Bulletin of the *Transilvania* University of Braşov • Vol. 16(65) No. 2 – 2023
Series I: Engineering Sciences
https://doi.org/10.31926/but.ens.2023.16.65.2.4

ADVANCED SURVEILLING DRONE: DESIGN, IMPLEMENTATION AND PERFORMANCE

C. PREDA[1] R. M. BLEOTU[2]

Abstract: *The usage of unmanned aerial vehicles (UAVs) has revolutionized various fields, including surveillance.*
This paper presents overall research on the development of an advanced surveillance drone system, focusing on its design, implementation, and performance in real-world applications. The research highlights important areas, including facial detection and recognition, communication infrastructure, and the integration of software and hardware components. The paper also investigates the suite of sensors and the software architecture that powers the drone, including the use of advanced algorithms for data processing.

Key words: *face identification, data processing, algorithms, face recognition, sensor integration, UAV.*

1. Introduction

Unmanned aerial vehicles (UAVs), commonly known as drones, have been created as mobile platforms for various applications ranging from aerial photography to agriculture, disaster management, and surveillance [3], [5]. The ability of drones to fly autonomously or be remotely piloted makes them invaluable tools for tasks that are either too dangerous or impractical for human operators [12]. In particular, surveillance drones offer unique advantages in monitoring and reconnaissance, providing valuable insights across diverse domains including security, law enforcement, and environmental monitoring.

Despite their potential, traditional surveillance drones have limitations in terms of range, endurance, and data processing capabilities [5]. To address these limitations, other authors propose an advanced surveilling drone system that integrates cutting-edge technologies in drone hardware, sensor fusion, communication protocols, and data processing algorithms.

In this paper, we provide an overview of the system design, implementation details, and performance evaluation.

[1] Department of Industrial Machinery and Equipment, Faculty of Engineering, Lucian Blaga University of Sibiu, Romania, cosmin.preda@ulbsibiu.ro
[2] Department of Industrial Machinery and Equipment, Faculty of Engineering, Lucian Blaga University of Sibiu, Romania

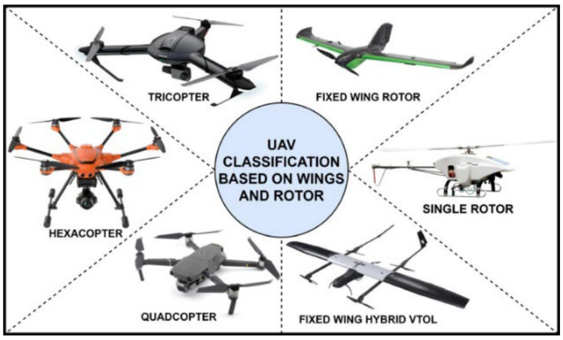

Fig.1. *UAV classification and types*

Traditional surveillance methods often rely on stationary cameras, manned aircraft, or ground patrols, each of which presents limitations in terms of coverage, accessibility, and operational costs [4], [6]. In contrast, drones offer a dynamic and flexible platform for surveillance, capable of rapid deployment and adaptation to changing circumstances, the drone system prioritizes capturing the frontal view of a person, as it provides the most accurate facial recognition results. The research addresses traditional surveillance limitations by enhancing the drone's capability to adjust its angle and position dynamically, ensuring that, when possible, the drone captures the front face of the subject [16].

The structure of the paper is as follows:

1. Introduction: an overview of the challenges and advancements in drone-based surveillance, describing the importance of facial recognition and the system's ability to adapt for optimal capture.
2. Drone platform and specifications: a detailed examination of the drone's hardware, including sensors, cameras, and motion control systems that allow it to adjust its position for better facial recognition, with an emphasis on its ability to track and capture the front face of a subject.
3. Software algorithms for multi-drone control: a description of the algorithms used to control multiple drones.
4. Facial detection and recognition: an analysis of the facial recognition technology used, discussing the system's adaptive capabilities to reposition the drone for frontal face capture, along with the technical and environmental factors that influence recognition accuracy and the algorithms created to achieve it.
5. Communication and security: an overview of the secure communication infrastructure, ensuring that the adaptive drone positioning and data transmission are protected from interference or hacking.

This paper will present an advanced surveilling drone system that which addresses the existing limits for traditional drone surveillance methods, for public use, by integrating a suite of sensors and equipment, which bring an improvement to the drone in various conditions and environments. The hardware and software systems detailed in this research will provide significant advancements in the following areas: search and rescue operations, border surveillance, infrastructure protection, urban surveillance, private security.

2. System architecture

Creating a diagram for a face detection system using a drone involves illustrating the components and their interactions. Below is a simple diagram outlining the key elements of such a system, presented in Figure 2.

