs for infrastructure inspection developed a passive and protective mechanism that will make a collision-resilient UAV.

II. UAV-Based Method of Visual Inspections

The advantage of using an Unmanned Aerial Vehicle for visually inspecting bridges is that traditional inspection methods don't require pre-inspection preparations. The UAV-based method uses only a UAV mounted with a wireless camera to inspect the bridge condition.

The traditional method manually photographs the bridge's entire condition and defects condition. The photographs will be appended to the report, and the inspector must use the blackboard and ribbon rods or measuring tape to get detailed information. The blackboard must be included in the photograph as well. But this process takes longer time spent on the field or area. That is why the use of UAVs not only systematically captures images from the data taken from the camera mounted on the UAV using a developed software application but also measures defects using the trace tool provided in the evaluation app. Moreover, a digital form provided by the software app includes inspection details to be filled out by the inspector. This information will be attached to the corresponding image gathered.

The disadvantage of using UAVs for inspection may be limited on the outside parts of the bridge, and some cracks will be difficult to detect because the UAV propeller might collide with unwanted materials or the bridge walls. Proximity inspection of the bridge may not be performed using UAV alone; thus, obstacle-avoidance techniques for the UAV were implemented in numerous studies to address the problem [7], [8], [9], [10]. The development of shelled UAVs addressed the collision problem while giving a good visual condition and increasing motion effectiveness. This is in the form of spherical shells that are passively rotating, which enables the UAV or drone to collide with obstacles without compromising its flight stability.

III. Shelled UAV for Infrastructure Inspections

Shelled UAV or shelled drone is enclosed with a protective shell mainly for protecting the drone propeller and its body during a close-proximity inspection. This eliminates the risk for the inspector to inspect a damaged infrastructure against safety and environmental hazards and prevents accidents if the shelled drone collides or is stuck and needs to be recovered. Per previous related studies, developed a shelled drone of the spherical shell structure.

Salaan et al. (2016) developed a design strategy for the protective spherical shell of the drone and further investigated the developed shelled drone from previous related studies on actual bridge conditions, as shown in Fig.2 [10]. Simulation and actual bridge inspection were undertaken to assess its effectiveness in gathering photos of various bridge components without any concern for collision. The spherical shell drone successfully displayed its capabilities, including moving through confined spaces and rolling in touch with smooth and uneven surfaces.

Librado et al. (2022) developed meshing and insulating procedures for close-proximity visual examination of distribution lines, the idea of adding meshed net for the spherical shell [11]. The shelled drone with a meshed net was used for distribution line proximal inspection, as shown in Fig. 3. The design concept of the study was based on the works of Mizutani et al. (2014) and Salaan et al. (2018). Together with its meshing and insulation strategies, the resiliency offered by the spherical shell alone is improved. The spherical shell with meshed net prevented the intrusion of relatively smaller objects and could perform efficiently in the test. Recently, Pao et al. (2023) investigated the performance of the shelled drone through aerodynamic and vibration analysis [12]. This additional meshed net of the shelled UAV further augments the safety and survivability of the drone.

IV. Overview on Unmanned Aerial System for Bridge Inspections

The UAV-based structural health monitoring system presented in this paper comprises three main components: the shelled UAV, the crack detection system, and the software application. The shelled UAV is used as the equipment for close visual inspection of the bridge. It allows the inspector to monitor the parts of the bridge that are difficult to access using traditional equipment. The crack detection system is trained using deep learning and will be deployed with the software application. The software application is developed as a tool that will aid the inspectors in evaluating and assessing the bridge condition.

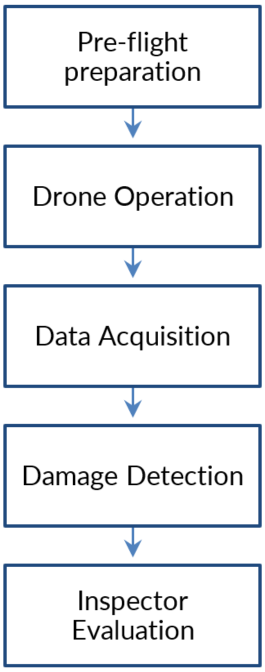


Fig. 4. 5-Step process of a UAV-assisted bridge inspection procedure with hardware and software applications.

TABLE I
REQUIREMENT SET BY DPWH

Category	***State***	***Requirements***
Hardware Drone	Drone Operation	• Provides a safe and secure operation • Can access narrow spaces and critical parts of the bridge • Able to capture images of the structure in proximity • Waterproof
Software Application	Data Acquisition	• Livestream from the drone camera • Real-time crack detections • Record the live stream for review • 3D mapping / localization of damage
	Damage Detection	• Automatic detection of cracks • Capable of detecting damages on the bridge
	Inspector Evaluation	• Review images of interest • Capable of measuring point-to-point distances of the damages

From these components, a five-step bridge inspection process is presented. The first step is the pre-flight inspection, wherein the shelled UAV is prepared for flight with safety checks before take-off. The second step is the drone operation, wherein an operator controls the shelled UAV to survey the infrastructure. The third step is data acquisition, combined with the drone operation. During this step, a software application shows a live stream of the drone's camera view and allows the inspector to record and download it for later review and evaluation. The fourth step is damage detection. For this step, a software application is developed to allow a review of the data acquired in the previous step. The inspector can capture and save specific images of defects and categorize their type of damage. The application also provides automatic crack detection of the data acquired. Once the data review is done, the next step is the evaluation. A software application is developed to view the captured images of defects. The application provides a crack annotation tool to allow the inspector to trace and measure the length and width of cracks of interest. The first and second steps utilize the shelled UAV, while the third, fourth, and fifth steps utilize the crack detection system and the software application, Fig. 4 shows the 5-step process.

Fig. 5. Waterproof SwellPro Fisherman FD1 Fishing Drone of 2-kg payload capacity.

Taking advantage of the UAV's capability to reach high places while remotely operated, the shelled UAV is designed to withstand high-impact collisions from sudden wind gusts and navigate through narrow passages while inspecting. Table 1 summarizes the requirements set by the DPWH.

V. DEVELOPMENT OF THE PROPOSED SHELLED UAV

This section details the methods employed by the researchers to build the whole system.

A. Selection of the Base UAV

A multirotor UAV, specifically the X-type quadcopter, was chosen as the base UAV unit. This type of drone offers stable flight, small size, and the ability to hover, which is suitable for the required application. Due to size constraints, small UAVs are recommended to have an adequate-size protective shell that can still enter narrow passages or spaces on the infrastructures to be inspected. In addition, the other payloads that should be catered for by the base UAV are the gimbal structure and the crack detection system. Therefore, it must have a high payload capacity and strong propulsion system to achieve a long flight duration and effectively perform its desired application.

Furthermore, as suggested by our industry partner (DPWH), the base UAV unit should also be waterproof since it is possible that the drone may crash due to unavoidable circumstances and falls into the water during a bridge inspection. In relation to this, it is preferable that the UAV can float and water takeoff when doing drone retrieval.

Based on these requirements, we chose an off-the-shelf drone, the SwellPro Fisherman FD1, since it is well suited for our application, as shown in Fig. 5. It has a 2kg payload capacity and a waterproof rating of IP67. Also, having a maximum flight time of around 12 mins with a 1.5kg load is adequate for the inspection procedure. Lastly. its capability to have stable flight even when experiencing strong wind

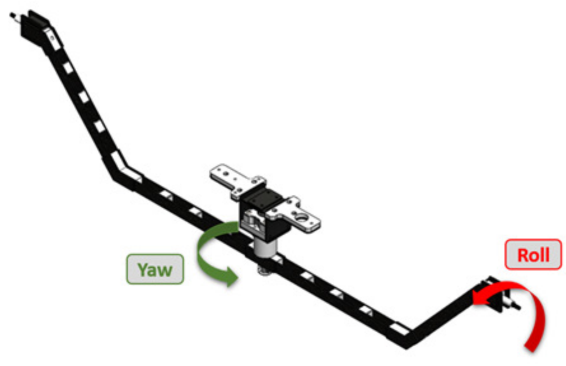


Fig. 6. 2-Degrees of Freedom Gimbal Structure allowing yaw and roll rotation.

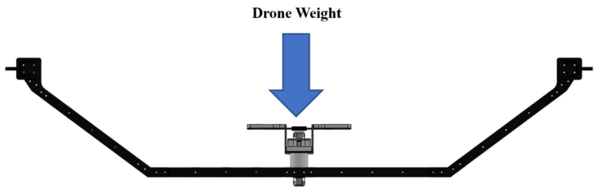


Fig. 7. Gimbal component supporting the drone's weight with the onboard camera.

gusts (up to Beaufort 7) is also an important feature of our application.

B. Design and Analysis of the Gimbal Structure

When designing the gimbal structure, the base UAV should be easily attached and removed from the gimbal unit, thus having a modular attribute that our industry partner sets. Moreover, instead of having a typical 3-axis gimbal system, a 2-axis gimbal unit was chosen to address this requirement.

This gimbal system is only composed of roll and yaw rotation. It spans 820 mm and is mostly made of strong, lightweight materials such as carbon and ABS plastic. Fig. 6 shows the three-dimensional CAD model of the 2-axis gimbal structure. Furthermore, as shown in Fig. 7, the gimbal component holds or supports the SwellPro Fisherman FD1 drone's weight; thus, the strength of the gimbal design should be examined.

C. Sizing and Analysis of the Shell with Nylon Mesh

Two important criteria should be met when sizing the protective shell of the shelled UAV. First, the shell should not be larger than the minimum girder spacing of the bridge, and the second one is to ensure that the spherical shell should not hit the propeller even if it undergoes deformation during the collision. The main girder spacing of large-scale bridges is between 1.1 and 1.7m, whereas small-scale bridges are only between 0.6 and 0.8m. However, only large-scale bridges are considered for the design since the elevation of those smaller bridges is low enough, wherein it does not need scaffolding when conducting the manual inspection. Moreover, considering the drone frame size (450 mm) plus the base UAV unit's propeller size (12 in), the allowable shell diameter should be greater than 760mm. Taking also into

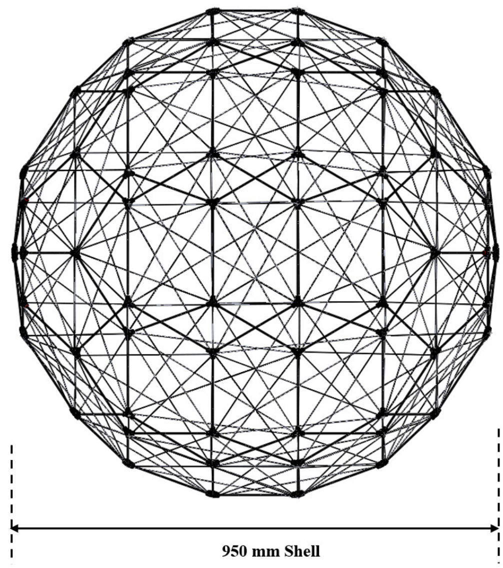


Fig. 8. The 950 mm diameter passive rotating spherical shell with nylon mesh.

