

Article

Coverage Path Planning with Semantic Segmentation for UAV in PV Plants

Andrés Pérez-González * , Nelson Benítez-Montoya , Álvaro Jaramillo-Duque 
and Juan Bernardo Cano-Quintero 

Research Group in Efficient Energy Management (GIMEL), Electrical Engineering Department,
Universidad de Antioquia, Calle 67 No. 53-108, Medellín 050010, Colombia;
nelson.benitez@udea.edu.co (N.B.-M.) alvaro.jaramillod@udea.edu.co (Á.J.-D.);
bernardo.cano@udea.edu.co (J.B.C.-Q.)

* Correspondence: afernando.perez@udea.edu.co; Tel.: +57-322-639-7758



Citation: Pérez-González, A.é.; Benítez-Montoya, N.; Jaramillo-Duque, Á.; Cano-Quintero, J.B. coverage path planning with Semantic Segmentation for UAV in PV Plants. *Appl. Sci.* **2021**, *11*, 12093. <https://doi.org/10.3390/app112412093>

Academic Editors:
Luis Hernández-Callejo,
Maria del Carmen Alonso García
and Sara Gallardo Saavedra

Received: 16 November 2021
Accepted: 15 December 2021
Published: 19 December 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Abstract: Solar energy is one of the most strategic energy sources for the world's economic development. This has caused the number of solar photovoltaic plants to increase around the world; consequently, they are installed in places where their access and manual inspection are arduous and risky tasks. Recently, the inspection of photovoltaic plants has been conducted with the use of unmanned aerial vehicles (UAV). Although the inspection with UAVs can be completed with a drone operator, where the UAV flight path is purely manual or utilizes a previously generated flight path through a ground control station (GCS). However, the path generated in the GCS has many restrictions that the operator must supply. Due to these restrictions, we present a novel way to develop a flight path automatically with coverage path planning (CPP) methods. Using a DL server to segment the region of interest (RoI) within each of the predefined PV plant images, three CPP methods were also considered and their performances were assessed with metrics. The UAV energy consumption performance in each of the CPP methods was assessed using two different UAVs and standard metrics. Six experiments were performed by varying the CPP width, and the consumption metrics were recorded in each experiment. According to the results, the most effective and efficient methods are the exact cellular decomposition boustrophedon and grid-based wavefront coverage, depending on the CPP width and the area of the PV plant. Finally, a relationship was established between the size of the photovoltaic plant area and the best UAV to perform the inspection with the appropriate CPP width. This could be an important result for low-cost inspection with UAVs, without high-resolution cameras on the UAV board, and in small plants.

Keywords: deep learning (DL); unmanned aerial vehicle (UAV); photovoltaic (PV) plants; semantic segmentation; coverage path planning (CPP)

1. Introduction

According to REN21, over the past two years, global photovoltaic (PV) plants capacities and annual additions have grown and expanded rapidly. For instance, 621 GW were installed in the year 2019 and 760 GW in the year 2020 [1], despite the reduction in electricity consumption and shifted daily demand patterns due to the COVID-19 pandemic [2]. Additionally, it has become one of the most profitable options and is an energy resource that has recently decreased in cost. As a result, solar electricity generation has grown in residential, commercial, and utility-scale projects [3]. The future of PV generation will focus on optimizing hybrid systems [4] and improving the performance of each element of the system, as well as reducing their cost due to large-scale production [5]. Furthermore, the sector trend has been asking for low prices, and the competitive market has encouraged investment in solar PV technologies across the entire value chain, particularly in solar cells and modules, to improve efficiencies and reduce the levelized cost of energy (LCOE) [1]. As a result, PV power plants could grow almost sixfold over the next ten years, reaching a

cumulative capacity of 2840 GW globally by 2030 and rising to 8519 GW by 2050, according to [6].

Thus, it is established that the number of PV plants and generated power have increased around the world. This implies specific technical challenges in their maintenance and operation (O and M) [7]. Some of these threats affect their production, which increases cost, and decreases profitability. The most common threats are the failures in the inverters and PV modules [8]. Through dirty equipment, the state of the environment, or manufacturing defects in the PV modules, PV plant energy generation can be curtailed by 31% in the worst cases [9–11].

One must consider that PV plants are commonly installed on roofs, rooftops, canopies, or facades for urban environments. Likewise, solar farms utilize rural environments, such as deserts, plains, and hills [7,12,13]. Depending on the location of the PV plant, the manual inspection tasks could be exhausting and take up to 8 h/MW, if the number of modules is considerable. The amount of inspection time increases for solar PV plants on rooftops or canopies, by virtue of aspects of their installation [14]. In addition, inspections to detect threats in the panels must be conducted by trained personnel. In some cases, problems occur in elevated installations, for which special training and certification is required. These jobs could put people and facilities at risk [15].

In recent decades, unmanned aerial vehicles (UAVs) have been increasingly used in inspection and patrol tasks [16–18]. UAV-based applications for PV plant inspections have many advantages in comparison with the manual inspection methods. The main advantages are flexibility, lower cost, larger area coverage, faster detection, higher precision, and the capacity to perform a superior and automatic inspections [11,17–21].

There are many approaches to performing an inspection with UAVs in PV modules. One of them used UAVs with a thermal imaging camera to take photos in the infrared spectrum to evaluate the UAVs parameters, such as height, speed, viewing angle, sun reflection, irradiance and temperature, all of which are necessary to perform defect inspection [22]. In a second approach, UAVs were implemented to inspect different solar PV plants, wherein analysis of the correlation between altitude and the pixel resolution was used to detect PV panel defects and features like shape, size, location, and color, among others, of a particular defect were also detected [7,23–26]. In a third approach, the authors proposed PV plant fault inspection with UAVs using image mosaics combined with orthophotography techniques to create a digital map; image mosaics were combined with multiple visible and/or infrared range images into a single mosaic image covering a large area [14,27,28]. Whereas the orthophotography technique is a vertical photograph that shows images of objects in true planimetric position [29]. Thus, these two techniques were integrated with previous works to achieve an advanced tool that allows monitoring and taking actions in the operation and maintenance (O and M) of PV plants [7]. In short, this tool has been used to detect defects and dust or dirt in PV modules [28]. Apart from this, some approaches have been developed for the detection of defects in PV modules using artificial vision techniques, machine learning, deep learning, and the integration therewith of the previous approaches [7,24,30–33]. Additionally, these approaches to planning the UAV's flight path were configured from a ground control station (GCS) program. Irrespective of these approaches, an important aspect is that the UAV should automatically follow the path to cover important points in PV plants. Many research efforts have been made to calculate the paths of and solve the waypoint planning problem for UAV inspection of solar PV plant applications [34–37], but none of them propose the coverage path planning (CPP) as a method to complete this task.

The CPP, given a region of interest (RoI) in a 2D environment, consists of calculating the path that passes through each one of the points that make up the desired environment and must be found considering the limitations of movement [38,39]. CPP is classified as a classical NP-hard problem in the field of computational complexity. These problems were initially analyzed for indoor environments with mobile robots. However, with the development of GPS, CPPs began to be used for missions with UAVs. Due to the envi-

ronment in which the task is performed and the obstacles present, precise localization in the environment is an arduous task, which makes the CPP a difficult problem [40]. Additionally, it is classified as a motion planning subtopic in robotics, and has two approaches, heuristic and complete. In heuristic approaches, the robots follow a set of rules defining their behavior, but do not present a guarantee for successful full coverage. These guarantee using the cellular decomposition of the area, which involves space discretization into cells to simplify the coverage in each sub-region, unlike complete methods, which cannot afford such processing. Another important issue mentioned by the authors of reference [39] is the flight time to fully cover the area, which can be reduced using multiple robots and by reducing the number of turning maneuvers. Finally, the available RoI information is important; several approaches accept previous knowledge of the robot's respecting the search RoI (offline), while sensor-based approaches collect such information in real-time along the coverage (online) [41].

In the literature, CPP approaches are needed in several application areas, such as floor cleaning [42], agriculture [43,44], wildfire tracking [45], bush trimming [46], power line inspection [47], photogrammetry [39], visual inspection [48], and many more. Additionally, many surveys regarding CPP present several approaches and techniques for performing missions with, mostly, land vehicles [41,49,50]. The research interest in aerial robots (indoor and outdoor) has surely motivated the research of CPP [51]. This can be implemented in many UAVs platforms, such as fixed-wing, rotary-wing, and hybrid UAV (VTOL) [39]. Rotary-wing UAVs are inexpensive and have good maneuverability, and their small payload capacity limits the weight of on-board sensors and flight time. Hence, they are more suitable for CPP missions on a small scale. Additionally, the increasing usage of UAVs in applications with complicated missions has led to CPP methods being a very active research area for single and multiple UAVs, especially recently [39,52–54]. As evidenced in a previous work [43], the classic taxonomy of coverage paths in UAVs are classified into no decomposition, exact cellular decomposition, and approximate cellular decomposition. The first performs the coverage with a single UAV, for which no decomposition technique is required, because the shape of the RoI has a non-complex geometry. The second divides the free space into simple, non-overlapping regions called cells. The union of all cells fills the free space completely. The cells in which there are no obstacles are easy to cover and can be covered by the robot with simple movements. The third is based on grids. They use a representation of the environment decomposed into a collection of uniformly squared cells [55], considering rectangular, concave, and convex polygons for RoIs. In addition to this, CPP performances were assessed with applied metrics according to [39]. The elemental approach most used to solve offline CPP problem is the area decomposition into non-overlapping sub-regions [56], to determine the appropriate visiting sequence of each sub-region and to cover each decomposed region in a back-and-forth movement to secure a complete coverage path. As a result, the methods for obtaining complete coverage of an RoI are the exact and approximate cellular decomposition methods [57–59].

On the other hand, image processing helps to obtain a map of the robot or RoI. Robots, such as UAVs, need to know the RoI before commencing CPP [60], which represents where the PV plants are and can be determined in a process called boundary extraction [61,62]. Then, the deep learning (DL) image segmentation technique, also known as semantic segmentation [63], is achieved by applying deep convolutional neural networks (CNN), such as the U-Net network model [64,65] or the FCN model [66], which dramatically enhance the segmentation results. Once the segmentation is done and the mask is obtained or the RoI is identified, the GCS calculates the CPP that guides the UAV in the automatic plant inspection, during which it captures images of PV plants [67]. Most of the failures occur at the centimeter or millimeter level, and that poses a challenge for the inexpensive sensors available today [68]. Tests of coverage paths it can be conducted with drones in a real or simulated environment. The most viable option for this stage of the work is simulation, as verified by other research [69–72]. Simulation has been recognized as an important research tool; initially, simulation was an academic research tool, but with the

advancement of computers, simulation has reached new levels. It is a remarkable tool that guarantees support in design, planning, analysis, and making decisions in different areas of research and development. Of course, robotics, as modern technology, has not been an exception [73,74].

This work is focused on implementing the best strategy of coverage path planning (CPP) over PV plants with UAVs using semantic segmentation in a deep learning server to obtain the RoI. The experimental results were obtained by simulating the CPP methods and using UAVs, such as the 3DR Iris and Typhoon aerial robots.

The key contributions of this work are as follows:

- This work proposes CPPs as a novel strategy for conducting an inspection flight over a PV plant with an UAV, since there are no previous reports of such work.
- This work used three CPP methods over three PV plants, which were modeled in a simulation environment to evaluate metrics and parameters. As a result, a relationship was found between the CPP width and energy consumption, and according to that, the best CPP method to implement.
- This work proposes a hybrid CPP method that uses image processing and a DL server to find the RoI quickly and accurately, becoming a semi-automatic process.
- A free simulation tool is provided, with an interface to simulate the inspection of PV plants with UAVs.

This paper is structured as follows. In Section 2, necessary definitions and the techniques used to obtain the results are described. In Section 3, the three CPP methods implemented are compared to show relationships among the CPP width, number of maneuvers, and energy consumption, with the aim of finding the best CPP method to implement. Finally, in Section 4, some conclusions are given.

2. Materials and Methods

In this work, three PV power plants were selected that met the image requirements of no light distortions, non-complex geometry, and grouped panels. These plants are in different parts of the world. The first PV plant has an area of 35,975 square meters, located in Brazil (-22.119948323621525 , -51.44666423682603), known as Usina Solar Unioeste 1 [75]. The second PV plant has an area of 25,056 square meters, located in Iran (34.0504329771808 , 49.796635265177294), known as Arak power plant [37]. The third plant is in the United States (38.55989816199527 , -121.42374978032115); the plant has 1344 square meters of area, and it is part of a PV plant located on the roof of the California State University Sacramento Library (CSUSL) [76].

These plants were subjected to a series of processing stages, as shown in Figure 1. The RoIs were obtained from Google Maps satellite images with a predefined altitude, and limits, from which the image (input image) of the desired PV system was obtained as the first stage, shown in Figure 1. In the second stage, the image was entered into a DL server that was developed in this work, taking into account the previous work [65]. The DL server was launched with TensorFlow [77] and Flask [78]. This stage obtains a mask (image segmentation) of the PV plant, as shown in Figure 1. In the third stage it was necessary to apply a series of OpenCV functions (post-processing), as referenced in Algorithm A1 in the Appendix A, to adjust the mask (output mask) to the PV plant area, as seen in Figure 1. Then, the output mask or RoI was introduced in the GUI interface, where the CPP method was selected to internally execute. Later, the path GPS positions (CPP computed) were sent to the UAV through MAVlink commands [79,80]. The UAV executed these commands in the Gazebo platform (CPP simulation) and, at the same time, the simulation data was fed back the GUI interface. Simultaneously all GPS points reached by the UAV were drawn in the GCS platform (CPP). In this stage, the trajectory was validated. Each stage is described in greater detail in the following sections.