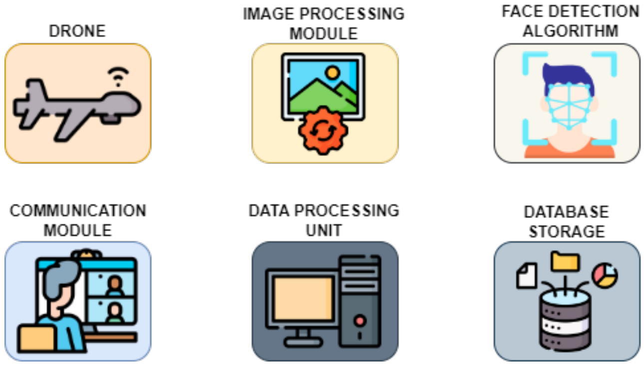

Fig.2. *Face detection diagram*

In this paper, we presented the design, implementation of an advanced surveilling drone system. The main objectives of the research are:

-to provide an overview of the system architecture, including the hardware components, sensor suite, communication infrastructure, and software algorithms.

-to describe the implementation details of the advanced surveilling drone system, including the integration of off-the-shelf components and custom-built software modules.

-to develop a system capable to sustain a safe surveillance for a drone system.

Fig.3. *Surveillance drone*

The advanced surveilling drone system comprises several interconnected components designed to work seamlessly together to achieve efficient surveillance capabilities. Figure 3 illustrates the assembly of the drone with auxiliary equipment.

2.1. Hardware components

The drone platform serves as the primary hardware component of the system. It is equipped with a variety of sensors, including high-resolution cameras, thermal imaging cameras, LiDAR (*Light Detection and Ranging*) and microphones for audio surveillance, as presented in Figure 4.

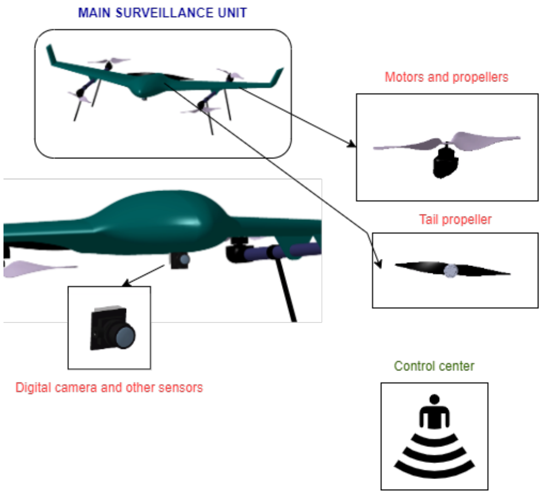

Fig.4. *UAV propulsion system and auxiliary system*

The drone's propulsion system, flight control unit, and onboard computing hardware are optimized for stability, endurance, and payload capacity and presented in the following chapters.

2.1.1. Airframe and propulsion system

The drones feature a hybrid airframe that integrates quadcopter rotors with a fixed-wing configuration, enabling both vertical take-off/landing and efficient horizontal flight [12], [16]. Key aspects of its design include:

Materials, such as lightweight and durable composite materials that provide structural integrity while minimizing overall weight, technical information are presented in Table 1.

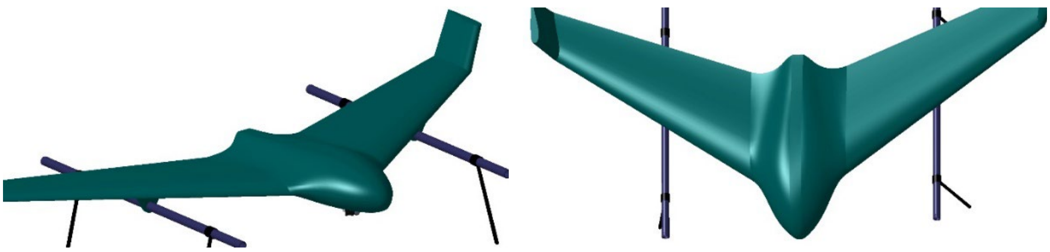

Fig.5. *Body of the drone assembly*

Aerodynamics, the streamlined fixed-wing design enhances aerodynamic efficiency, enabling longer flight durations and stable performance in various weather conditions, the body of the drone is presented in Figure 5 above [16].

Modular construction, the platform supports modular components and payloads, allowing for easy customization and maintenance. Quick-release mechanisms facilitate rapid deployment and field servicing.

Technical specification for drone Table 1

Category	Specification	Details
Dimensions and weight	wingspan [m]	2.3
	length [m]	1.3
	maximum takeoff weight [kg]	7
Performance specifications	payload capacity	up to 1.2 kilograms, depending on configuration
	maximum endurance	up to 110 minutes, depending on payload and environmental conditions
	maximum range	up to 100 kilometers on a single charge
	cruising speed [km/h]	60-90
	maximum speed [km/h]	100
	service ceiling	up to 4,500 meters above sea level

The propulsion system consists of:

Vertical lift motors, four electric brushless motors positioned in a quadcopter configuration provide vertical thrust for take-off, which are presented in Figure 6, for landing, and hover capabilities.

Fig.6. *Vertical lift motors used*

Horizontal flight motor, a rear-mounted pusher propeller driven by a high-efficiency electric motor enables forward flight, leveraging the aerodynamic lift of the fixed wings for energy-efficient cruising, as in Figure 7, which ensure the horizontal movement.

Fig.7. *Main motor for flight*

Power supply, the drone is powered by high-capacity lithium-polymer (*LiPo*) batteries, ensuring substantial endurance and reliable performance. The system includes redundant power management to maintain operational safety.

