axioms

MDPI

Article

Radio-Controlled Intelligent UGV as a Spy Robot with Laser Targeting for Military Purposes

Muhammad Zia Ur Rahman [1], Umair Raza [1], Muhammad Azeem Akbar [2,*], Muhammad Tanveer Riaz [1,3], Abdu H. Gumaei [4] and Nasir Ahmad [1]

[1] Department of Mechanical, Mechatronics and Manufacturing Engineering, University of Engineering and Technology Lahore Faisalabad Campus, Faisalabad 38000, Pakistan
[2] Department of Software Engineering, LUT University, 15210 Lappeenranta, Finland
[3] Department of Electrical, Electronic, and Information Engineering (DEI), Università di Bologna, 40136 Bologna, Italy
[4] Department of Computer Science, College of Computer Engineering and Sciences, Prince Sattam Bin Abdulaziz University, Al-Kharj 11942, Saudi Arabia
* Correspondence: azeem.akbar@lut.fi

Abstract: The main objective of this unmanned ground vehicle is to deal with the security issues like terrorist activities across the border and in various remote combat missions by reducing the involvement of soldiers. This unmanned ground robot comprises a wireless high-definition camera that can transfer live streams from the robot to headquarters using Wi-Fi. The robot's movement can be controlled with two modes; one of them is a radio controller working on 2.4 GHz frequency with seven independent channels. Secondly, its movement can also be controlled using a Python-based GUI application. Nowadays, different techniques have been used for face recognition; in our remotely piloted robot, we have used Haar-cascade classifiers in combination with the LBPH algorithm to implement real-time facial recognition. The robot uses a rack and pinion driving mechanism and an ATMEL Cortex-M3 CPU as a controller with 32-bit/s processing speed. In addition, a laser is installed on the turret to shoot the targets down, which can be used in an autonomous mode based on facial recognition results, or it can be used manually either through an RF controller or Python-based GUI. The turret moves in 2-DOF with the help of metallic geared servo motors. Both servo motors can rotate up to 180°. The driving mechanism of the robotic tank is just like DDR, with one difference, the two DC gear motors of the robot are connected diagonally.

Keywords: image processing; intelligent systems; RF communication; mobile robotics; rack and pinion mechanism; turret tilt and pan mechanism; LBPH algorithm; object detection; object localization; machine vision

check for updates

Citation: Rahman, M.Z.U.; Raza, U.; Akbar, M.A.; Riaz, M.T.; Gumaei, A.H.; Ahmad, N. Radio-Controlled Intelligent UGV as a Spy Robot with Laser Targeting for Military Purposes. *Axioms* **2023**, *12*, 176. https://doi.org/10.3390/axioms12020176

Academic Editors: Jaime Gallardo-Alvarado and Mario A. Garcia-Murillo

Received: 9 December 2022
Revised: 30 January 2023
Accepted: 4 February 2023
Published: 8 February 2023

Copyright: © 2023 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/).

1. Introduction

Human life is precious, and the loss of life can be reduced considerably by using robots in the armed forces. Nowadays, all military organizations use such robots to perform perilous tasks in chancy environments [1]. In the war field, we control our military tanks by personal computer wirelessly using Wi-Fi or RF (radio frequency) communication systems with live video transmission and image processing techniques that will target the felon automatically or manually using the laser gun. These kinds of robots are very helpful in conducting various remote combat missions and in identifying as well as eliminating different terrorist activities within the border, and across the border. In addition, night vision cameras can also be used on the robot to conduct surveillance even in the darkness of night [2,3].

Conventionally, military robots were not capable of live video transmission; they only captured video and stored it in their storage device, but now with live video transmission, the military can react in more detail presently. Getting information through live video

transmission is also helping them to be prepared for the situation ahead. Furthermore, this live video transmission can be used for different computations, as used in this project for facial recognition [4]. In addition, with the surveillance feature, this robot can also be used as a spy robot. Spy robots were first introduced in the early 1940s in World War II. The purpose of these robots was to reduce the number of victims of war, as the entrance of a person into the enemy territory was very difficult, and accessing sensitive information for secret operations was nearly impossible; hence these robots were introduced. However, the major flaw was that these robots were not wireless. Another flaw was the storage of video streaming which requires a large amount of disk storage [4].

Unmanned ground vehicles (UGV) were introduced in the early 1970s, a great revolution in the history of robotics. But there were also many flaws in these UGVs as they were wirelessly controllable over only a very small range like 50–100 m. The only advancement in these spy robots was that they were wireless, but the flaw in their video transmission was the same. This flaw has also been covered in this project; it can be controlled wirelessly over a range of approximately 1.2 km with a clear line of sight. However, if the line of sight is unclear, it might reduce the effective range of the wireless control system [5].

Then, in the early 2000s, modern UGVs were invented. They were controllable over a long range of hundreds of meters and capable of live video transmission. Still, they were controlled manually using joysticks and targeting the felon manually by viewing the target on the screen. To improve this flaw, automatic laser targeting has been introduced in our project. Automated laser targeting can also help eliminate ground-to-air targets manually and eliminate ground-ground human targets automatically and manually [4].

Along with the advancement of the control features of robots from wire transmission to RF and then with long-range RF communications, Wi-Fi-based control features have also been in use for a few years, in which mobile robots are controlled using graphical user interfaces based on different programming languages like Java, JavaScript, node, C^{++} and most famous Python. These GUIs are built on personal computers or laptops, and robots can be controlled by different commands from GUI [6].

A few years earlier, we saw some spy and normal robots, controllable by smartphones. The smartphone has been a great invention in human history. Modern smartphones have been equipped with GPS, compass, and some other great features which are very useful, and the use of the mobile phone in such projects makes it compact and makes the man–machine interface more attractive [7].

Advancement in live video transmission and computation of that video has increased daily because of continuous research in artificial intelligence, machine learning, computer vision, and robotics. This research urges researchers and developers to make robots controllable over a wide range of applications equipped with live video transmission, which can be used for various operations for the benefit of mankind. For example, monitoring flooded areas and estimating damages due to floods [8]. Another example of advancement in image processing leads to farming facilities in which avoiding diseases and checking pests and other harmful agents seems impossible without sensors. However, installing sensors on wide land is not cost-effective. Hence, image processing techniques have been used to identify the harmful agents by capturing pictures of leaves and sending a notification to farmers through IoT and other means to control actions [9].

The advancement in the field of image processing is not confined to the above applications but has been extended to military applications. For example, spy robots have been seen equipped with a color sensor, which helps them to camouflage based on the environment. Advanced robots now use image processing techniques to camouflage rather than using color sensors [5]. Another application in military robots has been seen in recent years: motion tracking. Based on motion tracking, several operations can now be done, for example, triggering a security alarm or remote elimination using a controllable weapon [10].

In recent years spy robots with live video transmission have been used to conduct highly dangerous remote operations, for example, surveillance in terrorist areas in which mines have been planted. In the early stages, these robots were used to defuse the mines

using remote operations. However, with the advancement of technology, they can also be used to conduct search operations on enemy ground to detect and defuse mines. Furthermore, mine detection and motion tracking were initially based on the image subtraction algorithm, also known as SAD [11].