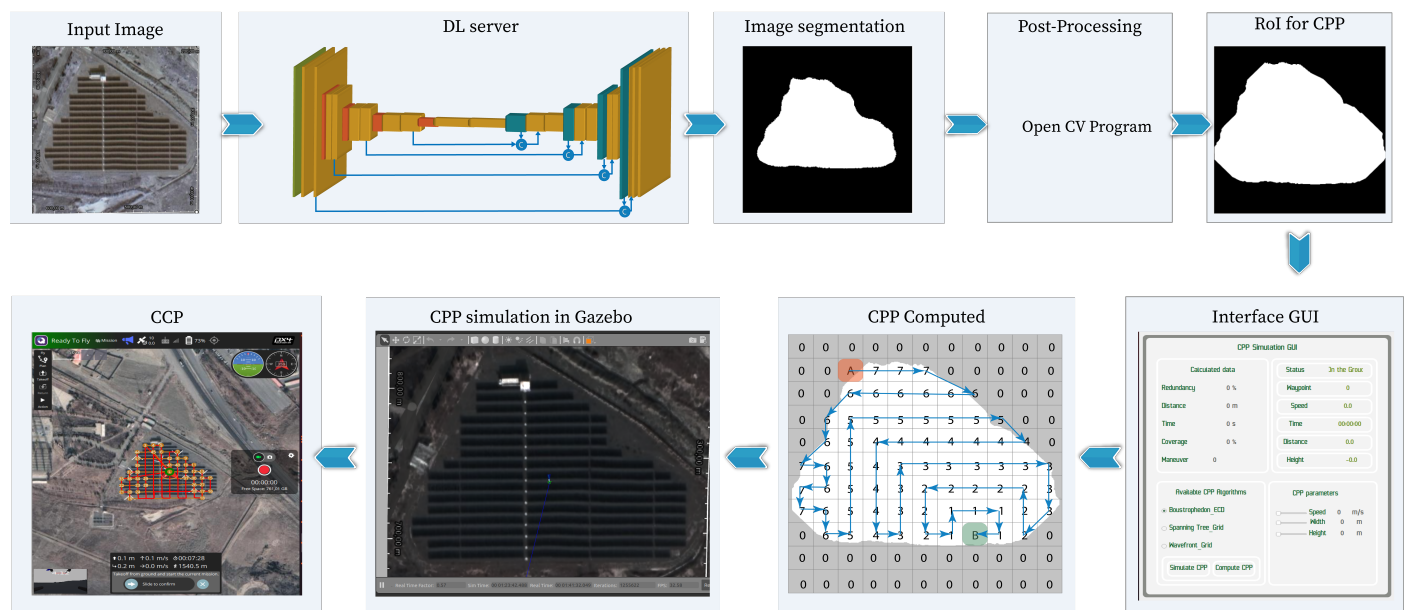


Figure 1. Stages for CPP with semantic segmentation for UAV in PV plants.

2.1. Deep Learning (DL) Server for Segmentation

A deep learning (DL) server for segmentation was necessary to extract the RoIs from the Google Maps images. Additionally, the server was used to achieve this task automatically with the process called semantic segmentation, wherein each pixel is labeled with the class of its enclosing object [65,66]. In previous work, a convolutional neural network was proposed, wherein a public database was used; this data was prepared, resized for training, and assessed with two network structures. The U-net network had the best performance, in terms of metrics, in the semantic segmentation task [65]. Then a DL server was employed to perform image segmentation and obtain the RoI [81].

2.2. Post Processing

In this step, a set of OpenCV functions were applied with Python 3.7. For example, in the first function, morphological operators like “Erode” and “Dilate” were applied to the images, then the “FindContours” function was applied to help extract their contours. The contour can be defined as a curve that joins all the continuous points at the boundary of the PV installation. So, the “ContourArea” function was then used to find the area of the previous contour. Following this pattern, the area was compared with 400 others to filter the bigger area and eliminate the little areas belonging to false positives. Then the “ApproxPolyDP” function was used to approximate a shape of the contour to another shape with fewer vertices. Subsequently, the “DrawContour” function was used to draw the resulting contour [82,83]. Finally, the “Erode” morphological operators were used again, to expand the known area and compensate for the limitations of the mask with regard to the CPP method and some faulty occurrences caused by the false positives of the DL server. The pseudocode of the openCV functions used is shown in Algorithm A1 in the Appendix A.

2.3. 2D Coverage Path Planning Method in the GUI Interface

In previous studies of CPPs, there were many existing methods from which to select to solve the CPP problem. In this work, three methods based on CPP were selected, considering the following criteria: time of execution, ease of implementation, and more, which were used to cover the RoI. The methods were selected according to [38,39].

The boustrophedon exact cellular decomposition (BECD), which was proposed by [84], was the first selected. The CPP method is delineated in Figure 2.

The second was grid-based spanning tree coverage (GBSTC), which works with cellular decomposition, first proposed by [85], and depicted in Figure 3.

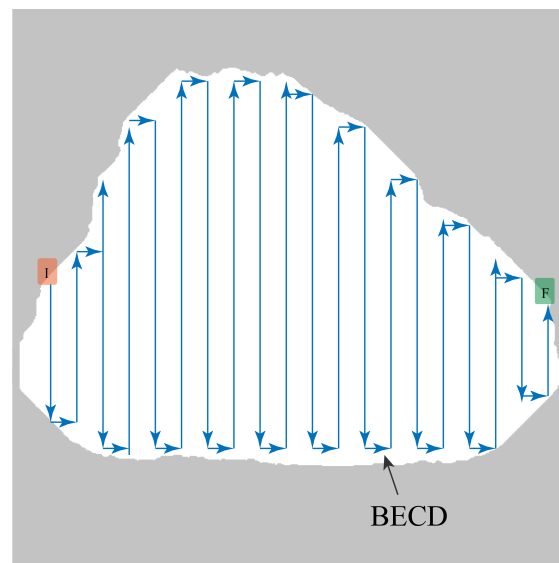


Figure 2. The RoI and the path generated from the initial point (I) to the final point (F) by BECD.

The third method selected for this project was grid-based wavefront coverage (GBWC), first proposed by [86]. The method is illustrated in Figure 4. Each of these CPP methods is explained in more detail in the following sections.

(a). The boustrophedon exact cellular decomposition (BECD) Method: This method takes the robot's free space and obstacles and splits them into cells. These cells are covered by the robot using a back-and-forth pattern from the initial point to the final point, using maneuvers of 90 degrees to change direction from south to north or vice versa, as shown in Figure 2. This method improves the trapezoid decomposition technique, as it exploits the structure of the polygon to determine the start and end of an obstacle, and thus is able to divide the free space into a few cells that do not require a redundant step, and it permits the coverage of curved areas [84].

(b). Grid-Based Spanning Tree Coverage (GBSTC): This method is based on approximate cell decomposition and differs from the previous method in that the following postulates were considered. First, the method divides the space into grids of side L . Second, the robot only moves in perpendicular directions to the sides of the grid. Third, every grid is subdivided into four grids of side $L/2$. Finally, GBSTC discards space that is partly occupied by obstacles. Consequently, considering these previous postulates, the method consists of several stages: in the first stage, a graphic structure is defined, $S(N, E)$, where N is nodes, defined as the central point of each grid, and E is edges, defined as the line segments connecting the centers of adjoining grids, as shown in Figure 3a. In the second stage, the method builds a spanning tree for S , and employs this tree to plan a cover path as follows. Starting in grid I with a sub-grid of side $L/2$, the robot begins by travelling between adjoining sub-grids along a path that moves around the spanning tree at a constant distance and in a counterclockwise direction, finishing when the initial sub-grid, I , is found again, which means it is also in the final point, F [85]. An example of this method is illustrated in Figure 3b. The approximation depends on the side length, L , of the grid.

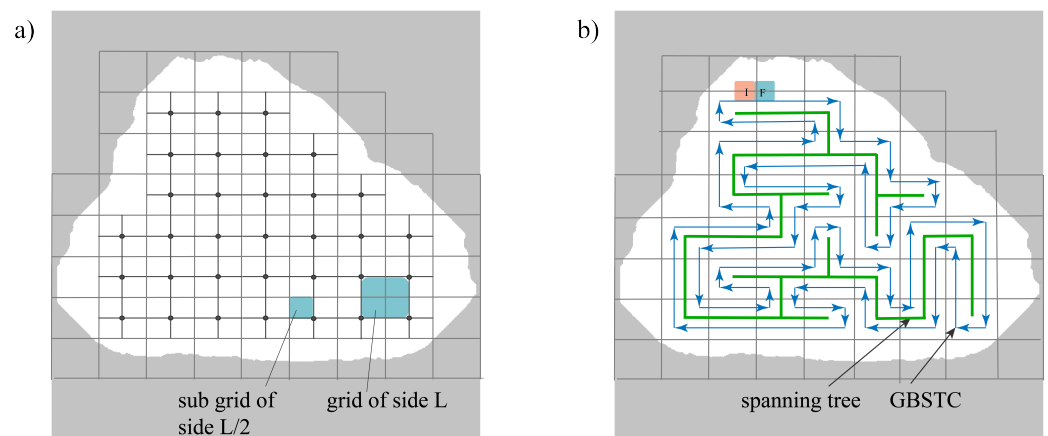


Figure 3. (a) Approximate cell decomposition in grids, and sub grids. (b), coverage path generated with the GBSTC method.

(c). grid-based wavefront coverage (GBWC): The first grid-based method proposed for CPP, GBWC is an offline method that uses a grid representation, in addition to applying a full CPP method. The method requires an initial grid, I , and a final grid, F . A distance transformation that propagates a wavefront from the final to the initial point is used to assign a specific number to each item on the grid. That is, the method first assigns a zero to the final item, and then a one to all its surrounding grids. Then all unmarked grids adjoining those marked one are numbered two. The process is incrementally repeated until the wavefront reaches the initial grid [86], as illustrated in Figure 4a.

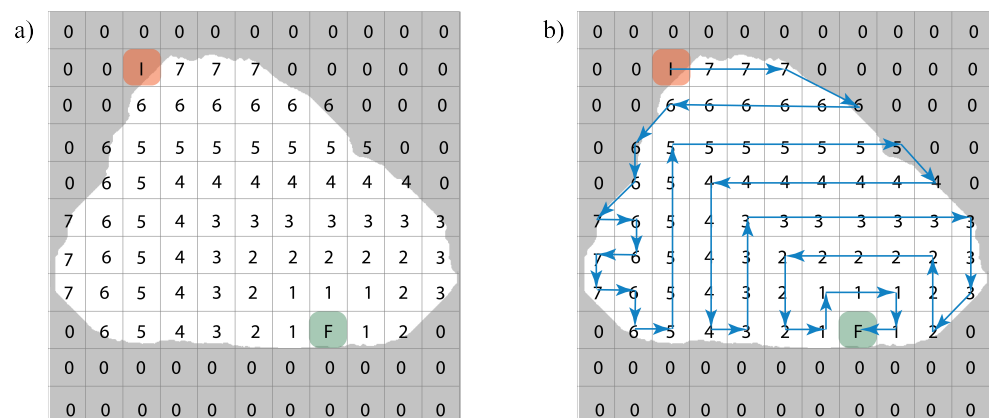


Figure 4. (a) Wavefront distance transforms for the selection of the initial position (I) and final position (F). (b) Coverage path generated using the wavefront distance transforms with the selection of the initial position (I).

Once the distance transformation is determined, a coverage path can be found by starting at the initial grid, I , and selecting the adjoining grid with the highest number that has not been explored. If two or more are unexplored, and adjoining grids share the same number, one of them is randomly selected, as shown in Figure 4b.

2.3.1. Metrics

The metrics evaluate the performance of the three CPP methods. Such assessment can be performed by considering five commonly used metrics to evaluate the effectiveness of the proposed CPP methods both theoretically and in simulation (dynamically) [7,41,43,62]. These five metrics are covered path length, flight time, energy consumption, redundancy of points traveled, and percentage of coverage of the total area. Each of these metrics is described below.

The L_{ct} metric (covered path length) is the length of the entire path covered by the UAV from the initial to the final point. For a trajectory in a 2D plane composed of n points, assuming the initial point as $(x_1, f(x_1))$ and the end point as $(x_n, f(x_n))$, the L_{ct} can be computed as shown in Equation (1):

$$L_{ct} = \sum_{i=1}^{n-1} \sqrt{(x_{i+1} - x_i)^2 + (f(x_{i+1}) - f(x_i))^2} \quad (1)$$

where $(x_i, f(x_i))$, in which $i = 1, 2, \dots, n$, representing all the n points of the UAV flight path in 2D coordinates. More details can be found in [87].

The flight time metric of the PX4 SITL Gazebo model is the time required to travel the total flight path (takeoff, path travel, and landing) with a dynamic speed that considers the inertia and the variation in speed due to UAV turns angles. These data are collected through sensors in the Gazebo plugins [88].