2.2. Sensor suite

The sensor suite is an important role in gathering data from the environment. The high-resolution cameras provide visual information, while the thermal imaging cameras enable detection of heat signatures, useful for identifying objects or individuals in low-light conditions. LiDAR sensors enhance the drone's ability to navigate and detect obstacles in its flight path, while microphones capture audio signals for analysis [2], [3].

Fig. 8. *Sensor suite for drone*

Electro-Optical and Infrared (*EO/IR*) camera systems are presented in Figure 8 above.

The primary surveillance capabilities are facilitated by advanced EO/IR camera systems:

- **High-resolution daylight camera**

Resolution: up to 42 megapixels, providing detailed imagery for precise analysis.

Optical zoom: integrated zoom lenses (up to 30x optical zoom) allow for close-up inspection from considerable altitudes and distances.

Stabilization: 3-axis gimbal stabilization ensures steady and clear images during flight, compensating for vibrations and movements.

Live streaming: real-time video transmission capabilities enable immediate situational awareness and decision-making.

- **Thermal imaging camera**

Resolution: high-sensitivity thermal sensors detect temperature variations for night-time operations and through obscurants like smoke and fog.

Dual-sensor integration: combined EO/IR payloads allow simultaneous capture of visual and thermal data, enhancing target detection and identification capabilities.

Temperature range detection: capable of detecting a wide range of temperatures, suitable for diverse applications such as search, rescue or monitoring of heat signatures.

- **LiDAR sensors**

The sensors are used for precise terrain mapping and obstacle detection, facilitating safe low-altitude flights and detailed environmental surveys.

2.3. Communication infrastructure

Efficient communication is essential for real-time monitoring and command/control of the drone. The system utilizes a combination of wireless communication protocols, including WI-FI, cellular networks, and satellite communication, depending on the operational environment and range requirements, as presented in Figure 9 below.

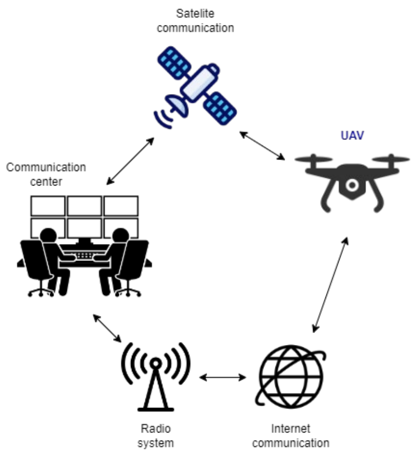

Fig.9. *Communication interface of drone*

The infrastructure typically includes ground control stations, communication links (both line-of-sight and beyond-line-of-sight), and onboard systems for data processing and transmission.

Communication links are essential for maintaining control over the drone and for data transmission. These links can be divided into:

Radio Frequency (*RF*) links, which are typically used for line-of-sight communication, RF links are reliable for short to medium distances, depending on the frequency band and environmental factors.

Beyond-Line-of-Sight (*BVLOS*) communication, for extended range operations, BVLOS communication is achieved through cellular networks, satellite links, or high-frequency RF bands. These links enable the drone to operate at great distances from the GCS, even in remote or obstructed environments.

The onboard communication systems are responsible for processing and transmitting data collected by the drone’s sensors back to the GCS, because enables real-time transmission of data, commands, and video feeds, ensuring that the operator maintains control and awareness of the drone’s operations [11], [16]. These systems include data encryption ensuring that data transmitted from the drone is securely encrypted to prevent unauthorized access or interception.

Redundant communication systems provide backup communication channels to ensure continuity in case the primary link fails.

2.4. Software algorithms

Efficient communication is essential for real-time monitoring and command/control of the drone. The system utilizes a combination of wireless communication protocols, including Wi-Fi, cellular networks, and satellite communication, depending on the operational environment and range requirements, diagram flow as in Figure 10.

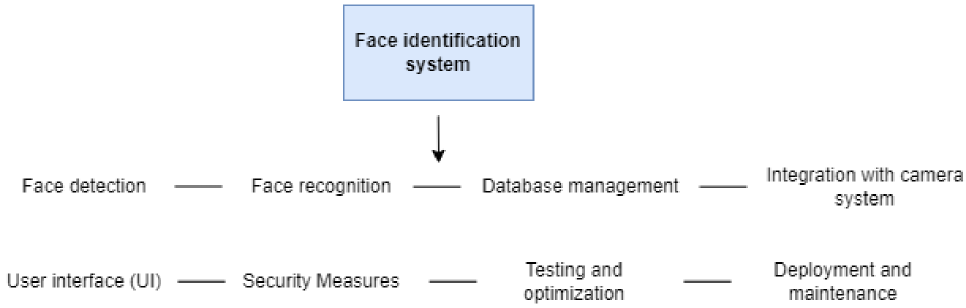

Fig.10. *Face identification mechanism steps*

Software algorithms are the key of the drone's decision-making processes. They control everything from the drone's stability during flight to the analysis of data captured by onboard sensors, such as cameras, GPS, and environmental sensors. For surveillance drones, these algorithms are optimized for real-time data processing, allowing the system to detect and recognize faces, track objects, and respond to dynamic changes in the environment.

