Journal of Engineering Science and Technology
Special Issue on SIET2022, May (2022) 132 - 156

AUTONOMOUS UAV INSPECTION SYSTEM FOR PREVENTIVE MAINTENANCE OF SOLAR FARMS

MOHAMMED GAMAL AL-GAWDA, CHEE YONG LAU*, NAI SHYAN LAI

School of Engineering, Asia Pacific University of
Technology and Innovation, Kuala Lumpur 57000, Malaysia
*Corresponding Author: laucheeyong@apu.edu.my

Abstract

The expansion in energy consumption led to the growth in providing more sustainable energy resources and inspection techniques to sustain their development. This expansion will require systematic routine inspections in solar facilities to maintain safe conditions and provide early check-in for natural occurring defects that deteriorate the power efficiency. However, large solar power plants are established in large areas. Traditionally, maintenance crews will personally check the modules or be sent to erroneous zones with thermal cameras, cranes, and scissors lift. It is time-consuming and challenging to walk through a wide area with heavy equipment. A full sweep will take several days for a team, making it a difficult task. Therefore, an autonomous Unmanned Arial Vehicle (UAV) inspection system for preventive maintenance is constructed to radically reduce the workload of people, injuries, money, and time. Using Artificial Intelligence (AI) and deep learning, the system will automatically detect and classify cell type, surface defects, and number of defects. simultaneously, the UAV will autonomously navigate across panel arrays using machine vision algorithms for trajectory tracking and path segmentation along with providing localization for transition between arrays and precision landing to enable wireless charging across the docking station. Multiple tests were performed on this system, giving it a 92.13% average accuracy, 80.33% average precision for the defect detection model, 96% average accuracy, 96.05%, and 96.05% average precision for the solar cell type classifier. The autonomous navigation provided 85% accuracy, while the precision landing using Augmented Reality University of Cordoba (ArUco) localization has a relatively accurate landing with a 9.12 cm average offset distance from the center of the docking station.

Keywords: Autonomous UAV, Defect detection, Inspection, Machine learning, Machine vision, Precision landing, Sustainable energy, Solar type classifier, Trajectory tracking, Wireless charging.

1. Introduction

Global energy consumption has risen dramatically since the industrial revolution, owing to the expansion of human activities, which has been powered by population and Gross Domestic Product (GDP). This expansion in energy consumption led to the growth in providing more sustainable energy resources to replace fossil energy sources.

Derived directly from the sun, solar energy is the cleanest and most plentiful renewable energy resource among other renewable energies. However, massive solar power plants are established in deserts, which is considered a suitable environment for the solar photovoltaic (PV) systems due to the sunshine concentration levels accompanied by low altitudes. Solar PV systems are usually mounted on the roof or the field, with the desired tilt angle. As a result, to produce the optimum amount of electricity, every part of the system must be in good working order. That is where Inspection mechanisms come into play [1-4].

Other traditional detection methods such as the thermal camera detection provides good hot spots and temperature classification but nonconstant monitoring. Moreover, the panel surface magnetic field measurement method provides a good cluster failure but nonconstant monitoring. On the other hand, the impedance measurement method provides an inadequate voltage, current, and impedance classification with nonconstant monitoring. Several limitations can be observed in the main categories of the Solar PV panels traditional inspection techniques, such as insufficient monitoring, lack of close vision in real-time and classification of factors that affect the efficiency. On the other hand, the manual inspection can cost more money, time, less covered space, and difficult mobility in such an environment [5-7].

This research and project are focused on developing an autonomous solar panels inspection system using a UAV with sensors and camera vision that have several machine vision capabilities to identify and classify different types of external factors that deteriorate solar panel efficiency, rather than using manual labour or traditional methods to inspect the solar farm arrays. The proposed system can be divided into two categories: autonomous navigation and inspection system. This can enhance mobility, reduce time, provide more coverage, and reduce manual inspection costs.

2. Proposed Methodology

A simple and low-cost DJI Tello quadrotor platform firstly takes off from the wireless charging docking station based on the python script that uses OpenCV to enable autonomous navigation based on several image filtering and edge detection techniques to centralize the drone along with the solar array. In addition to providing trajectory detection for the UAV using visual odometry and path segmentation, the designed system must employ autonomous localization using ArUco markers (fiduciary binary square markers used to estimate camera poses) to ease the transition from one array of solar panels to the successive one. Simultaneously, providing real-time defect detection and solar panel type classification in real-time using TensorFlow object detection

Application Programming Interface (API) that uses transfer learning, MobileNets and Single Shot Detector (SSD) architecture to identify and classify solar PV defects for maintenance purposes and the type of the detected solar panel. Furthermore, localization will provide precision landing across the wireless charging pad for more efficient autonomous operation. The system interface will provide active real-time monitoring and communication with the drone will be carried out over radio signal or Wi-Fi with the ability to introduce any manual control by the operator in case of emergency and generate a record of the detected defects followed by route mapping of the corresponding path as show in Fig. 1.

The main components of the system were divided into 5 sections as shown in Fig. 2; defect detector, the Solar PV type classifier, trajectory tracking using path segmentation and following, localization using ArUco markers to switch between arrays and apply precision landing across the wireless charging pad and drone enhancements that solved overheating as well as adding wireless charging compatibility to the UAV.

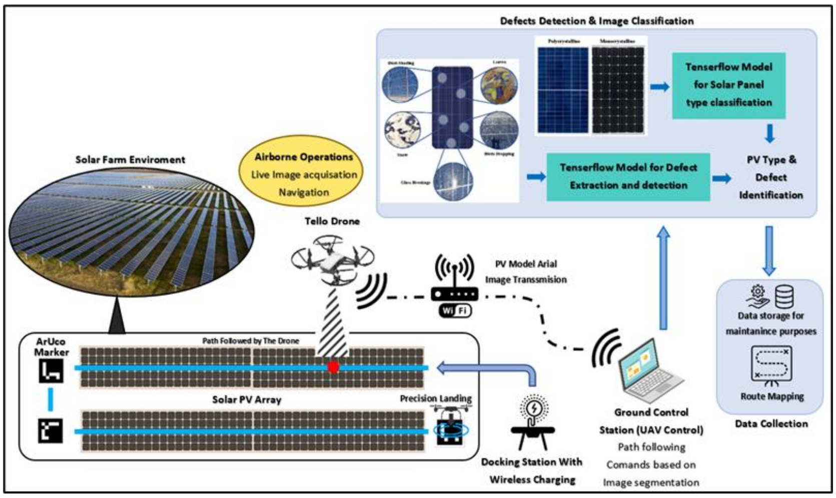

Fig. 1. Implemented system visualization.

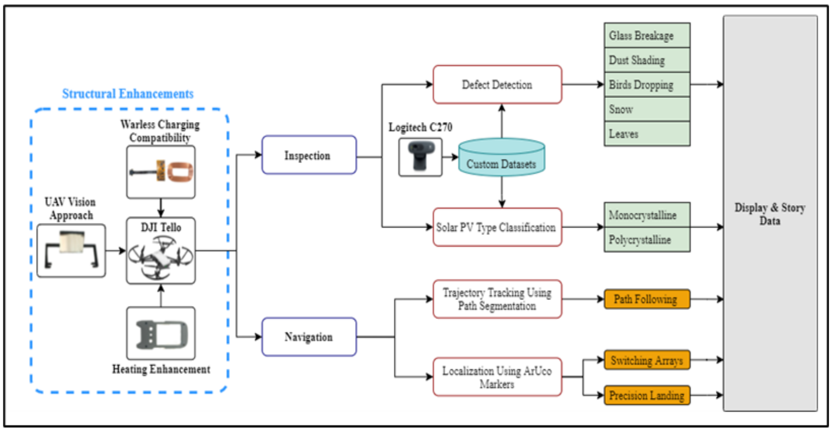

Fig. 2. Implemented system framework.

2.1. Defect detection

To train the model, custom dataset was required. custom dataset was created for the five types (Glass Breakage, Dust Shading, Birds Droppings, Snow and Leaves) with a total number of 2,046 images divided into 1,780 for training and 266 for testing. Data augmentation enhanced the performance and results of the machine learning model. Different backgrounds, such as sand, grass, and tile, were utilized to make the model more robust by generating environment variances that the algorithm may encounter in the actual world.

In this case, the training of a custom model was done using Tensorflow. The model was then deployed to the UAV for real-time detection once accepted results were obtained. Firstly, downloading the SSD MobileNet V2 FPNLite 320x320 architecture from Tensorflow Model Zoo was done, then the label map that specifies the five classes of defects (Glass Breakage, Dust Shading, Birds Droppings, Snow and Leaves) with their corresponding ids was created along with the TF records to help in speeding up the training of the custom model. The training process lasted for 6 hours and involved 10,000 training steps. The model is then saved to be deployed into the UAV for real-time detection once accepted results are obtained.

Once any defect has occurred, the model will draw a bounding box with the corresponding accuracy score and display it onto the system's Graphical User Interface (GUI). Once the model compatibility is achieved, the model will accept the input frame stream from the UAV then compare and generalize the probability of each defect as illustrated in Fig. 3.