Moreover, in armies of under-developed countries, no such projects have yet to detect targets automatically with the database feed. In our project, unlike a traditional gun, the robot uses a laser gun to detect the target using image processing automatically; then, it will shoot it with a laser gun. The laser gun is used in advanced military projects nowadays; high-power laser guns have a variety of applications [12]. For example, it has been used for the elimination of water-to-air targets. Similarly, advanced militaries are using laser guns to shoot down ground-to-air targets, i.e., enemy drones, etc. [13,14].

Moreover, in our project, the turret, equipped with a laser gun, can also be controlled manually in case the facial features are unclear or we find a potential risk in the live transmission that is not in our database.

As this robot is wireless and equipped with the advantage of live video transmission, there is no need to install expensive hard drives to store the data on the robot. It will also help reduce weight significantly. On the contrary, this video streaming can be used in real-time to act. We can also store the video on any PC in headquarters, not on the robot.

To improve the processing speed of images taken from the live video transmission and to implement more accurate and real-time facial recognition, an advanced algorithm has been used in this project which processes the images in a matter of seconds and not only detects the specified target but also processes real-time localization of the target within the frame of reference. This real-time localization helps calculate the exact position of the target in the structure of reference using the LBPH algorithm [15]. Some existing works also used the LBPH algorithm, where authors [16] used this algorithm in drone technology; the system provides 89.1% efficiency. In another research study [16], the LBPH algorithm is built into drone technology that exhibits 98% efficiency. In the proposed research, an intelligent LBPH algorithm is developed in radio-controlled UGV. The difference between the works [16,17] and ours is that although we employ many samples—roughly 50 for each object—and the number can be expanded to an endless number, the recognition accuracy improves as the number of samples increases. Therefore, a 99.8% precision factor is something we are capable of. Furthermore, since we have improved the database with many samples, we can identify numerous people simultaneously. In addition, we used a single high-definition camera on the robot because the mobile robot's visual control interface, just one camera, can make it simple to operate [18].

After exact coordinates are found, it draws a rectangle around the face of the target, and the coordinates of this rectangle are sent to the robot wirelessly over Wi-Fi. As the person moves in the graphical video frame of reference, an image processing technique named object localization is used to track the movement of the face in real-time [19–21].

Then, with basic mathematics, conversion is performed using the coordinates, and the aim of the turret is controlled. As mentioned earlier, this spy robot is controllable over a wide range of km over an RF line. Previously these kinds of spy robots could not communicate on secure RF lines, and this communication could be tempered at any time. Still, in this project, RF communication is performed over secure lines. As mentioned earlier, this robot can be controlled manually and has two modes. The first is a radio controller, and the other is a Python-based GUI. The radio controller comprises four analog channels and three digital channels. Two analog channels are used to control the movement of the robot, two for the movement of the turret, and digital channels are used for laser targeting, etc.

2. Methodology and Hardware

HMI for this robot is a Python-based GUI application and a radio controller used to control the robot's movement. This spy robot operates in 2 modes which are as follows:

1. Manual mode
2. Semi-Automatic mode.

In "manual mode", the position of the turret, the laser targeting, and the robot's movement are controlled manually using the GUI application or the radio controller. The radio controller comprises a joystick to control the movement of the robot and another joystick for the movement of the turret in 2 directions. However, in Semi-Automatic mode, the turret's position depends upon the face's position in the video frame, and the shooting down of the target whose ID is already in the "Database" is done automatically by the facial recognition algorithm using the laser weapon. The general working of our intelligent robot can be understood by the flowchart shown in Figure 1.

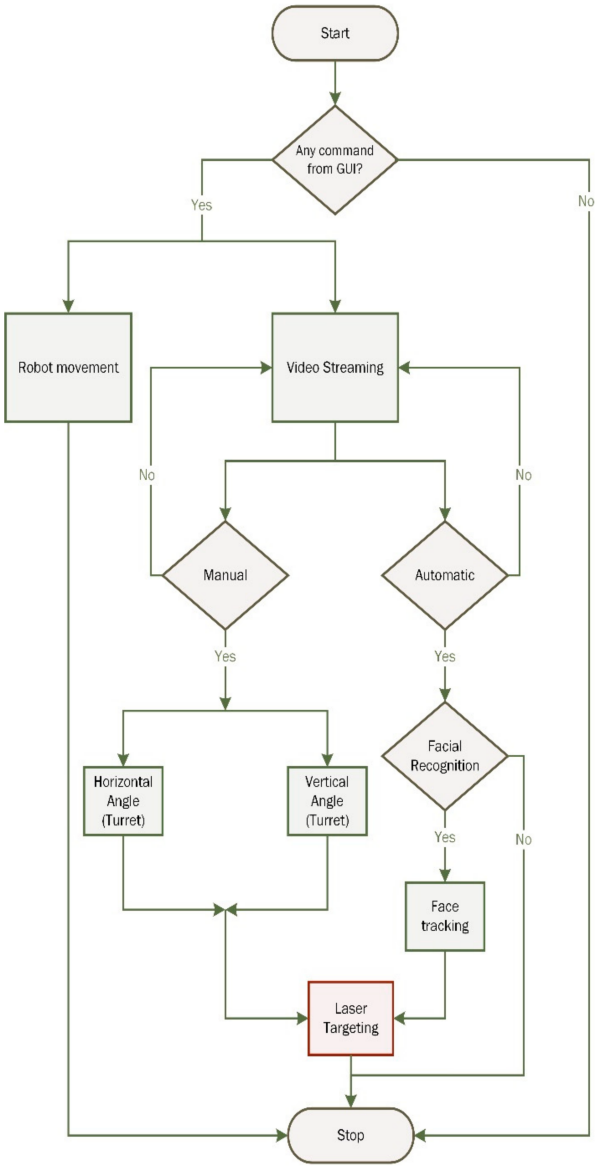


Figure 1. Flowchart.

The facial recognition algorithm is also based on Python; for compiling the algorithm, PyCharm was used. Haar-Cascade classifiers and the linear binary pattern histogram (LBPH) algorithm are used to recognize the face of the person in the image taken from the video. This whole procedure is divided into basically 3 sections:

1. Dataset
2. Trainer
3. Recognizer

In the 1st section of 'Dataset', the algorithm is provided with the sample images of the target. It assigns the target a unique ID and saves it in the database. Datasets can also be made in 2 different ways. The first method is that we have one picture to train our algorithm, but this method is less accurate. The second method is to capture several sample images of a single target and create a mini dataset of each target, saving samples with a unique ID. Currently, the algorithm in this project has been trained to capture and save up to 50 samples of each target in the directory, as shown in Figure 2.



Figure 2. Data Samples.

The picture is cropped to the boundary area of the face and is conducted using face detection based on the Haar-cascade algorithm; sample pictures do not include any background. As Haar-cascade is based on a linear binary pattern histogram, colored image samples are converted into grayscale images for quick processing compatible with LBPH [15].