2.5. Facial detection

Facial detection is an important step in numerous modern technologies, especially in security and surveillance applications. Unlike facial recognition, which identifies or verifies individuals, facial detection focuses on determining whether a face is present in an image or video frame [15].

While drones are already widely used in fields such as security, surveillance, and search-and-rescue missions, incorporating facial detection, the software significantly increases the scope and precision of their operations. In emergency scenarios such as natural disasters or missing persons cases, UAVs with facial detection capabilities provide an important tool for search and rescue missions [13].

Facial detection is the process of identifying human faces within a digital image or video. This technology enables systems to automatically detect the location of one or multiple faces, irrespective of the orientation, lighting conditions, or expressions. It serves as the essential first step before deeper analysis or processing, such as identification, tracking, or behavior recognition.

It plays an important role in security and surveillance systems, particularly when used with technologies like facial recognition, in large gatherings or protest situations, when drones equipped with facial detection can autonomously identify individuals within a crowd and track their movements [3], [9]. For instance, in high-security areas such as airports or military zones, surveillance cameras equipped with facial detection capabilities can monitor crowds, detect individuals of interest, or trigger alerts when faces are recognized from watchlists.

Additionally, drones equipped with facial detection can help track and locate individuals in large areas, identifying targets in search-and-rescue missions or monitoring suspicious behavior in public spaces.

2.4.2. Impact of altitude for facial detection and recognition

The effectiveness of these technologies is heavily influenced by the drone's altitude and the angle at which it observes the subject. This chapter explores the impact of altitude and angle on facial detection and recognition, providing a detailed analysis of the challenges and considerations that may occur in different operational scenarios. The frontal view is generally preferred for facial detection and recognition due to its high accuracy it is often used because most features of a person's physiognomy can be observed and it can be identified much more easily, compared to a side view, which is illustrated in Figure 11 [14], [15]. Surveillance drones equipped with pan-tilt-zoom (*PTZ*) cameras can be programmed to maintain a frontal view of the subject, improving recognition reliability.

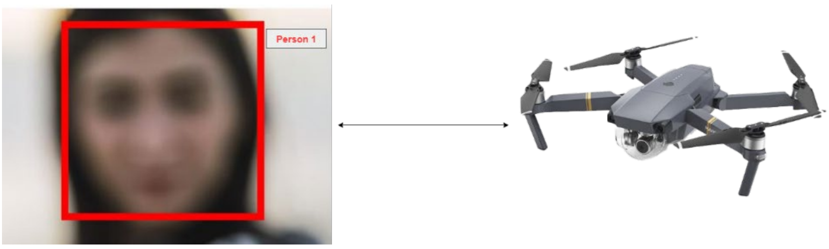

Fig.11. *Face detection using drone*

The combined effect of altitude and angle on facial recognition is complex. For example, at higher altitudes, even a slight deviation from a frontal angle can significantly reduce recognition accuracy due to the compounded loss of resolution and perspective distortion [3]. Conversely, at lower altitudes, the impact of angle may be less pronounced, but the need for constant repositioning of the drone can introduce new challenges.

An optimal altitude must balance the need for a wide field of view with the requirement for high-resolution facial images. Typically, altitudes between 50 and 100 meters offer a good compromise, providing sufficient detail for facial recognition while covering a broad area for surveillance [10].

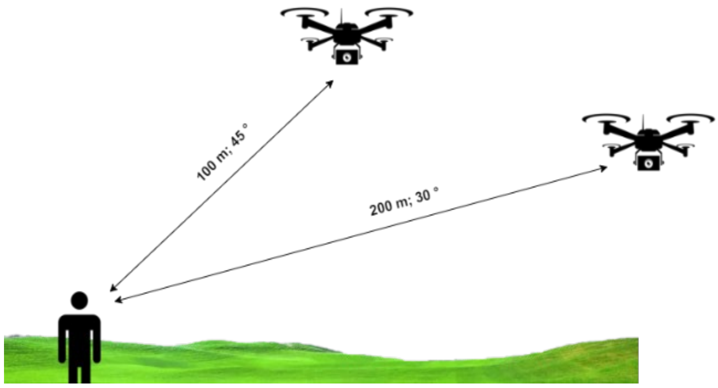

Fig.12. *Influence of the distance and angle for face detection*

At high altitudes (over 150-200m) the drone's camera captures a broader field of view but at the cost of reduced image resolution when focusing on individual faces. The greater the altitude, the more the camera must zoom in to capture detailed facial features, which can lead to pixelation and loss of critical details necessary for accurate detection and recognition, as in Figure 12 [3], [9].

Discuss algorithms like the Active Shape Model (*ASM*) or the Active Appearance Model (*AAM*) are used for landmark detection and their integration with facial detection systems.

2.5. Facial recognition

Recognition algorithms are a fundamental aspect of artificial intelligence and computer vision, enabling machines to interpret, understand, and categorize visual information in a manner similar to human perception [3], [4]. These algorithms are the key of many modern technologies, including facial recognition systems, autonomous vehicles, security surveillance, and augmented reality.

Recognition algorithms have a wide range of applications across various industries:

Security and surveillance: facial recognition systems are used in security checks, while object detection algorithms are vital for monitoring and identifying potential threats in real-time.

