future internet

MDPI

Article

Flying Watchdog-Based Guard Patrol with Check Point Data Verification

Endrowednes Kuantama [1,*], Avishkar Seth [2], Alice James [2] and Yihao Zhang [1]

[1] School of Computing, Faculty of Science and Engineering, Macquarie University, Sydney, NSW 2109, Australia; yihao.zhang3@students.mq.edu.au
[2] School of Engineering, Faculty of Science and Engineering, Macquarie University, Sydney, NSW 2109, Australia; avishkar.seth@mq.edu.au (A.S.); alice.james@mq.edu.au (A.J.)
* Correspondence: endrowednes.kuantama@mq.edu.au

Abstract: The effectiveness of human security-based guard patrol systems often faces challenges related to the consistency of perimeter checks regarding timing and patterns. Some solutions use autonomous drones for monitoring assistance but primarily optimize their camera-based object detection capabilities for favorable lighting conditions. This research introduces an innovative approach to address these limitations—a flying watchdog designed to augment patrol operations with predetermined flight patterns, enabling checkpoint identification and position verification through vision-based methods. The system has a laser-based data transmitter to relay real-time location and timing information to a receiver. The proposed system consists of drone and ground checkpoints with distinctive shapes and colored lights, further enhanced by solar panels serving as laser data receivers. The result demonstrates the drone's ability to detect four white dot LEDs with square configurations at distances ranging from 18 to 20 m, even under deficient light conditions based on the OpenCV detection algorithm. Notably, the study underscores the significance of achieving an even distribution of light shapes to mitigate light scattering effects on readings while also confirming that ambient light levels up to a maximum of 390 Lux have no adverse impact on the performance of the sensing device.

Keywords: drone; patrol; OpenCV; laser; monitoring; security; data communication; computer vision; IoT; watchdog

Citation: Kuantama, E.; Seth, A.; James, A.; Zhang, Y. Flying Watchdog-Based Guard Patrol with Check Point Data Verification. *Future Internet* **2023**, *15*, 340. https://doi.org/10.3390/fi15100340

Academic Editor: Andrey V. Savkin

Received: 18 September 2023
Revised: 12 October 2023
Accepted: 13 October 2023
Published: 16 October 2023

1. Introduction

In recent years, the flying Internet of Things, commonly called drones, has rapidly evolved, permeating various domains such as surveillance, defense, logistics, cartography, and search and rescue [1–5]. This research focuses on the specific application of drones in security monitoring. Within this context, drones play a pivotal role by either augmenting or substituting human involvement in routine patrols and perimeter inspections to preempt unauthorized access to restricted areas. Drones employed for security monitoring are equipped with sophisticated cameras and sensor arrays, enabling them to detect anomalous activities and issue timely alerts [6–9]. The primary advantage of aerial surveillance platforms is their ability to offer a comprehensive and adaptable field of view. Consequently, this technology is relevant in defense operations, spanning military and civilian installations, including warehouses, factories, farms, and mining facilities [10–14]. The drones designed for this purpose exhibit autonomous flight capabilities, executing predefined flight paths and even incorporating artificial intelligence algorithms for adaptive navigation. However, it is crucial to note that the term "fully autonomous" is somewhat misleading, as human intervention remains indispensable in several critical aspects of drone operation. These include the initial power-up procedures, pre-flight system assessments, hardware diagnostics, waypoint establishment, and task allocation. Drones are valuable for collaborating with human operators to ensure adequate security monitoring [15–19]. Operators are

responsible for overseeing drone flights, maintaining consistency in executing scheduled perimeter checks, and, in essence, utilizing these airborne platforms as an innovative form of flying closed-circuit television (CCTV). This symbiotic relationship between drones and human operators significantly enhances security measures in various sectors.

This research addresses the limitations inherent in drone-based and human patrol surveillance methods. Drones with camera-based detection systems often struggle in adverse environmental lighting conditions, restricting their object detection capabilities to favorable lighting scenarios [20,21]. Conversely, human patrols encounter challenges related to task consistency, particularly during perimeter checks. This study proposes implementing an automated system that coordinates the efforts of drones and human patrols through remote monitoring and perimeter sign-in methods. Currently, the prevalent technologies for drone check-ins rely on markers or QR code detectors, primarily applied in drone delivery contexts and limited by specific lighting conditions. Meanwhile, human patrol check-ins typically involve Radio Frequency Identification (RFID) stationary systems [22,23]. This research explores innovative solutions to enhance surveillance capabilities, mitigate limitations, and improve security measures in various operational scenarios.

With all existing technology and issues, we propose the creation of a cutting-edge Flying Watchdog System capable of location recognition using distinct glowing markers. As part of this system, drones employ laser data transmission to a Portable Glowing Pad (PGP) acting as the receiver. The laser's low-angle beam or single-point transmission ensures precise drone positioning directly above the designated check-in location. This development of a Flying Watchdog Guard Patrol system is structured around three primary focal points:

- Light pattern detection with vision-based technology;
- Laser data communication through the air;
- Drone check-in pad with remote monitoring capability.

A critical aspect of this research involves the utilization of LEDs with varying shapes and colors as markers. The OpenCV algorithm enables color and shape detection in low-light conditions, ensuring robust performance in the dark. Additionally, a D350 depth-sensing camera integrated with the drone facilitates the identification of three distinct patterns generated by LEDs of different colors. The experimental results will demonstrate the system's detection range, the lumens of LEDs detectable under diverse low-light environmental conditions, detection algorithms, and the optimization of detector response time. The drone has a laser transmitter featuring an 8-bit unique code exclusively for PGP authentication. The PGP, the check-in station, incorporates a self-powering mechanism through solar panels. It functions to activate the LED pattern and serve as the laser receiver while offering real-time remote monitoring capabilities to manage LED activation times, check-in times, and log-book data. Real-time monitoring also enables the possibility of triggering alarms in the event of patrol negligence. The number of PGPs can be tailored to meet the specific requirements of the surveillance operation, accommodating a variable quantity of checkpoints and perimeter locations for drone inspection. Verification entails conducting drone flights following predefined flight patterns with multiple checkpoints to ensure the system's reliability. The flight and check-in process can be conveniently monitored through remote access, ensuring the system's robustness and effectiveness in enhancing surveillance operations.

This system introduces an innovative approach featuring a specially designed luminous marker capable of being detected by a vision sensor, thereby supplanting the need for traditional QR codes or tags. Tags typically serve as reference points in drone applications, facilitating specific tasks like drone landings. However, these conventional tags face limitations in low-light conditions. The primary challenge is the development of a detection algorithm capable of operating optimally in favorable conditions, utilizing a compact LED emitting a small dot of light and following a particular LED pattern. Our study aims to assess the system's performance across varying environmental lighting conditions to gauge its reliability. We will also investigate how LED color affects detection distance when com-

bined with the detection algorithm and the efficacy of the filtering process in distinguishing genuine targets from false ones. This innovative approach draws inspiration from a detection model initially designed for identifying traffic lights in autonomous driving scenarios. Additionally, we explore the groundbreaking concept of information transmission via laser technology and its integration into drone systems. This research encompasses various aspects, including the alignment process between the laser transmitter and receiver, data transmission time, and the impact of ambient light on laser-based data transmission. This multifaceted investigation promises to yield valuable insights into the practicality and effectiveness of these novel techniques.

The remaining sections of this study are structured as follows. In Section 2, we explore prior research, providing valuable context to better appreciate our contributions' uniqueness. Moving on to Section 3, we present an in-depth overview of the system's design and architecture, elucidating the goals we have set for our proposed method. Section 4 comprehensively presents the test results and a thorough drone and PGP system analysis. Finally, in Section 5, we draw our conclusions and discuss avenues for future work.

2. Related Work

In the present study, the prevailing paradigm in drone-based security patrol systems revolves around path planning for individual drones and swarms, deploying detection methodologies for intruder discrimination, and utilizing sensor technologies optimized for low-light environments. Conventional checkpoint data validation, particularly concerning patrol check-in processes, predominantly relies on RFID technology. This research introduces an innovative approach employing a laser as a singular, concentrated beam for high-speed data transmission. A computer vision-based sensing apparatus is also implemented to recognize traffic signal colors.

The patrolling method can identify suspicious behavior by utilizing single or multiple object optimization techniques to anticipate human movement [24,25]. This approach leverages heuristic search strategies to explore the impact of various intruder behaviors on search performance [26,27]. Critical parameters in the pattern of search within this method encompass the size of the search area, the number of patrolling drones, camera positioning, and the behavior of potential intruders. In the context of night-time security and intruder detection, an infrared camera is employed with drones [28]. Furthermore, the study delves into the analysis of human behavior, specifically focusing on gait and even hand movements, by compiling a dataset encompassing diverse walking styles [29]. The primary research objectives revolve around night-time object detection, human behavior analysis, and intruder warning systems. Another facet of this field of study involves path planning utilizing multiple Unmanned Aerial Vehicles (UAVs) for intricate patrol missions [30]. The findings demonstrate that a strategy involving multi-layer nesting and random walk patterns outperforms the particle swarm optimization algorithm in the context of drone patrolling. The utilization of drones in security operations remains primarily centered on their capacity for search and detection, as evidenced by prior research. While incapable of entirely autonomous flight, drones are supportive tools that enhance security measures. This study, however, shifts its focus towards a novel perspective, emphasizing the role of patrol drones in actively safeguarding and fortifying security efforts.