The redundancy of points traveled, $R_{\%}$, corresponds to the number of points that are visited more than once from the total number of points that the path contains, Equation (2).

$$R_{\%} = \frac{P_{pc}}{P_{vmo}} \quad (2)$$

where P_{pc} , and P_{vmo} correspond to the number of points the trajectory contains, and the number of points visited more than once.

The percentage of coverage of the total area, $C_{\%}$, measures the number of effectively covered points within the total number thereof by the points the area to be covered contains, given by Equation (3).

$$C_{\%} = \frac{P_v}{P_T} \quad (3)$$

where P_v , and P_T correspond, respectively, to the total number of points visited and the total number of points in the area.

Moreover, the number of maneuvers metric, which is the number of turning maneuvers the UAV performs on a path, is often used as the main performance metric in coverage [89,90].

The energy consumption metric is computed from the voltage and current data from the power module of the PX4 SITL model [91]. Its value depends on parameters, such as the CPP width between lines and the speed and the height of the UAV at the time of implementing the CPP, which were configured in the interface and were simulated in Gazebo.

To validate the results of the methods described above, the BECD, GBSTC, and GBWC methods were implemented in two UAVs, simulated in Gazebo, and their performances were assessed by the metrics presented above [39]. The next section describes the results and compares the models in detail.

2.4. Simulation and Validation Platform

Based on [92], the Gazebo platform was selected to execute the simulation experiments, as it has extensive documentation on its webpage. In addition, it is the most mentioned and used simulation platform in previous work [73,74] that implemented path planning or CPP, used UAV sensors, or deployed several UAVs [92]. Gazebo also allows the modeling of commercial UAVs using the PX4 autopilot software [91], as shown in Figure 5. In addition, this figure shows one of the experiments conducted with the Typhoon UAV, flown over the CSUSL plant. The UAV sonar sensor, represented by blue lines, is also shown, as is the UAV camera, in the box at the upper right.

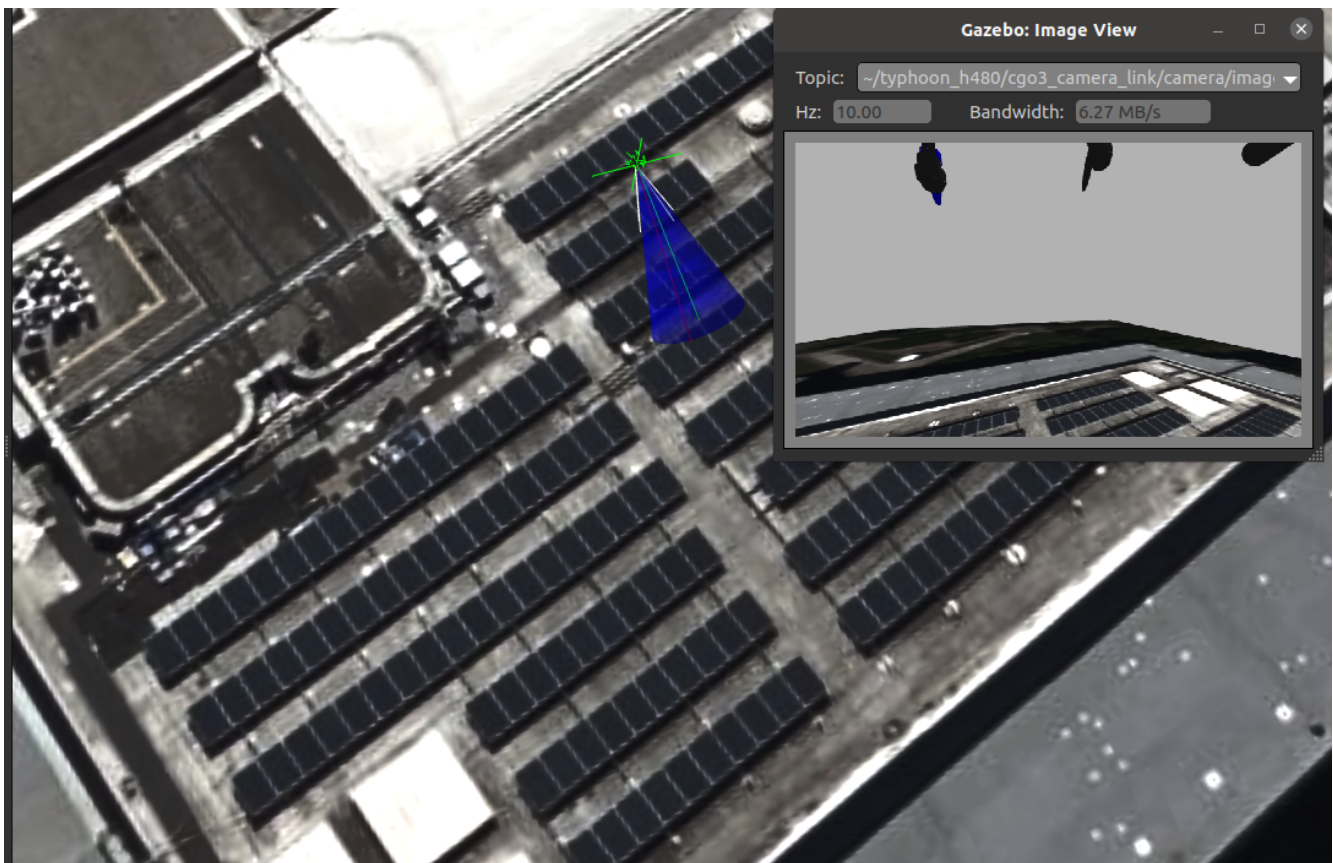


Figure 5. UAV over A PV plant, simulated in Gazebo.

The integration of Gazebo [92], PX4 [91], Python, and the CPP methods was implemented using ROS (Robot Operating System) as middleware [93]. This tool allows communication among nodes. The nodes are processes, and each node has a task associated with it, such as sending a Mavlink command to control the UAV trajectory and permitting reading messages from the UAV to discern its status in flight, using a simulation mode referred to as software in the loop (SITL). This simulator provides the ability to run different vehicles, such as a plane, copter, or rover, without a need for any microcontrollers or hardware [94]. In addition, two PX4 autopilot rotary wing UAVs, the 3DR Iris and Typhoon UAV, were chosen because they have good maneuverability and are more suitable for small-scale CPP missions. Furthermore, these UAVs have been widely used in other research [67,70,95].

The GCS software (QGroundControl) was selected to validate the CPP calculated for each UAV and PV plant obtained, because it is the most compatible tool with the PX4 autopilot. It is also recommended on the PX4 webpage. On other hand, another compatible tool, namely a GUI, was designed using Qt to modify and vary the parameters and to convert the way points from RoI pixels to geo-referenced points [96].

3. Results and Discussion

The procedure described above—DL-server, post-processing, CPP, and metrics evaluation—are represented in Figure 1, was applied to three different PV plant images. The results obtained are described in the following sections.

3.1. Results with Deep Server, and OpenCV Functions

OpenCV functions and a DL server were combined to accurately extract the mask and to trade off the DL server results of the PV plant area. Then, the CPP methods used the mask as a region of interest (RoI).

The stages to obtain the results of the RoI in the three images are shown in Figures 6–8. In the first stage, a high-resolution image from Google Maps, of predefined height and width, was obtained, and used as an input image in the DL server, depicted in Figure 6a. In the second stage, the output image of the DL server was the mask, shown in Figure 6b. In the third stage, the opening function was used in the mask, Figure 6c. In the fourth stage, the RoI was obtained using the draw contour method, as seen in Figure 6d. Finally, the RoI was blended with the input image with the aim of comparing the results in Figure 6e. The results were satisfactory and can be adapted, depending on the environment.

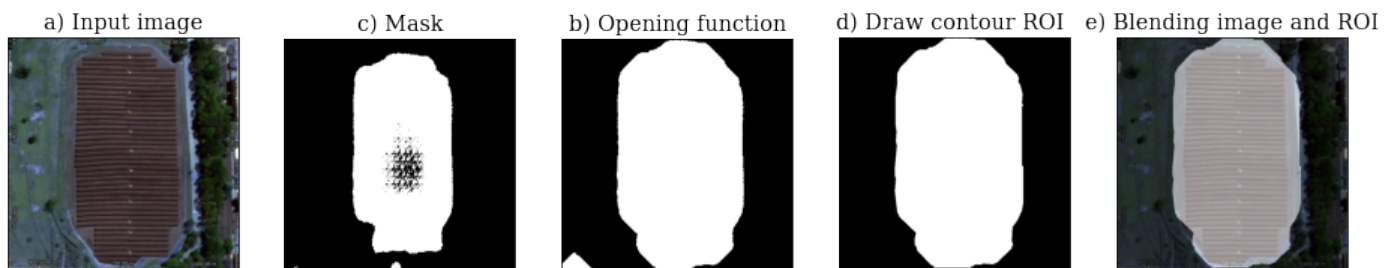


Figure 6. Steps of boundary extraction by the DL server and OpenCV functions in the Unioeste 1 PV plant.

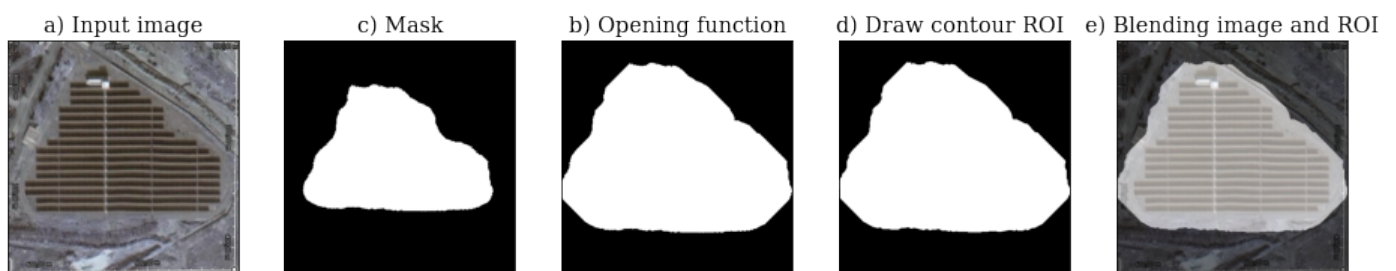


Figure 7. Steps of boundary extraction by DL server, and OpenCV functions in Arak PV plant.

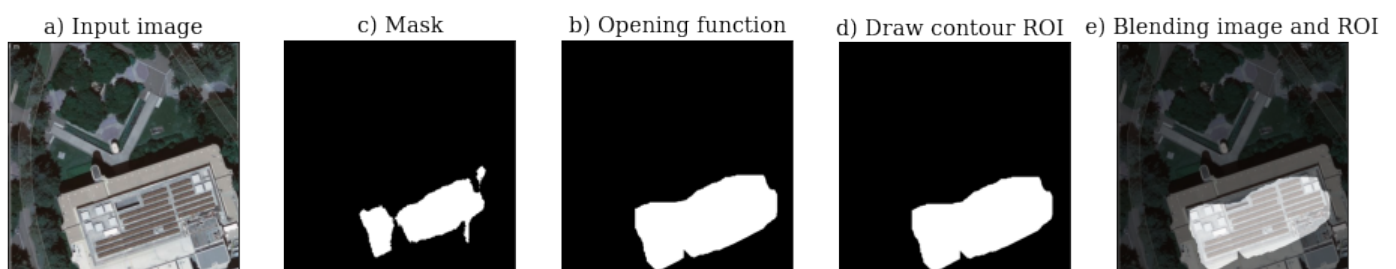


Figure 8. Steps of boundary extraction by DL server, and OpenCV functions in CSUSL PV plant.

3.2. Results of the CPP Method

The CCP method results were obtained from six experiments that implemented the five metrics previously described. Each experiment was conducted by selecting a PV plant and a UAV, then choosing the CPP method and the width, speed, and height of the UAV over the flight path. All of this was done in the implemented GUI interface. The six tests were conducted by varying the CPP width parameter, while other parameters were not varied, and the same experimental conditions were maintained throughout.

For these experiments two UAVs were selected, the Typhoon [97] and 3DR Iris [98], as mentioned in Section 2.4. Three simulated PV plants were also chosen, as highlighted in Section 2. The metrics referenced in Section 2.3.1 were assessed for each test. Battery consumption was obtained from the GCS software with the SITL parameter activated. The rest of the data was collected from the Gazebo simulations. The experiments are explained in detail in the following sections, and some results are discussed.

3.2.1. The Three First Experiments with a Typhoon UAV

The three first CPP experiments were conducted with a Typhoon UAV, varying the CPP width between 0 and 20 m, depending on the PV plant being covered. For example, in the first and second PV plants, Unioeste 1, and Arak, respectively, the CPP widths were varied between 5 and 20 m. In the third CSUSL PV plant, the width was varied between 1 and 8 m, due to the method's restrictions on running in small areas with a large width. Then, in each of these experiments, the resulting metrics were annotated in Tables, as shown in Appendix B.

The logged metrics were utilized to draw a clustered columns and lines chart for experiments 1, 2 and 3, performed with the Typhoon UAV, to highlight the most important information and to determine their correlations with each other. In these graphs the vertical axis was scaled logarithmically and the horizontal axis was labeled with the type of CPP method in use; in this way, it is possible to visualize the correlation of flight time, number of maneuvers, and path length covered with the energy consumption by each CPP method, as shown in Figures 9–11. In addition, the relationship between the redundancy metric and the CPP width is identified, showing that this metric is higher in BECD, but does not affect energy consumption, which is the primary metric of interest.