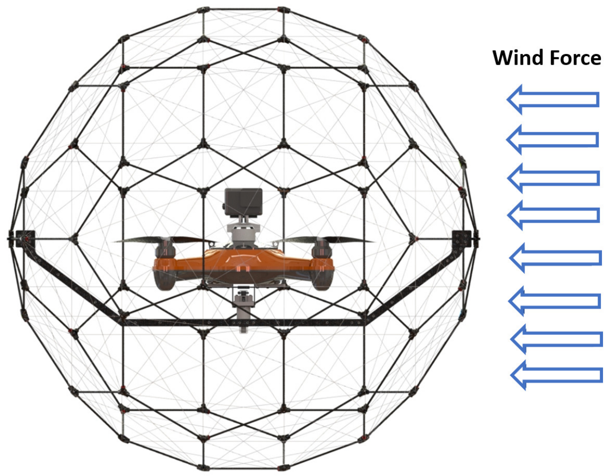


Fig. 9. Shelled UAV during bridge inspection encountering wind-induced drag forces.

consideration the span of the gimbal structure, the determined size of the shell is around 950mm, as shown in Fig. 8. Furthermore, the shell is created by utilizing strong and lightweight materials like carbon fibre rods and ABS 3D printed joints.

Salaan et al. (2016) developed a design strategy for the protective spherical shell of the drone. A fullerene spherical shell structure was made, giving lesser drag, higher strength, and reduced overall weight. The impact strength was used to assess and validate the choice of the base drone, analysis of the spherical shell's design and diameter, setting of the gimbal structure, and material choice as implemented in Fig. 9.

During bridge inspection, specifically on the girder area, the shelled UAV is exposed to a vacuum-like effect due to the bridge structure. This causes the shell to enter the girder gap at a relatively faster speed and may collide with the concrete wall. Thus, it is important to analyze the stiffness of the shell to ensure that its deformation is not high enough to touch the propeller. From the study of Librado et al. (2022), the meshing strategy wherein a nylon monoline was wrapped around the protective shell in a star-shaped pattern improved its stiffness. Probable solutions were also investigated to come up with the best combination between the carbon rod and nylon size with the goal of having an improvement on the overall stiffness of the shell while having only a slight increase in weight and drag force. It can also withstand high-impact force and, at the same time, dissipates more energy compared to the unmeshed shell. Drag force, system vibration and stability experienced by different shell configurations were also analyzed through the works of Pao et al. (2022). The fullerene-type structure shell with nylon mesh shows a promising characteristic; hence, it was applied on the designed shelled UAV in this study with one hexagon face of the shell left open as a provision for base UAV installation and maintenance purposes.

This study's proposed shelled UAV of 950mm shell diameter consists of a new gimbal design, a SwellPro Fisherman drone component, and an onboard camera. The proposed shelled UAV assembly will also experience significant drag force due to the wind during bridge inspections, as shown in Fig. 9. Therefore, the proposed shelled UAV's drag force and stability performance should be investigated. Moreover, the onboard camera's ability to capture clear images may be affected by the shell with nylon mesh; thus, the camera's image clarity during the flight will also be considered.

VI. Crack Detection Development Process

The crack detection system developed in this project aims to assist inspectors in detecting cracks when they visually inspect bridges. The crack detection system aims to localize cracks in the pixel level given an input image. In this research, the deep learning approach to semantic segmentation is employed.

A. Initial Testing and Data Gathering

The initial test flight was conducted on one of the target sites, which is the Mandulog Bridge in Iligan City, Philippines, as shown in Fig. 10. Precautionary measures were implemented before drone operation, such as the usage of personal protective equipment (PPE), conducting a pre-flight inspection checklist, and a safety net to catch the drone in case it drops due to abnormalities or other unavoidable circumstances. Spectators were also warned not to come close to the drone inspection area to prevent undesirable accidents. The shelled UAV was then operated and visually inspected the structure. Fig. 11 shows samples of high-quality images acquired by the drone during the said inspection.

B. Procedure in Developing the Crack Detection System

These can be used as additional testing images to train crack detection and evaluation. Moreover, the drone pilot did not experience any problems operating the drone since it flies controllably. Minor mechanical issues were also observed but were immediately addressed. Ongoing flight tuning and improvements to the drone have been made.

This application requires a fast and precise network architecture to perform semantic segmentation. Semantic segmentation's task is to predict each pixel's class label in an image. The output is an alpha mask with the same size as the input. Fig. 12 shows the process flow in developing the crack detection system.

Fig. 10. Shelled-UAV conducting a close visual inspection and precautionary measures such as using a safety net prepared in case the drone falls.

Fig. 11. Sample images captured by the Shelled-UAV on the girder of the Mandulog Bridge in Iligan City, Philippines.

The process starts by preparing the crack dataset, which is then randomly split into training and validation datasets. The selection of a network architecture suitable for the research's application follows. Once a network architecture is determined, the base model is implemented and trained to detect cracks. The model is trained for a number of epochs wherein after every epoch, the model is validated on the validation dataset. The loss between predicted and true crack pixels is computed during validation. Training continues until an acceptable loss is obtained. The model is then evaluated regarding precision, recall and F1 score. Finally, the results are visualized and observed.

C. Preparing the Dataset

The general flow of data preparation is shown in Fig. 13. The dataset comprises images of surface cracks with and without shell obstructions. The first step is to gather images of surface cracks. The images are obtained from the shelled UAVs and commercial drone video recordings during bridge inspections. Once data is gathered and collected, crack annotation follows, wherein crack pixels in the images are manually labeled. The images collected are then preprocessed by dividing them into 448x448 pixels using a sliding window split. This dimension sets the input size of the crack detection model. The final step in preparing the dataset is to randomly split the dataset into training and validation sets at a ratio of 4:1.

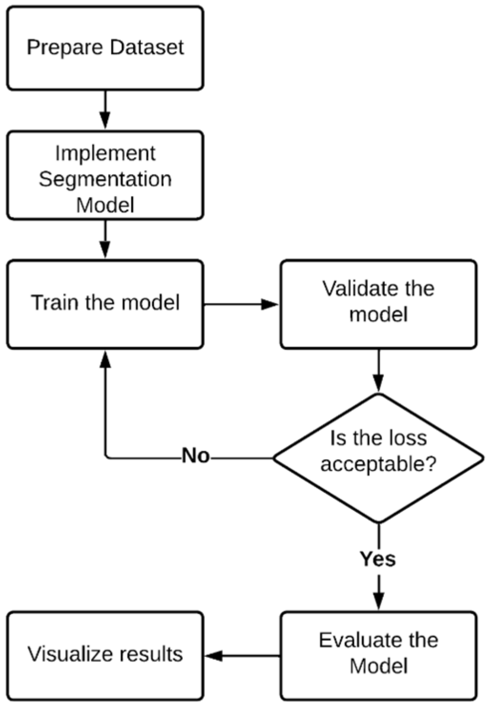


Fig. 12. The process flow in developing the crack detection system to assist inspectors in detecting cracks when visually inspect bridges.

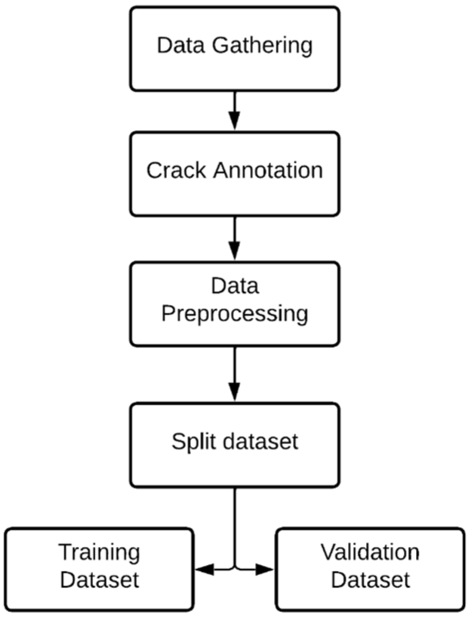


Fig. 13. Data preparation general flow with the dataset consists of images of surface cracks with and without shell obstructions.

In training the crack detection system, thousands of images of surface cracks are needed to produce a detection system that can generalize well to unseen data. For this application, the images obtained through the drone camera view contain shell obstructions because of the camera position and the design of the shelled UAV.

CrackUAS dataset is curated by obtaining images of surface cracks on concrete walls and bridges. To train a robust crack detection system that can detect surface cracks with or without the presence of obstructions, three types of images comprise the dataset: (i) images of unobstructed surface cracks which are collected using a commercial drone, (ii) images taken from the perspective of the shelled UAV that shows portions of the passively rotating shell, and (iii) synthetic images of surface cracks with shell obstructions.

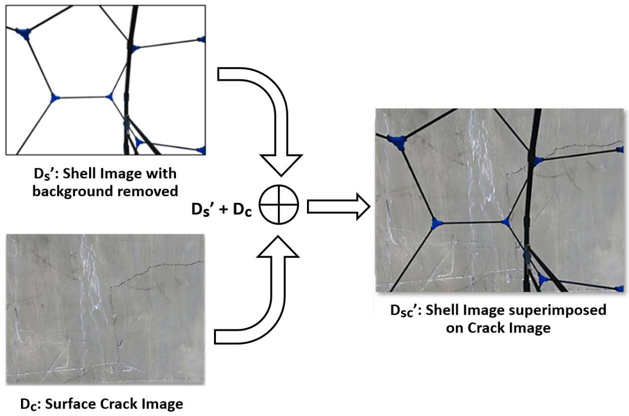


Fig. 14. The process of creating the synthetic data, synthetic images were generated by superimposing the pixels of the shell on unobstructed images of surface cracks.

To expand the dataset, synthetic images were generated by superimposing the pixels of the shell on unobstructed images of surface cracks. Fig. 14 shows the process of creating synthetic data. The advantage of adding synthetic data is that high-quality surface cracks of different forms and sizes can be used and matched with the shell pixels, thus, simulating images taken from the shelled UAV's perspective. This method produces images of surface cracks with shell obstruction, which are time-consuming and difficult to gather from actual structures. Public datasets of surface cracks from [13], [14], [15], [16], [17], [18] are also added to the curated dataset to increase the data size further.

D. Selection and Implementation of Segmentation

The state-of-the-art architecture used in semantic segmentation tasks is the U-Net. It was developed by Ronneberger (2015) for biomedical image segmentation. It won the ISBI cell tracking challenge in 2015. Their training strategy proves it can outperform the best methods with only a few images while strongly relying on data augmentation techniques. Moreover, segmenting a 512x512 image took less than a second on an NVIDIA Titan GPU with 6GB of memory. Because of its favourable speed and performance, the U-Net architecture will be used for our crack detection system, as described in Fig. 15. The U-Net architecture [19] is based on an encoder-decoder structure.

The encoder follows the typical architecture of a convolutional network but is structured as a contracting path. As the image passes through the layers in this part of the network, the dimensions of the feature maps are downsampled by half while the number of channels is doubled. This is performed using a max pooling operation. On the other hand, the decoder mirrors the encoder and is structured as an expansive path.

After each decoder layer, transposed convolutions are performed to up-sample the feature maps' dimensions and

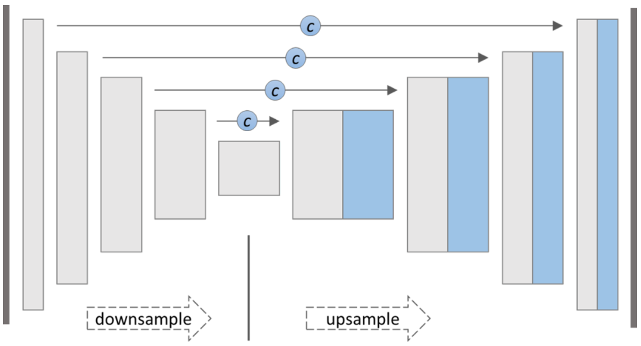


Fig. 15. The U-Net Architecture based on an encoder-decoder structure that will be used for the crack detection system.

compress the number of feature channels into half using transposed convolutions.

A skip connection concatenates the mirrored contracting path to its corresponding expansive path to provide local and global information during the up-sampling process. Consequently, the network is structured as a U-shape, hence the name, U-Net.