In the 2nd section, 'Trainer', the algorithm is trained for the sample images stored in the database. The first step moves to the dataset's directory, where sample images are stored. Then, it imports all the images stored in the directory, applies the LBPH recognizer to identify the faces, and applies the Haar-cascade classifier. After that, it stores the training file "yml" extension, which is further used by the recognizer.

In the final section, 'Recognizer,' the algorithm displays the live video streaming on the PC screen. If a person comes in front of the camera, the algorithm takes a picture of the person. Then, it starts matching that picture with the sample images of the unique ID stored in the dataset trainer file created in the previous section. The same method will be implemented as in the dataset in which it converts the images into grayscale and then implements the detector, which determines the features of the face and changes the area of focus to the face region, then it matches with the dataset. If the captured image matches the one stored in the dataset with the unique ID, then the name of the target is displayed on the screen.

When the facial recognition is done, the algorithm of 'Recognizer' performs another task of 'tracking.' The 'Recognizer' algorithm determines the coordinates of the face of the target on the screen and then sends those coordinates to the microcontroller by serial communication. These coordinates are then transmitted to the other microcontroller, which is the CPU of the robot, wirelessly, using RF-based secure lines over 2.4 GHz. To transmit these tracking coordinates or the other commands to control the robot movement, NRF modules are used. The PCB layout of the transmitting NRF module is shown below in Figure 3.

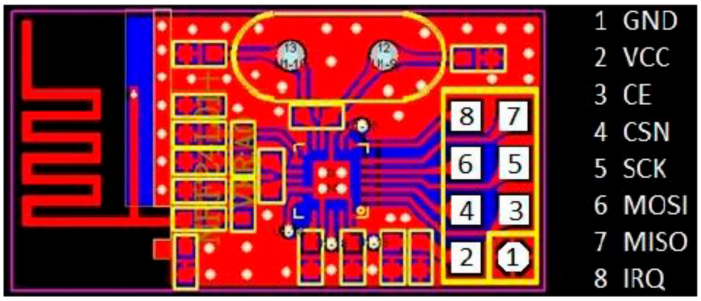


Figure 3. NRF PCB Layout.

NRF modules operate on 2.4 GHz and have a wide range of up to several km. The range of NRF modules can be increased using an antenna, as shown below in Figure 4.



Figure 4. Long-range NRF transceiver module.

These NRF modules operate over SPI (Serial Peripheral Interface) protocol. This protocol enables a server where several 'slave' devices can be controlled using a single 'Master' device. Specific pins of the microcontroller are used in this protocol; one of those pins is 'CE', which enables the 'Clock' pin for the specific 'slave' device. Another feature of the NRF modules is their 'bidirectional communication.' There is a wide range of air data rates supported by NRF modules, i.e., 250 Kb/s–2 Mb/s. Furthermore, the no. of channels of the NRF module is '32', meaning 32 devices can be controlled simultaneously.

Along with the NRF modules, NRF adapters are also used to regulate the supply voltage and to keep the data packets intact in case of any loss. These adapters are useful for microcontrollers like Arduino, ESP, and Atmel Cortex M3. Because these microcontrollers work on 5 V digital or analog Input/Output, but the NRF module operates on 1.9–3.6 V to regulate the supply voltage, adapters have been used. It also consists of bypass capacitors, increasing the module's communication reliability. The physical assembly of the NRF adapter module is shown below in Figure 5.

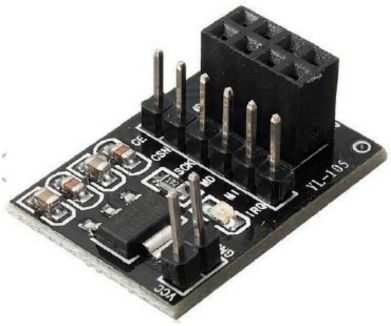


Figure 5. NRF Adapter module.

It also contains a 4*2 female connector for easy plug-in of NRF modules, one 6*1 male connector for ease of SPI communication with the microcontroller, and a 2*1 male connector for supply voltage. In addition, there is an LED, which also ensures the working of the NRF adapter. The overall specification of the NRF adapter module is given below in Table 1.

Table 1. Specification of NRF Adapter module.

Parameters	Details
Input DC Voltage	5 V DC
Output DC Voltage	1.9~3.6 DC
DC Current	12.3 mA
On-board Voltage Regulator	AMS1117-3.3 chip
Dimension	26 × 19 × 12 mm

2.1. Hardware Design

2.1.1. Mechanical Design

The mechanical model of the spy robot is a tank-type robot consisting of a rack and pinion as driving mechanisms. This driving mechanism has been used keeping in mind the application of the robot. Because this robot has applications in military operations and remote areas, which are mostly rough in surface, this rack and pinion driving mechanism will lead to more efficient movement on different surfaces [22–24]. Basically, 2 encoded DC gear motors have been installed in this project diagonally to control the robot's movement. Specifications of gears are given below in Table 2, and hardware assembly is in Figure 6.

Table 2. Specification of driving gears.

Parameters	Details
Diameter	5 cm
No. of Teeth	20
Teeth Thickness	0.2 cm
Pitch	0.553 cm
Center–Center Distance	3.1 cm
Length of Shaft Connector	4.1 cm



Figure 6. Driving gears.

There are also some bearing wheels to assist the driving mechanism of the caterpillar belt to work smoothly. This also helps them travel over rough surfaces. The bearing wheels of this robot can be visualized below in Figure 7.



Figure 7. Bearing wheels.

Specifications of the bearings are given in Table 3.

Table 3. Specifications of bearing wheels.

Parameters	Details
Diameter	4.8 cm
Length of Shaft Connector	4.1 cm
Center–Center Distance	3.1 cm

A 2-DOF "tilt-pan" mechanism is installed on the robot based on the metallic gear servo motors to control the turret's position. This involves 2 brackets in which servo motors can be assembled, as given below in Figure 8.

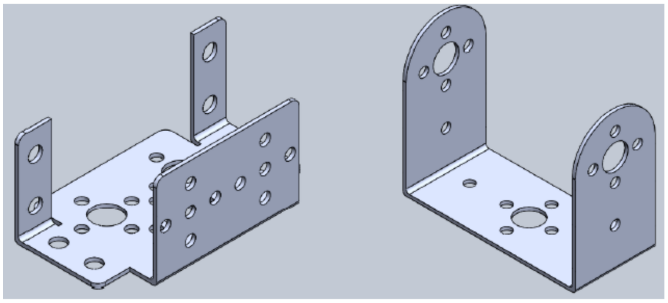


Figure 8. Servo Brackets for tilt-pan mechanism.

A plastic-based caterpillar belt has been used in this project to keep the weight of the robot within the limit. Specifications of the caterpillar belt are as follows in Table 4:

Table 4. Specifications of caterpillar belt.