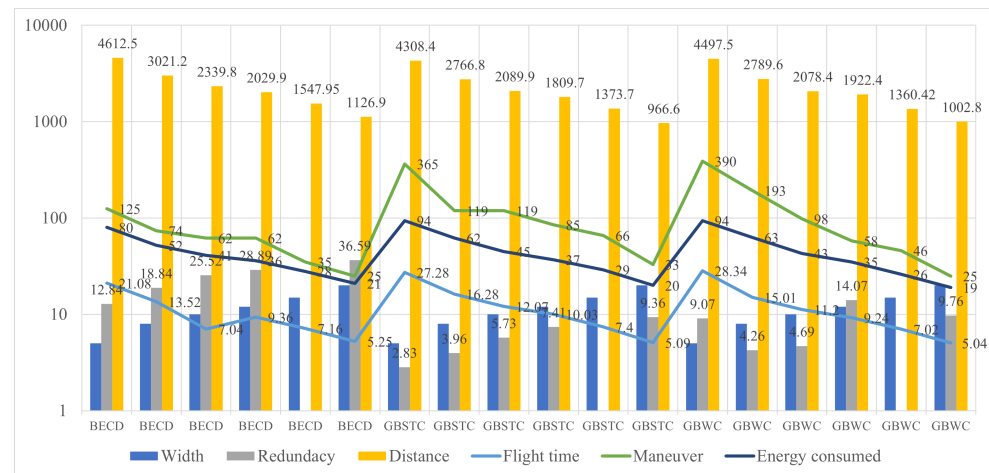


Figure 9. Performance metrics of Typhoon UAV on Unioeste PV plant (Experiment 1).

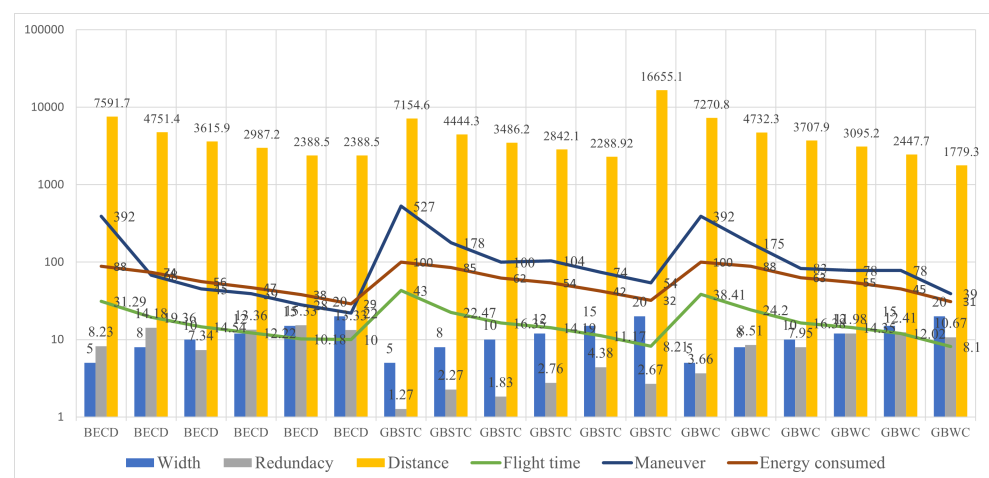


Figure 10. Performance metrics of Typhoon UAV on Arak PV plant (Experiment 2).

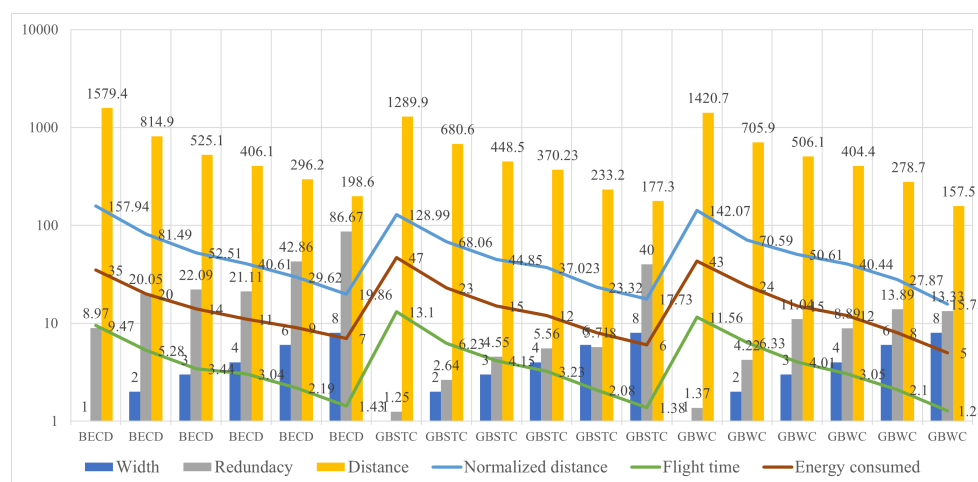


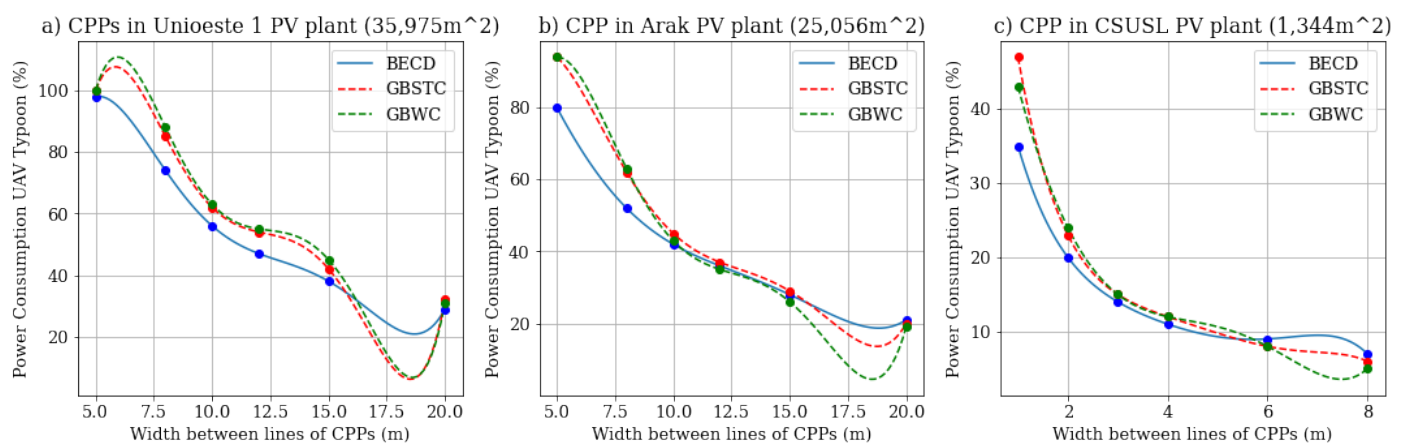
Figure 11. Performance metrics of Typhoon UAV on CSUSL PV plant (Experiment 3).

The UAVs' energy consumption, in conjunction with the CPP widths, were obtained from the energy consumption results logs, then a new Table 1 was composed of three large columns, corresponding to all the assessed PV plants, and each PV plant column is made up of three CPPs with their respective values of consumed energy. The rows of the table contain the CPP widths in increasing order from top to bottom, as shown in Table 1. Where it can be observed, some parameters, such as the energy consumption with regard to CPP width, were due to the size of the RoI. Additionally, it can be observed that the greater the CPP width, the lower the energy consumption, as seen in the columns of Table 1. For example, for the BECD CPP in Unioeste 1, with a width of 5 m, the percentage of energy consumed was 98%, while, for a width of 20, the energy consumed was 29%. It is also observed that the larger the PV plant, the UAV consumed all its energy in CPPs with narrow widths; on the other hand, if the plant is small, with the same width, the UAV does not have energy consumption problems. As can be seen in the row with a width of 8, where, for the Unioeste 1 and Arak PV plants, a lot of energy was consumed, between 88% and 52%, unlike the CSUSL plant, in which where the energy consumed was very little, between 7% and 5%. In the experiments for which energy consumption, with respect to CPP width, cannot be observed, (N/A) was used to annotate these results, which happened when the CPP width was very large with respect to the RoI, as this does not allow the generation of the route, or when the CPP width was very small with regard to the RoI causing a flight path in which the UAV consumes all its energy. In short, the UAV used can be undersized or oversized with respect to its intended PV plant.

The graphs in Figure 12 were constructed from Table 1 by quintic polynomial interpolations, which requires six data points to form a curve that passes through all given data points [99], where the abscissa axis is the CPP width and the ordinate axis is energy consumption. For the experiment conducted at the PV plant Unioeste 1, the graph in Figure 12a shows that the BECD method had the lowest energy consumption when the CPP width was in the range of 5 to 16 m, while GBSTC and GBWC had the lower energy consumption when the CPP width was in the range of 16 to 20 m. For the experiment tested at the Arak PV plant, the graph in Figure 12b shows that the BECD method had the lowest energy consumption when the CPP width was in the range of 5 to 10 m, while the GBSTC and GBWC had similar energy consumption when the CPP width was in the range of 10 to 15 m, and GBWC had the lowest energy consumption when the CPP width was in the range of 15 and 20 m. For tests conducted at the CSUSL PV plant, the graph in Figure 12c shows that the BECD method had the lowest energy consumption when the CPP width was in the range of 1 to 5.5 m, while the method GBSTC had the lowest energy consumption when the CPP width was in the range of 5.5 to 8 m. All plants were recreated in a simulation environment with their real dimensions.

Table 1. Comparison of three CPP methods with regard to CPP width, for a simulated Typhoon UAV at three PV plants.

PV Plant	Energy Consumption								
	Unioeste 1 (35,975 m ²)			Arak (25,056 m ²)			CSUSL (1344 m ²)		
	BECD	GBSTC	GBWC	BECD	GBSTC	GBWC	BECD	GBSTC	GBWC
Width (m)									
1	N/A	N/A	N/A	N/A	N/A	N/A	35%	47%	43%
2	N/A	N/A	N/A	N/A	N/A	N/A	20%	23%	24%
3	N/A	N/A	N/A	N/A	N/A	N/A	14%	15%	15%
4	N/A	N/A	N/A	N/A	N/A	N/A	11%	12%	12%
5	98%	100%	100%	80%	94%	94%	N/A	N/A	N/A
6	N/A	N/A	N/A	N/A	N/A	N/A	9%	8%	8%
8	74%	85%	88%	52%	62%	63%	7%	6%	5%
10	56%	62%	63%	41.5%	45%	43%	5%	N/A	5%
12	47%	54%	55%	36%	37%	35%	N/A	N/A	N/A
15	38%	43%	45%	27%	29%	26%	N/A	N/A	N/A
20	29%	32%	31%	21%	20%	19%	N/A	N/A	N/A

**Figure 12.** Performance test of Typhoon UAV varying the CPP width.

3.2.2. The Last Three Experiments with 3DR Iris UAV

The last three CPP experiments were conducted with a 3DR Iris UAV, varying the CPP width between 1 and 20 m, depending on the PV plant covered. For example, in the first and second PV plants, Unioeste 1 and Arak, respectively, the CPP widths were varied between 8 and 20 m; in the third PV plant, CSUSL, the width was varied between 1 and 8 m, due to the restrictions of the method requiring it be run in small areas with a large width. Then, in each of these experiments, the resulting metrics were annotated in tables, as shown in Appendix C.

As in the previous chapter, the recorded metrics were used to create bar and line graphs for Experiments 4, 5 and 6, which are those conducted with the 3DR Iris UAV, to highlight the most important information and to determine correlations among the metrics. In these graphs, the vertical and horizontal axes were scaled logarithmically and labeled in the same manner as above, with the aim of visualizing the correlations between flight time, number of maneuvers and path length covered, with regard to energy consumption by each CPP method, as shown in Figure 13–15. Furthermore, the relationship between the redundancy metric and CPP width is identified, showing behavior similar to the previous UAV's data.

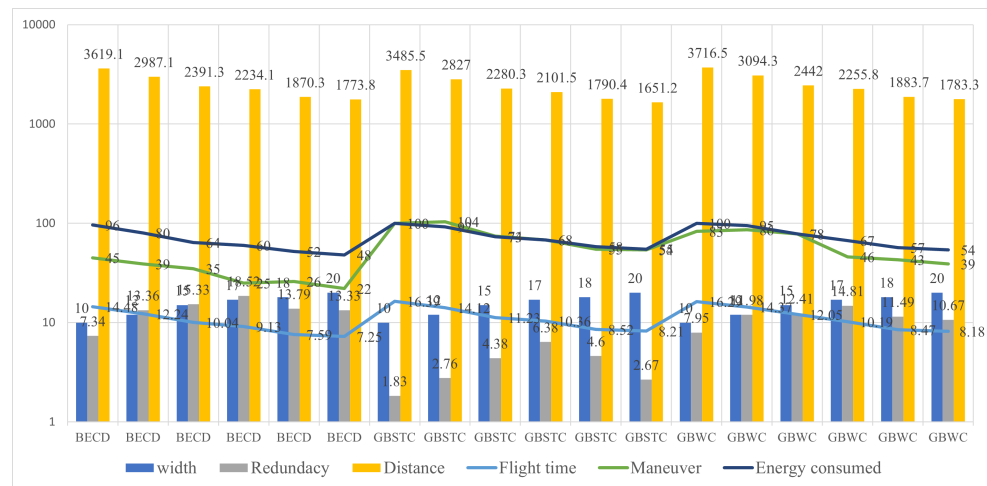


Figure 13. Performance metrics of 3DR Iris UAV on Unioeste PV plants (Experiment 4).