The U-Net architecture is implemented using the PyTorch library. It is an open-source machine learning framework that enables fast, flexible experimentation and efficient production. The PyTorch library easily selects between different encoders such as VGG16, ResNet34, or ResNet101.

E. Training Parameters

The VGG16 pre-trained on the ImageNet dataset is selected as the encoder for this application. Normalization is applied to the images using ImageNet's mean of [0.485, 0.456, 0.406] and standard deviation of [0.229, 0.224, 0.225]. The images are then converted to grayscale. The dimension of the input image is set to 448 x 448 pixels with an expected output of the same size.

The initial learning rate, learning rate decay factor and learning rate decay frequency are set to 0.001, 0.5 and 5, respectively. The optimizer used is the Stochastic Gradient Descent with momentum with the value of momentum set to 0.9.

For the loss function, binary cross entropy (BCE) with logits is used. This function is a more stable version of BCE as it combines a sigmoid layer before calculating the BCE loss. This function should be minimized during training where N is the batch size, yi is the ground truth, and (yi) is the predicted label and is given as:

$$BCE_{logits} = -\frac{1}{N}\sum_{i=0} y_i * log(\sigma(\hat{y}_i) + (1-y_i) * log(1-\sigma(\hat{y}_i)) \quad (1)$$

VII. Software Application Development Process

Software applications are developed to improve specific stages of the inspection procedure using the shelled UAV. These stages include data acquisition, damage detection and inspector evaluation. A software application for each stage is developed to improve and aid in the workflow of the inspectors. The software development process is shown in Fig. 16.

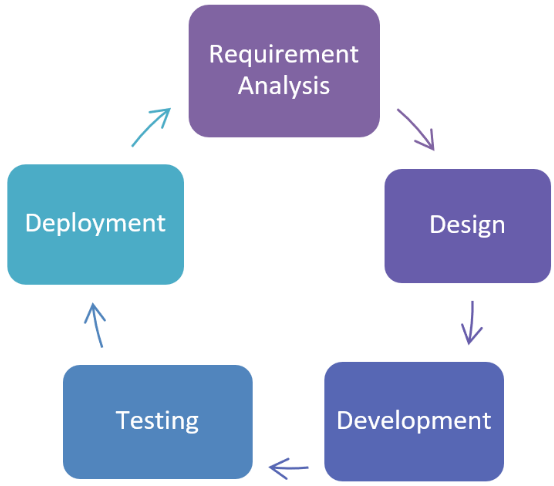


Fig. 16. The software development process includes data acquisition, damage detection and inspector evaluation.

The application for the data acquisition stage is used during field inspections. It shows a live stream of the video from the shelled UAV. An application that will aid in reviewing the data collected during the inspection is developed for the damage detection stage. The user can run crack detection on the video and then capture frames of interest for further review and annotation. Finally, the inspector evaluation application reviews the images of interest captured in the previous stage. The user can trace cracks on the image and let the application calculate the point-to-point distances of the trace. This allows the inspector to measure the crack's length and width.

The software application is written as a desktop application and developed using Python. The PyQt5 toolkit is used to develop the graphical user interface. It is developed to run on Windows systems. At the beginning of the development process, requirement analysis with potential stakeholders from DPWH is conducted. At this phase, the requirements are defined. Afterwards, a user flow diagram is designed to show how the features work together. Once the features are clearly defined, the design is implemented through coding. This is the development phase which outputs a working software application. After the development phase, the testing phase follows to examine the software's performance and functionality and address issues or bugs that will be encountered. It is then packaged and deployed for the user's work environment. This cycle repeats until all the requirements are met.

A. Software Requirements

One of the research goals of this study is to innovate the data collection, damage detection and evaluation processes that satisfy the needs of bridge inspectors. Table 2 summarizes the features of the three applications that will be developed based on the suggestions from DPWH personnel. The features encapsulate the needs of the inspectors, which is transformed into a digital platform.

TABLE II
SOFTWARE APPLICATION REQUIREMENT

Software Application	• Livestream drone camera view • Livestream operator's camera view • Show real-time crack detections • Record the drone-operation • Livestream • Input inspection details
Damage Detection	• Open videos were taken during the inspection. • Run crack detection on selected videos. • Show visualizations of crack detections • Show the frequency of crack detections for each frame under its corresponding timestamp. • Simultaneous playback of raw video and detection of video • Capture frame from the video
Evaluation	• Import frames from the Damage Detection App • View the imported frames one at a time. • Zoom in/out tool. • Pan tool • Trace tool • Calculate the crack length and width

B. Software Design: Data Acquisition App

The features listed in the previous section create a user flow diagram for each application. The diagram shows how the application will be used according to its purpose. The data acquisition application is used during field inspections. The application provides an inspection details form that will be filled out by the inspector. The details include the bridge's name, location, and description. These data are based on the Bridge Management System Manual provided by DPWH.

Before using the app for data gathering, connection to the drone camera via WIFI and the operator's camera via USB port should be set up. When both connections are established, a live feed from both cameras will be shown on the application. The drone camera shows a close-up view of the bridge structures as the drone is flown near the bridge. The operator's camera is ideally positioned to show the drone's location with respect to the bridge. This helps in localizing which part of the bridge the data is gathered. As the drone is operated to survey around the bridge, the inspector can start recording the live stream. If real-time detections are enabled, the application will show visualizations of the crack detections on the current frame. When the drone is done surveying a section or area, the inspector can stop the recording and download the high-resolution video from the drone camera.

Before saving the video and inspection details, the user can review and edit the data if there is mistyped information. If another survey inspection is performed, the user can record again. Fig. 17 shows the user diagram flow of the Data Acquisition App.

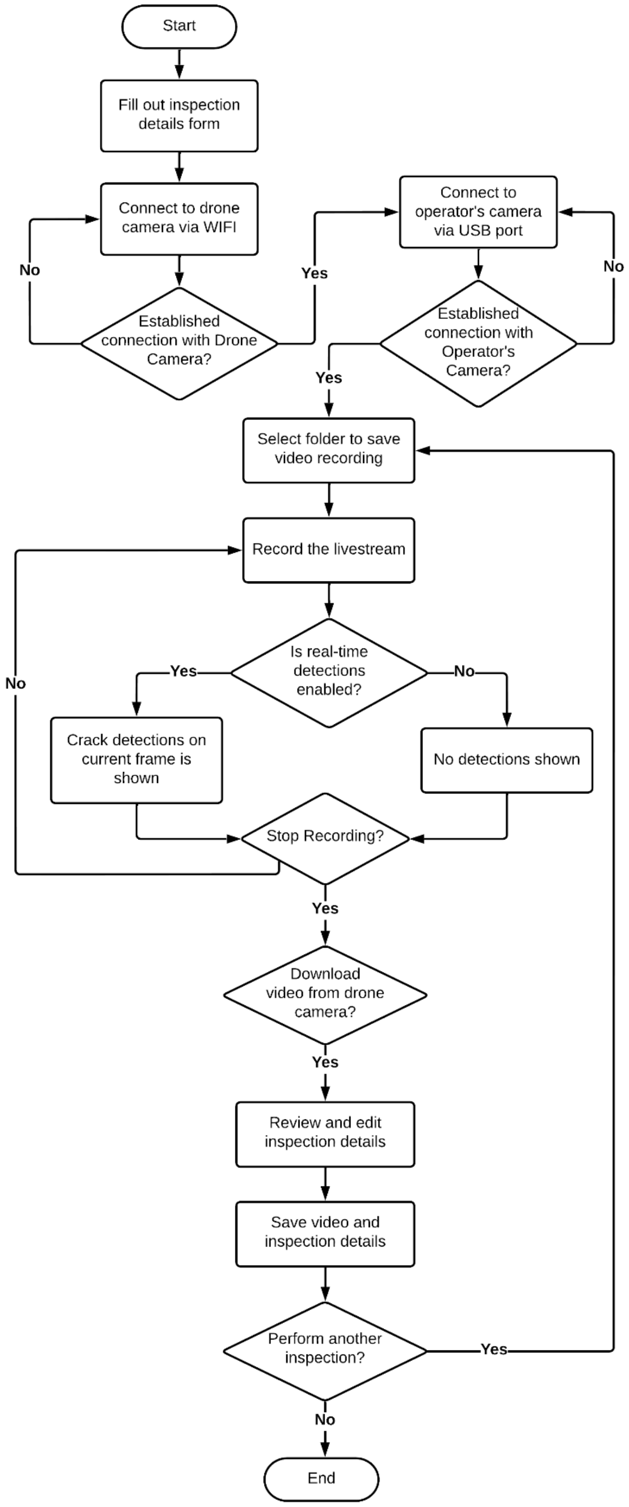


Fig. 17. The user diagram flow of the Data Acquisition App to be used during field inspections.

C. Software Design: Damage Detection App

The damage detection application is used after field inspection. Data collected from the Data Acquisition App can be imported into the app. After importing the data, the app automatically opens the video from the drone camera view and its corresponding view from the operator. If crack detection is not yet performed on the video, the user can run crack detection on the video. Otherwise, the application will automatically show the processed video with detections.

Before predicting cracks on the video, the user can select the detection threshold, either low, medium, or high. A low detection threshold sets the detection accuracy to 50%, showing more detections. A medium detection threshold sets the threshold to 75% accuracy and shows a medium number of detections. A high detection threshold sets the threshold to 90% accuracy but shows lesser detections; however, the detections shown are most likely true cracks.

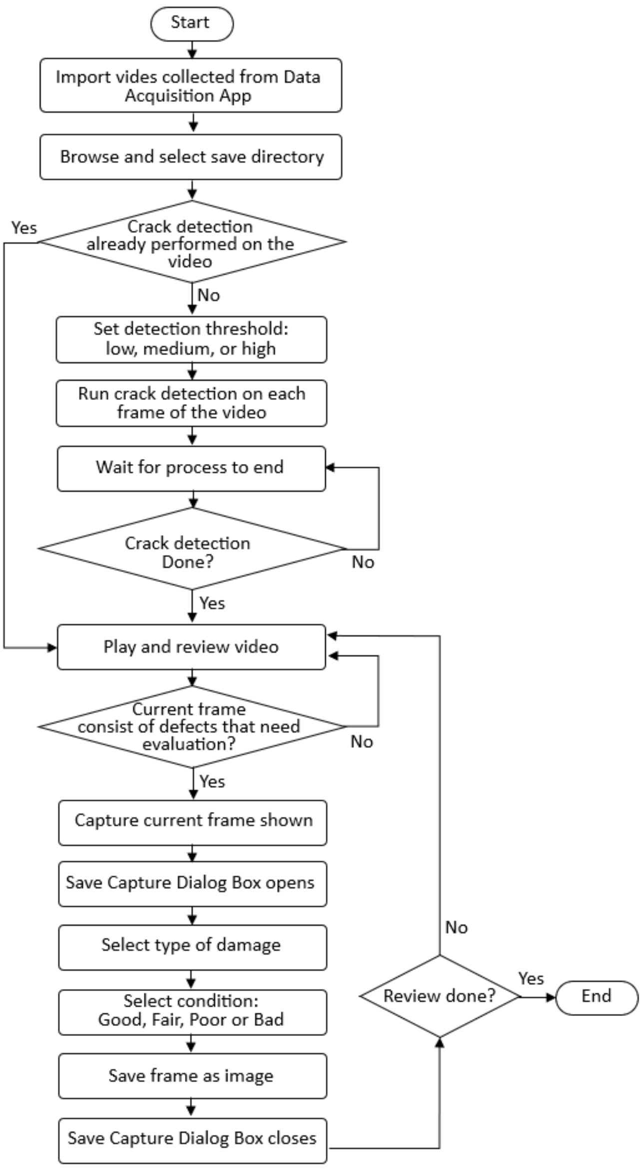


Fig. 18. The user diagram flow of the Damage Detection App to be used after field inspection.