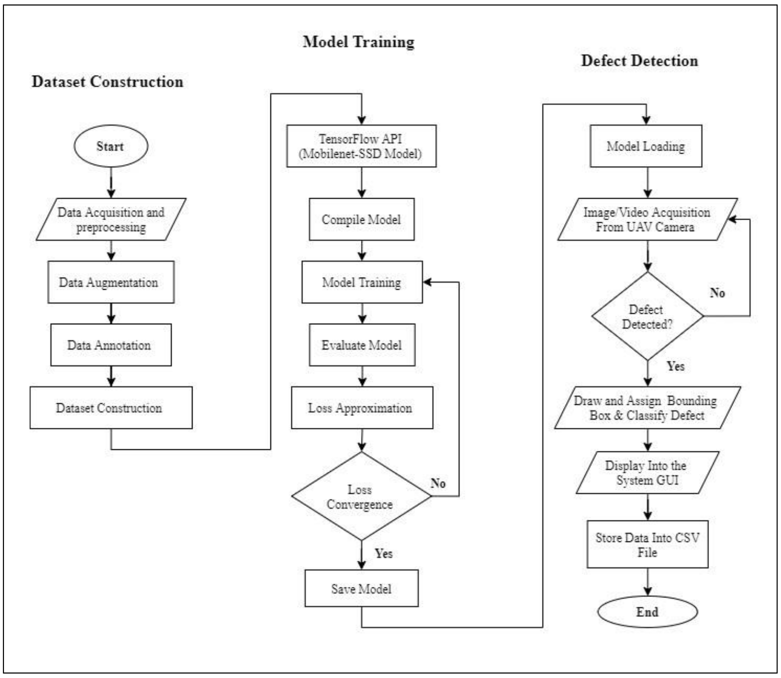

Fig. 3. Object detection flowchart.

2.2. Solar PV type classifier

The Solar PV type classifier also utilizes the concept of transfer learning but rather employ MobileNet only without using SSD architecture. Transfer learning will ensure fetching the gained knowledge from the base model and refines it using the captured training data for the solar panel types, as shown in Fig. 4.

Data augmentation is used to enhance the dataset with more unknown objects that might exist across solar farm environment. The total number of datasets used for training the classifier was 2,805 images divided into 1,377 for Monocrystalline and 1,428 for Polycrystalline.

After training of the model is completed and loss convergence is achieved, the model is exported and loaded into the main python script of the UAV as shown in Fig. 5. Once compatibility is achieved by adding all the required libraries to the main system virtual environment, the UAV is deployed, and the video frames are inputted into the model, the classifier will instantly try to generalize and display the type of solar panels displayed on the system GUI from the capture frames in real-time. Therefore, this will ensure that the UAV inspection system has acquired environmental awareness of the system surroundings, which is one of system anatomy's main fundamental components [8, 9].

2.3. Trajectory tracking using path segmentation

Autonomous system of the UAV does not depend on Global Positioning System (GPS) or on pre-mapped environment, or simulated environment like Robot Operating System (ROS) for navigation. Image processing approach to identify environmental features that exist across different types of solar farms environments and the Region of Interest (ROI) segmentation which is this case is the solar array to be employed in the navigation and localization of the UAV which make real-time mapping redundant in this case.

Navigation is initiated by obtaining the DJI Tello quadrotor's video frames in real-time. Several telemetry data will be recorded to help in plotting the route after the end of the mission. These include speed commands sent to the drone (approximated deg/s and cm/s) and pitch, roll, yaw, height Eulerian angles [7, 10, 11].

As shown in Fig. 6, after obtaining the frames, they are split into three regions (Left, Center, Right) then it will assign the region filled with 20% pixels as 1 and the area filled with less than 20% as 0 to simulate a vision senor. Image segmentation based on the conversion of RGB color into HSV color specs of the solar PV array is applied to extract and isolate the solar array from the surrounding environment. To prevent overshoot of the drone the vision sensor cases will apply faster movements when offset is significant.

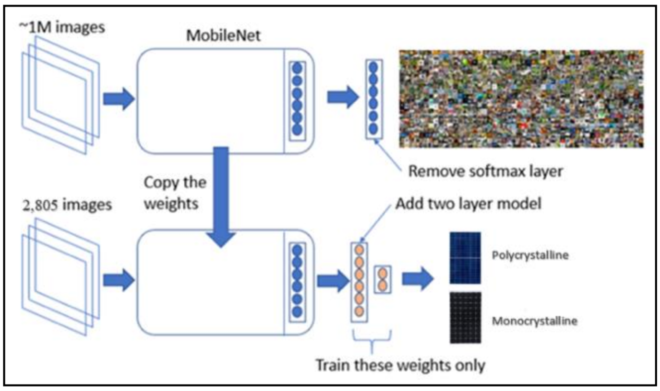

Fig. 4. Image recognition system flowchart.

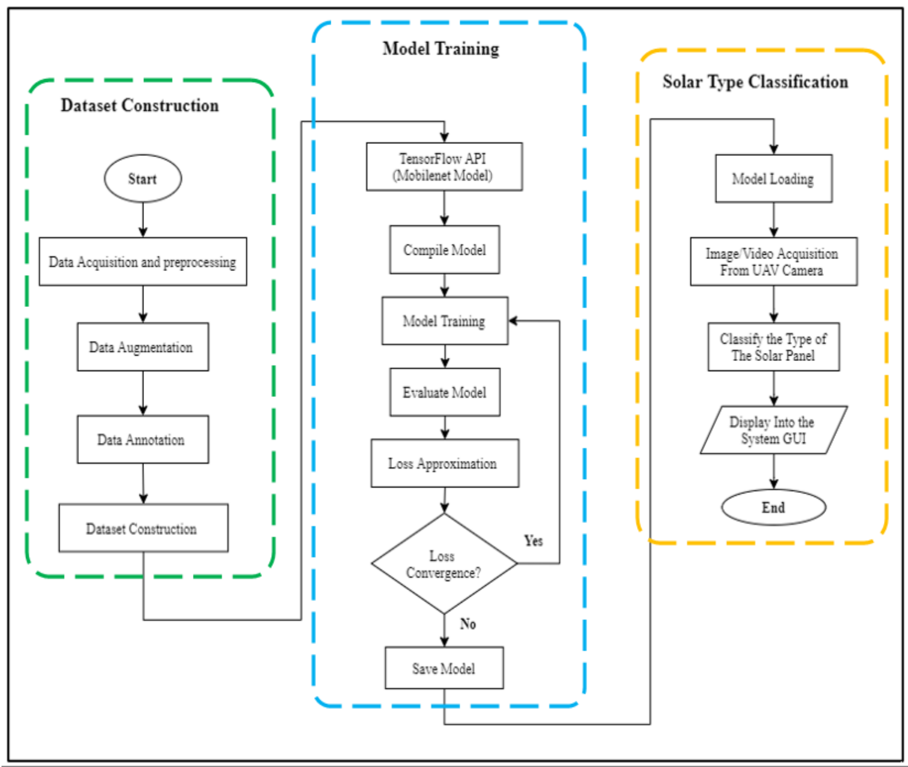

Fig. 5. Solar PV type classifier flowchart.

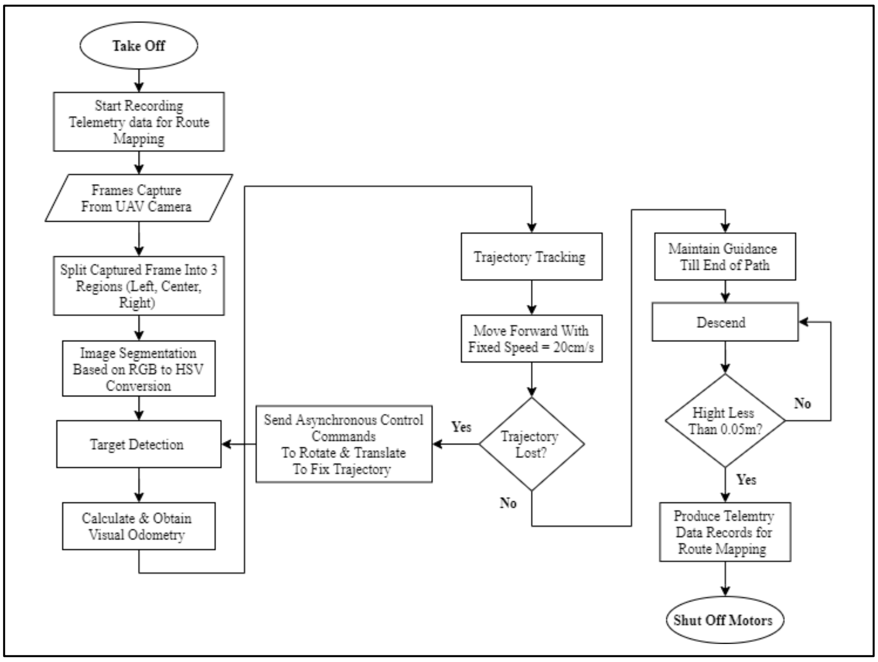

Fig. 6. Path segmentation flowchart.

2.4. Localization using ArUco markers

Localizing and directing the UAV across the ends of the solar array path is done using ArUco markers. The ArUco localization and control starts by obtaining the captured frames across the trajectory path while each ArUco marker has a unique ID that can be retrieved using the information stored in bits. The inputted frames are then converted into grayscale and thresholder to enable marker detection and encoding. There are a total of 1024 distinct designs that can be formed using this structure.

The proposed system uses the original dictionary-based markers for better detection, resulting in reduced false-positive identification of the markers. The system will return the Two-Dimensional (2D) position of each detected marker's ID (1, 2, 3…), center point, corners which are Three-Dimensional (3D) array of shape.

Furthermore, the system will calculate the center point of the UAV frame based on the width and height of the frame and use it to calculate the deviation across the x and y axes, then use the coordination point to calculate the x-y vector (hypotenuse) that connects the center point of the UAV frame with the maker center point as this is (ideally) where the UAV supposed to be kept using Pythagorean theorem as illustrated in Fig. 7.