Parameters	Details
Length	78 cm
Width	4 cm
Joint to Joint Horizontal distance	5.2 cm
No. of joints	100

Overall, the CAD model of the mechanical assembly of the robot base is given below in Figure 9.

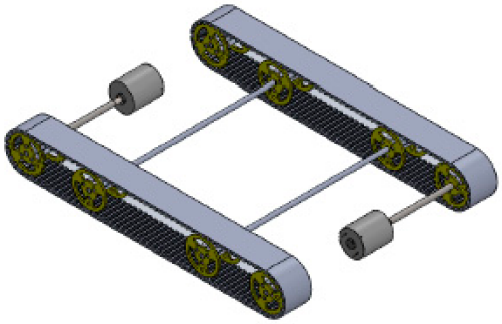


Figure 9. CAD Model of mechanical base.

2.1.2. Electronic Design

The major electronic part of this robot is NRF24L01 modules which transmit and receive the signals to perform required tasks. NRF adapter bases regulate the power supply to the NRF modules and smooth the data packets' transmission. The general working of the electronic system of our intelligent robot can be understood from the block diagram shown in Figure 10.

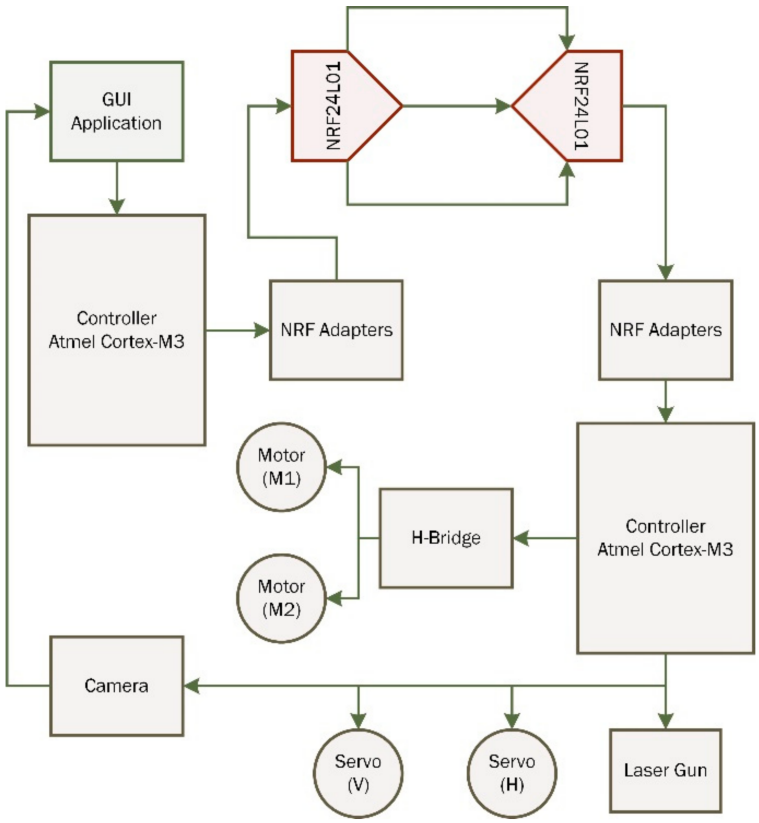


Figure 10. Electronic Design.

The robot's movement is controlled by an H-Bridge made of high-power BJTs. For safety purposes, 'catch diodes' and opto-couplers are used, which keeps the controller safe from any back EMF effect. The PCB layout of the H-bridge is shown in Figure 11, and 3D layout is in Figure 12.

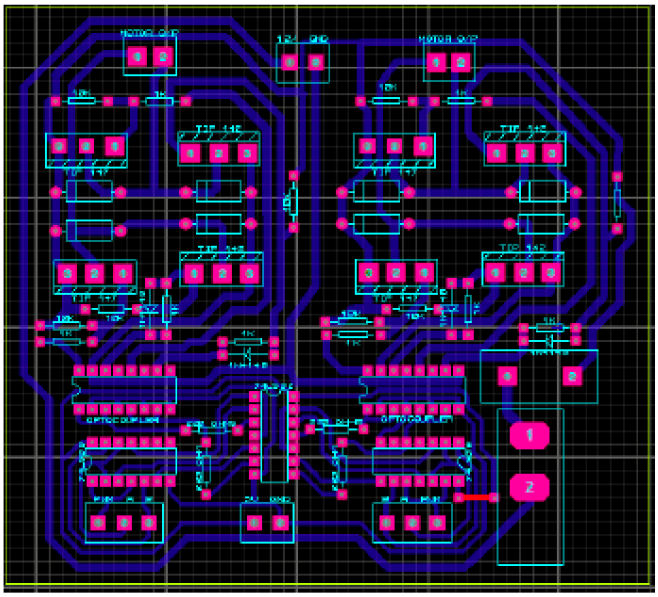


Figure 11. PCB Design of H-Bridge.

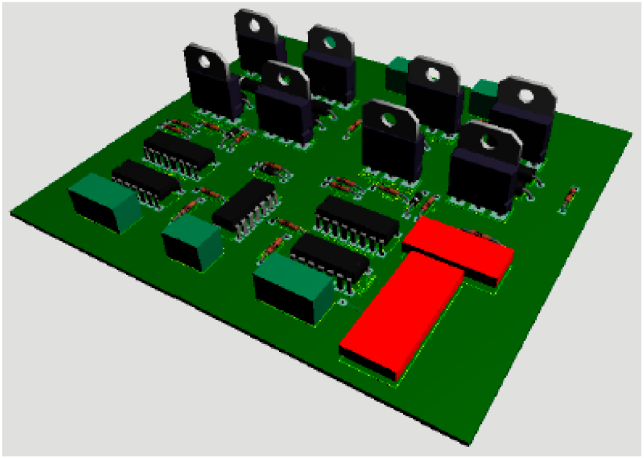


Figure 12. 3D Design of H-Bridge.

This project consists of a mini-IP camera having a built-in Wi-Fi chip to capture the video and transmit it to the personal computer in 1080p resolution. This camera is operated with a 3.7 V Ni battery. The battery and camera module are shown in Figure 13.



Figure 13. Mini Wi-Fi IP Camera.

The main microcontroller used in this project is an Atmel Cortex M3 installed on the robot. This microcontroller has a processing speed of 32-bit/. It controls the movement of the robot and turret and also communicates wirelessly through the NRF module with the PC, where image processing is performed. Upon reception of the command, it will execute the laser targeting. Atmel cortex M3 can be seen in Figure 14.

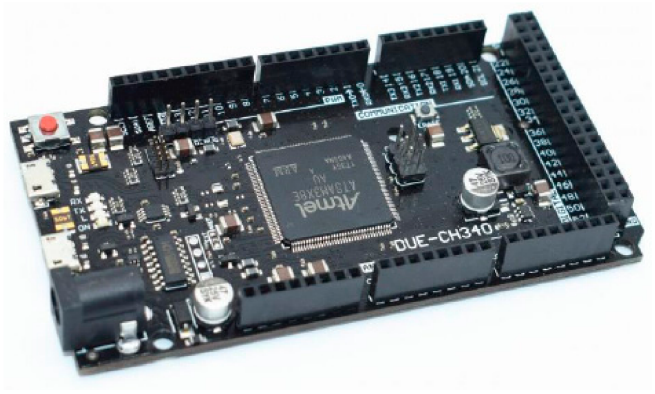


Figure 14. Atmel Cortex M3.