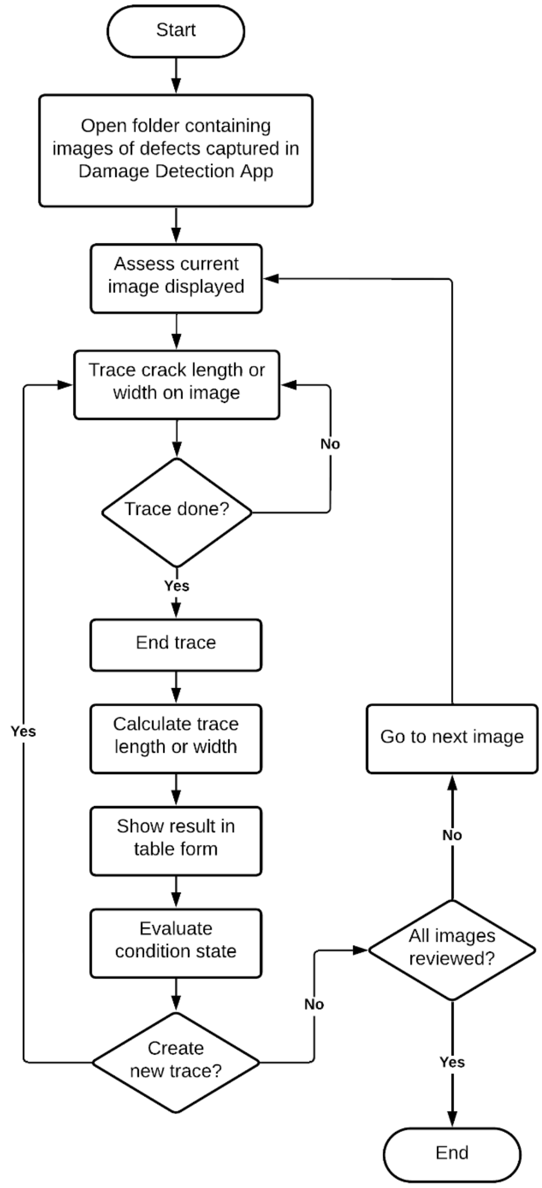


Fig. 19. The user diagram flow of the Evaluation App, specifically designed to annotate images with cracks as defects.

The user diagram flow of the Damage Detection App to be used after field inspection is shown in Fig. 18. When the app is done processing the detections, the processed video is automatically opened for playback and review. The inspector can play the video and closely observe which frames contain images of defects. Video playback controls such as play, pause, and skip are also provided. The user can also speed up or slow down the video. If the current frame shown consists of defects that need evaluation, the user can use the capture button to save the frame as an image. Before capturing, the user can choose the directory to save the images. When the capture button is clicked, a save capture dialogue box opens where the user can select the type of damage: crack, spalling/scaling/disintegration, rebar exposure/corrosion, honeycomb, water leakage or others. The user can also evaluate the condition state by indicating if it is good, fair, poor, or bad. The user can save the captured frame as an image if all the details are filled out.

D. Software Design: Evaluation App

The evaluation application is specifically designed to annotate images with cracks as defects. Instead of the traditional ribbon rods or measuring tapes used during field inspections to measure the length and width of cracks, the inspector can collate images of defects seen during the inspection and then organize them for annotation using the evaluation application. The user diagram flow of the Evaluation App, specifically designed to annotate images with cracks as defects, is shown in Fig. 19.

Multiple images captured in the Damage Detection App can be imported into the app. The user can review and go through each image one by one. A trace tool allows the user to trace cracks in the image. Traces for measuring crack lengths can be done by placing consecutive points that follow the geometry of the cracks which are of interest. A line that connects the points will be automatically drawn over the

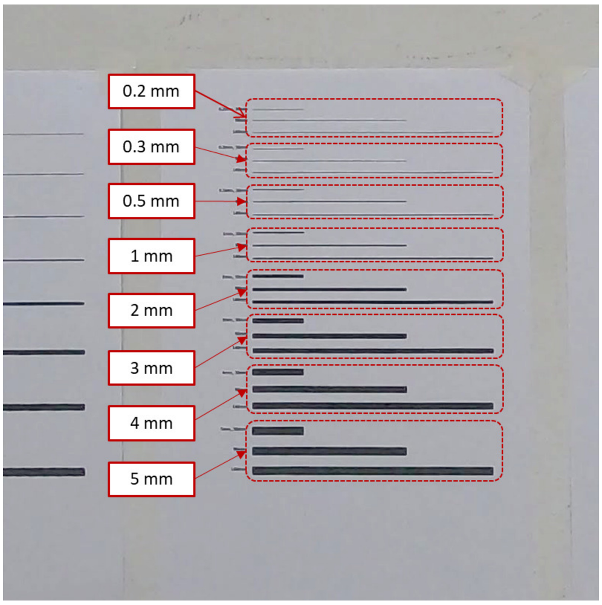


Fig. 20. Captured image at 450 mm from the wall using the installed camera on the drone.

points. The app derives and uses a function that converts the pixel distances into actual dimensions. The app automatically converts the trace into its actual dimension when the user clicks the calculate button. The results are summarized in a table format, wherein the dimension of a corresponding trace is listed. From the summary, the inspector can then assess the condition state of the cracks by indicating it in the app. Once the inspector reviews an image, the same process can be applied to the next images.

E. Pixel to Actual Dimension Calibration

The drone camera used for data acquisition is the Go Pro Hero 7. To get the pixel to the actual dimension formula, different lines with known dimensions are put up on the wall and taken by the camera at known distances. The subject placed on the wall has illustrations of lines with widths of 0.2mm, 0.3mm, 0.5mm, 1mm, 2mm, 3mm, 4mm, and 5mm. A sample of the captured image is shown in Fig. 20. The subject is taken at known wall-to-camera distances of 400mm to 800mm, with a step distance of 50mm. The equivalent pixel of the lines as seen on the digital image is recorded for each known dimension and distance.

To calculate the actual dimension of objects taken from the camera, a function with respect to the distance from the camera to the wall and the number of pixels is derived from the data gathered. The resulting formula derived is as follows:

$$dimension_{mm} = (0.00099*d+0.00142)*no.of pixels \quad (2)$$

VIII. Design Evaluation of the Newly Developed Shelled UAV

A. The Shelled UAV Components

After the meshed spherical shell and the gimbal unit were fabricated, these were assembled along with the base UAV. A slight modification was made to the base drone to attach it to the gimbal mechanism. Consequently, a Styrofoam ball

Fig. 21. Base drone SwellPro Fisherman FD1 unit) with the crack detection system.

Fig. 22. The shell with nylon mesh and attached gimbal inside the shell with drone holder.

was installed to enhance its floating capability and ability to perform water takeoff.

We selected the SwellPro Fisherman FD1 drone for the base UAV since it is highly suited for our purpose and meets our specifications, as shown in Fig. 21. It features an IP67 waterproof rating and a 2-kilogram payload capacity. The maximum flying period is 12 minutes with 1.5 kg weight sufficient.

The shell with attached nylon mesh has a specific intrusion area where no nylon net is attached, and inserted the drone and onboard camera inside the shell. The gimbal component holds the drone part, and the gimbal with the drone attached smoothly performs the roll rotation. The base drone holder also has a mechanism that allows drone rotation in the yaw axis. Hence, the newly developed shelled UAV features:

1. Passive rotating spherical shell with two-axis gimbal
2. Composed of two subassemblies – base drone (Fig. 21) and shell with a yaw-roll gimbal (Fig. 22)
3. Waterproof
4. Modular type
5. Capable of water takeoff and landing.
6. Integrated with the crack detection system.

Table 3 summarizes the weight budget of the newly

Fig. 23. The actual shelled UAV system consists of a shell with nylon mesh, a drone with an onboard camera and floater, and the gimbal component.

TABLE III
WEIGHT BUDGET OF SHELLED UAV

Shelled UAV Component	Weight (grams)
Spherical Shell	450
Gimbal Mechanism	370
Crack Detection System	155
UAV (without battery)	1552
Battery	616
Total Weight	3143

designed shelled UAV, having an overall system weight of 3143 g. The actual shelled UAV assembly consists of the drone with a camera mounted and supported by the gimbal mechanism, enclosed by the shell with nylon mesh, as shown in Fig. 23.

B. Simulation and Analysis of the Shelled UAV

This 1.5 kg weight of the drone is used for analyzing the strength of the gimbal component. For the gimbal mechanism, using FEA (Finite Element Analysis), the new gimbal system's structure was evaluated. The 2-DOF gimbal component design was put through simulations and theoretical investigations to determine its robustness. The drone and onboard camera weight were multiplied by 5 to assess if the gimbal component was safe at the greater weight.

The simulation stress results show greater stresses at the carbon fibre plate component, as shown in Fig. 24. The overall safety factor is equal to 4 since the impact factor of 2 for impact loading cases is multiplied by the recommended safety factor of 2 for the design. The allowable stress of the gimbal design used the overall safety factor of 4 multiplied by the stress concentration factor of 1.44. Based on maximum shear-stress failure theory, the maximum stress in shear from the simulation results due to the carbon plate component found is 3.345 x 107 N/m^2 which is less than the calculated allowable stress or allowable strength of the gimbal mechanism, 5.76 x 109 N/m^2. Thus, it was found that the gimbal design is overall safe.

Its drag force performance was investigated for the 950 mm diameter shell with nylon mesh. The CFD (Computational Fluid Dynamics) simulation is an excellent method for analyzing the shelled UAV system's aerodynamic performance. Fig. 25 shows the drag force comparison of the shelled UAV whole assembly, the shell with nylon mesh component (no drone and on-board camera), and the drone,

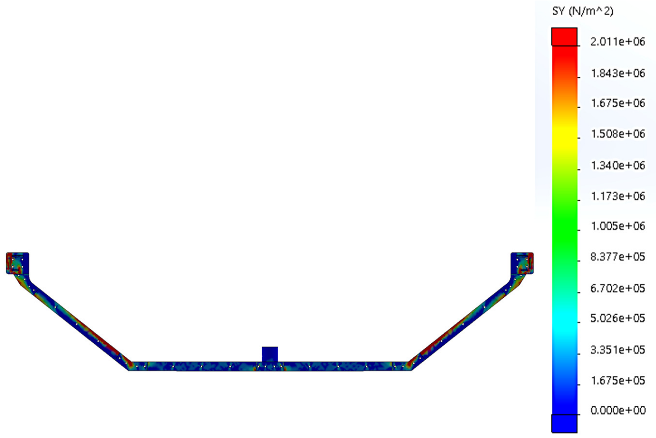


Fig. 24. The simulation stress results show greater stresses at the carbon fibre plate component.

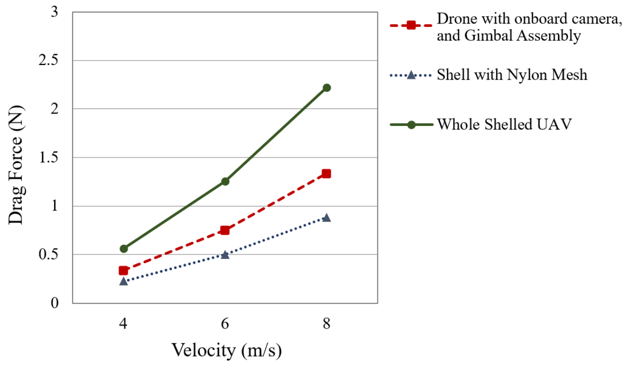


Fig. 25. Drag contribution comparison of the whole shelled UAV and its components due to wind velocity

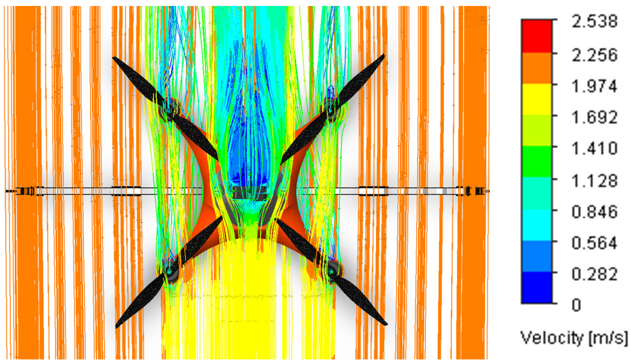


Fig. 26. Flow visualization showing wake regions on the onboard camera compared to the drone and gimbal assembly. But the gimbal mechanism minimized this effect, allowing rotations at pitch and roll directions.

on-board camera, and gimbal components (excluding the shell with nylon mesh component).