If a UAV achieves the minimum threshold distance, then depending on marker ID allocated across the solar farm, the UAV will perform the required maneuver to shift to the following array and start allocating the next marker as shown in Fig. 8. Simultaneously, the localization system will send Asynchronous Radio Control (RC) commands depending on the marker's threshold distance to centralize the drone on the detected marker [10].

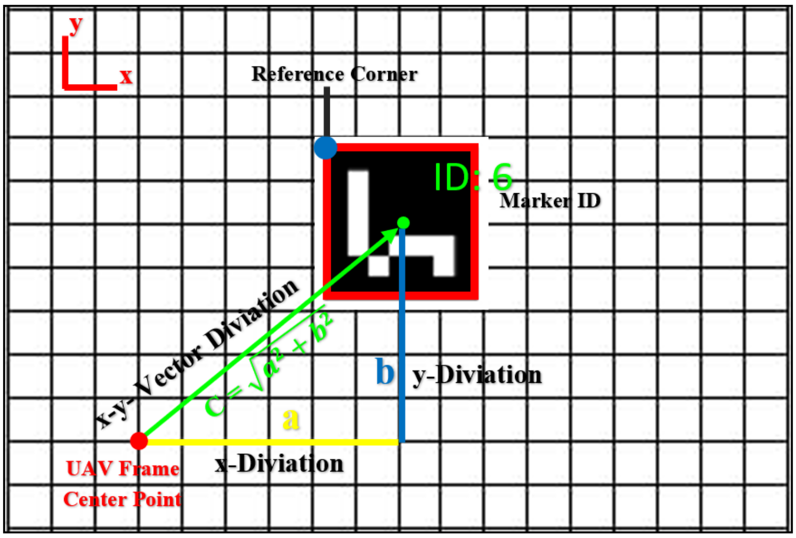

Fig. 7. ArUco marker localization.

When a Landing Marker is detected, the UAV will descend in accordance with the marker with the minimum threshold value. Eventually, once recorded height by the Tof sensor is less than 5 cm, it will land on the wireless charging docking station and begin charging process.

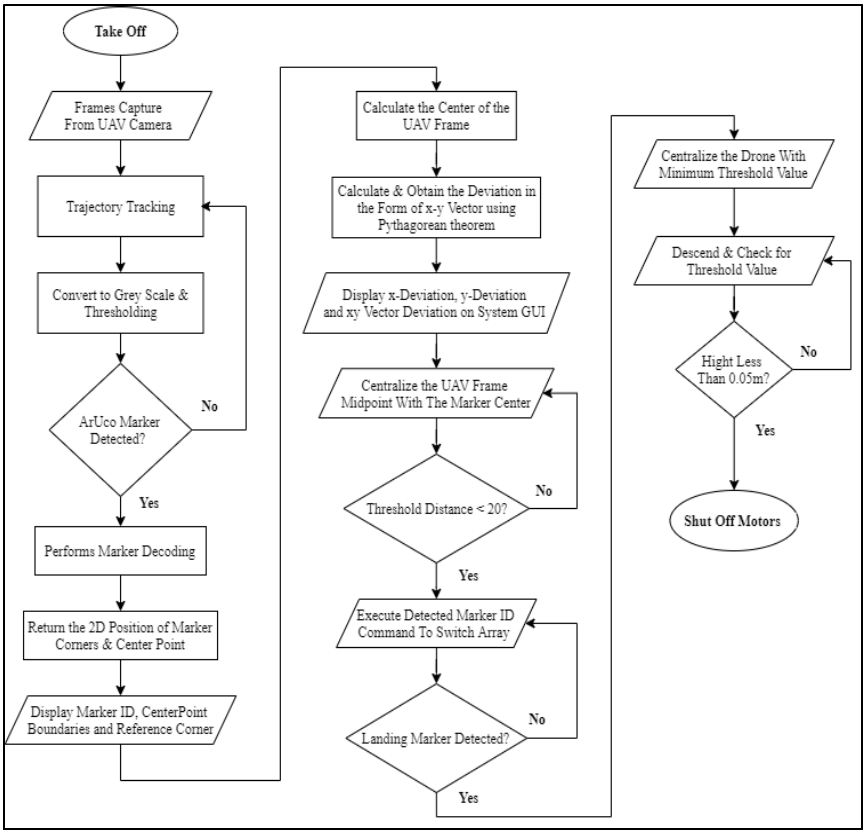

Fig. 8. ArUco marker flowchart.

2.5. Structural enhancements

However, by running software development applications and introducing the asynchronous communication that has been used to perform the autonomous navigation and simultaneously sending the captured video frames throughout the build in 2.4 GHz 802.11n Wi-Fi into the laptop, it will produce a vast computation burden on the UAV Central Processing Unit (CPU).

The Tello drone is designed for fast and continuous movement rather than steady movements operations, making the compact and encapsulant default bottom cover design offer lesser ventilation. This caused the drone to reach an approximate temperature of ~86 Celsius, which is considered high, and after reaching this benchmark, the communication between the drone and the laptop can experience some disturbance.

Therefore, three bottom cover designs were implemented which are: vented, cooling and hybrid designs as shown in Fig. 9. The vented design depends more on ventilation space to drive the air produced by the drone propellers to go through the sides of the bottom cover and apply direct cooling to the heat sink. On the other hand, the cooling design depends on an additional nano fan that relates to the UAV battery to cool the CPU. Lastly the hybrid design offers a combination of the vented design concept and uses the nano fan to cool down the CPU unit as well.

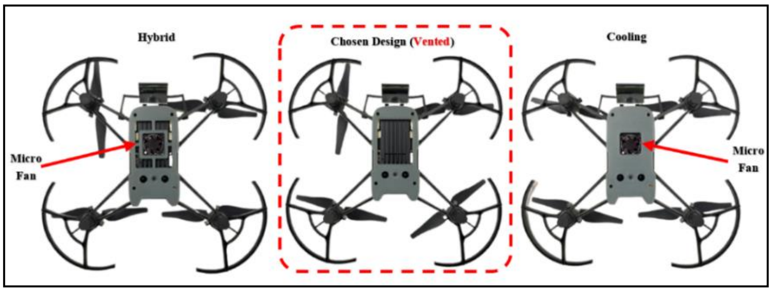

Fig. 9. CAM designs for vented, cooling & hybrid designs.

Wireless charging for drones allows for more flexible charging alignment, universal drone compatibility, safer connections, and endurance in harsh environments such as solar farms. Landing on the charging pad or charging station is all that is required to charge the UAV.

At the end of the path, after successful landing using the precision landing, the Tello drone is equipped with a wireless charging module that utilize type B connection that can receive 5 W with 5 V and 1000 mA input transmitted by the wireless charging transmitter. Transferring of the energy occurs by a transmitter coil (Tx) that transmit energy to the receiver coil (Rx) in the bottom of the drone via electromagnetic induction as shown in Fig. 10. The charger uses an induction coil to create an electromagnetic flux, which the receiver coil in the Drone converts it back into electricity to be fed into the Drone battery. If the Tello battery is completely drained the drone charging operation can take up to 90 minutes.

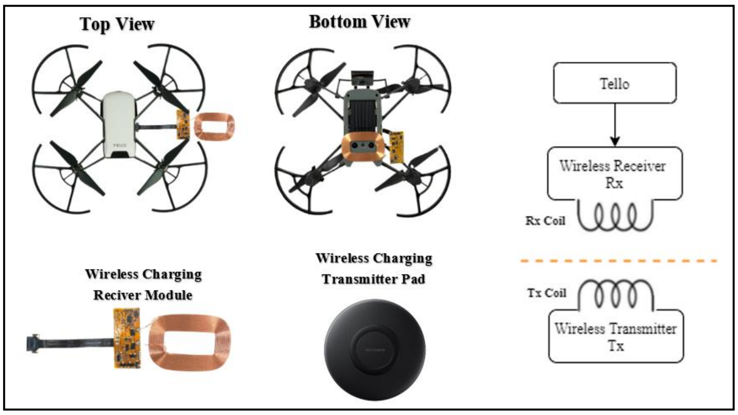

Fig. 10. Wireless charging mechanism with exploded view and operation flow.

3. System Testing

3.1. Accuracy, precision, recall and F1-score for defect detection

The testing results were recorded for both Monocrystalline and Polycrystalline. To simulate actual solar PV panels, patterns were printed out and inserted into a glass picture frame with a similar exterior to an actual solar panel, and then defects were artificially formed to match natural occurring defects since there was no access to actual PV modules. In a controlled environment, the system evaluated 60 images for each class with different background environments like grey tile, white tile and grassy area with 300 samples for the 5 classes. After correctly identifying the defects, the data was recorded to using the 5x5 Confusion Matrix as shown in Fig. 11.

As indicated in Fig. 12, the model had an average accuracy of 92.13%, with snow and birds dropping as the most accurate classes due to their distinctive nature and they do not cluster with the background texture like the dust shading, leaves and glass breakage, that get affected by the reflection of the covering glass and experience similar background texture. Moreover, the model offers an 80.33% average precision, with dust shading and leaves being the lowest values. Dust shading is mainly affected by the background environment and the lighting condition that in many cases misclassify it as glass breakage due to the reflection of the glass, while the leaves precision is only affected for Monocrystalline solar cell that has similar brownish texture. Since the glass breakage and leaves were misclassified for other defects, it caused a drop back in their recall percentage to fall between 65.05-66.66%. However, the model had an 82.74% average recall that is considered excellent. Lastly, the F1-score demonstrate an excellent balance between precision and recall producing an average F1-score of 80.33% as illustrated in Table 1.