2.1.3. Design Parameters and Mathematics

Design parameters and calculations will be categorized into 3 categories, i.e., base frame, load calculation, and motor selection according to load.

- Base Frame
  - Length: 38 cm
  - Width: 28 cm
  - Thickness: 0.3 cm
- Approximate Load
  - Atmel Cortex-M3 = 80 g
  - NRF = 90 g
  - Camera chip = 150 g
  - Battery = 500 g
  - Servo motors = 100 g
  - Brackets = 100 g
  - Laser module chip = 300 g
  - Gears = 200 g
  - Bearings = 600 g
  - Base = 1800 g
  - Motors = 600 g
  - Others = 250–450 g

  (Total = 4970 g approximate loads considering tolerance 5200 g)
- Motor Selection

Given that the radius of the gear wheel is 2.4 cm and required translational speed is 40 cm/s–60 cm/s.

$$v = r \times \omega \tag{1}$$

From Equation (1):

$$\omega = (v/r) \times 9.54592 \text{ rpm}$$

$$\omega = (50/2.4) \times 9.54592 \text{ rpm}$$

$$\omega = 198 \text{ rpm}$$

$$m = 5.2 \text{ kg}$$

$$\tau = 3.47 \text{ Nm}$$

$$V = 12\ \text{V}$$
$$P = 18\ \text{W} \quad (2)$$

From Equation (2):

$$I = \frac{P}{V} = \frac{18}{12} = 1.5\ A$$

2.2. Facial Recognition

2.2.1. Dataset

A dataset is created by determining the face in the image captured. A binary number is obtained by labeling each image pixel using the LBP operator to determine the exact facial features and their location in the image. Each image pixel is labeled by a threshold of 3×3 neighborhood from the center value [15]. The value of each label can be represented as

$$LBP_{P,R} = \sum_{P=0}^{P-1} s(g_p - g_c)2^P \quad (3)$$

where the gray value of each equally spaced pixel around the central pixel is represented by g_p and the gray value of each central pixel is g_c. With the help of this LBP operator, only that part of the image is cropped and saved in the dataset, which has facial features; the rest of the image is excluded.

2.2.2. Recognizer

Comparison between the stored dataset and the current facial pattern from the live video is determined by calculating the dissimilarity between the current histogram ζ created based on the face in the live video and the one created by the facial patterns stored in the dataset ψ. This dissimilarity is represented in terms of the square of the distance $x^2(\zeta, \psi)$ [15]:

$$x^2(\zeta, \psi) = \sum_{j=0}^{l-1} \frac{(\zeta_j - \psi_j)^2}{(\zeta_j + \psi_j)} \quad (4)$$

where the feature vector is represented by "l". The psedudo code of intelligent LBPH algorithm in combination with Haar-cascade classifiers in provided next (Algorithm 1).

Algorithm 1: Pseudo Code

```
• Dataset
START
preprocessing;
Import numpy
Import opencv
Import sys
Face_cascade = Import Cascade Classifier
Cam = Initialize the Camera and start live video
Id = Enter ID for the Target
initialization;
Sample_number = 0
IF not cam:
    Print ("Error Opening Capture Device")
    Sys.exit(1)
        ENDIF
While:
    Ret, Img = Read the camera
    Img = Flip the image
    Gray = Gray scale conversion of Img
    Faces = Conversion into multi grayscale
```

```
                For (x, y, w, h) in faces:
                    Draw rectangle for (x,y) faces
                    Roi_gray = calculating region of interest for face
                      Roi_color = cropping the region of interest for face
                      Sample_number = sample_number + 1
                      Save cropped image with sample ID and sample number
                      Show the sample images saved in directory
                IF enter Q then:
                      break
                Elif Sample_number > 49 then:
                      break
            Release the Camera
            Close live video
            END
```

- Trainer

```
            START
            import os
            import opencv
            import numpy
            Import img from PIL

            Recognizer = Create LBPH Recognizer
            detector = Importing cascade classifier
            path = Setting path to dataset directory
            Def Get Images from ID path
                imagePaths = setting paths to images
                Faces = []
                IDs = []
                For imagePath in imagePaths:
                      faceImg = Open images from path
                      faceNp = Converting images into array of numbers
                      ID = Getting ID from the sample image name
                      faces = Apply detector on faceNp array

                      For (x, y, w, h) in faces:
                            Appending faceNp array to determine the size of sample images
                            Appending ID of the sample images
                Output Faces, IDs

            faces, IDs = getImagesWithId('DataSet')
            Train LBPH Recognizer for Face array
            Save the training file in a desired path
            Close all windows
            END
```

- Tracking and Recognizer

```
            START
            Import os
            Import numpy
            Import opencv
            Import sys
```

```
Import serial
Import time
Import struct
Import urllib3

Print “Connecting to Arduino”
arduino = Set serial port to general COM with baudrate 9600
Sleep time for 3 s
Print message “Connection Established Successfully”
recognizer = Applying LBPH Recognizer
Read the Training file stored in directory
cascadePath = Import face cascades
faceCascade = Apply classifier to cascadePath
font = Setup Hershey_simplex font
cam = Start live video from Cam 1
IF not cam:
    Print message “Error Opening Capture Device”
    sys.exit(1)

while True:
    ret, img = Read from Camera
    img = Flip the images taken from live video
    gray = Conversion of colored pic to gray scale
    faces = Setting up Multiscale gray scale
    For (x, y, w, h) in faces:
        Draw rectangle around face in live video
        Id, confidence = Identification of person from ID through dataset
        IF (Id == 1):
        Id = Extract the name from ID and display in live video
            x1 = Calculate initial horizontal coordinate of face position
            y1 = Calculate initial vertical coordinate of face position
            x2 = Calculate horizontal angle of face position
            y2 = Calculate vertical angle of face position
            Send Center coordinates to Arduino
            Print message “Id =”
            Print (Id)
            Print message “Horizontal Angle =”
            Print (x1)
            Print message “Vertical Angle =”
            Print (y1)
        ELSE:
            Id = Display Unknown Person
        Draw rectangle around face
        Write text below rectangle
    Show images from live video
    k = Wait 30 s and check the entry key
    If k == 27 then
        Break
Release Camera
Close all windows
END
```

2.3. Developed GUI

GUI interface of the robot has 5 buttons, i.e., Forward, Left, Right, Back, and Stop, to control the direction of the robot, and there are two other buttons. One button starts the video streaming and the facial recognition, and the 2nd is for laser targeting. Graphical interface can be seen below in Figure 15.