Regarding light detection, the prevailing emphasis often centers on color detection monitoring, which finds applications in recognizing traffic lights for autonomous driving purposes and monitoring RGB images to assess light pollution levels [31–33]. The results of these investigations reveal that both OpenCV and the You Only Look Once (YOLO) algorithm are instrumental in detecting traffic lights and recognizing their colors through trained models [34]. Deep learning techniques, specifically those based on integral channel features, can identify shape, color, and texture attributes associated with traffic lights. Additionally, YOLOv4 and YOLOv5 demonstrate exceptional success rates exceeding 85% in detecting and recognizing traffic lights [35,36]. Besides computer vision, another approach to night-time illumination detection involves assessing night ground brightness

with optical devices for light mapping [37]. The existing research shows different techniques for detection with different algorithms depending on the application. The findings from prior studies underscore the significance of refining the filtering process to mitigate light dispersion effects. Many of these investigations are still ongoing, with a primary focus on enhancing process accuracy and minimizing the occurrence of false positives in target identification.

3. Proposed Flying Watchdog Architecture

The drone system is comprised of two distinct components: an aerial surveillance module designed to augment security patrols and a ground-based checkpoint platform serving as a check-in station for security personnel. The ground checkpoint, or PGP, has Wi-Fi connectivity to transmit pertinent information about patrol schedules, logging data, and alerts to web or mobile interfaces.

This Internet of Things (IoT)-based framework offers functionalities such as scheduling security patrols, recording drone-generated maps with GPS coordinates, and triggering alarms during security breaches. It is worth noting that while IoT applications of this nature have found widespread adoption, the primary focus of this study does not center on this application domain. The proposed architectural configuration is illustrated in Figure 1. The drone can fly with a predetermined flight pattern between checkpoints. Upon reaching each checkpoint, the drone engages in a search mode to identify PGP by executing a circular flight path with a 1 m radius, and the camera is directed downward at a 45° angle. Subsequently, upon PGP detection, the drone activates a laser for 20 s to transmit a security code, effectively halting the checkpoint timer. The PGP is intricately designed to establish a network connection, facilitating the transmission of check-in time data to the IoT Hub platform. If a check-in is not performed within the specified time frame, an alarm at the monitoring station is triggered. Importantly, this research underscores the adaptability of the approach, allowing for flexibility in the number of PGPs employed for checkpoints and customization of consequences in case of untimely check-ins by the drone.

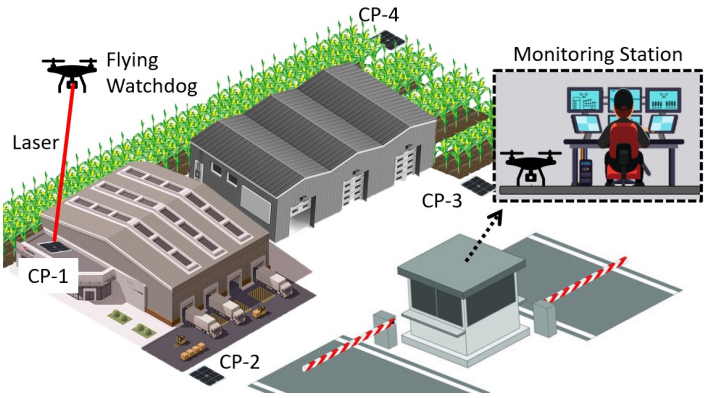

Figure 1. The illustration of the proposed method.

3.1. Flying Watchdog Model

The initial phase of this research is to develop a drone with a 680 mm wheelbase frame and PGP design. This study employs a meticulously designed drone measuring 48 cm in both length and width, featuring a robust 400 kV rotor, 13-inch propeller, and a high-capacity 6500 mAh 80C 14.8-volt Lithium Polymer (LiPo) battery, affording it an impressive flight duration of 45 min. The mechanical structure of the drone is seamlessly integrated with a flight controller boasting an array of sensors, including an accelerometer, gyroscope, digital compass, GPS, and lidar. This amalgamation equips the drone with the capability to operate semi-autonomously. Crucially, the drone establishes a vital connection with a monitoring station via a 915 MHz, 100 mW radio telemetry system, offering a transmission

range of up to 300 m. A comprehensive illustration of the drone's configuration is presented in Figure 2, providing valuable insights into its intricate design and componentry.

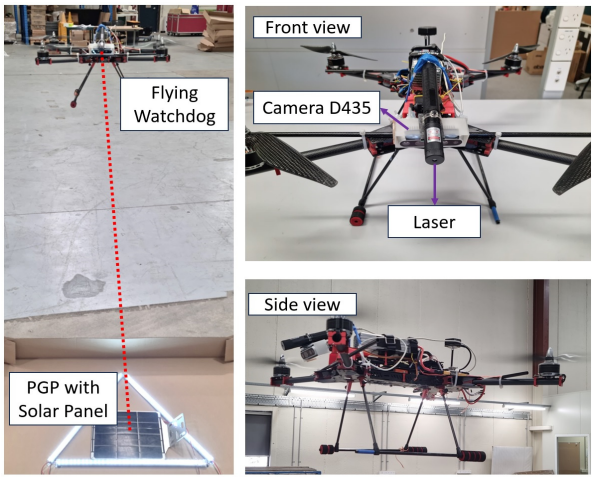

Figure 2. The architecture of flying watchdog.

The architectural framework of the aerial watchdog system is meticulously organized into three distinct modes: the flying mode, the searching mode, and the communication mode. The flying mode serves as the platform for executing predefined flight commands and overseeing the autonomous maneuvering of the drone en route to its destination. In the searching mode, a high-level processing computer, the 4 GB 64-bit NVIDIA Jetson Nano, is engaged. It is seamlessly connected to a flight controller via the universal asynchronous receiver transmitter (UART) communication protocol, operating at speeds of up to 921,600 bps (bits per second). In this configuration, hardware and telemetry compatibility are optimized at 115,200 bps.

Furthermore, the searching mode is integrated with a depth camera D435, serving as a sensory device to detect PGP based on color and shape criteria. Given the continuous monitoring requirements of both the PGP and flying mode, a third microcontroller is introduced to facilitate communication between the drone and the PGP. This microcontroller is also equipped with a laser for data transmission purposes. The communication protocol employed in this system utilizes half-duplex serial communication, whereby signals are transmitted from the NVIDIA Jetson to the Arduino microcontroller, initiating data transmission via the laser for a 20 s duration. A comprehensive visualization of the configuration of these three operational modes is thoughtfully depicted in Figure 3, providing valuable insights into the system's intricate design and functional interplay. Communication between the flight controller and NVIDIA Jetson is carried out using MAVROS, which bridges the MAVLink communication protocol from the flight controller with robot operating systems on NVIDIA Jetson. The drone configuration, monitoring, and predetermined flight pattern are carried out using the open-source Mission Planner platform.

3.2. PGP Detection Model

This study presents an innovative real-time pipeline that detects and recognizes PGP characterized by distinct colors and patterns. As detailed in Section 2, contemporary light detection technology predominantly finds application in traffic light detection. These systems are typically tailored for fixed-shaped objects, such as traffic lights, and primarily rely on three distinct detection colors coupled with specific algorithms for recognition. In our research, we adopted an edge light detection approach and a greyscale color translation algorithm to expedite the capturing and filtering of the emitted LED light. The proposed system leverages an Intel D435 camera, capable of capturing high-resolution 1080p videos, which is seamlessly interfaced with an NVIDIA Jetson, serving as the primary image processing unit. The system rigorously analyzes various lighting conditions to ensure

robust performance. In addition to its sophisticated image processing capabilities, the study also delves into the design and investigation of PGP variants, comprising three unique shapes and dot patterns, each distinguished by a diverse color palette, as exemplified in Figure 4. Specifically, white is adopted for the PGP shape, while the dot patterns exhibit a harmonious blend of blue, red, purple, yellow, and white hues.

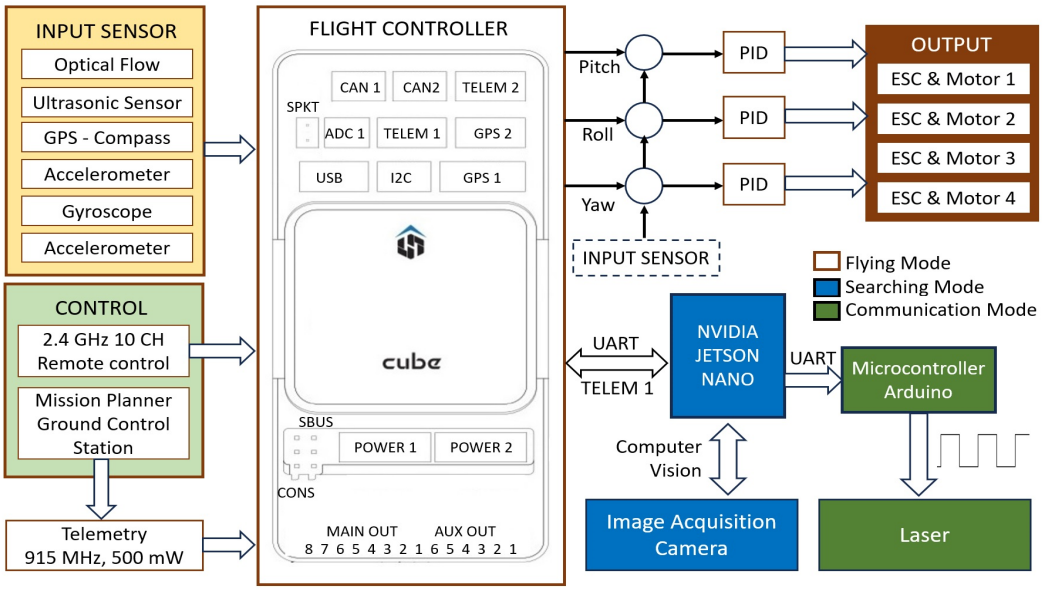

Figure 3. Flying watchdog schematic design.