	N = 300	Predicted Class				
		Glass Breakage	Dust Shading	Birds Dropping	Snow	Leaves
Actual Class	Glass Breakage	54	1	0	0	5
	Dust Shading	16	34	0	0	10
	Birds Dropping	1	0	50	2	7
	Snow	0	0	1	59	0
	Leaves	12	4	0	0	44

Fig. 11. Confusion matrix of defect classification using the trained model.

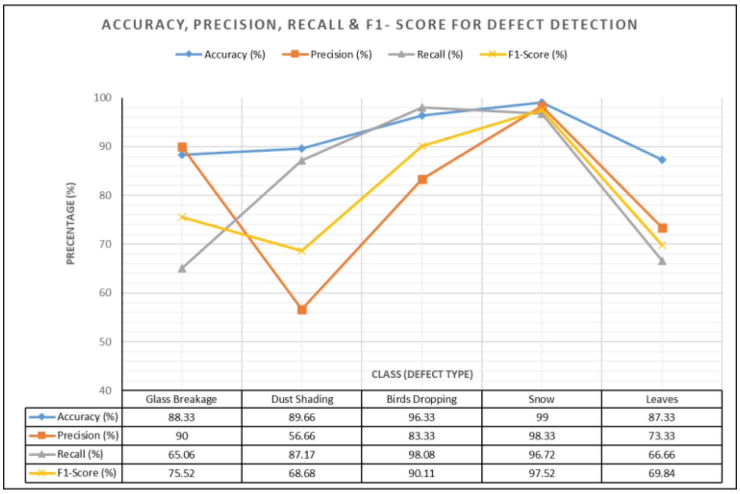

Fig. 12. Defect detection model accuracy, precision, recall and F1-score graph.

Table 1. Defect detection model average results.

Average Accuracy	Average Precision	Average Recall	Average F1-Score
92.13%	80.33%	82.74%	81.52%

3.2. Accuracy, precision, recall and F1-score for solar type classifier

The testing results were recorded for classifying Monocrystalline and Polycrystalline cells. During the testing, the model evaluated both regular panels that have no surface defects and defective panels that have texture deformation that, in some cases, can be a challenge to identify. Similar to the defect detection model testing artificially formed panels were used. In a controlled environment, the system evaluated 100 images for each type with different background environments like grey tile, white tile and wooden ground with 200 samples for the two types. The data was recorded to obtain the corresponding performance using 2x2 Confusion Matrix as shown in Fig. 13.

N = 200		Predicted Class	
		Monocrystalline	Polycrystalline
Actual Class	Monocrystalline	97	3
	Polycrystalline	5	95

Fig. 13. Confusion matrix of solar type classifier using the trained model.

The classifier model that uses MobileNets architecture provides an astonishing performance with a 96% average accuracy and 96.05% average precision, recall and F1-score as shown in Table 2. Even though the model is considered a lightweight model, providing a large dataset of 2,805 images with different panels (defective and regular panels) allowed it to classify the panels effectively. However, the only scenarios where the model failed to identify the panel type was under extreme dust shaded panels and panels covered with leaves which made panel's pattern extraction due to complete coverage of surface not possible.

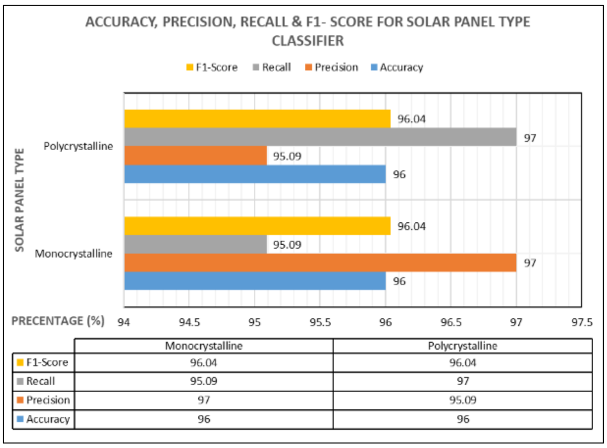

Fig. 14. Solar panel classifier accuracy, precision, recall and F1-score graph.

Table 2. Solar panel classifier model average results.

Average Accuracy	Average Precision	Average Recall	Average F1-Score
92.13%	80.33%	82.74%	81.52%

3.3. Precision landing test

The precision landing tests the ability to accurately land the autonomous UAV on the docking station to enable wireless charging compatibility.

The Fully autonomous precision landing leverages the customized vision reflector clip that provides a downward-looking camera for the Tello drone due to the lack of gimble. In each testing cycle, the UAV went through the complete path till the point of the landing marker at the endpoint of the track. The landing marker was attached to the center of two combined scaled A3 papers to precisely mark the x, y coordinates of the drone's landing position relative to the marker in the world coordinate system since the final landing locations determine the success of the accurate landing. The UAV went through 20 positioning cycles to conduct this test. In addition, the time spent by the drone in each positioning cycle to position itself over the marker and land was recorded since it reflects the efficiency of the precise landing approach.

The test cycles data is recorded for the deviation distance in x, deviation distance in y, the distance offset in x-y and the time consumption for every landing operation. The average result of the 20 positioning cycles is illustrated in Table 3.

Table 3. Test cycles offset distances and time consumption average results.

x-Distance (cm)	y-Distance (cm)	x-y Offset Distance $\sqrt{x^2 - y^2}$ (cm)	Time Consumption (s)
5.35	6.23	9.12	26.05

As illustrated in Figs. 15 and 16, landing locations using ArUco localization are considered relatively accurate since the landing locations of 12 points are relatively close to the target, with less than10 cm offsets on the x- and y-axis of the world coordinate system. Five landing locations were allocated between 18-10 cm radius offset distance from the marker centre point, while the last two drifted further away within a 22-20 cm radius. This type of distance offset occurs due to the delay time for the drone to execute the landing command while shifting from asynchronous centralization movement command to synchronous landing command. During this transition, the drone experiences a minor shift that amplifies while landing gradually. The precision landing has a function that recalls latest shift from the centre in case it drifts away, which help to maintain the target to resume the positioning cycle.

However, in three test cases and due to the shift and vibration that affect the vision reflector clip that lacks an active stabilizer, the UAV lost the marker and stayed hovering during the tests until timeout. This vibration also caused two test cycles time consumption to take over 40 seconds. Overall, the system provided an accurate landing with a 9.12 cm average offset distance that enable wireless charging compatibility and 26.05 seconds average time consumption.

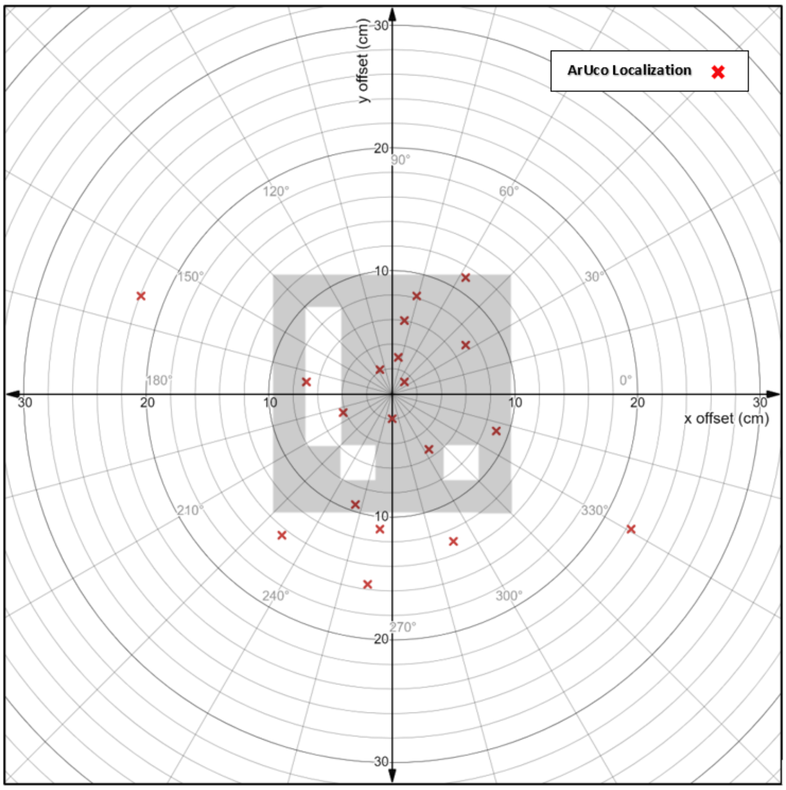

Fig. 15. Final landing locations coordinates relative to marker centre point.