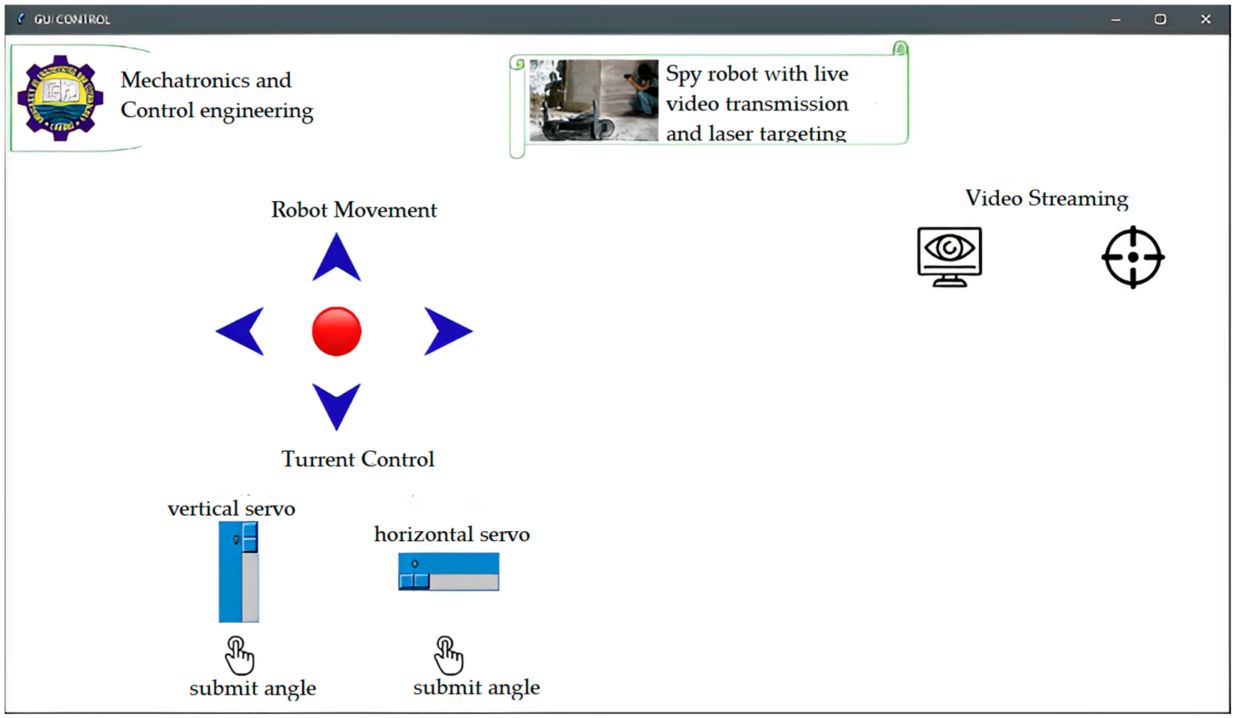


Figure 15. Python-based GUI.

There are two sliders, one is used to control the horizontal position of the turret, and the 2nd slider is used to control the vertical position of the turret manually; while in facial recognition mode, the turret automatically tracks the assigned target and changes its position according to the frame of reference.

2.4. Radio Controller

The control systems [25–27] are necessary to achieve the desired response. The designed radio controller for the robot operates on a 2.4 GHz frequency. The schemetic of radio controller is shown in Figure 16 and PCB layout of this controller is shown in Figure 17. The data transmission rate varies from 200 Kb/s–2 Mb/s, which affects the range of control. In total, 7 channels can transmit the data at the same time. However, these channels are limited due to microcontroller pins.

In this radio controller, almost 32 channels can be controlled simultaneously. In total, 7 tracks of the radio controller include 5 analog channels, of which 2 joysticks have 4 analog channels, two analog channels control robot direction, and the other two control turret position. There are 2 digital channels. One of them is used to fire the laser gun, while the rest of one digital channel, and the analog channel, is spared for more features to be added. Schematic, as well as the PCBA of the radio controller, has been designed in Eagle CAD platform. The microcontroller used in the radio controller is Atmega328p; it has an 8-bit processing speed and consists of multiple digital and analog input pins. The analog output can also be generated using pulse width modulation in programming. Programming of the radio controller has been done in the Arduino platform. Hardware assembly of the radio controller is shown in Figure 18.

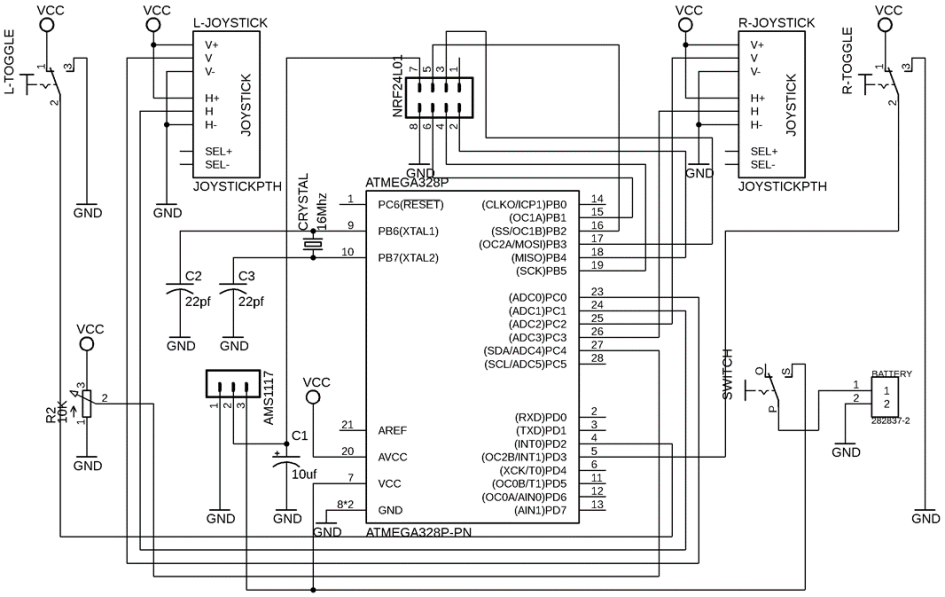


Figure 16. Schematic view of Radio Controller.

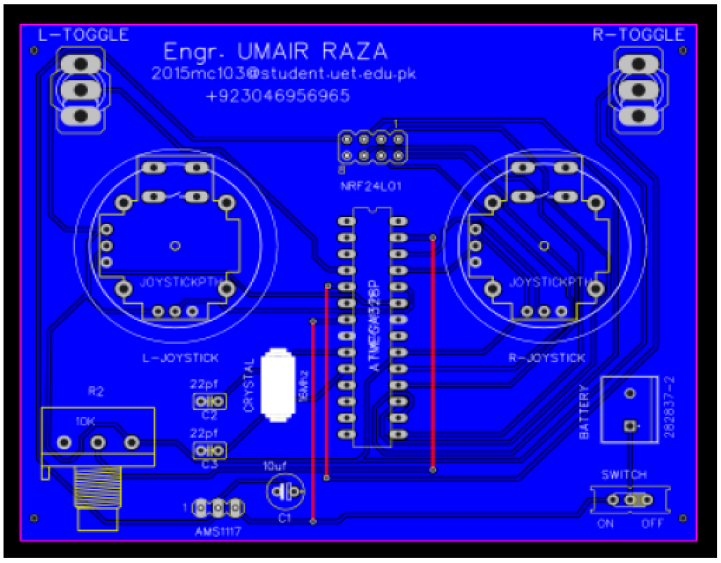


Figure 17. PCB Layout of Radio Controller.