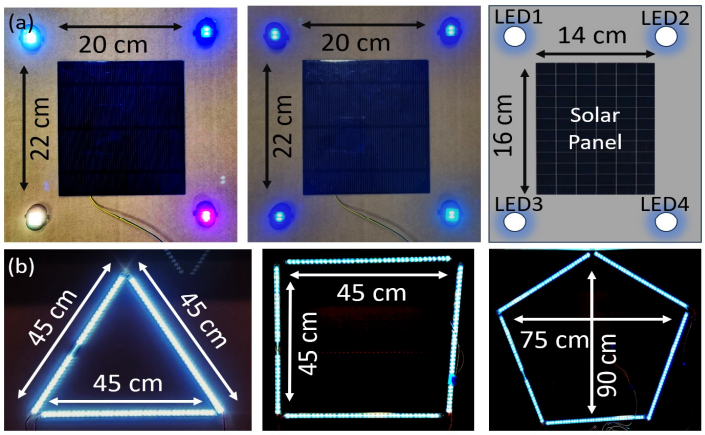

Figure 4. PGP based on: (**a**) dot square shape; (**b**) pattern (triangle, square, and pentagon).

The PGP detection model is constructed with a multi-stage approach encompassing essential preprocessing steps, color space transformation, segmentation via thresholding and contouring, and morphological operations. Within the preprocessing stage, the system identifies regions of interest, performs image denoising, and rectifies images, particularly in low-light settings. The threshold value is meticulously computed through the thresholding and contouring process to effectively distinguish between daytime and night-time scenarios, ensuring accurate image segmentation. Subsequently, the image is subjected to critical procedures, including feature extraction, classification, and verification. The verification stage represents the final step, validating the color and shape codes against predefined targets before transmitting data via a laser medium. The proposed PGP detection scheme, elucidating the intricate workings of this model, is thoughtfully depicted in Figure 5, providing valuable insights into the system's operational flow and critical components.

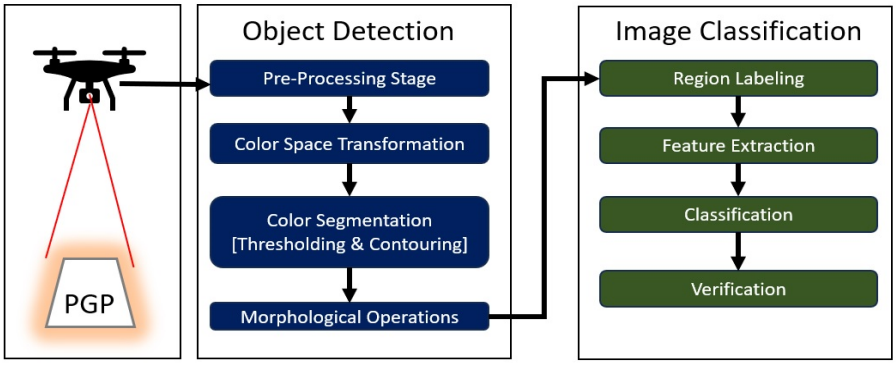

Figure 5. The PGP detection scheme.

The detection process involves thresholding and contouring, where the initial step focuses on segmenting the image into regions or objects of interest based on pixel intensity values. The image is first converted to grayscale to facilitate intensity-based analysis. Each pixel's intensity is quantified on a scale from 0 (representing black) to 255 (indicating white). Subsequently, a threshold value (T) is meticulously selected, considering the unique characteristics of the image and the desired segregation of objects from the background. This thresholding operation transforms the grayscale image into a binary image, with each pixel assuming a value of either 0 (black) if its intensity (Int) falls below the threshold (Int < T) or 255 (white) if its intensity surpasses or equals the threshold (Int $\geq$ T). Equation (1) represents the mathematical terms for a binary pixel (b).

$$f(b) = \begin{cases} 0, \text{ if Int} < T \\ 255, \text{ if Int} \geq T \end{cases} \quad (1)$$

In the second phase of the contouring process, we aim to identify and outline the boundaries of objects or regions within the binary image precisely. The identification process involves a series of sequential operations, commencing with the crucial edge detection step. Edge detection accentuates parts of rapid intensity change within the image, which directly correspond to the boundaries of objects. Among the array of edge detection algorithms available, the widely acclaimed Canny edge detector is frequently employed for its exceptional performance. This step is pivotal, as it is crucial in detecting encryption patterns characterized by various shapes. In this context, we use the Canny edge detector to detect edges within our images effectively. The procedure comprises several key stages, including Gaussian smoothing, gradient computation, non-maximum suppression, and edge tracking through hysteresis. The process commences with Gaussian smoothing, a critical step to reduce noise within the image. This operation involves applying a 2D Gaussian filter to the original photo, with the parameters of the Gaussian kernel determined by a formula that incorporates the standard deviation (σ), as seen in Equation (2). The outcome is a smoothed image rendition, setting the stage for subsequent and precise edge detection.

$$G(x, y) = \frac{1}{(2\pi\sigma^2)} \exp\left(-\frac{(x^2 + y^2)}{(2\sigma^2)}\right) \quad (2)$$

The Gaussian kernel object position in a two-dimensional axis (x, y) is represented with $G(x, y)$. The standard deviation of the Gaussian distribution controls the amount of smoothing. After applying Gaussian smoothing, we focus on calculating image gradients to pinpoint regions displaying significant pixel intensity changes, which serve as key indicators of potential edges. This gradient computation is conducted in a horizontal (I_x) and vertical (I_y) directions through specialized filters represent this state, as seen in Equation (3).

$$\begin{bmatrix} I_x \\ I_y \end{bmatrix} = G \begin{bmatrix} \text{img} \begin{bmatrix} -1 & 0 & 1 \end{bmatrix} \\ \text{img} \begin{bmatrix} -1 & 0 & 1 \end{bmatrix} \end{bmatrix} \quad (3)$$

In our analysis, the image obtained after Gaussian smoothing is denoted as 'G,' while the original image is symbolized as 'img'. We employ a concise 1×3 matrix to establish image boundaries, representing the [left, center, right] positions. The central value (0 in the matrix) is multiplied by the corresponding pixel in the image. The left value (-1 in the matrix) is multiplied by the pixel to the left. The correct value (1 in the matrix) is multiplied by the pixel to the right. We compute the gradient magnitude to gauge the intensity of edges at every pixel. This magnitude (mag) is derived as the square root of the sum of squared gradients in both the horizontal (x) and vertical (y) directions, which can be calculated using Equation (4). The gradient direction is computed using the tangent function to ascertain the orientation of edges, providing insight into edge orientations at each pixel, which can be seen in Equation (5). The symbol of 'dir' represented the gradient direction in radians.

$$\text{mag} = \sqrt{{I_x}^2 + {I_y}^2} \tag{4}$$

$$\text{dir} = \tan^{-1}\left(I_x, I_y\right) \tag{5}$$

In the pursuit of refining detected edges, we implement non-maximum suppression as a pivotal step. This technique selectively preserves local gradient maxima while discarding less pronounced gradients, effectively thinning the edges and maintaining detection accuracy. Subsequently, we employ edge tracking via hysteresis to connect neighboring strong edge pixels, thereby establishing continuous edges. This process hinges on two threshold values: a high threshold (T_{high}) and a low threshold (T_{low}). Pixels with gradient magnitudes exceeding T_{high} are classified as solid edges, while those falling between T_{low} and T_{high} are deemed weak edges. Weak edges are considered part of an edge if interconnected with solid edges, thus forming a coherent edge map. The edge detector is comprised of a multi-stage process that generates a binary edge map, where white pixels denote detected edges, as evidenced in encrypted LED pattern detection results. This comprehensive approach, encompassing Gaussian smoothing, gradient calculation, non-maximum suppression, and hysteresis thresholding, emerges as a robust and productive technique for precise edge detection within digital images.

Following the successful edge detection phase, our contouring algorithm takes the helm, systematically identifying connected components of edge pixels. This process assembles neighboring pixels that collectively delineate a common object boundary, giving rise to distinct groups. These groups, or contours, are subsequently represented as sequences of (x, y) coordinates, effectively outlining the boundaries of objects inherent in the binary image. Contours are invaluable assets in our endeavor to discern the LED pattern. They encompass a wealth of information, including metrics such as area and centroid, which can be calculated mathematically via contour analysis. This holistic approach, underpinned by the fusion of thresholding and contouring techniques, constitutes the bedrock for the efficacious detection and analysis of encrypted LED patterns within images, employing the versatile OpenCV framework.

3.3. Data Communication via Air

The challenging aspects of laser-based data transmission lie in the alignment procedure and the influence of ambient light. In our research, we devised an algorithm to facilitate automatic calibration, enabling a comparison between ambient and laser light. This comparison is particularly crucial as we employ solar panels as laser receivers. We implemented the Hamming encoder method for efficient data transmission, which processes 8-bit data. Data communication is designed using a laser direct link between the transmitter and receiver. The system employs a 650 nm red laser module with a 5 mW power rating and a 5–10 kHz frequency range. Controlled by an Arduino microcontroller, this communication setup adheres to safety regulations and employs an innovative communication protocol. The laser receiver, powered by a 5.5-volt, 3-watt solar panel, interfaces with the Arduino microcontroller within the PGP system, utilizing a 10-bit resolution Analog Digital

Converter (ADC). The communication protocol is based on a Simplex method using the Universal Asynchronous Receiver Transmitter (UART) serial communications standard, running at 9600 bits per second with a 64-byte First-In-First-Out (FIFO) serial buffer. This article explores the protocol's functionality, analogous to Morse code, where '0' and '1' code data represent 8 bits of information, with start and end bits serving as transmission flags.

The drone's laser system will initiate data transmission only after it successfully detects and verifies the PGP. The details of the data communication block diagram can be seen in Figure 6. The procedure for transmitting data via a laser follows a similar principle to conventional serial data transmission through a cable. The laser on the transmitting end remains active for 8 s, allowing the receiver ample time to gauge the laser input voltage. Subsequently, the analog-to-digital synchronization process, which includes start and stop bits, is executed. The communication protocol transmits data as an 8-bit string or array.