Healthcare: medical imaging uses recognition algorithms to detect abnormalities in X-rays, MRIs, and other diagnostic images.

Autonomous vehicles: self-driving cars rely on recognition algorithms to detect pedestrians, road signs, and other vehicles.

Retail: Recognition technology is used in retail for customer analytics, automated checkouts, and inventory management.

Augmented Reality (*AR*): recognition algorithms identify objects and surfaces, allowing virtual objects to interact with the real world seamlessly.

2.5.1. Template matching algorithms and computational offloading

Template matching algorithms identify objects by comparing input images with stored templates or patterns.

The algorithm slides the template across the input image and calculates a similarity measure at each position. The position with the highest similarity score is considered a match. This method is straightforward but can be computationally expensive for large images or templates.

Offloading requires a fast and reliable communication link between the drone and the external resource. Latency and packet loss during communication can affect the performance of offloaded tasks.

The decision to offload computations from a drone can be modeled mathematically to optimize energy consumption and task completion time.

T_{local} is the time taken to execute the task locally on the drone

$T_{offload}$ is the time taken to offload the task to an external server, including both transmission and execution time

E_{local} is the energy consumed by the drone to execute the task locally

$E_{offload}$ is the energy consumed for offloading the task to the external server.

Local execution time

$$T_{local} = \frac{C_{task}}{f_{drone}} \tag{1}$$

where,

C_{task} is the total computational complexity of the task in CPU cycles

f_{drone} is the CPU processing speed of the drone

Offloading time

$$T_{offload} = \frac{D_{task}}{B_{comm}} + \frac{C_{task}}{f_{server}} \tag{2}$$

where,

D_{task} is the size of the data to be offloaded

B_{comm} is the communication bandwidth between the drone and the external server

f_{server} is the CPU processing speed of the external server or cloud infrastructure.

Energy model

$$E_{local} = P_{drone} \times T_{local} \tag{3}$$

where,

P_{drone}- is the power consumption of the drone's CPU during task execution

$$E_{offload} = P_{comm} \times \frac{D_{task}}{B_{comm}} + P_{server} \times \frac{C_{task}}{f_{server}} \tag{4}$$

where,

P_{comm} is the power consumption for communication during offloading

P_{server} is the power consumption of the external server.

The decision to offload occurs when the time and energy consumed by offloading are less than the local execution.

$$T_{offload} < T_{local} \tag{5}$$

2.6. 3D facial modelling

3D facial modeling is the process of creating a three-dimensional digital representation of a human face. Unlike 2D facial recognition, which relies on flat images, 3D modeling captures the depth, contours, and structural details of a face, resulting in a more accurate and detailed representation. This technology is crucial for applications that require high precision and robustness, such as biometric identification, virtual reality, gaming, and facial recognition systems used in security and law enforcement.

Explain the advantages of 3D facial modeling in overcoming the limitations of 2D recognition systems. Discuss the process of creating and utilizing 3D models from multiple 2D images to enhance recognition accuracy, especially when dealing with varied angles and lighting conditions.

Photogrammetry is a technique that uses multiple 2D images taken from different angles to reconstruct a 3D model of a face.

The process involves capturing a series of photographs around the subject's face. Software then analyzes these images, identifying common points and features across different angles, and triangulates the data to create a 3D model.

2.6.1. Challenges of facial recognition

Discuss how natural variations in facial features over time, expressions, and accessories (e.g., hats, glasses, masks) pose challenges to accurate recognition. Explain how algorithms account for these variations to maintain high identification rates, a key factor.

Study how low-resolution or blurry images impair recognition performance. Discuss the importance of high-resolution sensors and image enhancement techniques that improve the quality of input data for recognition algorithms.

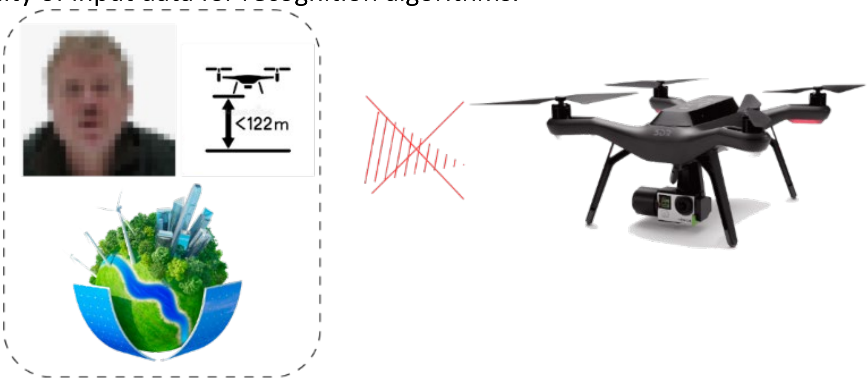

Fig.13. *Different factors that influence the face recognition*

Investigate how environmental factors like smoke, dust, and vegetation obstruct facial recognition processes, factors presented in Figure 13 above. Discuss mitigation strategies, such as multi-sensor fusion and robust algorithm design, to enhance recognition reliability in such conditions.