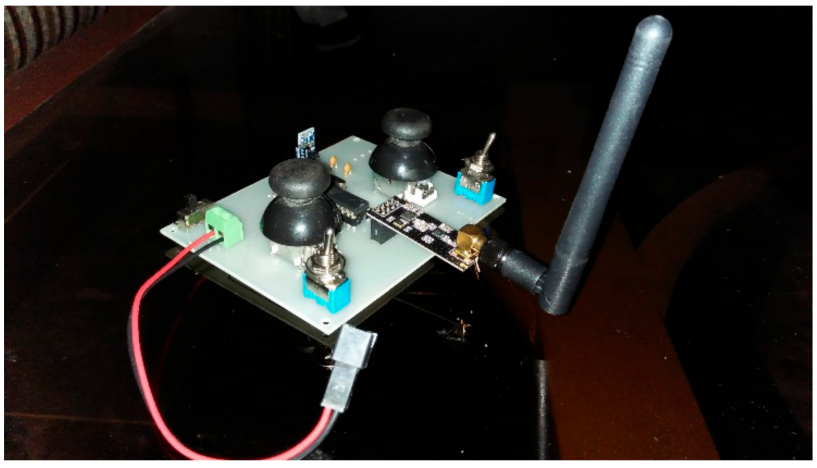


Figure 18. Hardware Assembly of Radio Controller.

3. Results and Discussion

Concerning RF communication, it is concluded that this UGV can be controlled wirelessly over a range of 1100 m with a clear line of sight, maintaining a data transfer rate of 200 Kb/s–2 Mb/s; if the line of sight is not cleared, then its range of operation is limited to 700–800 m. The air data rates for the NRF24L01+ that we used range from 250 Kbps to 2 MB/s. A 250 Kb/s data rate was chosen since it uses less power and has a greater range of more than 1 km, extending the battery life. Even though other wireless transmitters do not have many independent, bi-directional channels, the NRF24L01+ was chosen as the preferred option because it has 32 channels. As far as facial recognition is concerned, it takes 2–3 s to detect the target data already stored in our database.

There is an additional feature of making a dataset of an enemy target right on the spot and storing its images in our cloud system located in headquarters. It is helpful if a liable threat is detected during live video transmission. The limitation of facial localization is that it cannot track the movement of multiple targets and send the coordinates to our mobile robot. Facial recognition in live video transmission can be seen below in Figure 19. Figure 19 shows hundred percent accuracy in a real-time environment. If the sample photographs and the real-time were identical, we could attain 100%, which is unrealistic because one person can appear in different ways; for example, the beard is heavy in the sample, but not in real-time. In such circumstances, the similarity factor would vary. In conclusion, we achieved 99.8% accuracy in our experiment.

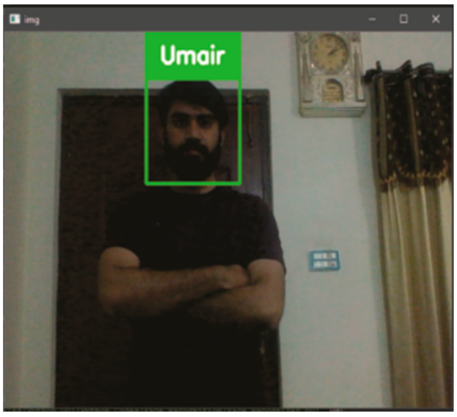


Figure 19. Facial Recognition.

When comparing the results in accordance with the datasheet, data packet loss at each stage and the distance-power curve for radio transmitters were taken into account. Higher power usage was also evident in the range comparison. In conclusion, we achieved improved facial recognition accuracy even with low resolution due to high sampling in the database, stable long range of more than 1 km with almost negligible data loss, and an enormous number of independent bi-directional communication channels.

4. Conclusions

This paper describes the complete model of an intelligent spy robot in detail. The robot comes with a receiver and transmission circuit based on 2.4 GHz bi-directional wireless communication and a high-definition camera, a source of live video transmission streaming over a Python server. All datasets can also be captured and maintained on a cloud server in headquarters. An intelligent machine learning algorithm is implemented on the live video stream, and the robot is controlled over both Python-based GUI and the remote controller in two modes in which the turret can be used automatically based on facial recognition or manually in case of potential risks which are not stored in the dataset.

In future work, one can add a 360° screen around the rover with cameras that will take images from the environment and display them on the screen to be hidden from drones and serve well in camouflage. This will allow us to implement a camouflage technique

in the mobile robot, using an instant cooling mechanism to avoid detection in IR cameras. Furthermore, as IoT (Internet of Things) technology provides real-time solutions [24], the proposed robots with GPS units attached may get geolocation data from the internet using GPS coordinates, as well as Google Maps photographs of its surroundings, which can then be stored in its database and used for camouflage. Moreover, IoT servers can be used because the database will be cloud-based and help the robot save its memory.

Author Contributions: Methodology, M.Z.U.R., U.R., A.H.G. and N.A.; Software, M.A.A.; Validation, M.Z.U.R., M.T.R. and A.H.G.; Formal analysis, M.Z.U.R. and U.R.; Investigation, A.H.G.; Data curation, M.T.R.; Project administration, M.A.A. and N.A. All authors have read and agreed to the published version of the manuscript.

Funding: This study is supported via funding from Prince Sattam bin Abdulaziz University project number (PSAU/2023/R/1444).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data is contained within the article. Any additional supporting guide can be provided upon request to the authors.

Acknowledgments: This study is supported via funding from Prince Sattam bin Abdulaziz University project number (PSAU/2023/R/1444).

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

Abbreviations

Abbreviation	Definition
DC	Direct Current
GPS	Global Positioning System
IoT	Internet of Things
GHz	Giga Hertz
RF	Radio Frequency
UGV	Unmanned Ground Vehicle
PCBA	Printed Circuit Board Assembly
CAD	Computer-Aided Design
Kb/s	Kilo-bits per second
Mb/s	Mega-bits per second
LBP	Local Binary Pattern
SPI	Serial Peripheral Interface
GUI	Graphical User Interface
PC	Personal Computer
LBPH	Local Binary Pattern Histogram
NRF	National Retail Federation
CPU	Central Processing Unit
ID	Identity
LED	Light Emitting Diode
CE	Clock Enable
DOF	Degree of Freedom

References

1. Ghouse, Z.; Hiwrale, N.; Ranjan, N. Military Robot for Reconnaissance and Surveillance using Image Processing. *Int. Res. J. Eng. Technol.* **2017**, *4*, 1767–1769.
2. Patoliya, J.; Mehta, H.; Patel, H. Arduino controlled war field spy robot using night vision wireless camera and Android application. In Proceedings of the 2015 5th Nirma University International Conference on Engineering (NUiCONE), Ahmedabad, India, 26–28 November 2016.