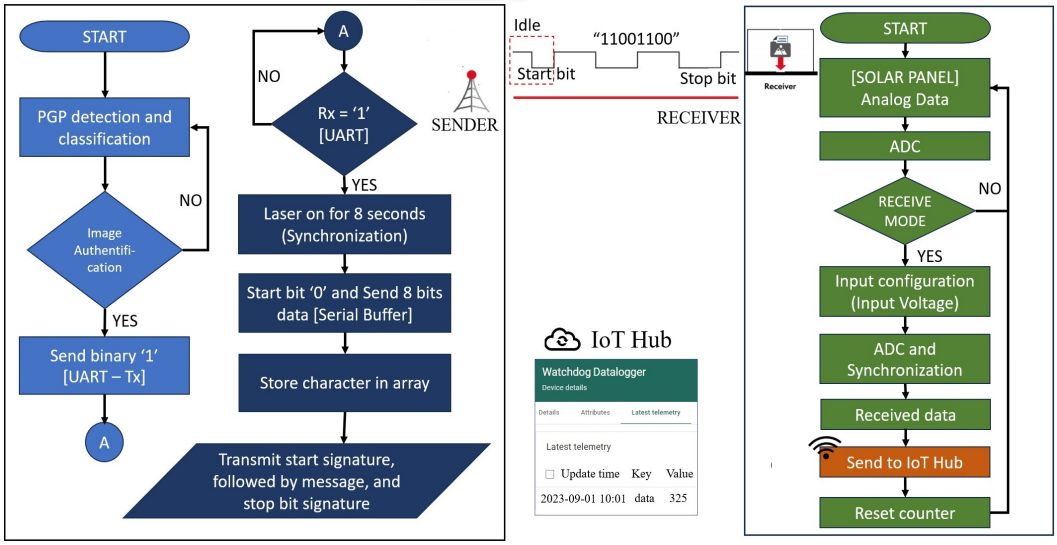

Figure 6. Block diagram of laser data communication.

Conversely, the solar panel, the laser receiver, employs an analog data comparison method. The received data manifest as analog information, with the voltage from the solar panel ascending in correlation with the intensity of the received light. To illustrate, if the solar panel receives light from its surrounding environment and registers a voltage range of 4.1 to 4.2 volts, the laser received by the solar panel will amplify the input voltage by approximately 5%, contingent on factors such as distance and laser power settings. This comparison process is essential for establishing a threshold value and configuring adaptive laser detection through software. This software effectively discerns between laser-generated light and ambient light. When the data received by the PGP system matches the information stored in the database, the IoT-based system integrated with PGP will transmit integer data (e.g., "325" indicating the location code from PGP) to the IoT hub. These shared data include a timestamp, as depicted in Figure 6. The PGP system is implemented using an ESP8266 NodeMCU, which boasts Wi-Fi connectivity and is linked to a Think-board functioning as the IoT hub for this setup. Communication within the PGP system is facilitated by the MQTT (Message Queuing Telemetry Transport) protocol. Moreover, a scheduling system for data loggers can be devised in more sophisticated IoT systems to accommodate multiple PGP systems.

The solar panel serves as the input receiver, and it is linked to the Analog-to-Digital Converter (ADC) within the microcontroller, facilitating the conversion of analog signals into digital ones. The solar panel exhibits sensitivity to incident light, and this investigation aims to elucidate the impact of low-power laser irradiation on the data reception procedure. The proposed system is meticulously devised to calibrate environmental parameters autonomously, including ambient light conditions and laser illumination. The outcome of this

calibration yields a discernible threshold value, and the associated algorithm is elucidated in Equation (6).

$$\overline{V}_{si} = \frac{\sum X_{si}(i)}{N} = \frac{\sum((a.t) + b)}{N(t)} \quad (6)$$

$$ADC_{si} = \frac{1}{V_i}.\left(2^{\alpha}.\overline{V}_{si}\right) \quad (7)$$

Solar irradiance exhibits constant fluctuations, resulting in variations in the solar panel's input voltage (V_{si}). Equation (6) is a tool to compute the average input voltage from the solar panel $(\overline{V}_{si})$, determined by the number of iterations or time-based iterations. Each input voltage measurement during an iteration (X_{si}) across the total iterations (N) contributes to calculating the average solar panel input voltage. Alternatively, one can compute the average value over a specific period, using N(t) to denote the number of iterations at a given time (t), where 't' is a continuous variable representing time. This time range typically spans from $t = t_1$ to $t = t_2$. In this context, 'a' symbolizes the rate of change, while 'b' represents the initial number of iterations at t = 0. The resultant average input voltage exists as an analog voltage value, which undergoes translation through an analog-to-digital converter. Equation (7) is employed to calculate this average input voltage. The onboard ADC microcontroller Arduino offers a 10-bit ADC value (α), supplied with a 5-volt input voltage (V_i). It is important to note that fluctuations in the input voltage originating from the solar panel will impact the ADC value, introducing variability in the readings.

$$\overline{V}_{(si+laser)} = \frac{\sum X_{(si+laser)}(i)}{N} = \frac{\sum((a.t) + b)}{N(t)} \quad (8)$$

$$ADC_{(si+laser)} = \frac{1}{V_i}.\left(2^{\alpha}.\overline{V}_{(si+laser)}\right) \quad (9)$$

The laser beam received by the solar panel induces fluctuations in irradiance levels. To ensure precise readings, it is essential to calculate the average value based on the number of iterations or within a specified time frame, as outlined in Equation (8). Furthermore, the average change in the solar panel's input value impacts the ADC (Analog-to-Digital Converter) reading, which can be computed using Equation (9). The data communication protocol's validation phase entails retrieving data through a serial monitor interface. Moreover, the examination also considers the potential influence of ambient lighting and the spatial separation between components on the data retrieval process.

4. Results and Discussion

In this research, our experimentation involved configuring the drone for automated flight, following a predetermined flight path with designated waypoints. Four specific checkpoints, namely CP-1, CP-2, CP-3, and CP-4, were integrated into the flight plan, as illustrated in Figure 1. To facilitate this, we harnessed an open-source platform, such as Mission Planner, to predefine the flight path and establish checkpoint locations for each Point of Interest (PGP). During the actual flight test, we activated the search mode when the drone reached the final checkpoint, CP-4. At this juncture, the drone autonomously navigated toward CP-4 and initiated a search mission to pinpoint the PGP's exact location. Leveraging computer vision technology based on distinct glowing patterns, the drone identified the PGP's position and hovered in place for 10 s to facilitate data transmission. The outcomes of our testing phase are graphically represented in Figure 7, which illustrates the drone's circular flight path around the CP-4 location, maintaining a radius of 1 m. The search maneuver was executed by controlling the drone's rotational angles, including pitch, roll, and yaw, while it remained in search mode for 30 s.

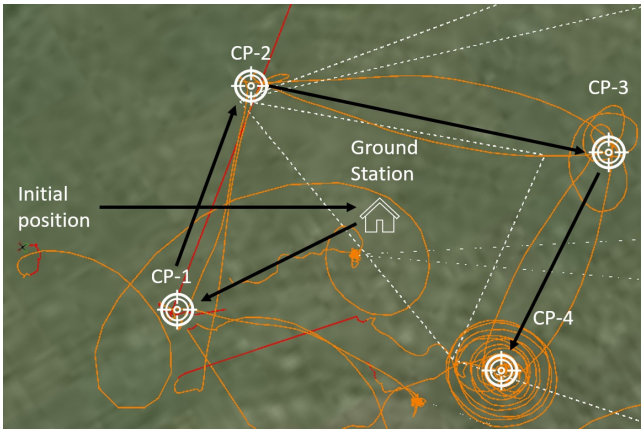

Figure 7. Flying watchdog check-in scenario and results.

The overview of the code snippets depicted in Figure 8 are crucial elements within the Drone Control system. These code snippets are scripted in Python and play a vital role within the context of the ROS package installed in the 'catkin_ws.' The primary focus here is the interaction between the NVIDIA Jetson and the flight controller, which involves establishing MAVLINK communication via MAVROS and configuring communication parameters. Furthermore, we delve into the utilization of MAVROS for tasks such as reading APM firmware configurations, managing the altitude of the delivery drone, and importing essential messages from 'mavros_msgs.' These messages facilitate reading APM firmware configurations, a task enabled through a 'yaml' file that provides crucial state information regarding the application, particularly on the watchdog drone. The initial code snippet (Figure 8, Code Syntax: Drone Control, Line 1) configures the MAVROS port for communication with the Flight Controller. It specifies the serial port as '/dev/ttyTHS1' and sets the baud rate, establishing a connection with Telemetry 2 or flight controller UART port.

Code Syntax: Drone Control

```
1 <!-- MAVROS Port -->
      <arg name="fcu_url" default="/dev/ttyTHS1: 921600"/>

2 # LED Detection python script
  Color Detection:
  frame = cap.read(), frame = cv2.resize(frame,(horizontal_res,vertical_res)) #camera FoV parameters
  hsv = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)
    lower_red = np.array([160, 100, 100])  || upper_red = np.array([180, 255, 255])
  # Create a mask to isolate red regions
    mask1 = cv2.inRange(hsv, lower_red, upper_red)   # Define the lower and upper bounds for the red color (in HSV)
  # Create another mask to handle the red hue wrapping around
    mask = mask1 + mask2   # Combine both masks to get the final mask for red
  contours, _ = cv2.findContours(mask, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)

  Shape Identification:
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY) # Convert the frame to grayscale
   blurred = cv2.GaussianBlur(gray, (5, 5), 0)  # Apply Gaussian blur to reduce noise and improve edge detection
    edges = cv2.Canny(blurred, 50, 150) # Perform edge detection using Canny
  if len(approx) == 3:
   cv2.drawContours(frame, [approx], 0, (0, 0, 255), 2)  # Draw detected triangles in red
  shape = "Triangle"  (sample for triangle detection)

4 from mavros_msgs.msg import AttitudeTarget, Set_Ned_Velocity, StatusMessage
```

Figure 8. Drone control and image detection source code.