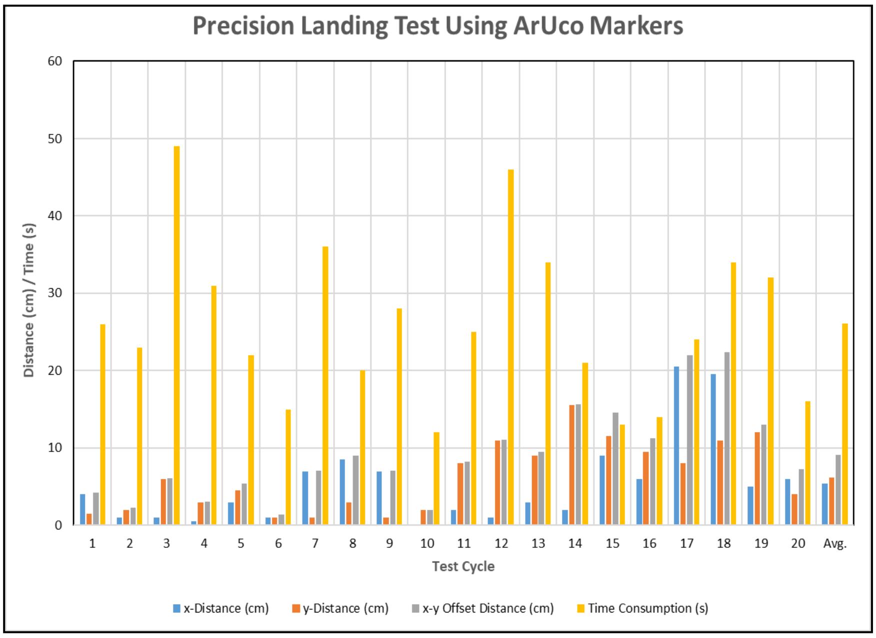

Fig. 16. Precision landing, test cycles offset distances and time consumption.

3.4. Accuracy in path repeatability test

The accuracy in path repeatability is the repeatability of the Path Following system is the extent to which successive attempts the UAV moves to the final point location without getting off track.

Two sets of tracks were prepared, by placing the solar panels in different arrangements similar to the solar arrays found on actual solar farms, but with the addition of uniformity to test how well the system can perform. Each track was ten cycles, with a total of 20 cycles for both tracks. In each testing cycle, the UAV went through the complete path till the endpoint of the track. The test evaluated the number of successes attempts to get to the endpoint without losing track.

Using the demonstrated data in Fig. 17, the accuracy percentage based on the successful and fail attempts can be calculate for both tracks as follow:

$$Accuracy\ \% = \left|\frac{Number\ of\ Successful\ Attempts}{Total\ Number\ of\ Attempts}\right| * 100 \quad (1)$$

$$Track\ 1\ Accuracy\ \% = \left|\frac{2-10}{10}\right| * 100 = 80\%$$

$$Track\ 2\ Accuracy\ \% = \left|\frac{1-10}{10}\right| * 100 = 90\%$$

$$Overall\ Accuracy\ \% = \left|\frac{3-20}{20}\right| * 100 = 85\%$$

As indicated in Figs. 17 and 18, in both track layouts, the system maintained more than 80% successful attempts rate. However, one of the test cycle take-offs failed due to a low battery percentage that forced the UAV to land shortly after taking off in Track 1 immediately. In the remaining two cases, the path following was subjected to deviation in tight corners when switching from asynchronous centralization movement command to synchronous movements command subjected to delay to allow the Tello drone User Datagram Protocol (UDP) protocol to receive the synchronous command once the marker threshold distance is achieved. In many cases, by using a powerful laptop, the Wi-Fi speed and Graphics Processing Unit (GPU) computational power can enhance or even eliminate both the UDP protocol communication and the Frames Per Second (FPS) rate, which provide smoother asynchronous movement commands based on the occurring errors. However, the laptop used was only able to achieve 7-9 FPS when connected to a power source and 5-6 FPS when running on battery.

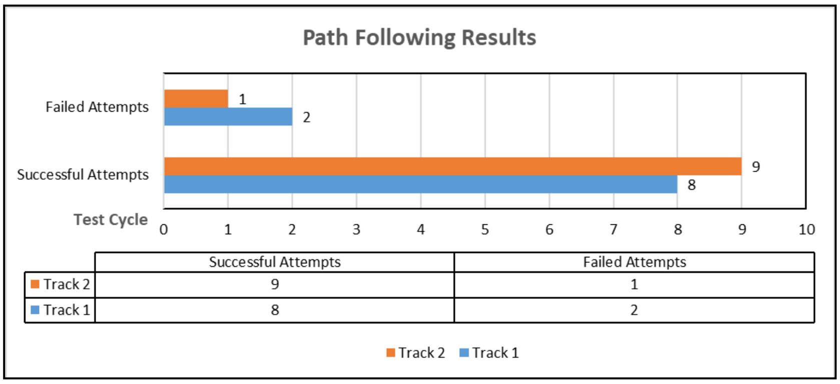

Fig. 17. Precision landing, test cycles offset distances and time consumption.

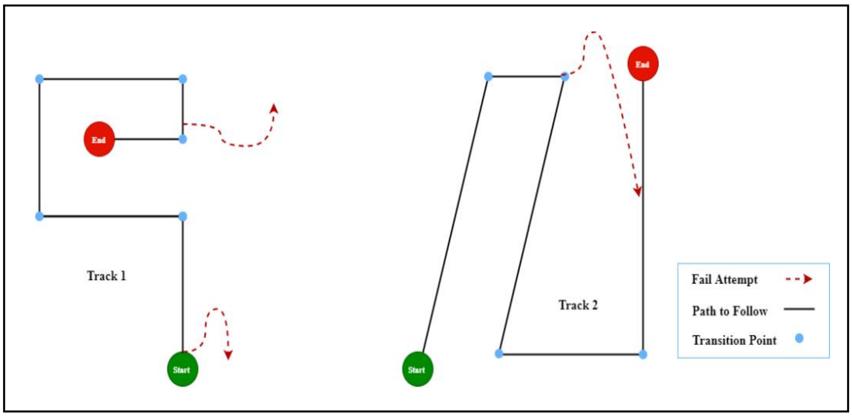

Fig. 18. Illustration of fail attempts in test tracks.

3.5. Thermal throttling test

Thermal throttling tests the amount of heat-generating by the Tello Drone Intel CPU and the corresponding flight time when using different bottom covers heat sink designs.

In an indoor controlled environment, the three bottom cover Computer-aided Design (CAM) designs (Vented, Cooling and Hybrid) illustrated, as well as the default bottom cover of the drone, were subjected to two sets of tests. The first test is an indoor static flight with a room temperature of 32 ℃, while the second test was a windy indoor static flight using a ceiling fan with approximately 1.3m/s wind speed and a temperature of 32 ℃. The room parameters were recorded using Benetech GM816 Thermo Anemometer. The test was carried out using a python script that logged all the flight data such as temperature, battery percentage and flight time for every sec using Tello Software Development Kit (SDK) into an excel file. The results show the temperature readings and the flight time for each bottom cover to find the best design in terms of ventilation.

As shown in Fig. 19, by analysing the temperature rise or the temperature response for indoor static flight using the slopes of the curves, the Vented design maintained the lowest temperature fluctuation under 85 ℃ with the longest flight time. On the other hand, the Cooling and Hybrid designs showed faster temperature rise above 85 ℃, with Hybrid having faster temperature rise and shorter flight time. The three CAM designs took about 1 min of rising time and 4 min to reach the steady-state phase of temperature fluctuation (83-86 ℃). However, the default design showed the worst heat response, with rapid increment in CPU heat the exceeded 87 ℃ in lesser than 2 min due to the encapsulant nature of the design, which caused the Tello to land by itself due to over temperature.

Moreover, by analysing Fig. 20, the indoor windy flight solidifies that the lowest temperature rise, and the longest flight time was achieved by the vented design that maintained lesser than 82 ℃ CPU temperature. However, the Cooling and Hybrid designs showed lesser temperature rise with longer flight time compared to the static flight with Cooling design having lesser temperature and more flight time and Hybrid having higher temperature and shorter flight time. Similarly, the default design showed the worst heat response, with rapid increment in CPU heat the exceeded 87 ℃ but with short increment of 20 sec flight time compared to the static flight. Eventually, the default design caused the Tello drone to land by itself due to over temperature.

To summarize, by examining Figs. 19 to 21, it can be clearly seen that the Vented design achieve the best flight time that reaches up to 7 min and 42 sec compared to the default design the provides approximately 2 min of flight time. The results can be justified due to the nature of the vented design that makes use of the air generated by the drone propellers and redirects it to the CPU without any power consumption, unlike the Cooling and Hybrid designs that rely upon the micro fan that adds additional burden and consume the drone battery. Nevertheless, the default design was designed for normal use that does not consider software development and consider faster movements which made the design more aerodynamic and more encapsulant which caused this type of results.

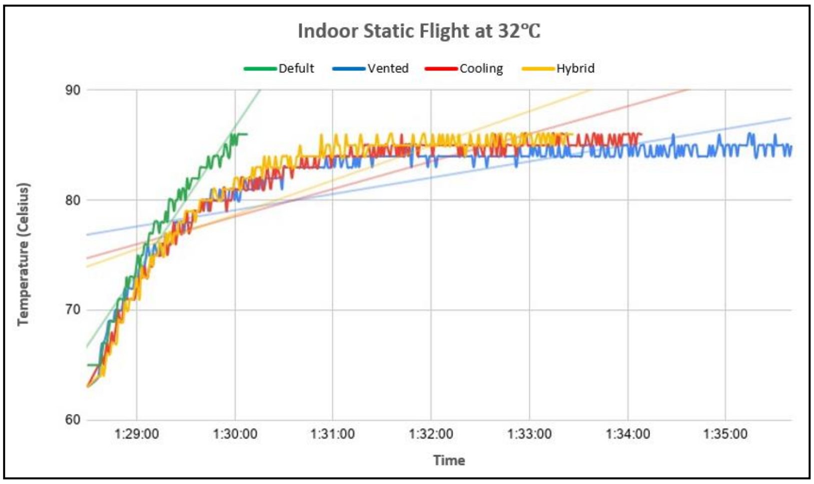

Fig. 19. Indoor static flight graph.