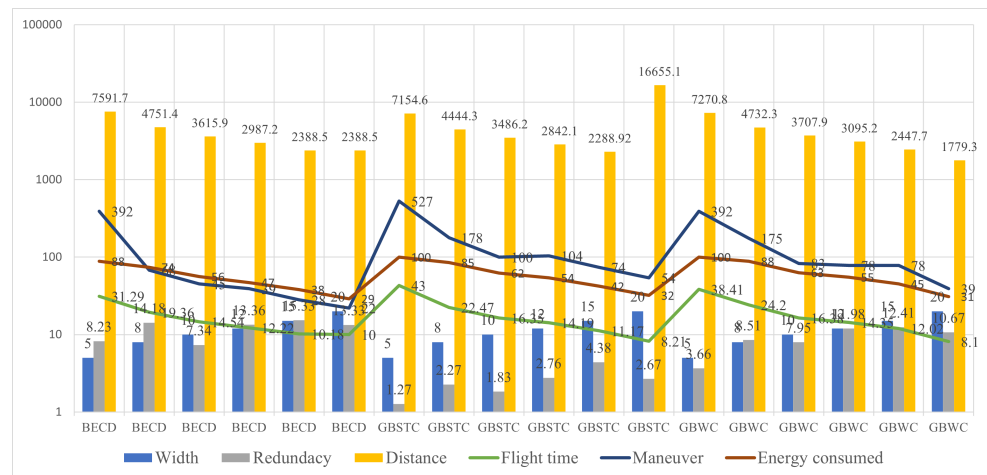


Figure 14. Performance metrics of 3DR Iris UAV on Arak PV plants (Experiment 5).

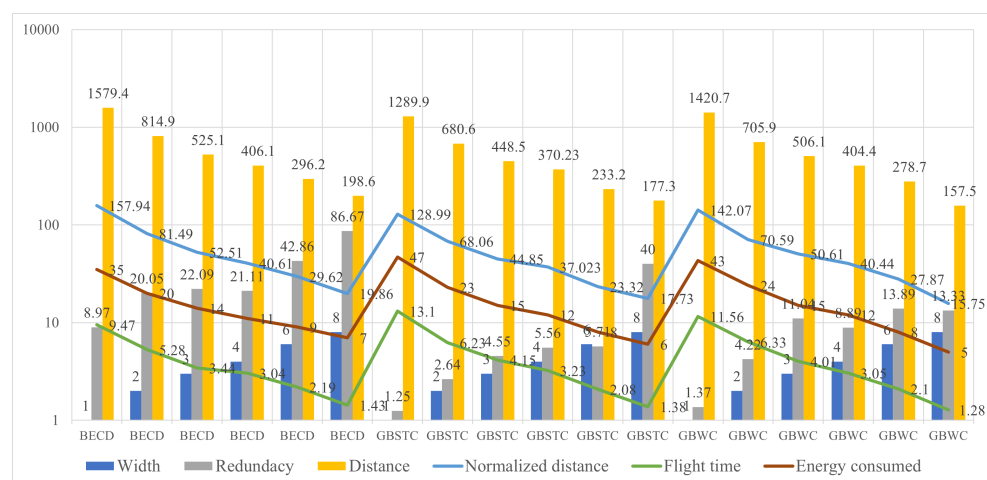


Figure 15. Performance metrics of 3DR Iris UAV on CSUSL PV plants (Experiment 6).

The UAV energy consumption and CPP widths were obtained from the energy consumption results logs, then a new Table 2 was established, in which three large columns correspond to the PV plants, where each PV plant column contains three CPPs with their respective values of consumed energy, and the rows of the table contain the CPP widths in an increasing direction from top to bottom. Here it can be observed, as in the last table, that

some parameters' values, such as the energy consumption by CPP width, are due to the size of the RoI. It can also be observed that the greater the CPP width, the lower the energy consumption, as seen in the columns of Table 2. For example, using BECD at Unioeste 1 with a width of 10 m, the percentage of energy consumed was 100%, while for a width of 20 m, the energy consumed was 48%. It is also observed that, for larger PV plants, the UAV consumed all its energy in a CPP with a small width, whereas, for small plants of the same width, the UAV did not have energy waste problems. As can be seen in the row corresponding to a CPP width of 8 m, at the Unioeste 1 and Arak plants a lot of energy was consumed, between 100%, and 88%, unlike at the CSUSL plant, where the energy consumed was very little, between 9%, and 8%. Similarly, in the previous experiments, wherein the UAV's energy was exhausted and the CPP width did not allow the generation of the route or when the CPP width was large with regard to the RoI, the results were scored with (N/A). To summarize, the UAV used is suitable for the last PV plant.

On the other hand, the 3DR Iris UAV, with its design, size, autonomy, and performance is suitable when the PV plant to be inspected is small, of an approximate size of 5000 square meters or less, such as the CSUSL PV plant, at 1344 square meters. Owing to such PV plant sizes, any coverage method used in this work can be used to cover an area with a CPP width of 1 meter, while minimizing the metrics of path length covered, flight time and maneuverability, to obtain lower energy consumption, as shown in the Table 2. In addition, this UAV did not perform well at large or medium-sized plants, since, to cover them using the CPP method, the width that must be provided is greater than 8 m; therefore, though the UAV's inspection of the PV plant had no high-resolution cameras on board, they were needed; without them, the inspection cannot be guaranteed.

Table 2. Comparison of three CPP methods with regard to CPP width, simulated in a 3DR Iris UAV on three PV plants.

PV Plant	Energy Consumption								
	Unioeste 1 (35,975 m ²)			Arak (25,056 m ²)			CSUSL (1344 m ²)		
	BECD	GBSTC	GBWC	BECD	GBSTC	GBWC	BECD	GBSTC	GBWC
Width (m)									
1	N/A	N/A	N/A	N/A	N/A	N/A	52%	72%	72%
2	N/A	N/A	N/A	N/A	N/A	N/A	30%	35%	38%
3	N/A	N/A	N/A	N/A	N/A	N/A	21%	24%	22%
4	N/A	N/A	N/A	N/A	N/A	N/A	17%	17%	15%
5	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
6	N/A	N/A	N/A	N/A	N/A	N/A	12%	12%	11%
8	100%	100%	100%	88%	98%	98%	9%	9%	8%
10	96%	100%	100%	70%	76%	71%	7%	N/A	7%
12	80%	92%	95%	60%	63%	59%	N/A	N/A	N/A
15	64%	73%	78%	45%	48%	43%	N/A	N/A	N/A
17	60%	68%	67%	N/A	N/A	N/A	N/A	N/A	N/A
20	48%	55%	54%	26%	26%	25%	N/A	N/A	N/A

The graphs in Figure 16 were constructed from Table 2 using quintic polynomial interpolations, wherein the abscissa axis is CPP width and the ordinate axis is energy consumption.

Additionally, the graph in Figure 16a shows that the BECD method had the lowest energy consumption; when the CPP width spanned the entire test range, the GBSTC and GBWC methods showed a higher energy consumption at the Unioeste 1 PV plant. The graph in Figure 16b shows that the BECD method had the lowest energy consumption when the CPP width was in the range of 8 to 10 m, whereas BECD and GBWC had similar energy consumption, leaving the GBWC method with the lowest consumption, when the

CPP width was in the range of 10 to 15 m. On the other hand, the GBWC method had the lowest energy consumption when the CPP width was in the range of 15 and 20 m; this test was conducted at the Arak PV plant. Finally, the graph in Figure 16c shows that the BECD method had the lowest energy consumption when the CPP width was in the range of 1 to 3 m, and the GBWC method had the lowest consumed energy when the CPP width was in the range of 3 to 8 m. The test was performed at CSUSL's PV plant.

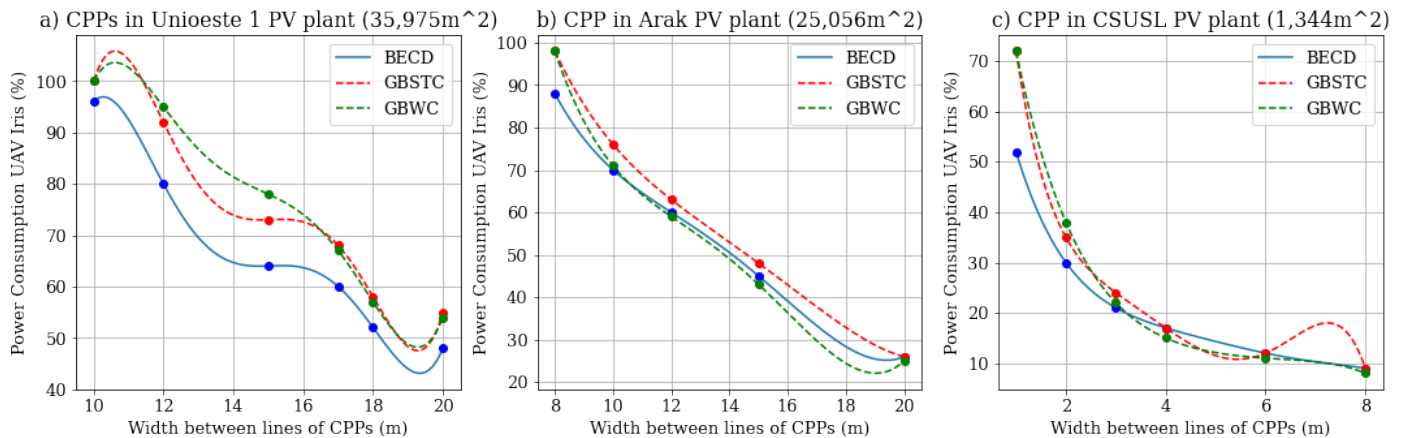


Figure 16. Performance test of UAV 3DR Iris varying the CPP width.

Summary Tables of the results of the metrics for each of the tests with the Typhoon UAV and 3DR Iris UAV are shown. All files, and logs for the experiments are available on GitHub at [100].

3.3. Discussion

The proposed strategy allows a semi-automatic and faster solution for achieving effective results when inspecting PV plants in geometrically simple areas, with certain limitations, although some results obtained in this work are theoretical results, such as from CPP width, for example; in practice, a camera of 14 megapixels is not adequate to inspect a PV plant with a CPP width of 20 m [37,101].

The results obtained in this work indicate that the most adequate method is BECD for a specific range of CPP widths, although it also shows adequate performance for various CPP widths in some RoIs. The GBWC showed a good performance when using a CPP width greater than 7 m in some RoIs, and all showed a relationship with the area to be covered, as shown in Tables 1 and 2.

An analysis of the metrics used in this work, such as redundancy, $R_{\%}$, which does not represent a significant factor when comparing these three CPP methods, showed that, although the BECD was the method with the highest redundancy, it was also the method with the lowest consumption of energy. On the other hand, the other metrics, such as the L_{ct} metric, flight time metric, and the number of maneuvers, are directly related to energy consumption, as can be seen in Appendix B and C. It also helps to conclude that the BECD method is more suitable for widths in a range between 0 and 7 m, due to the lower number of maneuvers in these ranges, as other authors have also mentioned [102,103]. The percentage of coverage of the total area, $C_{\%}$, always showed that coverage was total.

The L_{ct} metric, flight time and number of maneuvers were directly related to the consumption of energy for all the experiments performed, as seen in Figures 9–11 and 13–15, and as confirmed by other authors [89,90], in addition to helping to reinforce that the most appropriate CPP for certain ranges is the BECD.

The BECD method is the best method among the three methods tested, in a specific width range, from 0 to 7 m, for all the RoIs tested, which means that for a 10 Megapixel camera with a horizontal field of view of 7 m, the CCP method's good images, in terms of what to inspect of a PV plant can be obtained [23].

The implemented BECD method divided the RoI into small regions, and then, over these regions, it implemented the round-trip coverage pattern, called boustrophedon [38,39]. This pattern allows spending less energy consumption with small widths, due to its low number of maneuvers compared with the other methods, as shown in Appendix B and C.

On the other hand, the GBWC method allowed lower energy consumption, according to the results of the simulations, in widths greater than a certain value, but depending on RoI size; this method also allows an approximate coverage outside the RoI, a characteristic that is very important in this type of application, and that, perhaps, is not very attractive for terrestrial robotics, from which this type of method originates [86].

A RoI with many concave points is a great challenge for performance metrics, since they increase with the distance of travel, and also increase the number of maneuvers and therefore increase energy consumption, according to these results in the following works [67,103], a more detailed analysis on these characteristics was made.

In future works, the methods (BECD, GBWC) could be implemented in UAVs with characteristics similar to those used in this work; these characteristics are shown in Tables A1 and A5. Additionally, with these UAVs, an inspection could be conducted in at least one PV plant. Where one can think about the implementation of an expert system that selects the coverage path between the two types of methods (BECD, GBWC), according to the CPP width size, required for a camera with a certain resolution, and focal length.