The code snippet featured in Figure 8, under the sections "LED Detection Python script" and "Shape Identification," is responsible for processing camera images to detect color and shape. This code snippet commences by capturing a frame from a camera source using the 'cap.read()' function. The frame is then resized according to predefined horizontal and vertical resolution parameters, ensuring uniformity in image processing. For color-based object detection, the frame converts from the default BGR color space to HSV (Hue,

Saturation, Value) using 'cv2.cvtColor.' HSV is the preferred color space for segmentation because it isolates color information (Hue) from brightness (Value) and saturation. Color segmentation is achieved by establishing lower and upper HSV bounds for the target color, which, in this case, is red. These bounds are defined as 'lower_red' and 'upper_red.' Pixels falling within these boundaries represent the red color, for instance. Two masks, 'mask1' and 'mask2,' are generated to isolate primary and secondary red regions based on the specified HSV bounds. These masks are then merged to form a final mask ('mask') that accommodates variations in red hues. This last mask is employed to identify contours within the image via 'cv2.findContours.' Contours are continuous curves that delineate object boundaries within the frame. In this context, the contours signify regions of red color.

The code subsequently proceeds with shape identification, although the specific code for shape recognition is not provided. It seems to involve transforming the frame into grayscale, applying Gaussian blur to reduce noise, and employing the Canny edge detection algorithm. This code exemplifies one facet of shape identification, and similar structures are used for square and pentagon shapes during testing for various luminance levels. While this code is indispensable for tasks such as LED and shape recognition, it holds the potential for broader applications within a vision-based control system. The code depicted in Figure 8 harnesses MAVROS to govern the altitude of the watchdog drone. It employs a ROS publisher-subscriber transform node to manage altitude control. The process entails publishing commands to the MAVROS node responsible for altitude control. The code imports essential MAVROS messages from the 'mavros_msgs' package, which include 'AttitudeTarget,' 'Set_Ned_Velocity,' and 'StatusMessage.' These messages serve as the communication and control conduits between the NVIDIA Jetson and the flight controller, enabling functions such as setting attitude targets, velocity control, and drone status monitoring.

The system verification process comprises two distinct stages: firstly, the detection of luminous patterns utilizing OpenCV, and secondly, the transmission of data through the air. This research examines two critical parameters: the detection distance between the drone and the Point of Interest (PGP) and the data transmission range achieved using a low-power red laser.

4.1. Light Color and Pattern Detection Based on OpenCV

The computer vision processing in this study is segmented into two distinct phases: object detection and classification. Object detection, in turn, is further subdivided into three different scenarios: low light, darkness, and standard lighting conditions. These divisions assess the reliability and performance of the developed detection software under varying environmental conditions, including the impact of ambient light on detection range. We utilized a 0.06 A 5 V LED to create a distinct point pattern, as depicted in Figure 4. Each LED emitted light in a manner that produced a circular scattering pattern. Our research incorporated LEDs in five colors, red, purple, yellow, blue, and white, to investigate potential variations in detection distances. For precision in our measurements and experimental setup, the experiments were conducted horizontally, aligning the camera with the target object. The results of these experiments are presented in Figure 9, shedding light on the outcomes of our investigations.

The primary objective of the airborne watchdog system is to operate effectively during nocturnal hours or in environments with limited ambient light. Experimental trials were conducted under two distinct light intensity conditions, precisely 20 lux (considered meager light) and 35 lux (classified as low light). The findings from these experiments are presented in Figure 10. All four LEDs were subjected to identical power settings; however, it was observed that only the yellow LED remained undetectable, while the white LED exhibited the most extended detection range in low-light conditions, spanning approximately 18 to 20 m.

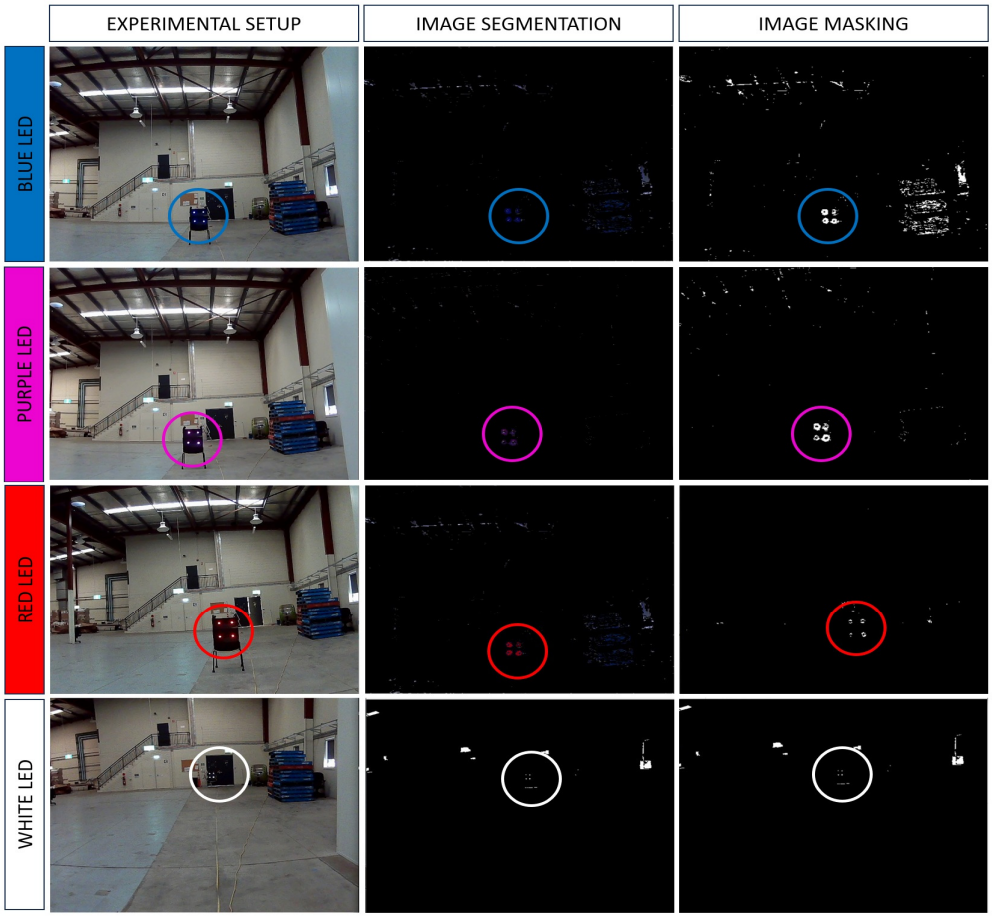

Figure 9. Color detection with a dot square pattern.

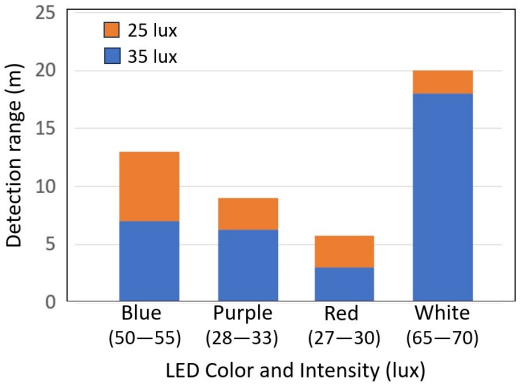

Figure 10. Color detection range for different colors.

Notably, with a 15 lux difference between meager light conditions, only the blue LEDs demonstrated a substantial difference in detection distance, extending up to 5 m. In comparison, the red and purple LEDs exhibited a more modest difference of only 2 m. The results indicate that detection distances increase as ambient lighting conditions become darker. Consequently, white light outperforms other LED colors in terms of detection range, making it the preferred choice for the LED pattern in the upcoming stages of development. Furthermore, these findings underscore the potential of single LEDs, or combinations of single LEDs arranged in different designs, to substitute traditional markers effectively. In conclusion, when the data readings align with the pre-established database, the system transmits a message to the serial monitor. It activates the data transfer process through a laser.

4.2. Light Shape Pattern Detection

Three distinct shapes, a square, a triangle, and a pentagon, were meticulously crafted utilizing 12-volt 1-watt LED strips, each measuring 45 cm long. Detecting the light shape patterns on these objects was executed by identifying the luminous outlines encompassing each shape, employing the methodology elucidated in Section 3.2. Subsequently, we embarked on an object classification procedure to distinguish genuine identifications from erroneous ones, as exemplified in Figure 11. The segmentation process using OpenCV serves the sole purpose of identifying triangles while excluding other shapes. However, it's worth noting that the system still identifies square-shaped rooftops among the results. The next step involves applying a kernel function filtering mechanism and employing light edge detection to address this. This step eliminates false positives, as Figure 10 depicts. Upon completion of the classification process, it becomes evident that only the triangle shape meets the criteria for legitimate identification in this scenario. Verification is subsequently conducted through serial monitors to confirm the accuracy of the title.