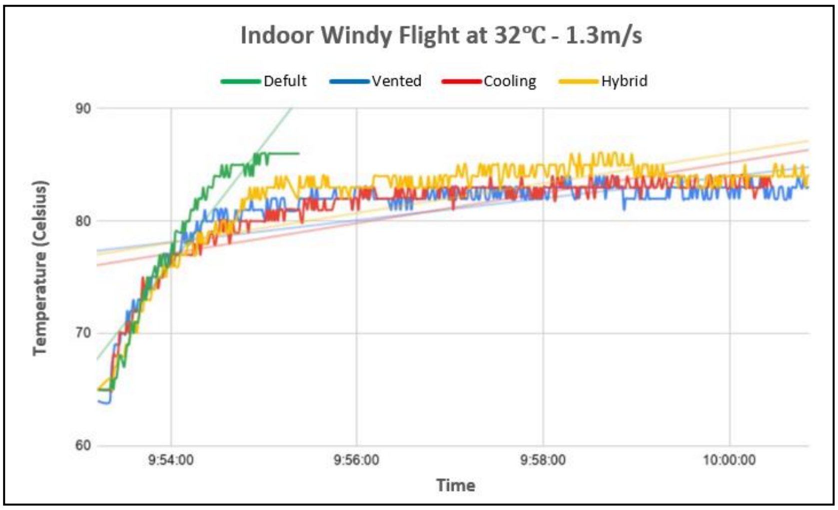

Fig. 20. Indoor windy flight graph.

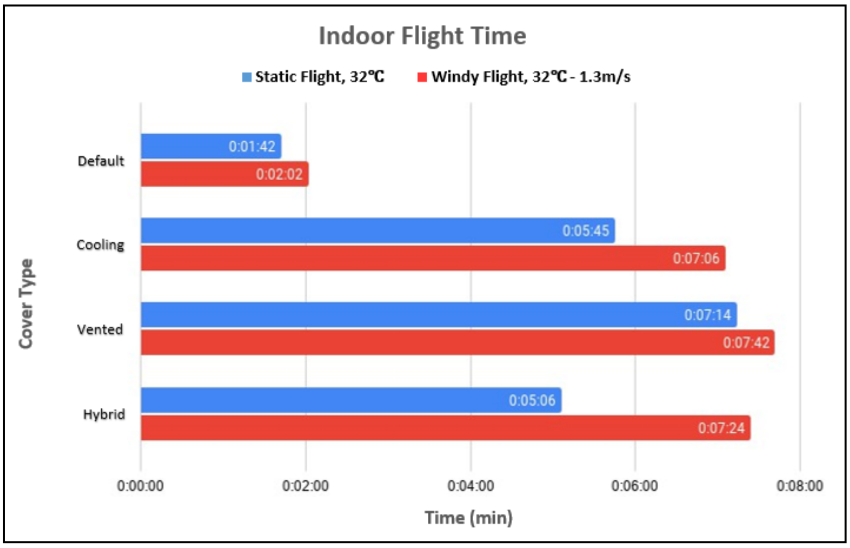

Fig. 21. Indoor flight time graph.

4.Experimental Results

The GUI screen is divided into three main components, as shown in Figs. 22 to 26. For example, when a defect occurs, the left segment will capture the occurring defect with a corresponding bounding box and its accuracy score.

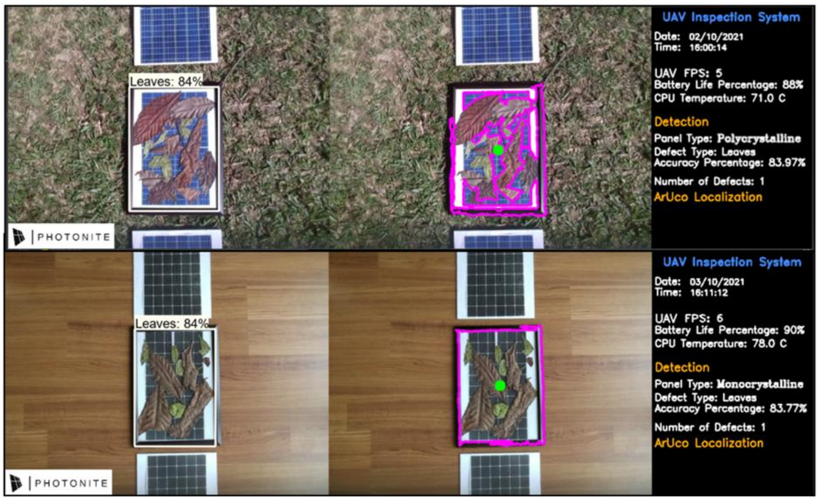

Fig. 22. Leaves defect detection for polycrystalline and monocrystalline solar panels.

The middle segment of the GUI illustrates the trajectory tracking using path segmentation which highlights the captured panel boundaries with the center point of deviation.

The right segment of the GUI is a dashboard that lists down all the important parameters such as date, time, obtained FPS, battery life percentage and the drone CPU temperature. Moreover, the detection parameter will highlight the captured panel type, defect type, defect accuracy score, and the overall number of captured defects. In addition, the ArUco localization will display the x, y distances and the center point deviation from the drone Point of View (POV).

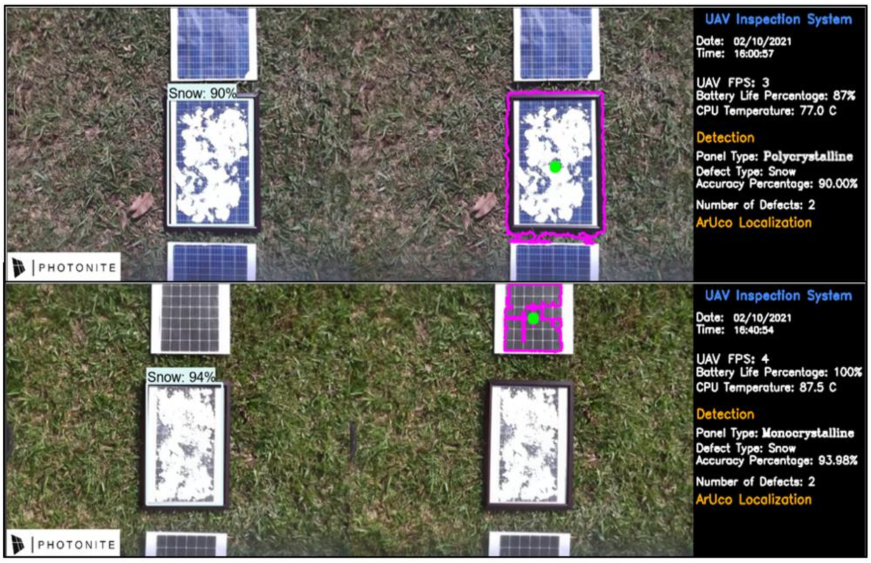

Fig. 23. Snow defect detection for polycrystalline and monocrystalline solar panels.

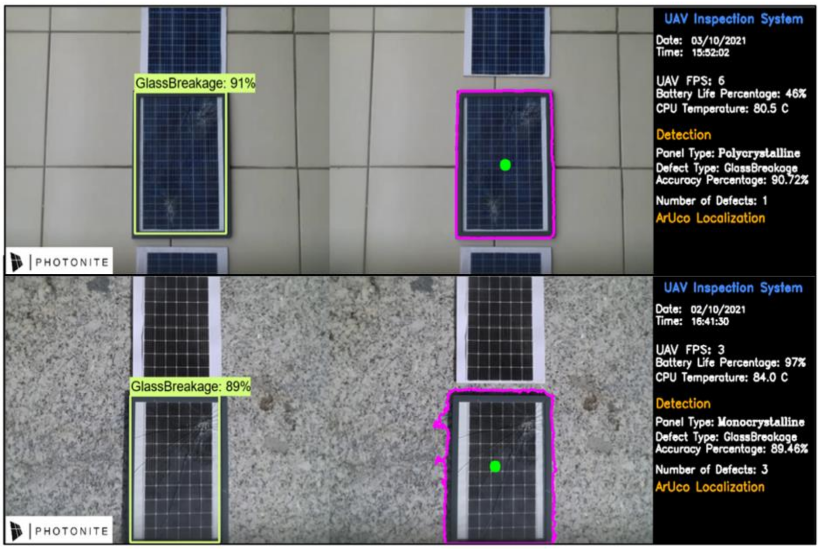

Fig. 24. Glass breakage defect detection for polycrystalline and monocrystalline solar panels.

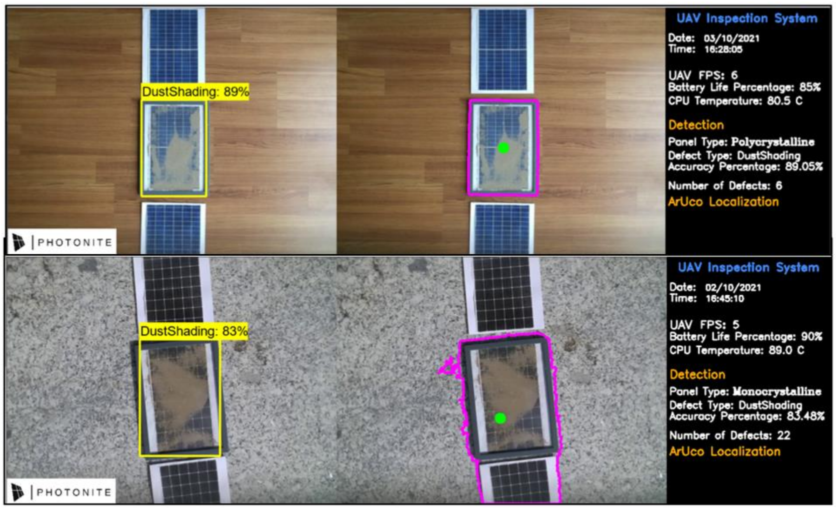

Fig. 25. Dust shading defect detection for polycrystalline and monocrystalline solar panels.