3. Implementation details

The implementation of the advanced surveilling drone system involves integrating off-the-shelf components with custom-built software modules. The drone platform is based on a commercial UAV model, modified to accommodate additional sensors and computing hardware. The software stack includes firmware for flight control, drivers for sensor integration, and application-level software for data processing and mission execution. Communication protocols are implemented using standard networking libraries, with provisions for encryption and secure transmission of data.

In order to achieve facial recognition for people, a mechanism was created that involves going through several steps, which are presented in Algorithm 1 below.

The software part was made using *Visual Studio Code* program, and the code was created to allow the video camera mounted on the surveillance drone to detect people, for the purpose for which it was made.

Algorithm 1 *for face recognition program*

```
import cv2
import dlib
import numpy as np
# Load pre-trained face detection model
detector = dlib.get_frontal_face_detector()
# Load pre-trained face recognition model
shape_predictor = dlib.shape_predictor("shape_predictor_68_face_landmarks.dat")
Face_recognizer =
dlib.face_recognitor_model_v1("dlib_face_recognition_resnet_model_v1.dat")
# Load a sample image for testing
img = cv2.imread("test_image.jpg")
# Convert the image to grayscale for face detection
gray = cv2.cvColor(img, cv2.COLOR_BGR2GRAY)
# Detect faces in the grayscale image
faces = detector(gray)
# Iterate through detected faces
        for face in faces:
        # Determine facial landmarks
        shape = shape_predictor(gray, face)
        # Extract face descriptor
        face_descriptor = np.array(face_descriptor)
        # Perform face recognition (compare this descriptor to a database of know face
descriptors)
        # Print the descriptor
        print("face Descriptor:", face_descriptor)
        #Draw rectangle around face
        cv2.rectangle(img, (face.left(), face.top(), (face.right(), face.bottom()), (0, 255, 0, 2)
        # Display the image with detected faces
        cv2.imshow("Face Recognition", img)
        cv2.waitKey(0)
        cv2.closeAllWindows()
```

The following Python code presents a simplified implementation of a paired system combining face recognition with an additional camera system. Using the computer vision libraries like *OpenCV* and a hypothetical face recognition module, this system demonstrates the integration of real-time face detection and recognition capabilities with live camera feeds.

After the facial detection program was created, the next step was to create a program and code for the facial identification of the detected persons. Thus, a Python code was created which is presented in Algorithm 2; it presents a simplified implementation of face identification using the *OpenCV* and *dlib* libraries. Face recognition is an important technology in computer vision, enabling systems to identify individuals from images or video streams automatically [1, 12].

By integrating facial recognition technology into the Java interface, the system can identify individuals of interest based on pre-trained models. The drones capture live footage, process it to detect and recognize faces, and relay that information back to the ground station.

Algorithm 2 *for face identification database*

```
import sqlite3
class FaceDatabase:
        def__init__(self, db_name='face_database.db');
               self.conn = sqlite.connect(db_name)
               self.cursor = self.conn.cursor()
               self.create_table()
        def create_table(self):
                self.cursor.execute(""CREATE TABLE IF NOT EXISTS faces
(id INTEGER PRIMARY KEY, name TEXT, embedding BLOB)"")
                self.conn.commit ()
        def add_face(self, name, embedding):
                self.cursor.execute(""INSERT INTO faces (name, embedding) VALUES (1,1)"", (name,
ebmbedding))
                self.conn.commit()
        def get_face_by_name(self, name):
                self.cursor.execute(""SELECT * FROM faces WHERE name=Andrei"", (name,))
                return self.cursor.fethone()
        def get_all_faces(self):
         self.cursor.execute(""SELECT * FROM faces"")
         return self.cursor.fetchall()
       det delete_face(self, name):
        self.cursor.execute(""DELETE FROM faces where name=Andrei"", (name, ))
        self.conn.commit()
        def close(self):
        self.conn.close()
# Example usage:
if __name__ == "__Andrei__":
db = FaceDatabase()
# Add face
```

```
    db.add_face('Andrei', b'example_embedding_data')
# Get face by name
    print(db.get_face_by_name('Andrei'))
# Get all faces
    print(db.get_all_faces())
# Delete face
    db.delete_face('Andrei')
    db.close()
```

An interface was created for the integration of the components and to control the entire system of drones, which is presented in Algorithm 3 and Algorithm 4. The interface provides an individual control for each drone, with audio-video images and with the possibility of identifying people according to the existing database. It allows operators to issue commands, monitor drone status, and receive real-time updates from all drones in the system.

Algorithm 3 *for drone software management and communication*

```
Import java.util.ArrayList;
Import java.util.HashMap;
Import java.util.Map;

// Drone class to handle basic functionalities of a drone
class Drone {
     private String id;

     public Drone(String id) {
          this.id = id;
          this.isInAir = false;
     }
     public String getId() {
          return id;
     }
     public Boolean isInAir() {
          return isInAir;
     }
     public void takeOff() {
          isInAir = true;
          System.out.println("Drone" + id + "is taking off.");
     }
     public void land() {
          isInAir = false;
          System.out.println("Drone" + id + "is landing");
     }
     public void moveToCoordinate(double latitude, double longitude) {
          if (isinAir) {
     System.out.println("Drone" + id + "is moving to coordinates: " + latitude + ", " +
longitude);
          } else {
```

```
        System.out.println("Drone" + id + "is not in the air, please take of first.");
            }
        }
}
```

The key idea behind Algorithm 4 is to allow a single operator or software system to manage multiple drones through an organized communication framework. This enables drones to share data, divide tasks, avoid collisions, and execute complex missions or searches collaboratively. This multi-drone control system improves operational efficiency and allows coverage of larger areas more effectively than a single drone could achieve [8].