3. Sharma, D.; Chauhan, U. War Spying Robot with Wireless Night Vision Camera. In Proceedings of the 2020 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), Greater Noida, India, 18–19 December 2020; pp. 550–555.
4. Khurshid, J.; Bing-rong, H. A Glimpse from Today and Tomorrow. In Proceedings of the ICARCV 2004 8th Control, Automation, Robotics and Vision Conference, Kunming, China, 6–9 December 2004; pp. 771–777.
5. Version, I.; Jain, P.; Firke, P.N.; Kapadnis, K.N.; Patil, T.S.; Rode, S. RF Based Spy Robot. *Int. J. Eng. Res. Appl.* **2014**, *4*, 6–9.
6. Conbere, M.; Dodds, Z. Toys and Tools: Accessible Robotics via Laptop Computers—iRobot' s Create: Indoor Autonomy. pp. 3–7. 2007. Available online: http://robotics.cs.brown.edu/aaai07/materials/hmc_paper.pdf (accessed on 1 November 2022).
7. Liedblad, B.M.; Thiemann, H.; Liedblad, B.M.; Thiemann, H.; John, R. Modified polynucleotides for the production of oncology-related proteins and peptides. U.S. Patent 9,587,003 B2, 7 March 2017.
8. Popescu, D.; Ichim, L.; Stoican, F. Unmanned aerial vehicle systems for remote estimation of flooded areas based on complex image processing. *Sensors* **2017**, *17*, 446. [CrossRef] [PubMed]
9. Lloret, J.; Bosch, I.; Sendra, S.; Serrano, A. A wireless sensor network for vineyard monitoring that uses image processing. *Sensors* **2011**, *11*, 6165–6196. [CrossRef] [PubMed]
10. Khode, S.K.; Jaiswal, S.R.; Patil, A.J.; Neve, T.S.; Joshi, S.A. Image Processing Based Enemy Scan and Shoot. *IJESC* **2017**, *7*, 12855–12860.
11. Bhat, S.; Meenakshi, M. Vision Based Robotic System for Military Applications—Design and Real Time Validation. In Proceedings of the 2014 Fifth International Conference on Signal and Image Processing, Bangalore, India, 8–10 January 2014; pp. 20–25.
12. Kaushal, H.; Kaddoum, G. Applications of Lasers for Tactical Military Operations. *IEEE Access* **2017**, *5*, 20736–20753. [CrossRef]
13. Roso, N.A.; da Moreira, R.C.; Oliveira, J.E.B. High power laser weapons and operational implications. *J. Aerosp. Technol. Manag.* **2014**, *6*, 231–236. [CrossRef]
14. Kalisky, Y.; Kalisky, O. Applications and performance of high power lasers and in the battlefield. *Opt. Mater.* **2011**, *34*, 457–460. [CrossRef]
15. Hadid, A.; Pietikäinen, M. Combining appearance and motion for face and gender recognition from videos. *Pattern Recognit.* **2009**, *42*, 2818–2827. [CrossRef]
16. Li, W.; Siddique, A.A. Facial recognition system using LBPH face recognizer for anti-theft and surveillance application based on drone technology. *Meas. Control.* **2020**, *53*, 1070–1077.
17. Shanthi, K.G.; Vidhya, S.S.; Vishakha, K.; Subiksha, S.; Srija, K.K.; Mamtha, R.S. Haar Cascade Face Detection and Local Binary Pattern Histogram Face Recognition based Drone. In Proceedings of the 2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 25–26 March 2022; IEEE: Piscataway, NJ, USA, 2022; Volume 1.
18. Chao, C.-T.; Chung, M.-H.; Chiou, J.-S.; Wang, C.-J. A simple interface for 3D position estimation of a mobile robot with single camera. *Sensors* **2016**, *16*, 435. [CrossRef] [PubMed]
19. Ouadiay, F.Z.; Bouftaih, H.; Bouyakhf, E.H.; Himmi, M.M. Simultaneous object detection and localization using convolutional neural networks. In Proceedings of the 2018 International Conference on Intelligent Systems and Computer Vision (ISCV), Fez, Morocco, 2–4 April 2018; Volume 2018, pp. 1–8.
20. Sun, B.; Wei, J.; Tang, X. The Optimization of Object Detection and Localization in Complex Background for Vision-based Robot. In Proceedings of the 2020 IEEE International Conference on Integrated Circuits, Technologies and Applications (ICTA), Nanjing, China, 23–25 November 2020; pp. 176–177.
21. Harzallah, H.; Jurie, F.; Schmid, C. Combining efficient object localization and image classification. In Proceedings of the 2009 IEEE 12th International Conference on Computer Vision, Kyoto, Japan, 29 September–2 October 2009; pp. 237–244.
22. Keerthi Vaasan, S.V.; FaizaL Khan, A.; Gowthaman, N.; Kavin, L. Design of Reciprocating Rack and Pinion Mechanism Dual Working Mechanism. In Proceedings of the Recent Advancements in Mechanical Engineering—RAME'18, Delhi, India, 14–15 October 2016; pp. 416–420.
23. Ur Rahman, Z.; Usman, M.; Farea, A.; Ahmad, N.; Mahmood, I.; Imran, M. Vision-Based Mobile Manipulator for Handling and Transportation of Supermarket Products. *Math. Probl. Eng.* **2022**, *2022*, 3883845. [CrossRef]
24. Sailaja, M.; Roy, M.R.; Kumar, S.P. Design of Rack and Pinion Mechanism for Power Generation at Speed Breakers. *Int. J. Eng. Trends Technol.* **2015**, *22*, 356–362. [CrossRef]
25. Rahman, M.Z.U.; Liaquat, R.; Rizwan, M.; Martin-Barreiro, C.; Leiva, V. A robust controller of a reactor electromicrobial system based on a structured fractional transformation for renewable energy. *Fractal Fract.* **2022**, *6*, 736. [CrossRef]
26. Ur Rahman, M.Z.; Riaz, M.T.; Al Mahmud, M.M.S.; Rizwan, M.; Choudhry, M.A. The prescribed fixed structure intelligent robust control of an electrohydraulic servo system. *Math. Probl. Eng.* **2022**, *2022*, 5144602. [CrossRef]
27. Ur Rahman, M.Z.; Raza, A.H.; AlSanad, A.A.; Akbar, M.A.; Liaquat, R.; Riaz, M.T.; AlSuwaidan, L.; Al-Alshaikh, H.A.; Alsagri, H.A. Real-time artificial intelligence based health monitoring, diagnosing and environmental control system for COVID-19 patients. *Math. Biosci. Eng.* **2022**, *19*, 7586–7605. [CrossRef] [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.