It was found that the drag contribution due to the shell with nylon mesh component gave an average of 33.40%, while the drone with an onboard camera and gimbal gave an average drag contribution of 40%. The SwellPro Fisherman FD1 drone structure is designed to lessen drag effects. That is why the drag contribution of the drone alone is lesser than the other components. However, the onboard camera contributes to more drag force for the drone to experience as these two components are treated as one assembly for visually inspecting bridges.

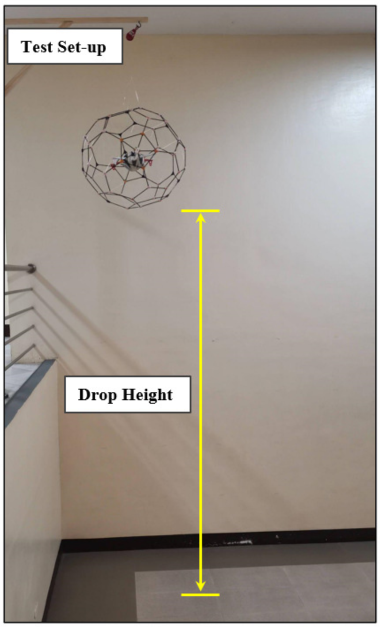


Fig. 27. Drop test experimental setup for the shell with nylon mesh and was implemented at specific drop heights.

TABLE IV
DROP TEST RESULT FOR THE SHELL WITH NYLON MESH

Drop Height (meters)	
0.5	No damage
1.5	No damage
2	No damage
2.5	No damage
3	Minor damage; One joint has a small crack

At 8 m/s wind velocity, the drone with an onboard camera gave a 0.229 N drag difference compared to the gimbal mechanism. The design of the gimbal mechanism gave more drag force because its design has rectangular plates perpendicular to the wind direction, or its projected area is greater. That is why there is a 0.45 N difference for the drone with camera and gimbal components compared to the shell with nylon mesh.

It can be seen in Fig. 26 that the onboard camera has wake regions compared to the drone and gimbal assembly. But the gimbal mechanism minimised this effect, allowing rotations at pitch and roll directions. And the drag force contributed by the onboard camera and gimbal is a good trade-off for the effective flight operation of the shelled UAV. This can be verified as well by considering the stability of the shelled UAV design during field demonstration.

C. Experimental Test and Evaluation

Since the spherical shell serves as the physical protection of the main drone, it is expected to experience collision or impact during operation. As shown in Fig. 27, a drop test was implemented for the shell with the nylon mesh. It was found that no significant damage was spotted until it reached 3-meter height, wherein one carbon rod was broken. This significant damage may be due to fatigue experienced by the shell with nylon mesh since it had already dropped from 2 meters up to 3 meters, as summarized in Table 4. To verify this result, the researchers conducted a drop test on the 3-meter height using a new shell, and minor damage was observed on the shell with nylon mesh. This minor damage is caused by a joint forming a small crack. But this small crack is insignificant because the material used, ABS plastic, can withstand up to 43 kg. force per square centimeter area equivalent to greater than 3-meter drop height [11].

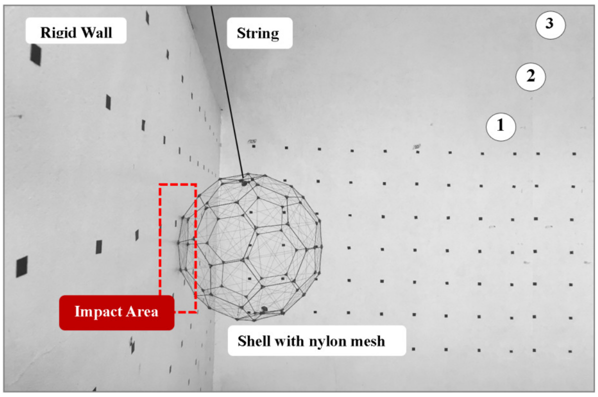


Fig. 28. The collision test setup for the shell with nylon mesh to further verify the strength of the shell in the drop test. The shell is released from a position where the center of mass is dropped at varying collision drop angles while maintaining a taut string.

TABLE V
COLLISION TEST RESULT FOR THE SHELL WITH NYLON MESH

Setup	*Collision Drop Angle*	
1	60 degrees	No damage
2	75 degrees	No damage
3	90 degrees	No damage; Deformation is more localized around the impacted joint

D. Experimental Test and Evaluation

A collision test with varying values of collision drop angle was conducted further to verify the strength of the shell with nylon mesh. The shell is released from a position where the center of mass is dropped at collision drop angles 60, 75, and 90 degrees while maintaining a taut string. The string was attached to the fixed, rigid wall while the other end was on the shell. The shell was made to collide at the impact area, as shown in Fig. 28, like a pendulum motion. Moreover, the resultant force due to the 90-degree angle with respect to the wall yields maximum impact force. As summarized in Table 5, it was found that the extent of deformation was observed at the maximum collision drop angle of 90 degrees, where the string and the drop height formed a 90-degree angle. The shell was not damaged, but the deformation is more localized around the impacted joint at maximum collision drop angle. This localized deformation found no damage to the shell component. Thus, proved that the nylon mesh added to the shell contributes to the overall stiffness and resilience of the shell alone due to the material of the nylon mesh.

IX. DEVELOPED CRACK DETECTION SYSTEM

The crack detection model is trained on an NVIDIA GeForce RTX 2070 GPU with 8 GB memory. The curated

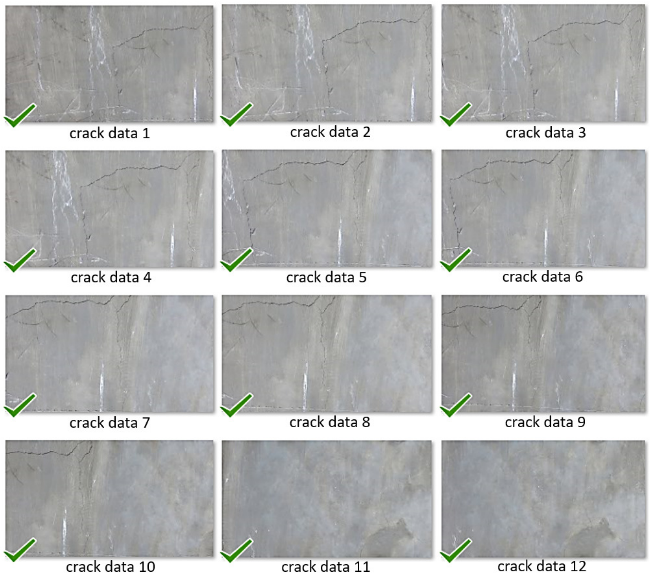


Fig. 29. Some images of concrete walls with cracks, with a total of 2,667 images, were collected for the crack dataset.

dataset consists of 3,820 images of surface cracks with and without shell obstructions. Since the curated dataset only consisted of thin surface cracks, an additional 11,298 images taken from public datasets with varying sizes of cracks were incorporated into the training. The final dataset consists of unobstructed and obstructed images of surface cracks. Images without cracks were also included. The model was trained for 55 epochs with a batch size of 4.

A. Crack Dataset

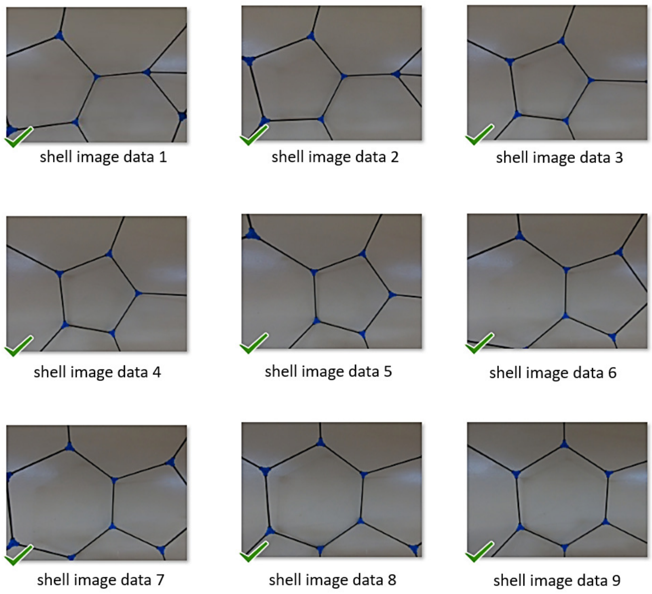
The images gathered for the crack dataset were collected in MSU-IIT. The images are frames extracted from video recordings of wall cracks. Fig. 29 shows some images of concrete walls with cracks. A total of 2,667 images are collected for the crack dataset. The dataset is composed of unobstructed images of cracks on concrete walls. And to gather images in tall structures, the data gathering system employed a camera mounted on a 15-ft custom-made wooden adjustable stick.

B. Shell Pixels Dataset

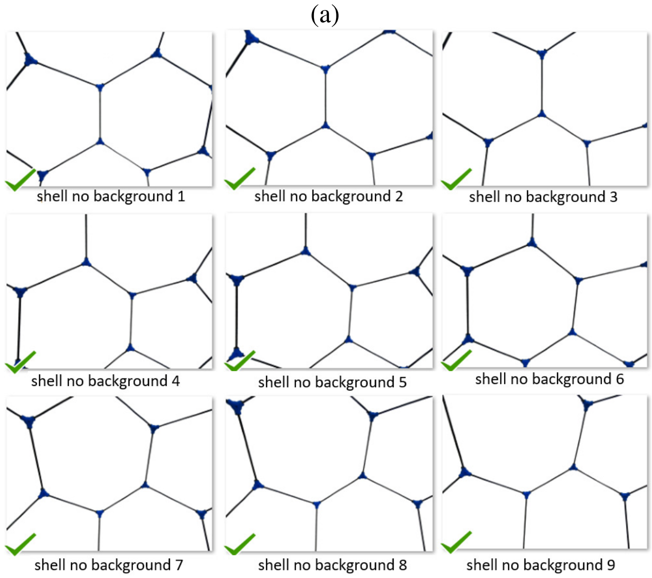
A preliminary protective shell was built for the purpose of data gathering. A video with the protective shell rotating is captured to obtain different orientations and positions of the protective shell. Frames from the video were extracted. A total of 4,296 shell images were extracted from the videos. Fig. 30a shows some of the shell images collected. Background removal was done in Photoshop. 1,215 images were edited and had their background removed as implemented in Fig. 30b.

C. Crack + Shell Pixels Dataset

Images from the Crack Dataset, Dc, are superimposed with images from the Shell Pixels Dataset, Ds', to build the Crack + Shell Pixels Dataset, Ds' + Dc. The image processing flow to build this dataset is shown in Fig. 30. A script is written for batch processing of the images in Photoshop. This allows for a faster workflow in building the synthetic data set, as shown in Fig. 31.