Algorithm 4 *for multiple drone control*

```
// Communication System to handle the control of multiple drones
        class CommunicationSystem {
            private ArrayList<Drone> drones = new ArrayList<>();
        public void addDrone(Drone drone) {
            drones.add(drone);
            System.out.println("Drone " + drone.getId() + "added to the system");
        }
        public void takeOffAll() {
            for (Drone drone : drones) {
            drone.takeOff();
        }
    }
        public void landAll() {
            for (Drone drone : drones) {
            drone.land();
        }
    }
        public void moveDroneToCoordinates(StringdroneId, double latitude, double longitude)
{
            for (Drone drone ; drones) {
                if (drone.getId().equals(droneId)) {
                    drone.moveToCoordonates(latitude, longitude);
                    return;
                }
            }
            System.out.println("Drone " + droneId + " not found");
        }
    // Facial Recognition System mock
        class FaceRecognitionSystem {
            private Map<String>, String> registeredFaces;
            public FaceRecognitionSystem() {
                registeredFaces = new HashMap<>();
// Register 3 persons in the system
registeredFaces.put(key"Person1", value:"Preda Cosmin");
registeredFaces.put(key"Person1", value:"Bleotu Robert");
registeredFaces.put(key"Person1", value:"Preda Madalin");
```

```
}
        public String recognizeFace(String personId) {
// Simulate face recognition by returning the person name based on personId
        if (registeredFaces.containKey(personId) {
```

As drone technology continues to evolve, efficient communication and control systems are critical for managing multiple drones in coordinated tasks.

This is especially important in surveillance, where drones need to gather data, perform facial recognition, and relay information back to ground stations in real-time. A software interface is important to facilitate easier communication, coordination, and task execution between multiple drones and the control system.

Algorithm 5 outlines the software interface developed to manage the control of multiple drones within a UAV system. The interface is designed to allow an operator to easily command and monitor drone operations, view real-time video feeds, and capture images during missions.

Algorithm 5 *for software interface that control the drones*

```
// Main class to bring everything together
public class DroneSystem {
Run | Debug
public static void main(String[] args) {
// Create the communication system
CommunicationSystem communicationSystem = new CommunicationSystem();
// Add 3 drones to the system
drone drone1 = new Drone(id:"Drone1");
drone drone2 = new Drone(id:"Drone 2");
drone drone3 = new Drone(id:"Drone 3");
communicationSystem.addDrone(drone1);
communicationSystem.addDrone(drone2);
communicationSystem.addDrone(drone3);
// Control the drones
communicationSystem.takeOffAll();
communicationSystem.moveDroneToCoordinates(droneIt:"Drone 1", latitude:40.7129, -
74.0060); //Coordinates for Sibiu
communicationSystem.moveDroneToCoordinates(droneId:"Drone 2", latitude:34.0522, -
118.2437); //Coordinates for Selimbar
communicationSystem.moveDroneToCoordinates(droneId:"Drone 3", latitude: 51.5032, -
0.1254); // Coordinates for Bucharest
// Facial Recognition System
FaceRecognitionSystem faceRecognitionSystem = new FaceRecognitionSystem();
// Simulate identifying 3 persons
System.out.println(x:"Recognizing faces:");
System.out.println("Drone 1 identified: " + faceRecognitionSystem.
recognizeFace(personId:"Person1")));
System.out.println("Drone 1 identified: " + faceRecognitionSystem.
recognizeFace(personId:"Person2")));
```

```
System.out.println("Drone 1 identified: " + faceRecognitionSystem.
recognizeFace(personId:"Person3")));
// Land all drones after the task
communicationSystem.landAll();
```

The interface facilitates communication between the drones and the ground control station, ensuring that commands and data flow seamlessly. In surveillance operations, for instance, drones can capture and transmit video feeds, facial recognition data, and environmental information back to the operators, an aspect highlighted in Figure 14, which present the final variant for the surveillance system.