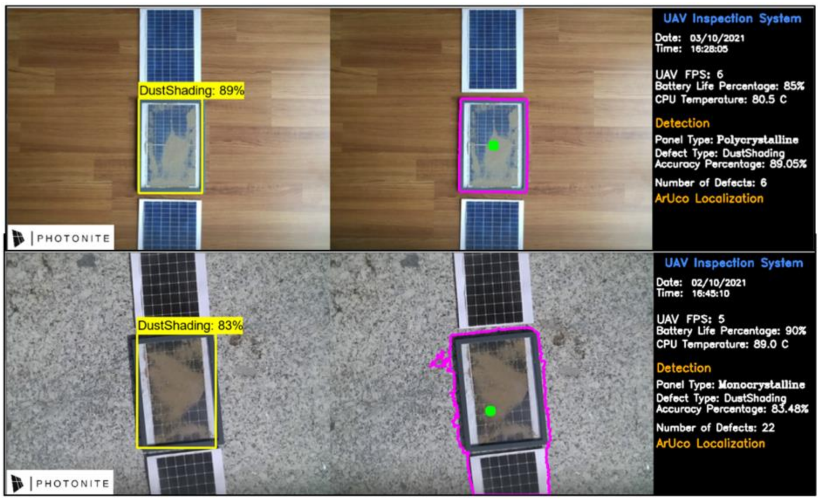

Fig. 26. Birds dropping defect detection for polycrystalline and monocrystalline solar panels.

When the drone reaches the transition points allocated across the edges of the solar array. The ArUco localization process is initiated, as illustrated in Fig. 27, and the deviation is displayed on the parameter list. Then, the drone will start approaching the threshold centre deviation distance to instruct the UAV with the corresponding required movements according to the detected marker ID to shift to the next array.

When the drone reaches the Transition points allocated across the edges of the solar array, the ArUco localization process is initiated, as illustrated in Fig. 27, and the deviation is displayed on the parameter list. Then, the drone will start approaching the threshold centre deviation distance to instruct the UAV with the corresponding required movements according to the detected marker ID to shift to the next array.

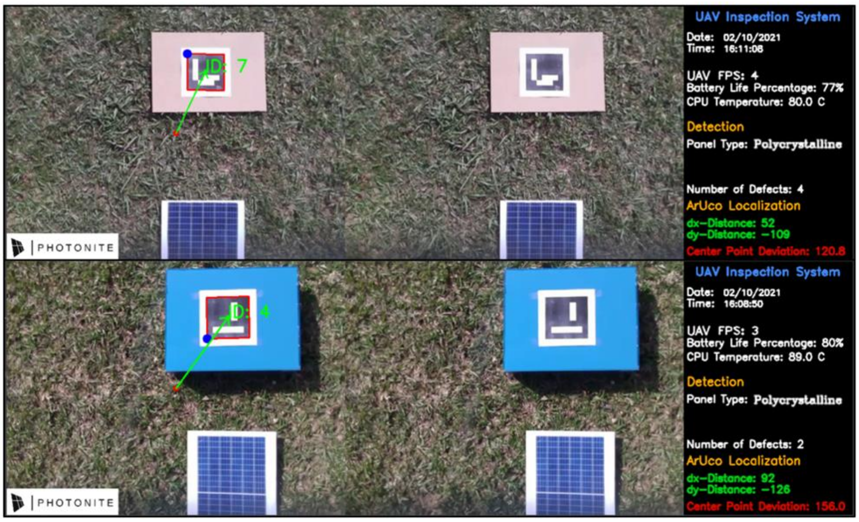

Fig. 27. ArUco marker detection in the transition and landing points.

The landing process will ensure landing with minimum distance deviation from the center of the ArUco marker that satisfies the wireless power transfer standard known as Qi which is the inductive charging to transmit electricity wirelessly over up to 4 cm by the wireless charging pad in the docking station as shown in Fig. 28.

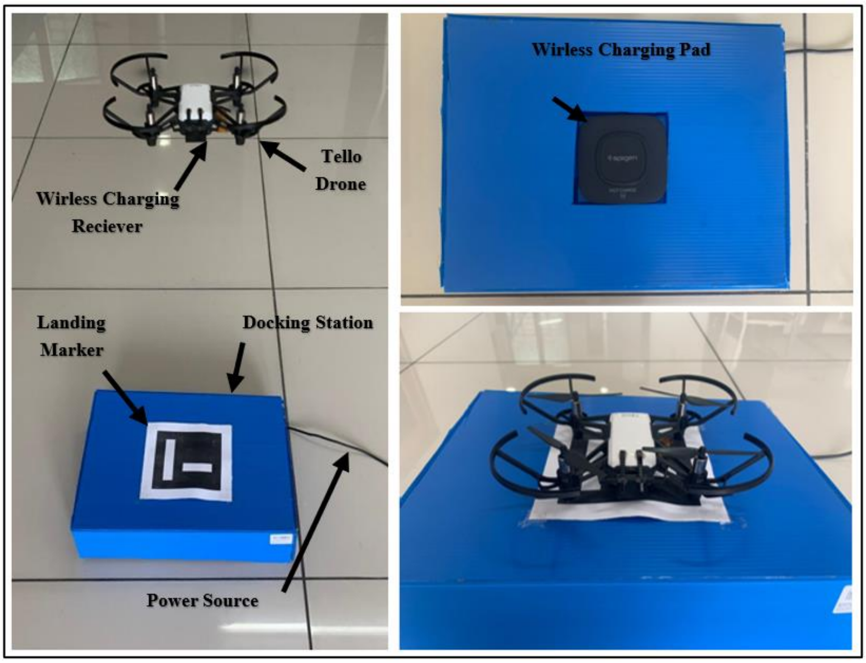

Fig. 28. Docking station landing setup with ArUco marker.

Data collection and storage will summarize the operation in an easy-to-read CSV file. The system will also automatically store a recorded video of the entire preformed inspection process.

The system plots the 3D position of the UAV during the inspection process using telemetry data and the camera vision relative position. The transition points, the take-off and landing points were highlighted manually for better visualization as shown in Figs. 29 to 30.

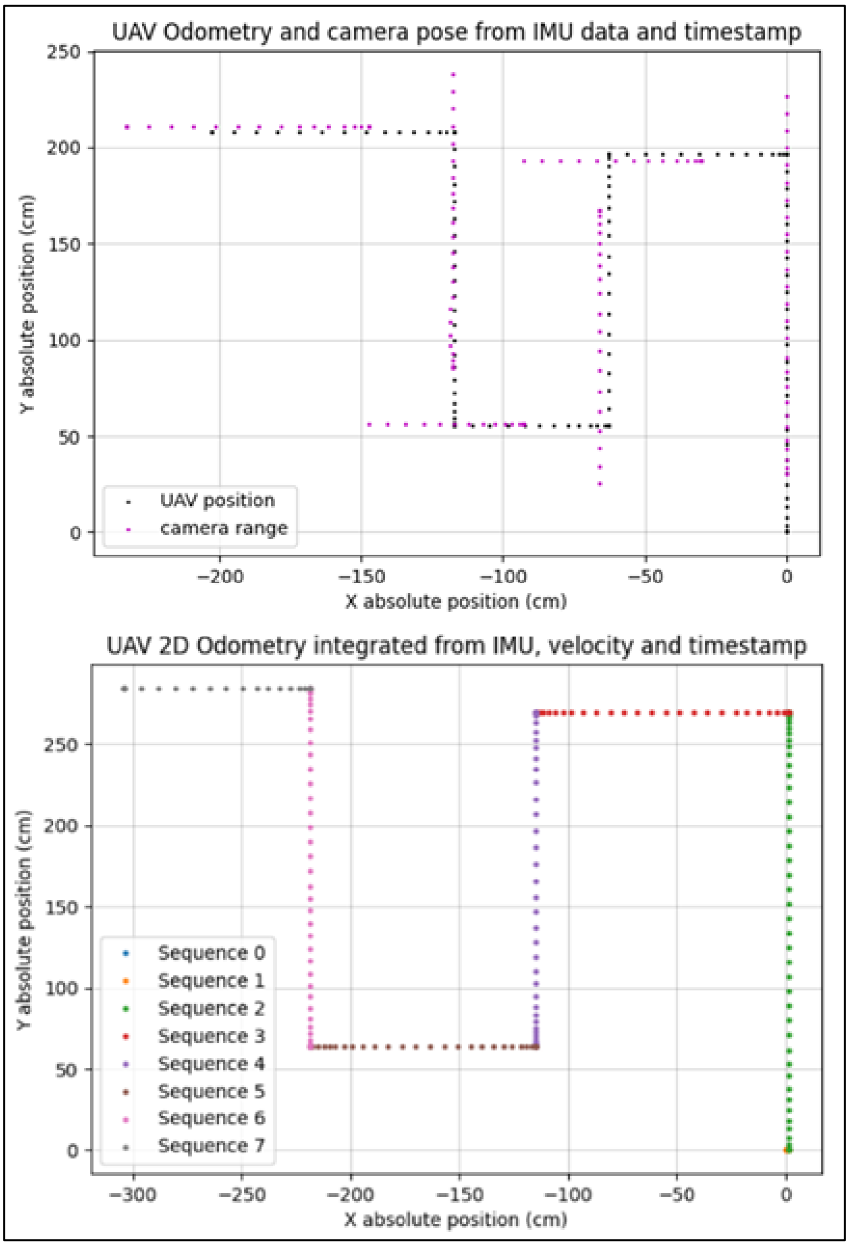

Fig. 29. UAV 2D odometry mapping.