(a)



(b)

Fig. 30. Shell images were collected (a) 1,215 images were edited and had their background removed (b).

D. Training Results

The result of the training is shown in Fig. 32. The graph illustrates a decreasing trend of training and validation losses. This shows that the model was able to learn from the dataset. At epoch 55, the least validation loss of 0.04862 was obtained with the value of the training loss at 0.04629.

E. Performance Evaluation

In evaluating the performance of the trained model, the precision, recall and F1-score are calculated. Precision indicates the model's performance in terms of its accuracy in predicting true positives out of all total predicted positives. This metric is calculated as:

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

Recall calculates the model's accuracy in predicting true

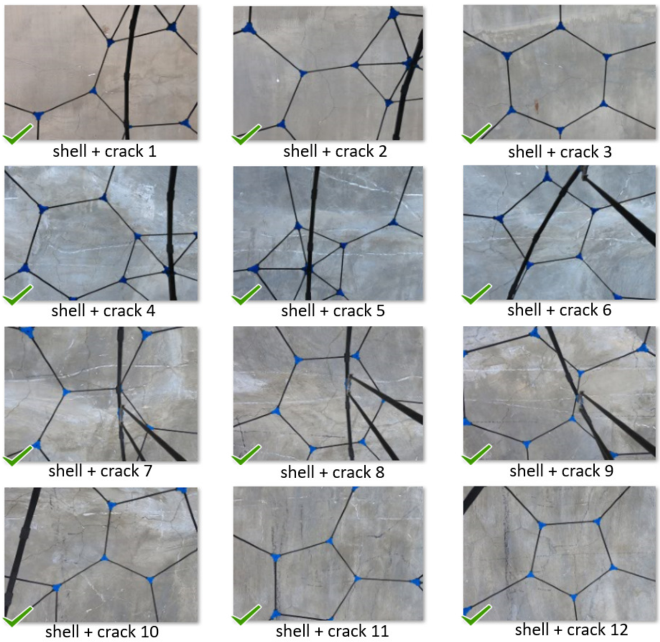


Fig. 31. The shell pixels superimposed on crack images (synthetic data).

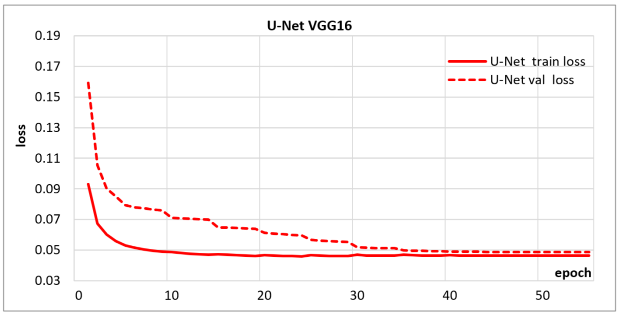


Fig. 32. Training and validation Loss of U-Net VGG16 trained on the CrackUAS dataset.

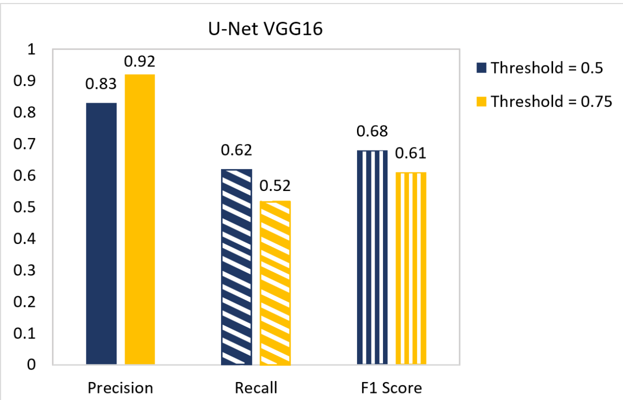


Fig. 33. Precision, recall, and F-1 score results, the trained model achieved an F1 score of 68% at a threshold of 0.5 and decreased to 61% as the threshold was increased to 0.75.

positives out of all actual positives. This metric is calculated as:

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

The F1 score metric considers both precision and recall and is often used in cases where there is an uneven class

TABLE VI
NETWORK LATENCY OF U-NET VCG16

Network Latency (ms)	
300 samples	83.94
500 samples	85.13
1000 samples	85.63
Average Network Latency	84.90

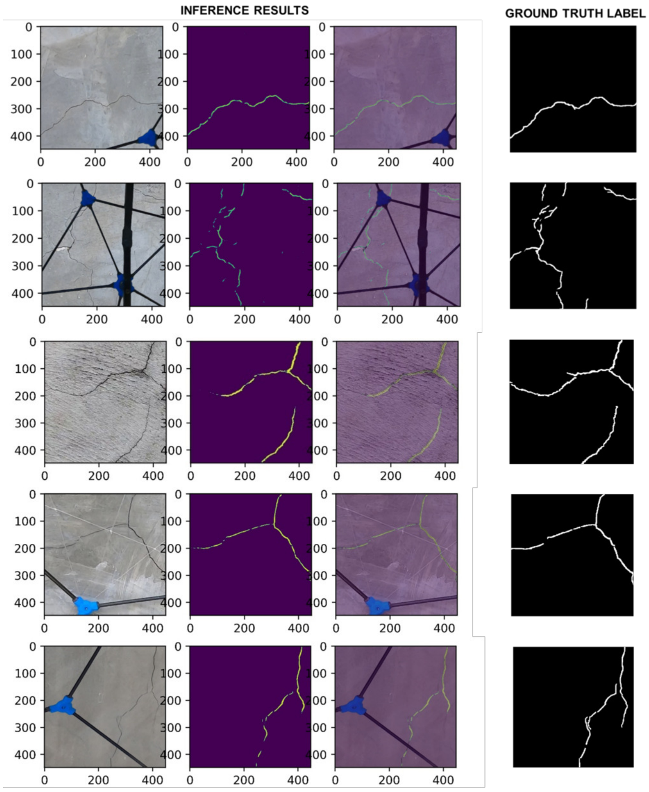


Fig. 34. Inference results of training on the CrackUAS dataset where images shown are from the results of the inference on the validation dataset with the threshold set to 0.75.

distribution. In the case of the CrackUAS dataset, there is a larger number of pixels belonging to the background class and lesser pixels that are cracked. The F1-score metric is calculated as follows:

$$F_1 = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{5}$$

The trained model was evaluated on the validation dataset. The results of the predictions at two thresholds are then compared, with values: 0.5 and 0.75. Pixel predictions with a probability higher than the threshold are labeled as a crack. Otherwise, it is labeled as background. Fig. 33 summarizes the evaluation results regarding precision, recall, and F1-score.

Precision measures how many of the predicted crack pixels are true cracks. In terms of precision, the results show that the precision increases as the threshold increases. This is expected since the high threshold ensures that most positive predictions are correct. During deployment, the inspector can adjust and fine-tune the threshold parameter according to their preference.

Recall determines how often the model misses out on the true cracks. A high recall means that the model can determine most of the true cracks given an image.

The results show a decreasing recall trend as the threshold increases from 0.5 to 0.75. This is expected since the pixels with low probabilities are classified as background instead. Recall determines how often the model misses out on the true cracks. A high recall means that the model can determine most of the true cracks given an image. The results show a decreasing recall trend as the threshold increases from 0.5 to 0.75. This is expected since the pixels with low probabilities are classified as background instead. The F1 Score summarizes the overall performance of the trained model as it considers both precision and recall. The trained model achieved an F1 score of 68% at a threshold of 0.5, as shown in Fig. 33. It decreased to 61% as the threshold increased to 0.75.

1) Network Latency: To evaluate the speed of the model, the network latency is calculated. It is measured as the time it takes for an input to go through the feed-forward of the network. The inference is done on a batch of 300, 500, and 1000 samples to get the average network latency. Table 6 shows that the model's average network latency is 84.90ms.

When running on an NVIDIA GTX 2070 GPU, the hardware can process a maximum batch size of 12. The network can process this optimal batch size in the given hardware. Given this optimal batch size, the number of images the network can process in a second is 16 samples. This is also called throughput. This value is dependent on the data, model, and device.

2) Visualization Results : A visualization of the inference on the CrackUAS dataset is shown in Fig. 34. The images show the inference results on the validation dataset with the threshold set to 0.75. Without obstructions, the trained model can detect actual cracks in the images. Comparing the predicted cracks with the ground truth could discriminate the shell obstructions from the cracks. Fundamentally, this model is deployed in the software applications developed for bridge inspections.

X. Developed Software Application

The software applications developed for data acquisition, damage detection and inspector evaluation were implemented according to the requirements set by the bridge inspectors. This section demonstrates the working application developed. The applications were tested according to the features implemented.

A. Data Acquisition App GUI

The main window of the Data Acquisition Application is divided into 4 panels: Drone Camera View, Operator's Camera View, Detections, and Inspection Details. The application's graphical user interface is shown in Fig. 35. The Drone Camera View and Operator's Camera View have dropdown boxes to select which camera to connect to and a Connect button to establish the live stream connection. A record button is placed in the Drone Camera View panel. The record button, when pressed, sends a record command to the drone camera and records the operator's view simultaneously.

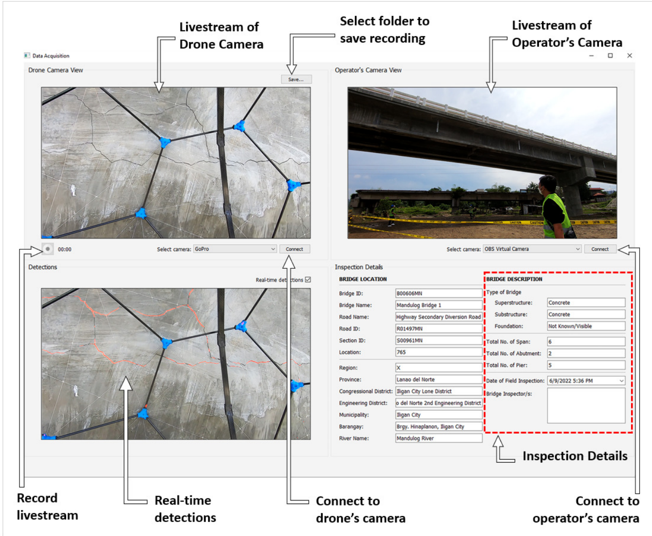


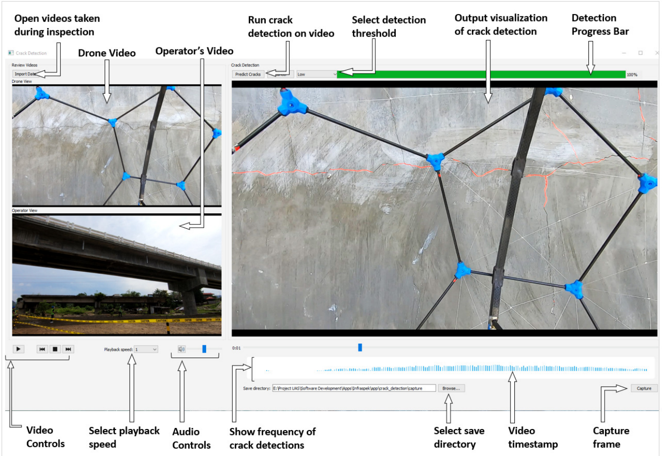
Fig. 35. The application's graphical user interface, the main window of the Data Acquisition Application, is divided into 4 panels: Drone Camera View, Operator's Camera View, Detections, and Inspection Details.

The same button is clicked to stop the record. Real-time crack detections are shown in the Detections panel. Because performing real-time detections in every live stream frame is computationally expensive, the sampling frequency in running crack detection is set to 1 frame per second. The Inspection Details panel is where the inspector inputs details of the bridge location and bridge description.