Finally, the results obtained with the BECD, and GBWC methods differ from the results obtained by other authors [34–37]. Who did not implement CPPs to solve these types of problems; they used other types of solutions that have restrictions when inspecting PV plants with UAVs. On the contrary, this work considers the CPPs, and obtained interesting results for future real implementation. The proposed CPP will increase the possibility of using inexpensive UAV systems for the inspection of PV systems on roofs of houses, and commercial buildings, and also, of the use of CPP with small widths to complete inspections at centimeter scales of the panel where the flaws can be better seen.

4. Conclusions

In this work, a method for implementing CPP in UAV for PV plant inspection was presented. The method consisted of a series of steps, one of these was the deployment of a DL-based U-net model to establish a DL service system from which to extract the limits of PV plants by extracting the boundaries of PV plants from an image. To summarize, the method was accurate, and fast without depending on the image, with low request latency and response.

This experiment focused on three novel path planning methods in the PV inspection missions in order to find the best path for covering each of the three PV power plants with less energy consumption. A GUI interface was used to order the UAV's maneuvers in the inspection of the simulated PV plants. The results of each CPP method in the simulation were compared. The best CPPs was the BECD, for a range of CPP widths of 0 to 7 m. These path planning algorithms can be performed by any multirotor UAV that receives Mavlink commands, can carry a camera sensor, and transmit real-time video to GCS.

Performance on the CPP tasks was measured using two different types of flying robots, a Typhoon UAV and a 3DR Iris UAV. It was demonstrated that the Typhoon UAV (or one with similar characteristics) is better suited for large or medium-sized PV plants (such as those in deserts, plains, and hills); instead, the drone-like 3DR Iris UAV is more suitable for small-sized PV generation (such as on roofs and rooftops, canopies, and facades). The proposed strategy allows such comparisons to be made and enables the selection of the most suitable UAV for each type of installation.

The values obtained for the metrics collected from each of the tests show a correlation between covered path length, flight time, number of maneuvers as regards energy consumption, and ensuring that the CPPs implemented in UAVs to inspect photovoltaic plants could be similar when implemented in real plants. The results also help to predict the energy consumption of a given UAV when performing a plant inspection.

Author Contributions: Conceptualization, A.é.P.-G., N.B.-M., Á.J.-D. and J.B.C.-Q.; methodology, A.é.P.-G.; software, A.é.P.-G., N.B.-M.; validation, A.é.P.-G.; formal analysis, A.é.P.-G.; investigation, A.é.P.-G., N.B.-M.; resources, A.é.P.-G., Á.J.-D. and J.B.C.-Q.; data curation, A.é.P.-G.; writing—original draft preparation, A.é.P.-G.; writing—review and editing, A.é.P.-G., N.B.-M., Á.J.-D. and J.B.C.-Q.; visualization, A.é.P.-G.; supervision, Á.J.-D. and J.B.C.-Q.; project administration, A.é.P.-G., Á.J.-D. and J.B.C.-Q.; funding acquisition, A.é.P.-G., Á.J.-D. and J.B.C.-Q. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Colombia Scientific Program within the framework of the so-called Ecosistema Científico (Contract No. FP44842-218-2018).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The models used in the computational experiment are available at GitHub in [100].

Acknowledgments: The authors gratefully acknowledge the support from the Colombia Scientific Program within the framework of the call Ecosistema Científico (Contract No. FP44842-218-2018). The authors also want to acknowledge Universidad de Antioquia for its support through the project “estrategia de sostenibilidad”.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

BECD	boustrophedon exact cellular decomposition
GBSTC	grid-based spanning tree coverage
GBWC	grid-Based wavefront coverage
DL	deep learning
UAV	unmanned aerial vehicle
PV	photovoltaic
O and M	operation and maintenance
CAGR	compound annual growth rate
LCOE	levelized cost of energy
List of Symbols	
L_{ct}	covered path length
$R\%$	redundancy of points traveled
$C\%$	percentage of coverage of the total area

Appendix A. Algorithm

In this step, a series of OpenCV functions, in python 3.7 and in jupyter-lab, were applied.

Algorithm A1: OpenCV functions.

```

1. input :A image  $I_r$  of size  $w \times l$ 
2. output:A Map for robot
3. initialization
4. import cv 2, np, flask, tensorflow, matplotlib
5. do
6.    $Im \leftarrow cv2.imread(Mask)$ 
7.    $th \leftarrow cv2.Threshold(I_m, 128, 255, THRESH_BINARY, THRESH_OTSU)$ 
8.    $k \leftarrow cv2.getStructuringElement(MORPH_RECT, (1, 1))$ 
9.    $j \leftarrow cv2.getStructuringElement(MORPH_RECT, (5, 5))$ 
10.   $lerosion \leftarrow cv2.erode(th2[w, l], k, iteration = 2)$ 
11.   $ldilation \leftarrow cv2.dilate(lerosion[w, l], j, iteration = 20)$ 
12.   $ThImage \leftarrow Idilation.astype(np.uint8)$ 
13.   $cnt \leftarrow cv2.findContours(ThImage[w, l], RETRE_XTER, CHAIN_APX_NONE)$ 
14.  for  $i$  in  $cnt$ :
15.     $area \leftarrow cv2.contourArea(cnt)$ 
16.    if ( $area > 400$ ):
17.       $apx \leftarrow cv2.approxPolyDP(cnt, 0.0010 * cv2.arcLength(cnt, True), True)$ 
18.       $cv2.drawContours(image\_copy, [apx], -1, (0, 0, 255), 7)$ 
19.   $mask \leftarrow cv2.zeros([w, l])$ 
20.   $cc \leftarrow cv2.drawContours(mask, cnt, -1, (255, 255, 255), -1, FILLED)$ 
while  $True$ 

```

Appendix B. Tables of Typhoon UAV**Table A1.** Yuneec Typhoon UAV technical specification.

Typhoon UAV	
	
Dimensions	520 × 457 × 310 mm (20.5 × 18 × 12.2 inches)
Weight	1980 g (69.8 ounces)
Battery	5400 mAh 4S/ 14.8 V (79.9 Wh)
Camera	12.4 megapixels, 14mm/F2.8
Flight Time	up to 25 min
Flight Speed	200 m/s
Payload Capacity	10.400 g
Motor	AC YUNH520120

Table A2. Experiment 1 with three CPP methods implemented using Typhoon UAV over Unioeste 1 PV Plant.

Width	CPP	Redundacy	Distance	Flight Time	Maneuver	Energy Consumed
5	BECD	8.23	7591.7	31.29	392	88
8		14.18	4751.4	19.36	68	74
10		7.34	3615.9	14.54	45	56
12		13.36	2987.2	12.22	39	47
15		15.33	2388.5	10.18	28	38
20		13.33	2388.5	10	22	29
5	GBSTC	1.27	7154.6	43	527	100
8		2.27	4444.3	22.47	178	85
10		1.83	3486.2	16.35	100	62
12		2.76	2842.1	14.19	104	54
15		4.38	2288.92	11.17	74	42
20		2.67	16,655.1	8.21	54	32
5	GBWC	3.66	7270.8	38.41	392	100
8		8.51	4732.3	24.2	175	88
10		7.95	3707.9	16.38	83	63
12		11.98	3095.2	14.35	78	55
15		12.41	2447.7	12.02	78	45
20		10.67	1779.3	8.1	39	31

Table A3. Experiment 2 with three CPP methods implemented using Typhoon UAV over Arak PV Plant.

Width	CPP	Redundacy	Distance	Flight Time	Maneuver	Energy Consumed
5	BECD	12.84	4612.5	21.08	125	80
8		18.84	3021.2	13.52	74	52
10		25.52	2339.8	7.4	62	41
12		28.89	2029.9	9.36	62	36
15		12.84	1547.95	7.16	35	28
20		36.59	1126.9	5.25	25	21
5	GBSTC	2.83	4308.4	27.28	365	94
8		3.96	2766.8	16.28	119	62
10		5.73	2089.9	12.07	119	45
12		7.41	1809.7	10.03	85	37
15		8.75	1373.7	7.4	66	29
20		9.36	966.6	5.09	33	20
5	GBWC	9.07	4497.5	28.34	390	94
8		4.26	2789.6	15.01	193	63
10		4.69	2078.4	11.2	98	43
12		14.07	1922.4	9.24	58	35
15		6.25	1360.42	7.02	46	26
20		9.76	1002.8	5.04	25	19

Table A4. Experiment 3 with three CPP methods implemented using Typhoon UAV over CSUSL PV Plant.

Width	CPP	Redundancy	Distance	Flight Time	Maneuver	Energy Consumed
1	BECD	8.97	1579.4	9.47	192	35
2		20.05	814.9	5.28	86	20
3		22.09	525.1	3.44	62	14
4		21.11	406.1	3.04	38	11
6		42.86	296.2	2.19	27	9
8		86.67	198.6	1.43	20	7
1	GBSTC	1.25	1289.9	13.1	542	47
2		2.64	680.6	6.23	154	23
3		4.55	448.5	4.15	88	15
4		5.56	370.23	3.23	58	12
6		5.71	233.2	2.08	30	8
8		40	177.3	1.38	19	6
1	GBWC	1.37	1420.7	11.56	462	43
2		4.22	705.9	6.33	168	24
3		11.04	506.1	4.01	69	15
4		8.89	404.4	3.05	44	12
6		13.89	278.7	2.1	24	8
8		13.33	157.5	1.28	14	5

Appendix C. Tables of 3DR Iris UAV

Table A5. 3D Robotics UAV 3DR Iris technical specification.

3D Robotics 3DR Iris	
	
Dimensions	10 cm in height, 55 cm motor-to-motor
Weight	1282 g
Battery	5100 mAh 3S
Camera	N/A
Flight Time	15–20 mins
Flight Speed	11 m/s
Payload Capacity	400 g
Motor	AC 2830, 950 kV

Table A6. Experiment 4 with three CPP methods implemented using UAV 3DR Iris over Unioeste 1 PV Plant.

Width	CPP	Redundacy	Distance	Flight Time	Maneuver	Energy Consumed
10	BECD	7.34	3619.1	14.48	45	96
12		13.36	2987.1	12.24	39	80
15		15.33	2391.3	10.04	35	64
17		18.52	2234.1	9.13	25	60
18		13.79	1870.3	7.59	26	52
20		13.33	1773.8	7.25	22	48
10	GBSTC	1.83	3485.5	16.39	100	100
12		2.76	2827	14.12	104	92
15		4.38	2280.3	11.23	74	73
17		6.38	2101.5	10.36	68	68
18		4.6	1790.4	8.52	55	58
20		2.67	1651.2	8.21	54	55
10	GBWC	7.95	3716.5	16.29	83	100
12		11.98	3094.3	14.37	86	95
15		12.41	2442	12.05	78	78
17		14.81	2255.8	10.19	46	67
18		11.49	1883.7	8.47	43	57
20		10.67	1783.3	8.18	39	54

Table A7. Experiment 5 with three CPP methods implemented using the UAV 3DR Iris over Arak PV Plant.

Width	CPP	Redundacy	Distance	Flight Time	Maneuver	Energy Consumed
7	BECD	15.42	3292.7	15.31	80	95
8		18.84	3020.2	14.2	74	88
10		25.52	2339.3	12.52	62	70
12		28.89	2030.4	9.57	50	60
15		31.25	1548.5	7.27	35	45
20		36.59	1133.3	5.21	25	26
7	GBSTC	5.73	2090.4	43	527	100
8		3.96	2757.8	17.06	179	98
10		5.73	2090.4	12.36	119	76
12		7.41	1809.3	10.18	85	63
15		8.75	13,674.2	8	66	48
20		9.76	971	5.07	33	26
5	GBWC	3.66	7270.8	38.41	392	100
8		4.69	2078.8	11.47	98	2
10		4.69	2078.8	11.47	98	71
12		14.07	1922.7	9.43	58	59
15		6.25	1360.08	7.07	46	43
20		6.25	1360.08	7.07	46	25

Table A8. Experiment 6 with three CPP methods implemented using the UAV 3DR Iris over CSUSL PV Plant.