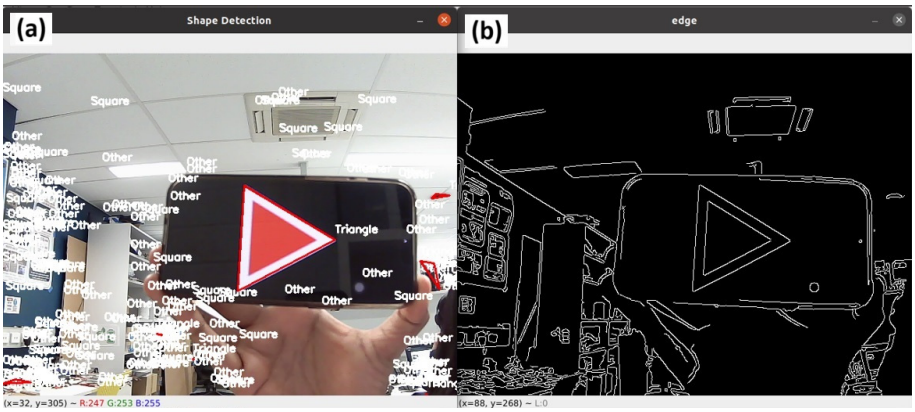

Figure 11. Object detection capabilities: (**a**) shape detection; (**b**) filtering.

The subsequent phase entails the identification of three distinct pattern shapes: triangles, squares, and pentagons. This endeavor aims to determine which condition can be detected over the most extended range. To achieve this, a series of trials involving a horizontal detection approach amid various environmental objects and under diverse lighting conditions were conducted. Figure 12 illustrates the experimental setup, encompassing the image segmentation process and the application of light edge filtering. The outcomes of the detection process are showcased on the serial monitor and subsequently transmitted from NVIDIA Jetson to the Arduino microcontroller, initializing the data transmission process through laser technology.

The detection process underwent examination under three distinct environmental lighting conditions: low light (8.9 lux), moderate light (168 lux), and standard illumination (390 lux). It is worth noting that the LED strip's luminance is 400 to 500 lux. Considering the detection distances, it becomes evident that the square shape surpasses the triangle and pentagon in terms of its range. Remarkably, the pentagon also exhibits a longer dimension than the square. Detailed findings and the outcomes of the classification process are tabulated in Table 1. These findings were validated by comparing the obtained results with the database.

The OpenCV algorithm has proven to be a dependable tool in the detection process, and applying Gaussian Kernel filtering has yielded positive results in the object classification procedure. Choosing between four LED dots or a square shape hinges on the specific application's requirements regarding pattern usage. Both systems exhibit the remarkable capability of providing a detection range of up to 20 m, and this range can be further extended by modifying the size of the LEDs and LED strips used. The subsequent phase of this endeavor involves object tracking, where the detected object will be realigned to the

center of the camera's field of view. This adjustment streamlines the process of transmitting data, enhancing overall efficiency.

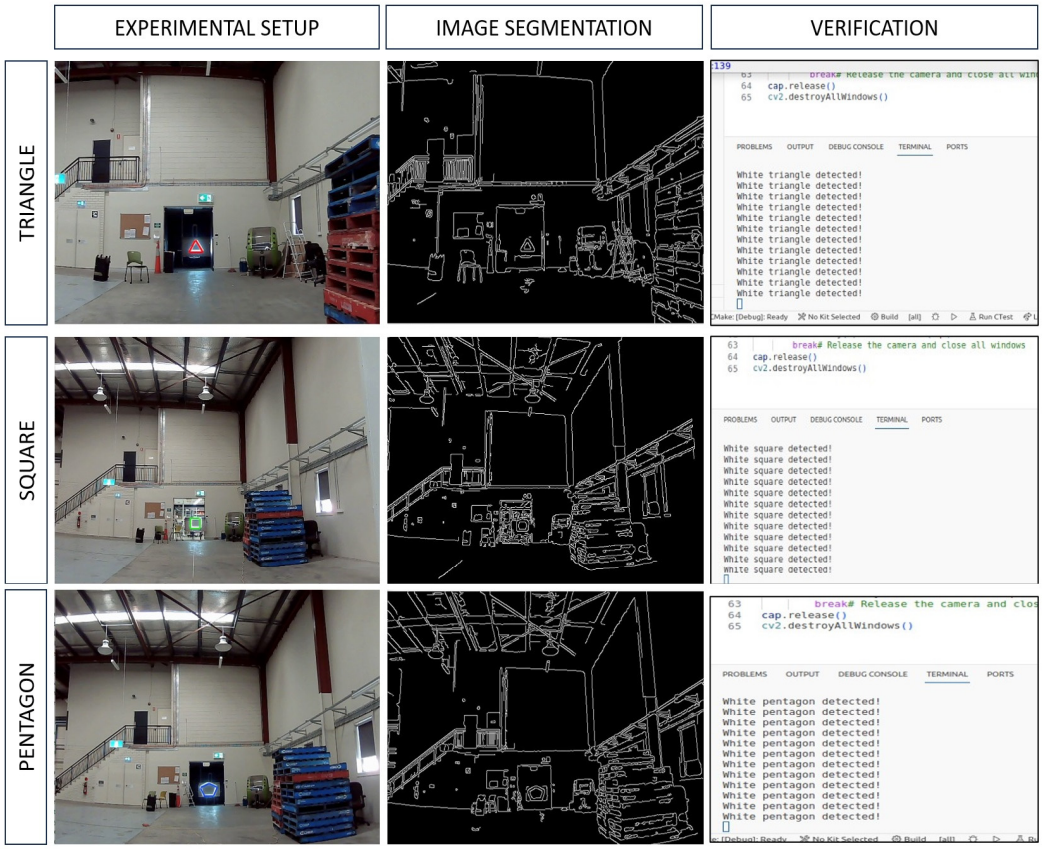

Figure 12. Object classification results.

Table 1. Light shape detection results.

Environmental Light Intensity (lux) I_o	Detection Range (m)		
	Triangle (I_x = 398 lux)	**Square (I_x = 450 lux)**	**Pentagon (I_x = 496 lux)**
0 − 1	14	20.5	15.3
160 − 170	13	18.4	14.3
380 − 390	10.1	17	12.5

4.3. Capability of PGP as a Remote Monitoring System

The adaptive software algorithm formulated using Equations (6) to (9) is instrumental in our approach. The initial step entails determining the solar irradiance value without a laser. This is achieved by calculating the average value ($\overline{V}_{si}$) over 256 iterations or within the time frame spanning from t_1 = 0 s to t_2 = 5 s. Subsequently, this value is translated into an ADC (Analog-to-Digital Converter) value, which falls within 0 to 1023 discrete analog levels. The outcome of the $\overline{V}_{si}$ calculation establishes the threshold value. If there is a significant change of more than 40 ADC units, the system reads ($\overline{V}_{(si+laser)}$) over 256 iterations to ensure the obtained value stabilizes before synchronization for data reception. The threshold value of 40 ADC was determined through testing, as depicted in Figure 13. These tests were conducted under three distinct lighting conditions (Ix): dark (Ix = 4 lux), low light (Ix = 293 lux), and normal light (Ix = 392 lux). Measurements were taken from 20 m, considering that the drone's operational range typically spans 10 to 20 m. This range ensures the effectiveness of the camera's field of view for monitoring suspicious

activities. It is important to note that the solar panel's input voltage value plays a pivotal role in these calculations in the absence of laser reception.

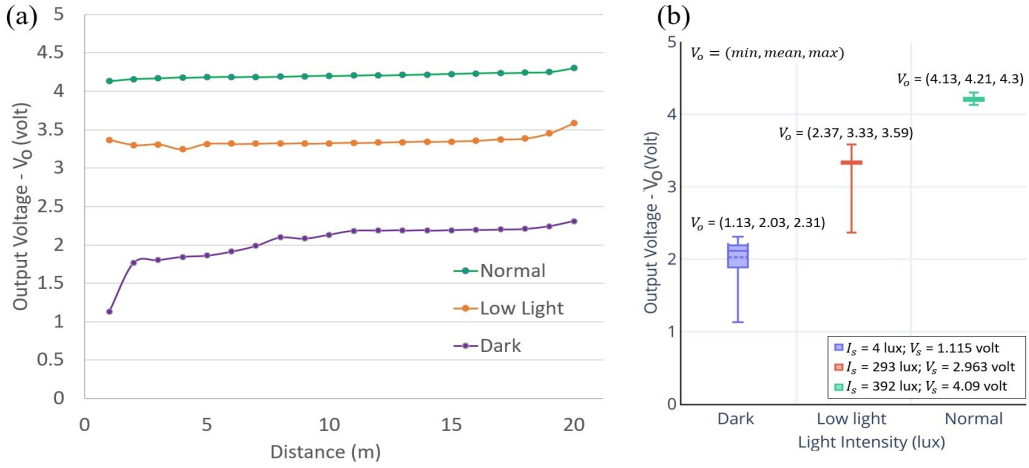

Figure 13. The effect of laser on a solar panel: (**a**) distance; (**b**) average output voltage.

The findings reveal that, at reading distances exceeding 10 m, the disparity between $(\overline{V}_{(si+laser)})$ and $\overline{V}_{si}$ exceeds 200 mV or surpasses 30 ADC discrete analog levels. These results significantly facilitate calculations to distinguish between ambient light and laser-generated light. Moreover, it is observed that as the distance between the laser and the solar panel increases, the solar panel's input voltage experiences a corresponding rise. This phenomenon is attributed to the larger radius of laser beam divergence, as illustrated in Figure 14. Additionally, under darker environmental conditions, the disparity value becomes more pronounced, enhancing the precision of the readings.

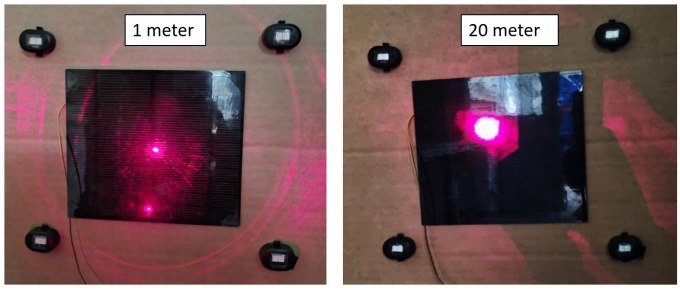

Figure 14. Laser beam divergence according to distance.