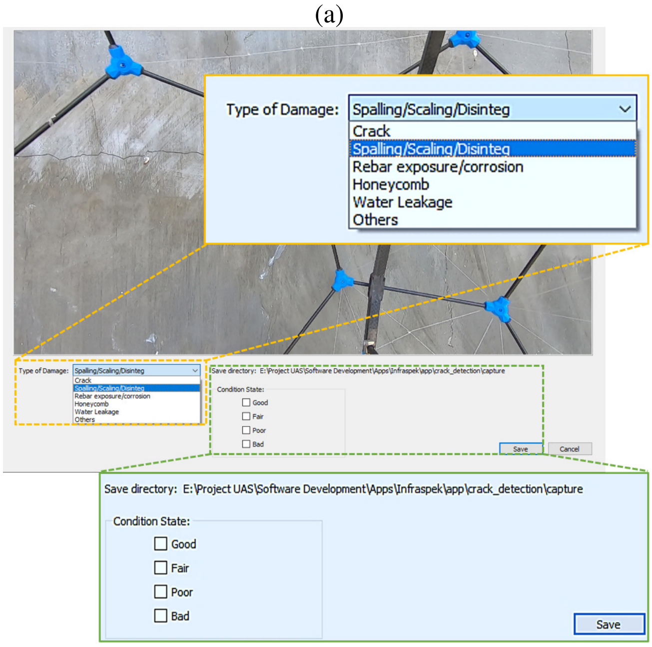
B. Damage Detection App GUI

shows 3 main panels: The Drone View, Operator View, and Crack Detection, as shown in Fig. 36. The Drone View shows the video gathered using the Data Acquisition App and the Shelled-UAV. The Operator View shows the video recording of the operator's perspective, which is recorded simultaneously with the Drone's view. The Crack Detection panel shows the results of running crack detection on the Drone View. A drop-down menu to select the detection threshold is shown at the panel's top. A progress bar is also placed adjacent to the dropdown menu. It indicates the progress of converting the drone view video into a video with detections. Video playback, speed and audio controls are also provided below the main panels. A video timeline is provided below the detection video. Beneath the timeline is a graph that shows the frequency of crack detections under that corresponding timestamp. A save directory browser is also placed below the main window where the user can browse and select the file directory to save the captured images. The capture button is provided at the bottom right corner of the window.

When the capture button is clicked, a Save Capture Dialog box opens. In the dialogue box, a dropdown menu of the type of damage is provided to allow the user to categorize the defect observed on the frame. The user can also check the condition state of the damage seen on the image. This selection is based on the inspector's assessment of the defects. Finally, a save button will save the image on the save directory indicated by the user.



(a)



(b)

Fig. 36. The main windows of the Damage Detection Application (a) and select the type of damage, condition state, and save directory (b).

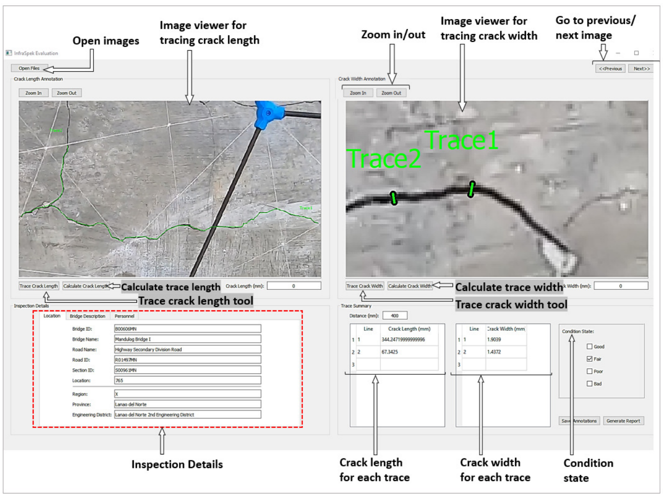


Fig. 37. The main windows of the Evaluation Application are divided into 4 panels: Crack Length Annotation, Crack Width Annotation, Inspection Details, and Trace Summary.

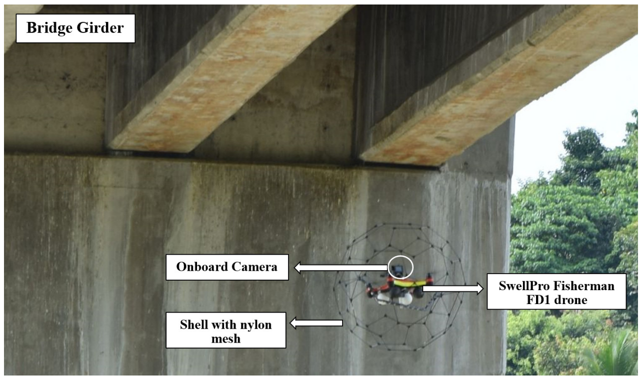


Fig. 38. The shelled UAV with onboard camera inspecting the bridge girder. The onboard camera provides real-time images of the bridge that the software application used for the Data Acquisition App.

C. Evaluation App GUI

The main window of the Evaluation App is divided into 4 panels: Crack Length Annotation, Crack Width Annotation, Inspection Details and Trace Summary, as shown in Fig. 37. An Open File button is in the top left corner of the app. This button opens a File Dialog box where the user can select multiple images which can be viewed and annotated in the app. The Crack Length and Crack Width Annotation panels consist of similar controls. Both panels display the same current image selected. The difference is that the left image is provided for annotations of crack lengths, while the right image is used to annotate crack widths.

Each panel has its own zoom-in/out buttons. Hovering the mouse on the image viewer and then clicking while dragging helps the user to pan around the image. Each panel provides a Trace Crack tool where the user can click on the image to trace the path of the crack. To start a trace, click the Trace Crack Length or Width button. Once a trace is done, the Trace Crack button is converted into a Done button. Clicking on this ends the trace, and the app automatically assigns the trace number. The user can then click the Calculate Crack Length/Width button to show the crack length on the Trace Summary Panel. The Inspection Details panel shows details of the bridge linked to the data gathered. Finally, the user can select the condition state of the image before going to the next image. To go through the selected images, the Previous and Next buttons are provided in the upper right corner of the window.

XI. Actual Bridge Assessment

An actual field assessment of the shelled UAV was made on a bridge in Iligan City, Philippines. The shelled UAV inspected the bridge girder for possible cracks detected, as shown in Fig. 38. A maximum wind gust of 6 m/s was recorded. The developed shelled UAV with an onboard camera provided real-time images of the inspected bridge and a live stream of the drone camera view on a physical computer. These real-time images and video transmissions were implemented in the developed software application.

A. Software Application Functionality Assessment

Table 7 lists the characteristics of the implemented Data Acquisition App. Each feature's functioning was evaluated

TABLE VII
FEATURES OF DATA ACQUISITION APPLICATION

Features	*Status*	*Remarks*
Select the camera to connect to the drone/operator's view from a list of available cameras	Passed	The application can connect to the drone's camera via WIFI and the operator's camera via a USB port
Livestream Drone Camera View	Passed	A live stream from the drone's camera can be viewed in the app
Livestream Operator's Camera View	Passed	A live stream from the drone's camera can be viewed in the app
Show real-time crack detections	Passed	Frames taken from the live stream are processed for crack detection Visualization of the detections is shown.
Record the drone-operator Livestream	Passed	The user can record the live stream using the record button.
Inspection Details	Passed	An inspection details form is provided to be filled out by the inspector. It is exported into an XML file whenever the live stream is recorded.
Save original video resolution from Drone Camera View	Passed	A video recording of the drone's live stream is saved locally on the computer
Save original video resolution from Operator's Camera View	Passed	A video recording of the operator's live stream is saved locally on the computer
Support multi-camera livestream	Passed	The drone's camera and the operator's camera can be viewed simultaneously in the app.

to see if it produced the desired result. and found that each feature passed the test and was functional.

Moreover, the features of the Damage Detection App implemented for the shelled UAV are listed in Table 8. Each feature was tested and verified if the expected response or outcome is met. The table below shows that all features were functional and have passed the test.

Table 9 includes a list of the Inspection Details App's features that were implemented. Each feature was tested to see if it produced the desired response or result. The table summarizes that each feature passed the test and was also functional. Thus, the software application integrated for the shelled UAV is overall functional.

B. Shelled UAV Performance Assessment

Since the gimbal component holds the drone part, and the gimbal with the drone attached performs the roll rotation, the base drone holder also has a mechanism that allows the rotation of the drone in the yaw axis. Thus, rotation data of the drone was gathered. Fig. 39a and 40a show how the onboard camera gave clear images of the area that is being inspected.

The onboard camera image demonstrates how bridge cracks were discovered during the bridge's actual flight test. Moreover, the rotational data of the drone at a 200-second time frame were investigated. These images were evaluated in the integrated software application for the shelled UAV.

TABLE VIII
FEATURES OF DAMAGE DETECTION APPLICATIONS

Features	*Status*	*Remarks*
Open videos that were taken during the inspection	Passed	The user can open the XML file generated from the data acquisition app to review the data gathered during the inspection.
Run crack detection on videos and save visualizations and masks as videos	Passed	The app can convert drone videos into video with visualizations of crack detections.
Show visualizations of crack detections after predicting cracks on the drone video	Passed	Red segmented masks on the drone video are drawn to represent the crack detections.
Select detection threshold	Passed	The user can select three levels of detection: • Low (default): 50% detection accuracy but shows more detections. • Medium: 75% detection accuracy, shows a medium number of detections. • High: 90% detection accuracy, shows lesser detections; however, the detections are most likely true cracks
Show the frequency of crack detections under the time slider of the video.	Passed	A bar graph of the number of crack pixels of each frame is shown under the time slider of the video.
Simultaneous playback of raw video and detection of video.	Passed	The drone, operator and The drone, operator and be played simultaneously time slider of the video.
Capture frame from video and export along with the detections.	Passed	The user can capture frames from the video. A dialogue box pops up wherein the user can specify the type of damage shown on the image.
Audio controls: mute, volume up/down. detections.	Passed	The user can mute or change the volume of the video
Video controls: play, pause, stop, skip backwards / forward, playback speed.	Passed	The user can play, pause, stop, or skip the video backwards / forward and change the playback speed.

TABLE IX
FEATURES OF THE INSPECTION DETAILS APPLICATION

Features	*Status*	*Remarks*
Open exported frames from Crack Detection App	Passed	• The user can open the images in a selected folder • All images in the selected folder can be viewed in the app one at a time • The user can navigate through the set of images by choosing which image to viewcurrently
Calculate crack length and width Detection App	Passed	The app can calculate length and width of the crack for each path created by the user
Crack Annotation Tool	Passed	The user can trace the cracks by creating pathways along the cracks of interest
Zoom in/out tool	Passed	The user can zoom in/ out of the images
Pan tool	Passed	The user can pan around the images
Export annotations	Not Implemented	For future improvements
Generate summary report	Not Implemented	For future improvements

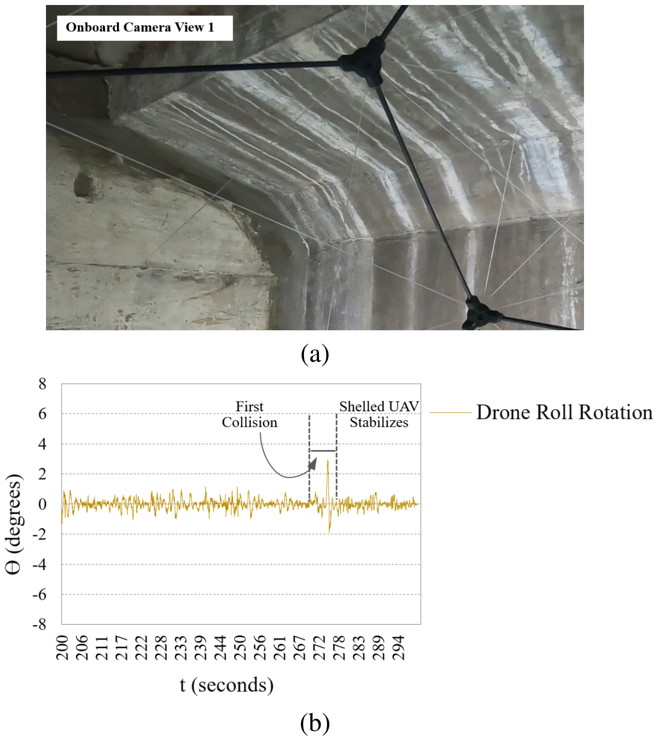


Fig. 39. Image clarity of the onboard camera view of the drone during the first collision time frame (a) and the gathered drone rotation data of the drone showing system stability after the event of the first collision, where shelled UAV went through the bridge girder's interior (b).