Width	CPP	Redundancy	Distance	Flight Time	Maneuver	Energy Consumed
1	BECD	10.06	1724.8	10.39	211	52
2		15.38	978	6.12	95	30
3		19.3	597.2	4.18	62	21
4		26.67	441.5	3.33	47	17
6		47.22	316.7	2.3	30	12
8		66.67	197.4	1.47	20	9
1	GBSTC	1.77	1416.4	15.31	626	72
2		3.61	697.4	7.2	220	35
3		5.85	508.6	5	108	24
4		5.56	373.4	3.28	65	17
6		19.44	270.1	2.22	32	12
8		40	198.8	1.52	19	9
1	GBWC	3.94	1414.9	15.3	626	72
2		5.53	817.4	7.36	218	38
3		7.6	553	4.36	86	22
4		8.89	404.4	3.05	44	15
6		13.89	274.7	2.13	24	11
8		20	184.9	1.41	16	8

References

- REN21®. GLOBAL STATUS REPORTS. 2021. Available online: <https://www.ren21.net/reports/global-status-report/> (accessed on 14 August 2021).
- Alkhraijah, M.; Alowaifeer, M.; Alsaleh, M.; Alfaris, A.; Molzahn, D.K. The Effects of Social Distancing on Electricity Demand Considering Temperature Dependency. *Energies* **2021**, *14*, 473. [\[CrossRef\]](#)
- Mey, A. Most U.S. Utility-Scale Solar Photovoltaic Power Plants Are 5 Megawatts or Smaller—Today in Energy—U.S. Energy Information Administration (EIA). Available online: <https://www.eia.gov/todayinenergy/detail.php?id=38272#> (accessed on 3 October 2021).
- Maghrabie, H.M.; Abdelkareem, M.A.; Al-Alami, A.H.; Ramadan, M.; Mushtaha, E.; Wilberforce, T.; Olabi, A.G. State-of-the-Art Technologies for Building-Integrated Photovoltaic Systems. *Buildings* **2021**, *11*, 383. [\[CrossRef\]](#)
- Hao, P.; Zhang, Y.; Lu, H.; Lang, Z. A novel method for parameter identification and performance estimation of PV module under varying operating conditions. *Energy Convers. Manag.* **2021**, *247*, 114689. [\[CrossRef\]](#)
- IRENA. Future of Solar Photovoltaic: Deployment, Investment, Technology, Grid Integration, and Socio-Economic Aspects. 2019. Available online: <https://www.irena.org> (accessed on 24 October 2021).
- Grimaccia, F.; Leva, S.; Dolara, A.; Aghaei, M. Survey on PV modules common faults after an O and M flight extensive campaign over different plants in Italy. *IEEE J. Photovoltaics* **2017**, *7*, 810–816. [\[CrossRef\]](#)
- Poulek, V.; Šafránková, J.; Černá, L.; Libra, M.; Beránek, V.; Finsterle, T.; Hrzina, P. PV Panel and PV Inverter Damages Caused by Combination of Edge Delamination, Water Penetration, and High String Voltage in Moderate Climate. *IEEE J. Photovoltaics* **2021**, *11*, 561–565. [\[CrossRef\]](#)
- Jamil, W.J.; Rahman, H.A.; Shaari, S.; Salam, Z. Performance degradation of photovoltaic power system: Review on mitigation methods. *Renew. Sustain. Energy Rev.* **2017**, *67*, 876–891. [\[CrossRef\]](#)
- Kaplani, E. PV cell and module degradation, detection and diagnostics. In *Renewable Energy in the Service of Mankind Vol II, Selected Topics from the World Renewable Energy Congress WREC, London, UK, 3–8 August 2014*; Springer: Cham, Switzerland, 2016; pp. 393–402.
- Di Lorenzo, G.; Araneo, R.; Mitolo, M.; Niccolai, A.; Grimaccia, F. Review of O and M Practices in PV Plants: Failures, Solutions, Remote Control, and Monitoring Tools. *IEEE J. Photovoltaics* **2020**, *10*, 914–926. [\[CrossRef\]](#)
- Donovan, C.W. *Renewable Energy Finance: Funding the Future of Energy*; World Scientific Publishing Co. Pte. Ltd.: Singapore, 2020.
- Narvarte, L.; Fernández-Ramos, J.; Martínez-Moreno, F.; Carrasco, L.; Almeida, R.; Carrêlo, I. Solutions for adapting photovoltaics to large power irrigation systems for agriculture. *Sustain. Energy Technol. Assess.* **2018**, *29*, 119–130. [\[CrossRef\]](#)
- Grimaccia, F.; Leva, S.; Niccolai, A.; Cantoro, G. Assessment of PV plant monitoring system by means of unmanned aerial vehicles. In Proceedings of the 2018 IEEE International Conference on Environment and Electrical Engineering and 2018 IEEE Industrial and Commercial Power Systems Europe (EEEIC/I&CPS Europe), Palermo, Italy, 12–15 June 2018; pp. 1–6.
- Guerrero-Liquet, G.C.; Oviedo-Casado, S.; Sánchez-Lozano, J.; García-Cascales, M.S.; Prior, J.; Urbina, A. Determination of the Optimal Size of Photovoltaic Systems by Using Multi-Criteria Decision-Making Methods. *Sustainability* **2018**, *10*, 4594. [\[CrossRef\]](#)

16. Grimaccia, F.; Aghaei, M.; Mussetta, M.; Leva, S.; Quater, P.B. Planning for PV plant performance monitoring by means of unmanned aerial systems (UAS). *Int. J. Energy Environ. Eng.* **2015**, *6*, 47–54. [\[CrossRef\]](#)
17. Shen, K.; Qiu, Q.; Wu, Q.; Lin, Z.; Wu, Y. Research on the Development Status of Photovoltaic Panel Cleaning Equipment Based on Patent Analysis. In Proceedings of the 2019 3rd International Conference on Robotics and Automation Sciences (ICRAS), Wuhan, China, 1–3 June 2019; pp. 20–27.
18. Azaiz, R. Flying Robot for Processing and Cleaning Smooth, Curved and Modular Surfaces. U.S. Patent App. 15/118,849, 2 March 2017.
19. Libra, M.; Daneček, M.; Lešetický, J.; Poulek, V.; Sedláček, J.; Beránek, V. Monitoring of Defects of a Photovoltaic Power Plant Using a Drone. *Energies* **2019**, *12*, 795. [\[CrossRef\]](#)
20. Li, X.; Li, W.; Yang, Q.; Yan, W.; Zomaya, A.Y. An unmanned inspection system for multiple defects detection in photovoltaic plants. *IEEE J. Photovoltaics* **2019**, *10*, 568–576. [\[CrossRef\]](#)
21. Alsafasfeh, M.; Abdel-Qader, I.; Bazuin, B.; Alsafasfeh, Q.; Su, W. Unsupervised fault detection and analysis for large photovoltaic systems using drones and machine vision. *Energies* **2018**, *11*, 2252. [\[CrossRef\]](#)
22. Gallardo-Saavedra, S.; Franco-Mejia, E.; Hernández-Callejo, L.; Duque-Pérez, Ó.; Loaiza-Correa, H.; Alfaro-Mejia, E. Aerial thermographic inspection of photovoltaic plants: Analysis and selection of the equipment. In Proceedings of the 2017 Proceedings ISES Solar World Congress, IEA SHC, Abu Dhabi, UAE, 29 October–2 November 2017.
23. Leva, S.; Aghaei, M.; Grimaccia, F. PV power plant inspection by UAS: Correlation between altitude and detection of defects on PV modules. In Proceedings of the 2015 IEEE 15th International Conference on Environment and Electrical Engineering (EEEIC), Rome, Italy, 10–13 June 2015; pp. 1921–1926.
24. Aghaei, M.; Dolara, A.; Leva, S.; Grimaccia, F. Image resolution and defects detection in PV inspection by unmanned technologies. In Proceedings of the 2016 IEEE Power and Energy Society General Meeting (PESGM), Boston, MA, USA, 17–21 July 2016; pp. 1–5.
25. Quater, P.B.; Grimaccia, F.; Leva, S.; Mussetta, M.; Aghaei, M. Light unmanned aerial vehicles (UAVs) for cooperative inspection of PV plants. *IEEE J. Photovoltaics* **2014**, *4*, 1107–1113. [\[CrossRef\]](#)
26. Oliveira, A.K.V.; Aghaei, M.; Madukanya, U.E.; Rüther, R. Fault inspection by aerial infrared thermography in a pv plant after a meteorological tsunami. *Rev. Bras. Energ. Sol.* **2019**, *10*, 17–25.
27. Tsanakas, J.A.; Chrysostomou, D.; Botsaris, P.N.; Gasteratos, A. Fault diagnosis of photovoltaic modules through image processing and Canny edge detection on field thermographic measurements. *Int. J. Sustain. Energy* **2015**, *34*, 351–372. [\[CrossRef\]](#)
28. Niccolai, A.; Grimaccia, F.; Leva, S. Advanced Asset Management Tools in Photovoltaic Plant Monitoring: UAV-Based Digital Mapping. *Energies* **2019**, *12*, 4736. [\[CrossRef\]](#)
29. Thrower, N.J.; Jensen, J.R. The orthophoto and orthophotomap: Characteristics, development and application. *Am. Cartogr.* **1976**, *3*, 39–56. [\[CrossRef\]](#)
30. Yao, Y.Y.; Hu, Y.T. Recognition and location of solar panels based on machine vision. In Proceedings of the 2017 2nd Asia-Pacific Conference on Intelligent Robot Systems (ACIRS), Wuhan, China, 16–18 June 2017; pp. 7–12.
31. Leva, S.; Aghaei, M. Power Engineering: Advances and Challenges Part B: Electrical Power. In *Chapter 3: Failures and Defects in PV Systems Review and Methods of Analysis*; Taylor & Francis Group, CRC Press: Boca Raton, FL, USA, 2018.
32. Kim, D.; Youn, J.; Kim, C. Automatic photovoltaic panel area extraction from uav thermal infrared images. *J. Korean Surv. Soc.* **2016**, *34*, 559–568. [\[CrossRef\]](#)
33. Li, X.; Yang, Q.; Lou, Z.; Yan, W. deep learning Based Module Defect Analysis for Large-Scale Photovoltaic Farms. *IEEE Trans. Energy Convers.* **2019**, *34*, 520–529. [\[CrossRef\]](#)
34. Ding, Y.; Cao, R.; Liang, S.; Qi, F.; Yang, Q.; Yan, W. Density-Based Optimal UAV Path Planning for Photovoltaic Farm Inspection in Complex Topography. In Proceedings of the 2020 Chinese Control And Decision Conference (CCDC), Hefei, China, 22–24 August 2020; pp. 3931–3936.
35. Luo, X.; Li, X.; Yang, Q.; Wu, F.; Zhang, D.; Yan, W.; Xi, Z. Optimal path planning for UAV based inspection system of large-scale photovoltaic farm. In Proceedings of the 2017 Chinese Automation Congress (CAC), Jinan, China, 20–22 October 2017; pp. 4495–4500. [\[CrossRef\]](#)
36. Salahat, E.; Asselineau, C.A.; Coventry, J.; Mahony, R. Waypoint Planning for Autonomous Aerial Inspection of Large-Scale Solar Farms. In Proceedings of the IECON 2019–45th Annual Conference of the IEEE Industrial Electronics Society, Lisbon, Portugal, 14–17 October 2019; Volume 1, pp. 763–769. [\[CrossRef\]](#)
37. Sizkouhi, A.M.M.; Esmailifar, S.M.; Aghaei, M.; De Oliveira, A.K.V.; Rüther, R. Autonomous path planning by unmanned aerial vehicle (UAV) for precise monitoring of large-scale PV plants. In Proceedings of the 2019 IEEE 46th Photovoltaic Specialists Conference (PVSC), Chicago, IL, USA, 16–21 June 2019; pp. 1398–1402.
38. Choset, H.M.; Lynch, K.M.; Hutchinson, S.; Kantor, G.; Burgard, W.; Kavraki, L.; Thrun, S.; Arkin, R.C. *Principles of Robot Motion: Theory, Algorithms, and Implementation*; MIT Press: Cambridge, MA, USA, 2005.
39. Cabreira, T.M.; Brisolara, L.B.; Ferreira Jr, P.R. Survey on coverage path planning with unmanned aerial vehicles. *Drones* **2019**, *3*, 4. [\[CrossRef\]](#)
40. Sebbane, Y.B. *Intelligent Autonomy of UAVs: Advanced Missions and Future Use*; CRC Press: Boca Raton, FL, USA, 2018.
41. Choset, H. Coverage for robotics—a survey of recent results. *Ann. Math. Artif. Intell.* **2001**, *31*, 113–126. [\[CrossRef\]](#)
42. Veerajagadheswar, P.; Ping-Cheng, K.; Elara, M.R.; Le, A.V.; Iwase, M. Motion planner for a Tetris-inspired reconfigurable floor cleaning robot. *Int. J. Adv. Robot. Syst.* **2020**, *17*, 1729881420914441. [\[CrossRef\]](#)