In this experimental phase, air-based data transmission involves the transmission of only 8 bits of data. The successful transmission and reception of identical data are verified through the serial monitor. These data functions are akin to RFID technology, enabling the extraction of information from PGP (Prescribed Ground Point), including the capability for PGP to relay timestamps to the cloud, serving as check-in information. The research demonstrates that due to the ample surface area of the solar panel, the system can receive data even when the laser experiences shifting or vibration caused by the drone's unstable hovering. It is important to note that this preliminary investigation does not prioritize factors such as bandwidth or delivery speed; these aspects will be explored in subsequent phases. From this initial trial, it can be deduced that the size of the solar panel and the choice of LED shape significantly influence the ease of detecting and transmitting information.

4.4. Validation and Testing in an Outdoor Environment

The objective of the outdoor testing phase was to confirm the efficacy of the developed aerial watchdog system for PGP detection, utilizing a unique red dot square pattern. In

Figure 10, we can observe the system's ability to identify a red LED even at a height of up to 6 m, even when operating in low-light conditions. The method leverages the power of the NVIDIA Jetson platform, offering drone control over three rotational axes: pitch, roll, and yaw angles. Figure 15 provides an overview of the outdoor testing environment, while Figure 7 illustrates sample outcomes during the search mode. Our research reveals that when the target falls within the camera's field of view, the sensing camera exhibits remarkable responsiveness, swiftly detecting and interpreting the LED as a square dot in just 1 millisecond. The findings are illustrated through a binary edge map within the mask frame, where white pixels signify detected edges. The fusion of previously discussed methods, including Gaussian smoothing, gradient computation, non-maximum suppression, and hysteresis thresholding, enhances the robustness of this image detection and classification technique. This rapid detection capability facilitates seamless laser data transmission, allowing for an uninterrupted 8-s window for data transmission.

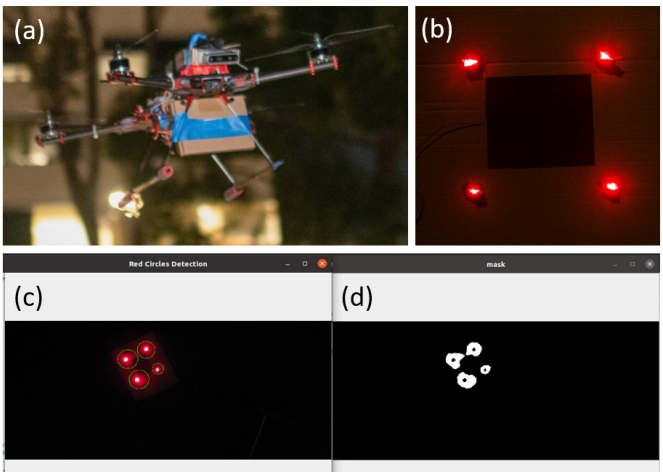

Figure 15. Flying watchdog outdoor test. (**a**) Flying watchdog; (**b**) PGP; (**c**) detector; (**d**) masking.

The Arduino-based laser communication system transmits data in 8-bit chunks to the PGP. Initially, it takes each incoming data byte received from the serial interface and splits it into two 4-bit nibbles. These nibbles are encoded into 8-bit integers using Hamming encoding, significantly enhancing the system's error detection and correction capabilities. Following this encoding step, the system introduces modulation by adding start and stop bits to ensure synchronization. During the modulation process, each bit is transformed into two half-bits, each accompanied by a clock pulse. Logical '1's are represented as 'on' states, while '0's are depicted as 'off' states, ensuring a robust and reliable data representation. The reconstructed byte is transmitted seamlessly through the serial interface, completing the communication cycle. Figure 16 illustrates the transmission process of continuous 8-bit data, specifically '101011' and '1101001,' with the start and stop bits represented by a logic '0'. This dedicated transmission method is capable of sending 8 bits of data at a rate of 117 Hz or within 8.5 milliseconds. This transmission speed is well-suited for seamless data reception and interpretation by the microcontroller, thereby ensuring error-free communication. The results of this process can be conveniently monitored through a serial monitor tool.

In this study, using LEDs as a marker poses challenges regarding light dispersion, influenced by various factors such as LED size, color, and type. Similarly, regarding vision detectors, the viewing angle and diffusers can significantly impact the reading process. Implementing a kernel function filtering mechanism and an edge detection method to address the issue of light dispersion, particularly in the context of wide-angle LEDs, has proven successful. These findings underscore that even a 5 mm LED with a 0.06 A current and a 5 V power supply can be detected by a vision sensor at distances of up to 20 m, mainly when the LED emits white light. However, in our investigation of the detection of traffic lights, the LEDs utilized are more extensive and equipped with LED reflectors, resulting in

brighter light emissions. In data transmission via laser technology, we departed from using laser optical communication equipment and opted for a more straightforward system. Our primary focus shifted towards developing programming algorithms to distinguish between laser-generated and ambient light, achieving more precise and reliable data transmission.

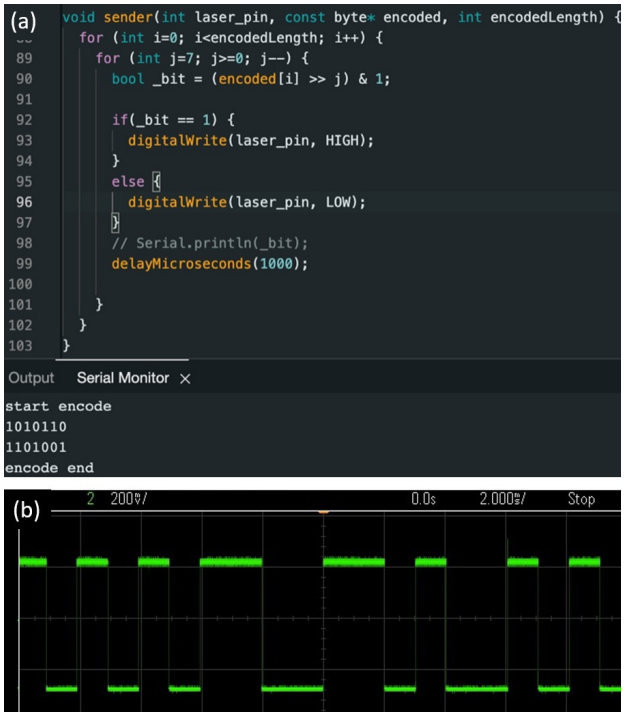

Figure 16. Laser data transfer: (**a**) transmitter data; (**b**) receiver data.

5. Conclusions

This research presents an innovative vision-based flying watchdog system designed to elevate the capabilities of human security-based guard patrol systems. This system's ability to identify and verify checkpoints under challenging lighting conditions and its real-time data transmission capabilities significantly contribute to security technology. It is important to note some limitations in our research. Firstly, our camera sensing position is fixed 45 downward, and the search pattern only forms a circular area with a radius of approximately 1 m from the waypoint. Despite these constraints, our light detection results reveal that a single white LED with specifications of 0.06 A, 5 volts, and an intensity ranging from 27 to 70 lux can achieve the longest detection distance, reaching up to 20 m, even in low-light conditions (9 lux) to standard lighting (390 lux). In contrast, other colors, such as blue, red, and purple lights, exhibit shorter detection distances. Notably, yellow light cannot be detected due to its proximity to white on the color spectrum. Additionally, we observed that the square shape outperforms the pentagon and triangle patterns regarding detection distance despite the pentagon having more LED strips and a larger size. Our research findings demonstrate the feasibility of utilizing LEDs with different colors and patterns for marker detection, particularly in scenarios where drone vision is deployed under favorable lighting conditions. We successfully implemented the OpenCV algorithm, incorporating thresholding and contouring with edge and color detection, effectively detecting and classifying objects. Furthermore, the Gaussian kernel algorithm proved effective in image smoothing. With an adaptive algorithm, the laser-based data transmission process exhibited robust performance, unaffected by variations in the distance between the drone and the PGP, as well as changes in beam intensity and divergence. Data transmission remained efficient, with 8-bit data packets utilized to expedite the process. In practical applications, when the transmitted data match the database on the detector, it triggers data upload to the cloud, thereby serving as part of the check-in verification process.

This study primarily focuses on low-speed data communication, currently lacking both encoders and decoders. Furthermore, the vision camera's orientation remains stationary, pointing downward. Consequently, forthcoming research initiatives should aim to enhance the PGP search method. Prospective improvements may encompass transitioning from a fixed camera angle to a dynamic one and incorporating sophisticated detection algorithms like YOLO to evaluate confidence levels. Furthermore, augmenting the laser-based data transmission technique by introducing data encoding, expanding capacity, and accelerating transmission speed could unlock fresh opportunities for advancing this groundbreaking security technology.

Author Contributions: Conceptualization, E.K.; methodology, A.S., A.J. and E.K.; software, A.J. and Y.Z.; validation, E.K.; formal analysis, A.J., Y.Z. and E.K; investigation, E.K. and A.S.; resources, A.S. and A.J.; data curation, A.J., E.K. and A.S.; writing—original draft, A.S. and E.K.; writing—review and editing, E.K., A.S., A.J. and Y.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not applicable.

Acknowledgments: The authors acknowledge the continued support from Macquarie University through the Computing and Engineering Faculty for providing the resources and space required for this research. We also acknowledge the technical staff at Macquarie for providing support and hardware tools required for prototyping.