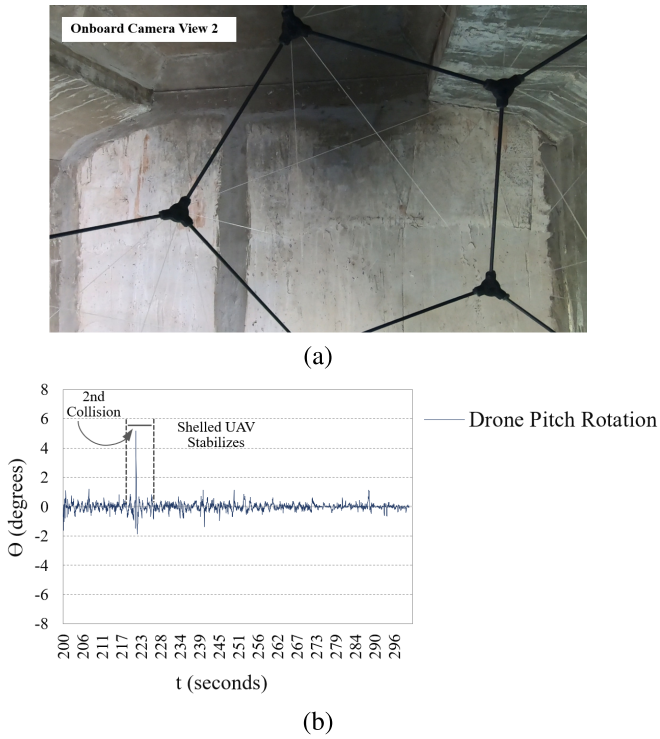


Fig. 40. Image clarity of the onboard camera view of the drone during the second collision time frame (a) and the gathered drone rotation data of the drone showing system stability after the event of the second collision, where shelled UAV went through the bridge girder's interior (b).

As shown in Fig. 39b and 40b, it was found that the shelled UAV demonstrated system stability following the impact (from the first collision). As the system approaches its stable state, the resulting roll rotation maximum amplitude values steadily decrease to less than 1 degree only.

The shelled UAV is waterproof; thus, it has water take-off and landing features. In bridge inspections, after inspecting the bridge girders with the river at the bottom of it, there is a possibility that the shelled UAV may experience signal losses or other unexpected scenarios. That is why, in the actual field test, the shelled UAV was dropped into the river, as shown in Fig. 41. The shelled UAV then took off after 10 seconds of landing still in the water to show that the system is capable of water take-off.

XII. Conclusion

In this paper, a shelled UAV system that structural inspectors can use to perform a close visual inspection of the infrastructure was developed. The newly developed shelled UAV features a passive rotating shell with a two-axis gimbal, composed of a base drone and shell with the yaw-roll gimbal. The shelled UAV system is waterproof, modular, capable of water takeoff and landing, and integrated with a crack detection system. A functional shelled UAV was fabricated, and test flights were successfully conducted.

The computational simulations and actual experiments showed that the shelled UAV could tolerate great impact forces during a collision and withstand drag due to wind gusts. The gimbal design is overall safe and effective. The drag force contributed by the onboard camera and gimbal is a good trade-off for the effective flight operation of the shelled UAV. Moreover, the actual flight test of the shelled

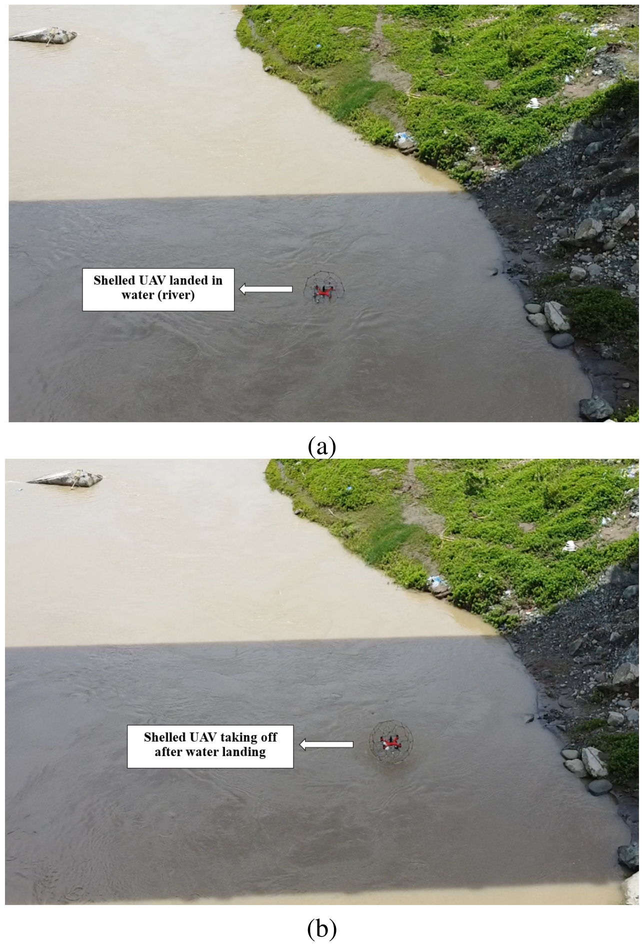


Fig. 41. The shelled UAV demonstrated its capability of the water landing (a) and take off (b).

UAV showed system stability, especially during a collision. The onboard camera also gave clear images of the area that was being inspected.

In this study, crack detection systems and software applications were developed. A curated dataset, CrackUAS, was also produced to train the crack detection system. The U-Net architecture was used as the segmentation model trained on the dataset. The trained model produced a good performance and could predict and segment cracks in the gathered dataset images. The software applications developed for data acquisition, damage detection and inspector evaluation were also implemented. The Damage Detection App converted the drone videos into a video with visualizations of the crack detections. Frames taken from the live stream were successfully processed for crack detection, and visualization of the detections was shown. The drone, operator and crack detection videos were played simultaneously as well. The Inspection Details App successfully calculated the length and width of the crack for each path created. Furthermore, each Damage Detection App and Inspection Details App feature was tested and verified. The results showed that all shelled Unmanned Aerial system features were functional and effective.

References

[1] Bridge Inspections. Flyability (2022). Available online: https://www.flyability.com/bridge-inspections/

[2] Bridge Management System; Bridge Inspection Manual. DPWH (2007). Available online: https://www.dpwh.gov.ph/dpwh/

[3] B. Chan, H. Guan, J. Jo, and M. Blumenstein, "Towards UAV-based bridge inspection systems: a review and an application perspective," Struct. Monit. Maint., vol. 2, no. 3, pp. 283–300, 2015, http://dx.doi.org/10.12989/smm.2015.2.3.283

[4] Miswar, Suhardi and N. B. Kurniawan, "A Systematic Literature Review on Survey Data Collection System," 2018 International Conference on Information Technology Systems and Innovation (ICITSI), Bandung, Indonesia, 2018, pp. 177-181, doi: 10.1109/ICITSI.2018.8696036.

[5] Dorafshan, S., Maguire, M. (2018). Bridge inspection: human performance, unmanned aerial systems and automation. Journal of Civil Structural Health Monitoring, 8, 443 - 476.

[6] Dorafshan, S., Maguire, M., Hoffer, N., Coopmans, C. (2017). Challenges in bridge inspection using small unmanned aerial systems: Results and lessons learned. 2017 International Conference on Unmanned Aircraft Systems (ICUAS), 1722-1730.

[7] Roberts, J.F., Stirling, T.S., Zufferey, J.-C., and Floreano, D., Quadrotor Using Minimal Sensing for Autonomous Indoor Flight, Eur. Micro Air Vehicle Conf. and Flight Competition (EMAV2007), 2007.

[8] Bachrach, A., He, R., Roy, N. (2009). Autonomous Flight in Unknown Indoor Environments. International Journal of Micro Air Vehicles, 1, 217 - 228.

[9] Salaan, C.J., Tadakuma, K., Okada, Y., Takane, E., Ohno, K., Tadokoro, S. (2017). UAV with two passive rotating hemispherical shells for physical interaction and power tethering in a complex environment. 2017 IEEE International Conference on Robotics and Automation (ICRA), 3305-3312.

[10] Mizutani, S., Ohno, K., Yanagimura, K., Okada, Y., Takeuchi, E., Tadokoro, S. (2014). 2A1-E04 Quadrotor with a Rotating Spherical Shell Which Can Fly in the Complex Structure (Aerial Robot and Mechatronics (1)).

[11] Librado, L.G., Cahig, C.Y., Banglos, C.A., Pao, J., Salaan, C.J. (2022). Meshing and Insulation Strategies for Shelled UAVs in Proximal Inspection of Distribution Utility Lines. IEEE Robotics and Automation Letters, 7, 5278-5285.

[12] Jeanette Pao, Carl John Salaan, Charles Alver Banglos, Karl Martin Aldueso, Lester Librado, and Jonathan Maglasang, "Shelled Drone with Meshed Net Performance Evaluation through Aerodynamic Analysis and Vibration Response Investigation," Engineering Letters, vol. 30, no.4, pp1573-1587, 2022

[13] Y. Liu, J. Yao, X. Lu, R. Xie, and L. Li, "Deepcrack: A deep hierarchical feature learning architecture for crack segmentation," Neurocomputing, vol. 338, pp. 139–153, 2019.

[14] Eisenbach, M., Stricker, R., Seichter, D., Amende, K., Debes, K., Sesselmann, M., Ebersbach, D., Stoeckert, U., Groß, H. (2017). How to get pavement distress detection ready for deep learning? A systematic approach. 2017 International Joint Conference on Neural Networks (IJCNN), 2039-2047.

[15] Dorafshan, S., Thomas, R.J., Maguire, M. (2018). SDNET2018: An annotated image dataset for non-contact concrete crack detection using deep convolutional neural networks. Data in Brief, 21, 1664 - 1668.

[16] Ronneberger, O., Fischer, P., Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. ArXiv, abs/1505.04597.

[17] Chaurasia, A., Culurciello, E. (2017). LinkNet: Exploiting encoder representations for efficient semantic segmentation. 2017 IEEE Visual Communications and Image Processing (VCIP), 1-4.

[18] Lin, T., Dollár, P., Girshick, R.B., He, K., Hariharan, B., Belongie, S.J. (2016). Feature Pyramid Networks for Object Detection. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 936-944.

[19] Zhou, Z., Siddiquee, M.M., Tajbakhsh, N., Liang, J. (2019). UNet++: Redesigning Skip Connections to Exploit Multiscale Features in Image Segmentation. IEEE Transactions on Medical Imaging, 39, 1856-1867.