43. Coombes, M.; Fletcher, T.; Chen, W.H.; Liu, C. Optimal Polygon Decomposition for UAV Survey coverage path planning in Wind. *Sensors* **2018**, *18*, 2132. [\[CrossRef\]](#) [\[PubMed\]](#)
44. Islam, N.; Rashid, M.M.; Pasandideh, F.; Ray, B.; Moore, S.; Kadel, R. A Review of Applications and Communication Technologies for Internet of Things (IoT) and unmanned aerial vehicle (UAV) Based Sustainable Smart Farming. *Sustainability* **2021**, *13*, 1821. [\[CrossRef\]](#)
45. Pham, H.X.; La, H.M.; Feil-Seifer, D.; Deans, M. A distributed control framework for a team of unmanned aerial vehicles for dynamic wildfire tracking. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 6648–6653.
46. Kaljaca, D.; Vroegindeweij, B.; van Henten, E. Coverage trajectory planning for a bush trimming robot arm. *J. Field Robot.* **2020**, *37*, 283–308. [\[CrossRef\]](#)
47. Chang, W.; Yang, G.; Yu, J.; Liang, Z.; Cheng, L.; Zhou, C. Development of a power line inspection robot with hybrid operation modes. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 973–978.
48. Mansouri, S.S.; Kanellakis, C.; Fresk, E.; Kominiak, D.; Nikolakopoulos, G. Cooperative coverage path planning for visual inspection. *Control Eng. Pract.* **2018**, *74*, 118–131. [\[CrossRef\]](#)
49. Galceran, E.; Carreras, M. A survey on coverage path planning for robotics. *Robot. Auton. Syst.* **2013**, *61*, 1258–1276. [\[CrossRef\]](#)
50. Bormann, R.; Jordan, F.; Hampp, J.; Hägele, M. Indoor coverage path planning: Survey, implementation, analysis. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, 21–25 May 2018; pp. 1718–1725.
51. Nam, L.; Huang, L.; Li, X.; Xu, J. An approach for coverage path planning for UAVs. In Proceedings of the 2016 IEEE 14th International Workshop on Advanced Motion Control, AMC 2016, Auckland, New Zealand, 22–24 April 2016; pp. 411–416. [\[CrossRef\]](#)
52. Dai, R.; Fotadar, S.; Radmanesh, M.; Kumar, M. Quality-aware UAV coverage and path planning in geometrically complex environments. *Ad Hoc Netw.* **2018**, *73*, 95–105. [\[CrossRef\]](#)
53. Yao, P.; Cai, Y.; Zhu, Q. Time-optimal trajectory generation for aerial coverage of urban building. *Aerosp. Sci. Technol.* **2019**, *84*, 387–398. [\[CrossRef\]](#)
54. Majeed, A.; Lee, S. A new coverage flight path planning algorithm based on footprint sweep fitting for unmanned aerial vehicle navigation in urban environments. *Appl. Sci.* **2019**, *9*, 1470. [\[CrossRef\]](#)
55. Elfes, A.; Campos, M.; Bergerman, M.; Bueno, S.; Podnar, G. A robotic unmanned aerial vehicle for environmental research and monitoring. In Proceedings of the First Scientific Conference on the Large Scale Biosphere-Atmosphere Experiment in Amazonia (LBA), LBA Central Office, CPTEC/INPE, Belém, Pará, Brazil, 25–30 June 2000; pp. 12630–12645.
56. Saeed, A.; Abdelkader, A.; Khan, M.; Neishaboori, A.; Harras, K.A.; Mohamed, A. On realistic target coverage by autonomous drones. *ACM Trans. Sens. Netw. (TOSN)* **2019**, *15*, 1–33. [\[CrossRef\]](#)
57. Lingelbach, F. Path planning using probabilistic cell decomposition. In Proceedings of the IEEE International Conference on Robotics and Automation, 2004. Proceedings, ICRA'04, New Orleans, LA, USA, 26 April–1 May 2004; Volume 1, pp. 467–472.
58. Khanam, Z.; Saha, S.; Ehsan, S.; Stolkin, R.; McDonald-Maier, K. coverage path planning Techniques for Inspection of Disjoint Regions With Precedence Provision. *IEEE Access* **2021**, *9*, 5412–5427. [\[CrossRef\]](#)
59. Khiati, W.; Moumen, Y.; Habchi, A.E.; Zerrouk, I.; Berrich, J.; Bouchentouf, T. Grid Based approach (GBA): A new approach based on the grid-clustering algorithm to solve a CPP type problem for air surveillance using UAVs. In Proceedings of the 2020 Fourth International Conference On Intelligent Computing in Data Sciences (ICDS), Fez, Morocco, 21–23 October 2020; pp. 1–5. [\[CrossRef\]](#)
60. Juliá, M.; Gil, A.; Reinoso, O. A comparison of path planning strategies for autonomous exploration and mapping of unknown environments. *Auton. Robot.* **2012**, *33*, 427–444. [\[CrossRef\]](#)
61. Rodriguez-Esparza, E.; Zanella-Calzada, L.A.; Oliva, D.; Heidari, A.A.; Zaldivar, D.; Pérez-Cisneros, M.; Foong, L.K. An efficient Harris hawks-inspired image segmentation method. *Expert Syst. Appl.* **2020**, *155*, 113428. [\[CrossRef\]](#)
62. Garcia-Garcia, A.; Orts-Escolano, S.; Oprea, S.; Villena-Martinez, V.; Martinez-Gonzalez, P.; Garcia-Rodriguez, J. A survey on deep learning techniques for image and video semantic segmentation. *Appl. Soft Comput.* **2018**, *70*, 41–65. [\[CrossRef\]](#)
63. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
64. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*; Springer: Munich, Germany, 2015; pp. 234–241.
65. Pérez-González, A.; Jaramillo-Duque, A.; Cano-Quintero, J.B. Automatic Boundary Extraction for Photovoltaic Plants Using the Deep Learning UNet Model. *Appl. Sci.* **2021**, *11*, 6524. [\[CrossRef\]](#)
66. Sizkouhi, A.M.M.; Aghaei, M.; Esmailifar, S.M.; Mohammadi, M.R.; Grimaccia, F. Automatic boundary extraction of large-scale photovoltaic plants using a fully convolutional network on aerial imagery. *IEEE J. Photovoltaics* **2020**, *10*, 1061–1067. [\[CrossRef\]](#)
67. Di Franco, C.; Buttazzo, G. coverage path planning for UAVs photogrammetry with energy and resolution constraints. *J. Intell. Robot. Syst.* **2016**, *83*, 445–462. [\[CrossRef\]](#)
68. Zefri, Y.; ElKettani, A.; Sebari, I.; Ait Lamallam, S. Thermal infrared and visual inspection of photovoltaic installations by UAV photogrammetry application case: Morocco. *Drones* **2018**, *2*, 41. [\[CrossRef\]](#)

69. Coopmans, C.; Podhradský, M.; Hoffer, N.V. Software-and hardware-in-the-loop verification of flight dynamics model and flight control simulation of a fixed-wing unmanned aerial vehicle. In Proceedings of the 2015 Workshop on Research, Education and Development of Unmanned Aerial Systems (RED-UAS), Cancun, Mexico, 23–25 November 2015; pp. 115–122.
70. Roggi, G.; Niccolai, A.; Grimaccia, F.; Lovera, M. A Computer Vision Line-Tracking Algorithm for Automatic UAV Photovoltaic Plants Monitoring Applications. *Energies* **2020**, *13*, 838. [\[CrossRef\]](#)
71. Coates, E.M.; Fossen, T.I. Geometric Reduced-Attitude Control of Fixed-Wing UAVs. *Appl. Sci.* **2021**, *11*, 3147. [\[CrossRef\]](#)
72. Tullu, A.; Endale, B.; Wondosen, A.; Hwang, H.Y. Machine Learning Approach to Real-Time 3D Path Planning for Autonomous Navigation of unmanned aerial vehicle. *Appl. Sci.* **2021**, *11*, 4706. [\[CrossRef\]](#)
73. Pitonakova, L.; Giuliani, M.; Pipe, A.; Winfield, A. Feature and performance comparison of the V-REP, Gazebo and ARGoS robot simulators. In *Annual Conference Towards Autonomous Robotic Systems*; Springer: Bristol, UK, 25–27 July 2018; pp. 357–368.
74. Akcakoca, M.; Atici, B.M.; Gever, B.; Oguz, S.; Demirezen, U.; Demir, M.; Saldiran, E.; Yuksek, B.; Koyuncu, E.; Yeniceri, R.; et al. A simulation-based development and verification architecture for micro uav teams and swarms. In *AIAA Scitech 2019 Forum*; AIAA: San Diego, CA, USA, 2019; p. 1979.
75. Unoeste Terá Maior Usina Solar de Geração Distribuída de SP-Unoeste. Available online: <http://www.unoeste.br/noticias/2019/3/unoeste-tera-maior-usina-solar-de-geracao-distribuida-de-sp> (accessed on 11 October 2021).
76. ABA Newsletter. Available online: <http://www.csus.edu/aba2/newsletters/fall2012/abagreennews.html> (accessed on 11 October 2021).
77. Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; et al. TensorFlow: A System for Large-Scale Machine Learning. In *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*; USENIX Association: Savannah, GA, USA, 2016; pp. 265–283.
78. Flask. 2021. Available online: <https://flask.palletsprojects.com/en/2.0.x/> (accessed on 31 October 2021).
79. Mavlink/Mavlink. 2021. Available online: <https://github.com/mavlink/mavlink> (accessed on 24 October 2021).
80. Li, J.; Zhou, Y.; Lamont, L. Communication architectures and protocols for networking unmanned aerial vehicles. In Proceedings of the 2013 IEEE Globecom Workshops (GC Wkshps), Atlanta, GA, USA, 9–13 December 2013; pp. 1415–1420. [\[CrossRef\]](#)
81. Sarker, I.H. Machine Learning: Algorithms, Real-World Applications and Research Directions. *SN Comput. Sci.* **2021**, *2*, 160. [\[CrossRef\]](#) [\[PubMed\]](#)
82. Bradski, G. The openCV library. *Dobb's J. Softw. Tools Prof. Program.* **2000**, *25*, 120–123.
83. OpenCV: OpenCV-Python Tutorials. Available online: https://docs.opencv.org/master/d6/d00/tutorial_py_root.html (accessed on 18 October 2021).
84. Choset, H.; Pignon, P. coverage path planning: The boustrophedon cellular decomposition. In *Field and Service Robotics*; Springer: London, UK, 1998; pp. 203–209.
85. Gabriely, Y.; Rimón, E. Spanning-tree based coverage of continuous areas by a mobile robot. *Ann. Math. Artif. Intell.* **2001**, *31*, 77–98. [\[CrossRef\]](#)
86. Zelinsky, A.; Jarvis, R.A.; Byrne, J.; Yuta, S. Planning paths of complete coverage of an unstructured environment by a mobile robot. In Proceedings of the 1993 IEEE/Tsukuba International Workshop on Advanced Robotics, Tsukuba, Japan, 8–9 November 1993; Volume 13, pp. 533–538.
87. Ceballos, N.M.; Valencia, J.; Giraldo, A.A. Simulation and assessment educational framework for mobile robot algorithms. *J. Autom. Mob. Robot. Intell. Syst.* **2014**, *8*, 53–59. [\[CrossRef\]](#)
88. Jaeyoung, L. PX4 Gazebo Plugin Suite for MAVLink SITL and HITL. 2021. Available online: https://github.com/PX4/PX4-SITL_gazebo/blob/ffb87ef4a312564cf91791bd5a9d683aacd085a6/models/iris/iris.sdf.jinja (accessed on 7 December 2021).
89. Torres, M.; Pelta, D.A.; Verdegay, J.L.; Torres, J.C. coverage path planning with unmanned aerial vehicles for 3D terrain reconstruction. *Expert Syst. Appl.* **2016**, *55*, 441–451. [\[CrossRef\]](#)
90. Araújo, J.; Sujit, P.; Sousa, J. Multiple UAV area decomposition and coverage. In Proceedings of the 2013 IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA), Singapore, 16–19 April 2013; pp. 30–37. [\[CrossRef\]](#)
91. PX4 Drone Autopilot. 2021. Available online: <https://github.com/PX4/PX4-Autopilot> (accessed on 23 September 2021).
92. Koenig, N.; Howard, A. Design and use paradigms for Gazebo, an open-source multi-robot simulator. In Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566), Sendai, Japan, 28 September–2 October 2004; Volume 3, pp. 2149–2154. [\[CrossRef\]](#)
93. Quigley, M.; Conley, K.; Gerkey, B.; Faust, J.; Foote, T.; Leibs, J.; Wheeler, R.; Ng, A.Y. ROS: An open-source Robot Operating System. In Proceedings of the ICRA Workshop on Open Source Software, Kobe, Japan, 12–17 May 2009; p. 5.
94. Silano, G.; Iannelli, L. CrazyS: A Software-in-the-Loop Simulation Platform for the Crazyflie 2.0 Nano-Quadcopter. In *Robot Operating System (ROS): The Complete Reference (Volume 4)*; Koubaa, A., Ed.; Springer International Publishing: Cham, Switzerland, 2020; pp. 81–115. [\[CrossRef\]](#)
95. Salameh, F.E.; Karabiyik, U.; Rogers, M.K. RPAS Forensic Validation Analysis Towards a Technical Investigation Process: A Case Study of Yuneec Typhoon H. *Sensors* **2019**, *19*, 3246. [\[CrossRef\]](#)
96. QGroundControl Ground Control Station. 2021. Available online: <https://github.com/mavlink/qgroundcontrol> (accessed on 31 October 2021).
97. Yuneec Typhoon H-Yuneec Futurhobby. Available online: <https://yuneec-futurhobby.com/yuneec-typhoon-h-pro-realsense> (accessed on 28 October 2021).

-
98. 3DR Iris-The Ready to fly UAV Quadcopter. Available online: <http://www.arducopter.co.uk/iris-quadcopter-uav.html> (accessed on 28 October 2021).
 99. Tan, S.H.; Md Ali, J. Quartic and quintic polynomial interpolation. In Proceedings of the 20th National Symposium on Mathematical Sciences: Research in Mathematical Sciences: A Catalyst for Creativity and Innovation, Putrajaya, Malaysia, 18–20 December 2012; Volume 1522, pp. 664–675.
 100. Pérez-González, A. Andresperez86/CPP_GUI. 2021. Available online: https://github.com/andresperez86/CPP_GUI (accessed on 29 October 2021).
 101. Jordan, S.; Moore, J.; Hovet, S.; Box, J.; Perry, J.; Kirsche, K.; Lewis, D.; Tse, Z.T.H. State-of-the-art technologies for UAV inspections. *IET Radar Sonar Navig.* **2018**, *12*, 151–164. [[CrossRef](#)]
 102. Ghaddar, A.; Merei, A.; Natalizio, E. PPS: Energy-Aware Grid-Based coverage path planning for UAVs Using Area Partitioning in the Presence of NFZs. *Sensors* **2020**, *20*, 3742. [[CrossRef](#)] [[PubMed](#)]
 103. Öst, G. Search Path Generation with UAV Applications Using Approximate Convex Decomposition. 2012; p. 52. Available online: <http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-77353> (accessed on 29 October 2021).