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

References

1. Wang, X.; Yao, F.; Li, A.; Xu, Z.; Ding, L.; Yang, X.; Zhong, G.; Wang, S. DroneNet: Rescue Drone-View Object Detection. *Drones* **2023**, *7*, 441. [CrossRef]
2. Shahmoradi, J.; Talebi, E.; Roghanchi, P.; Hassanalian, M.A. Comprehensive Review of Applications of Drone Technology in the Mining Industry. *Drones* **2020**, *4*, 34. [CrossRef]
3. Mogili, U.R.; Deepak, B.B.V.L. Review on Application of Drone Systems in Precision Agriculture. *Procedia Comput. Sci.* **2018**, *133*, 502–509. [CrossRef]
4. Mohd Daud, S.M.S.; Mohd Yusof, M.Y.P.; Heo, C.C.; Khoo, L.S.; Chainchel Singh, M.K.; Mahmood, M.S.; Nawawi, H. Applications of Drone in Disaster Management: A Scoping Review. *Sci. Justice* **2022**, *62*, 30–42. [CrossRef]
5. Mohd Noor, N.; Abdullah, A.; Hashim, M. Remote Sensing UAV/Drones and Its Applications for Urban Areas: A Review. In Proceedings of the IOP Conference Series: Earth and Environmental Science, Kuala Lumpur, Malaysia, 24–25 April 2018. [CrossRef]
6. Ali, H.; Hang, L.Y.; Suan, T.Y.; Polaiah, V.R.; Aluwi, M.I.F.; Zabidi, A.A.M.; Elshaikh, M. Development of Surveillance Drone Based Internet of Things (IoT) for Industrial Security Applications. In Proceedings of the Journal of Physics: Conference Series, Perlis, Malaysia, 19–20 October 2021. [CrossRef]
7. Subbarayalu, V.; Vensuslaus, M.A. An Intrusion Detection System for Drone Swarming Utilizing Timed Probabilistic Automata. *Drones* **2023**, *7*, 248. [CrossRef]
8. Alrayes, F.S.; Alotaibi, S.S.; Alissa, K.A.; Maashi, M.; Alhogail, A.; Alotaibi, N.; Mohsen, H.; Motwakel, A. Artificial Intelligence-Based Secure Communication and Classification for Drone-Enabled Emergency Monitoring Systems. *Drones* **2022**, *6*, 222. [CrossRef]
9. Kumar, A.; Yadav, A.S.; Gill, S.S.; Pervaiz, H.; Ni, Q.; Buyya, R. A Secure Drone-to-Drone Communication and Software Defined Drone Network-Enabled Traffic Monitoring System. *Simul. Model Pract. Theory* **2022**, *120*, 102621. [CrossRef]
10. Derpich, I.; Rey, C. Drone Optimization in Factory: Exploring the Minimal Level Vehicle Routing Problem for Efficient Material Distribution. *Drones* **2023**, *7*, 435. [CrossRef]
11. Shah, S.A.; Lakho, G.M.; Keerio, H.A.; Sattar, M.N.; Hussain, G.; Mehdi, M.; Vistro, R.B.; Mahmoud, E.A.; Elansary, H.O. Application of Drone Surveillance for Advance Agriculture Monitoring by Android Application Using Convolution Neural Network. *Agronomy* **2023**, *13*, 1764. [CrossRef]
12. Hafeez, A.; Husain, M.A.; Singh, S.P.; Chauhan, A.; Khan, M.T.; Kumar, N.; Chauhan, A.; Soni, S.K. Implementation of Drone Technology for Farm Monitoring and Pesticide Spraying: A Review. *Inf. Process. Agric.* **2023**, *10*, 192–203. [CrossRef]
13. Iqbal, U.; Riaz, M.Z.B.; Zhao, J.; Barthelemy, J.; Perez, P. Drones for Flood Monitoring, Mapping and Detection: A Bibliometric Review. *Drones* **2023**, *7*, 32. [CrossRef]
14. Benes, F.; Stasa, P.; Svub, J.; Alfian, G.; Kang, Y.S.; Rhee, J.T. Investigation of UHF Signal Strength Propagation at Warehouse Management Applications Based on Drones and RFID Technology Utilization. *Appl. Sci.* **2022**, *12*, 1277. [CrossRef]

15. Juang, J.G.; Tu, G.T.; Liao, Y.H.; Huang, T.H.; Chang, S.I. Drone Patrol Using Thermal Imaging for Object Detection. *Infrared Sens. Devices Appl.* **2020**, *11503*, 1–7. [CrossRef]
16. Xu, B.; Zhao, K.; Luo, Q.; Wu, G.; Pedrycz, W. A GV-Drone Arc Routing Approach for Urban Traffic Patrol by Coordinating a Ground Vehicle and Multiple Drones. *Swarm Evol. Comput.* **2023**, *77*, 101246. [CrossRef]
17. Bollard, B.; Doshi, A.; Gilbert, N.; Poirot, C.; Gillman, L. Drone Technology for Monitoring Protected Areas in Remote and Fragile Environments. *Drones* **2022**, *6*, 42. [CrossRef]
18. Li, Y.; Karim, M.M.; Qin, R. A Virtual-Reality-Based Training and Assessment System for Bridge Inspectors with an Assistant Drone. *IEEE Trans. Hum. Mach. Syst.* **2022**, *52*, 591–601. [CrossRef]
19. Alwateer, M.; Loke, S.W.; Zuchowicz, A.M. Drone Services: Issues in Drones for Location-Based Services from Human-Drone Interaction to Information Processing. *J. Locat. Based Serv.* **2019**, *13*, 94–127. [CrossRef]
20. Liu, Q.; He, Z.; Li, X.; Zheng, Y. PTB-TIR: A Thermal Infrared Pedestrian Tracking Benchmark. *IEEE Trans. Multimed.* **2020**, *22*, 666–675. [CrossRef]
21. Guo, X.; Hu, Q. Low-Light Image Enhancement via Breaking Down the Darkness. *Int. J. Comput. Vis.* **2023**, *131*, 48–66. [CrossRef]
22. Chen, Y.Y.; Jan, J.K.; Tsai, M.L.; Ku, C.C.; Huang, D.C. On the Security of RFID-Based Monitoring Mechanism for Retail Inventory Management. *KSII Trans. Internet Inf. Syst.* **2012**, *6*, 515–528. [CrossRef]
23. Fernández-Caramés, T.M.; Fraga-Lamas, P.; Suárez-Albela, M.; Castedo, L. Reverse Engineering and Security Evaluation of Commercial Tags for RFID-Based IoT Applications. *Sensors* **2017**, *17*, 28. [CrossRef] [PubMed]
24. Piciarelli, C.; Foresti, G.L. Drone Swarm Patrolling with Uneven Coverage Requirements. *IET Comput. Vis.* **2020**, *14*, 452–461. [CrossRef]
25. Stolfi, D.H.; Brust, M.R.; Danoy, G.; Bouvry, P. CONSOLE: Intruder Detection Using a UAV Swarm and Security Rings. *Swarm Intell.* **2021**, *15*, 205–235. [CrossRef]
26. Moltajaei Farid, A.; Mei Kuan, L.; Kamal, M.A.S.; Wong, K. Effective UAV Patrolling for Swarm of Intruders with Heterogeneous Behavior. *Robotica* **2023**, *41*, 1673–1688. [CrossRef]
27. Patrinopoulou, N.; Daramouskas, I.; Meimetis, D.; Lappas, V.; Kostopoulos, V. A Multi-Agent System Using Decentralized Decision-Making Techniques for Area Surveillance and Intruder Monitoring. *Drones* **2022**, *6*, 357. [CrossRef]
28. Yao, C.B.; Kao, C.Y.; Lin, J.T. Drone for Dynamic Monitoring and Tracking with Intelligent Image Analysis. *Intell. Autom. Soft Comput.* **2023**, *36*, 2233–2252. [CrossRef]
29. Kakiuchi, R.; Tran, D.T.; Lee, J.H. Evaluation of Human Behaviour Detection and Interaction with Information Projection for Drone-Based Night-Time Security. *Drones* **2023**, *7*, 307. [CrossRef]
30. Xiang, H.; Han, Y.; Pan, N.; Zhang, M.; Wang, Z. Study on Multi-UAV Cooperative Path Planning for Complex Patrol Tasks in Large Cities. *Drones* **2023**, *7*, 367. [CrossRef]
31. Hassan, E.; Khalil, Y.; Ahmad, I. Learning Deep Feature Fusion for Traffic Light Detection. *J. Eng. Res.* **2023**, *in press*. [CrossRef]
32. Yang, L.; Ma, R.; Zakhor, A. Drone Object Detection Using RGB/IR Fusion. *arXiv* **2022**, arXiv:2201.03786. [CrossRef]
33. Kashiyama, T.; Sobue, H.; Sekimoto, Y. Sky Monitoring System for Flying Object Detection Using 4k Resolution Camera. *Sensors* **2020**, *20*, 7071. [CrossRef] [PubMed]
34. Sarhan, N.H.; Al-Omary, A.Y. Traffic Light Detection Using OpenCV and YOLO. In Proceedings of the 2022 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies, Bahrain, 20 November 2022. [CrossRef]
35. Niu, C.; Li, K. Traffic Light Detection and Recognition Method Based on YOLOv5s and AlexNet. *Appl. Sci.* **2022**, *12*, 10808. [CrossRef]
36. Wang, Q.; Zhang, Q.; Liang, X.; Wang, Y.; Zhou, C.; Mikulovich, V.I. Traffic Lights Detection and Recognition Method Based on the Improved Yolov4 Algorithm. *Sensors* **2022**, *22*, 200. [CrossRef] [PubMed]
37. Massetti, L.; Paterni, M.; Merlino, S. Monitoring Light Pollution with an Unmanned Aerial Vehicle: A Case Study Comparing RGB Images and Night Ground Brightness. *Remote Sens.* **2022**, *14*, 2052. [CrossRef]