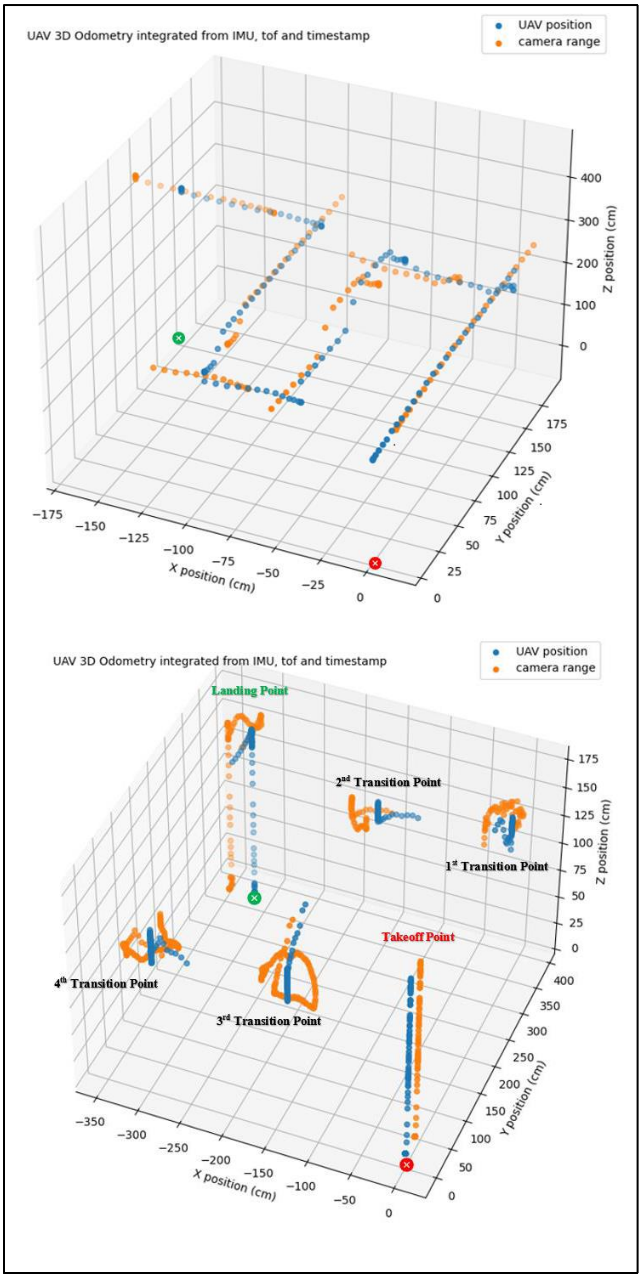

Fig. 30. UAV 3D odometry mapping.

5. Conclusion

This study presented a novel approach to preventive maintenance measures by early detection of naturally occurring defects that happen over time to eliminate the deterioration of solar PV cells and the power efficiency of solar modules. The presented prototype utilizes an autonomous UAV in the inspection process across solar arrays, mainly using machine vision for navigation and machine learning for defect detection to maintain a safe condition in the solar energy production cycle.

Two algorithms were implemented to provide defect detection of solar cells surface and cells type classification using transfer learning and custom training and testing datasets. The detection model had 92.13% average accuracy, 80.33% average precision, 82.74% average recall and 81.52% average F1-Score. Moreover, the classifier model illustrated 96% average accuracy, 96.05% and 96.05% average precision, recall and F1-score.

Tello drone was employed and equipped with path following based on image segmentation and localization using ArUco markers algorithms to achieve the autonomy. The Autonomous path following illustrated 85% accuracy and a relatively accurate landing with a 9.12cm average offset distance that enabled wireless charging compatibility.

Due to the restriction policies of data and ongoing pandemic, the implemented system didn't leverage training or testing based on collected data from real solar farm facilities to reflect its performance in real scenarios. However, as an alternative, the system used synthetic panels to be used as a training and testing dataset which intended to be as close as possible to actual panels, which relatively affected the validation of the performance.

The Tello drone by DJI is considered a downscaled version of the actual sized drones that are used across inspection facilities. The mini drone size made it light weighted, which affect its performance in windy conditions. It also had a decent camera that lack physical video stability and lacked a free axis gimbal.

The used computer for developing and training the inspection system, had relatively weak hardware specs that made training of the deep learning models take different approach due to inability to train more complex architectures. The precision landing lacks PID controllers or other supporting algorithms that can actively enhance the precision of the landing. Additionally, The Tello Drone communication depend on UDP protocol that is transmitted over Wi-Fi, which relatively limit the range of the drone depending on the Wi-Fi range rather than depending on Radio Frequency (RF) signals that have wider range

The autonomous UAV inspection system demonstrated promising results despite the faced problems and limitations. Therefore, the possibility of taking this project to further stages is feasible in many ways.

One of the significant hardware recommendations that can be applied is using higher standard UAVs like the Parrot ANAFI Ai 4G robotic UAV with 48 MP imaging precision and intelligent obstacle avoidance for autonomous navigation and photogrammetry missions and an aboard SDK for software development.

Additionally, a more robust ground control station computer can be used to train more complex deep learning architectures that provide higher accuracy, precision, and better image segmentation like Faster R-CNN or Mask R-CNN. Moreover, the powerful computer can significantly enhance the real-time operation in terms of FPS speed, Wi-Fi communication, and faster response, dramatically boosting navigation and inspection performance.

Autonomous navigation can be enhanced by employing optimization algorithms like the SLAM algorithm, which provides better state estimation for the

UAV trajectory. In addition, the autonomous navigation did not consider obstacle avoidance since the operation ROI is across solar arrays that have no existence of humans or obstacles, as well as not having obstacle avoidance sensors like ultrasonic and laser sensors on the Tello drone. However, it is better to consider additional safety precautions by enabling obstacle avoidance algorithms and sensors in case of any possible intersection occurrence.

Lastly, to validate the system performance, actual captured data from solar power generation facilities could significantly enhance the system performance in actual scenarios.

Acknowledgment

I would like to say a special and deepest thanks to Ir. Ts. Dr. Lau Chee Yong for taking me on as his student for this FYP, for which all knowledge and experiences would not have been obtained and learnt without his proper guidance and the tons of help that he had given me throughout this journey. Additionally, I also would like to thank Ts. Dr. Shankar Duraikannan for his feedbacks, support, guidance during this project that helped me throughout my journey.

References

1. Upadhyay, J.; Rawat, A.; and Deb, D. (2021). Multiple drone navigation and formation using selective target tracking-based computer vision. *Electronics*, 10(17).
2. Niccolai, A.; Grimaccia, F.; Leva, S.; and Eleftheriadis, P. (2021). Photovoltaic plant Inspection by means of UAV: current practices and future perspectives. *IEEE International Conference on Environment and Electrical Engineering and 2021 IEEE Industrial and Commercial Power Systems Europe*. Bari, Italy, 1-6.
3. Liao, K.C.; and Lu, J.H. (2020). Using Matlab real-time image analysis for solar panel fault detection with UAV. *Journal of Physics: Conference Series*, 1509(1).
4. Henry, C.; Poudel, S.; Lee, S.-W.; and Jeong, H. (2020). Automatic detection system of deteriorated PV modules using drone with thermal camera. *Applied Sciences*, 10(11).
5. Sandino, J.; Maire, F.; Caccetta, P.; Sanderson, C.; and Gonzalez, F. (2021). Drone-based autonomous motion planning system for outdoor environments under object detection uncertainty. *Remote Sensing*, 13(21).
6. Huang, Z.-J.; and Lau, C.-Y. (2020). Camera vision foreground detection employing frame differencing and background subtraction in solar farm monitoring application. *Solid State Technology*, 63(1), 883-892.
7. Irfan, M.; Dalai, S.; Kishore, K.; Singh, S.; and Akbar, S.A. (2020). Vision-based guidance and navigation for autonomous MAV in indoor environment. *Proceedings of the 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*. Kharagpur, India, 1-5.
8. Chen, W.; Xie, D.; Zhang, Y.; and Pu, S. (2019). All you need is a few shifts: Designing efficient convolutional neural networks for image classification. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7241-7250.

9. Li, X.; Yang, Q.; Lou, Z.; and Yan, W. (2018). Deep learning based module defect analysis for large-scale photovoltaic farms. *IEEE Transactions on Energy Conversion*, 34(1), 520-529.
10. Lu, Y.; Xue, Z.; Xia, G.-S.; and Zhang, L. (2018). A survey on vision-based UAV navigation. *Geo-Spatial Information Science*, 21(1), 21-32.
11. Krishnan, R.A.; Jisha, V.R.; and Gokulnath, K. (2018). Path planning of an autonomous quadcopter based delivery system. *Proceedings of the International Conference on Emerging Trends and Innovations in Engineering And Technological Research (ICETIETR)*. Ernakulam, India, 1-5.