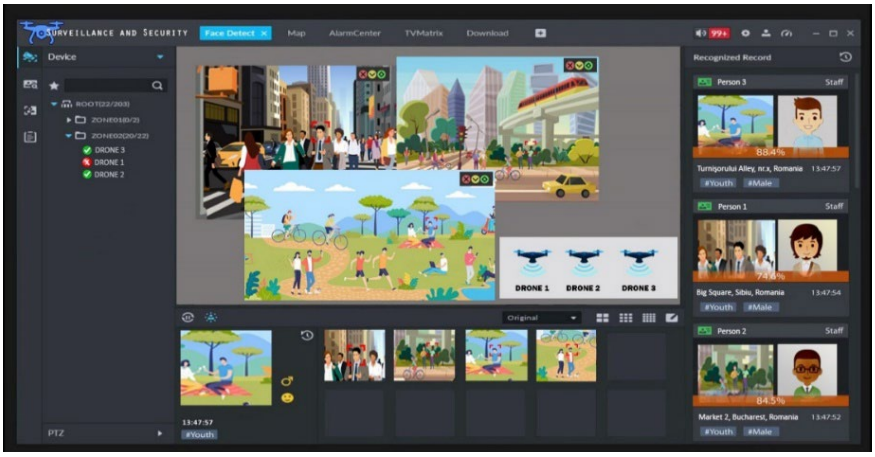

Fig.14. *Surveillance system interface created*

To obtain the performance of the advanced surveilling drone system, it is necessary to perform a series of field tests in various surveillance scenarios. The Java interface allows operators to monitor the live feed, run recognition algorithms, and receive alerts when a face matches a predefined set of individuals. This integration is particularly useful for security, where recognizing specific individuals in a crowd can lead to proactive intervention.

4. Security and considerations

Drones are susceptible to signal interception and jamming, which can disrupt operations or result in the loss of control over the drone. Key risks include also:

Interception: unauthorized parties can potentially intercept communication signals, gaining access to sensitive data or even taking control of the drone [10].

Jamming: deliberate jamming of communication signals can incapacitate the drone, forcing it to return to base or causing it to crash.

Data theft: unauthorized access to stored or transmitted data can lead to the exposure of critical information.

Data manipulation: altered data can mislead operators, leading to incorrect decisions or compromised missions.

Blockchain technology offers a good solution to enhance the security of drone communication infrastructure by providing a decentralized, tamper-proof system for data management and transmission.

Audit trails, in which blockchain provides a transparent and traceable audit trail for all data and communications [7], [13].

5. Conclusions and upcoming research

In this paper it was presented the design, implementation, and performance analysis of an advanced surveilling drone system. The system offers great capabilities for surveillance, exploring the field in drone hardware, sensor integration, communication protocols, and data processing algorithms.

The following remarks were highlighted during this research:

- Advanced design and specifications, including their dimensions, weight, and performance, is important to the operational success.
- 3D facial modelling in the research highlights the importance of this in achieving high accuracy in facial recognition tasks. By capturing the depth and contours of facial features, 3D modeling provides a more reliable means of identification, particularly in challenging conditions.
- Recognition algorithms are at the key of any surveillance system.
- Communication infrastructure and security are the key for any operation of surveillance drones.
- The research highlights the need for secure and reliable communication channels, with blockchain technology emerging as a potential solution to enhance security against hacking attempts.

The development and deployment of advanced surveillance drones are rapidly evolving, driven by technological advancements in hardware, software, and communication systems. As these systems become more sophisticated, they offer unprecedented capabilities for monitoring, identification, and data collection.

References

1. Bo, Q., Nidal, A. S., Bo, D.: New technologies for UAV navigation with real-time pattern recognition. In: Ain Shams Engineering Journal 15 (2024), 102480.
2. Fox, S. J.: The ‘risk’ of disruptive technology today (A case study of aviation – Enter the drone). In: Technology in Society 62 (2020), 101304.
3. Hsu, Hwai-Jung and Chen, Kuan-Ta: Face Recognition on drones: Issues and Limitations. In: Proceedings of MobiSys'15: The 13th Annual International Conference on Mobile Systems, Applications, and Services. The First Workshop on Micro Aerial Vehicle Networks, Systems, and Applications for Civilian Use, Florence. New York. Association for Computing Machinery, 2015, 39-44, ISBN 978-1-4503-3501-0.
4. Johnson, T. L., Johnson, N. N., McCurdy, D., Olajide, M. S.: Facial recognition systems

in policing and racial disparities in arrests. In: Government Information Quarterly 39(4) (2022), 101753.

5. Jurevičius, R., Goranin, N., Janulevicius J., Nugaras, J., Suzdalev, I., Lapušinskij, A.: Method for real time face recognition application in unmanned aerial vehicles. In: Aviation 23 (2019), p. 65-70.
6. Lee, H., Kim, M.-U., Kim, Y., Lyu, H., Yang, H. J.: Development of a Privacy-Preservings UAV System with Deep Learning-Based Face Anonymization. In: IEEE Access 9 (2021), 132652-162662.
7. Muhammad Ahmad Nawaz Ul Ghani, Kun She, Muhammad Arslan Rauf, Masoud Alajmi, Yazeed Yasin Ghadi, Abdulmohsen Algarni: Securing synthetic faces: A GAN-blockchain approach to privacy-enhanced facial recognition. In: Journal of King Saud University – Computer and Information Sciences 36(4) (2024), 102036.
8. Mohsen, R., Amirhamzeh, F., Hashem, P.: Deep learning-based face detection and recognition on drones. In: Journal of Ambient Intelligence and Humanized Computing 15 (2024), p. 373-387.
9. Shanthi, K.G., Sivalakshmi, P., Sesha, V.S., Sangeetha, L.K.: Smart drone with real time face recognition. In: Materials Today: Proceedings 80(3) (2023), p. 3212-3215.
10. Tri, N., Risto, K., Tuan, N.G.: An advanced Internet-of-Drones System with blockchain for improving quality of service of search and rescue: A feasibility study. In: Future Generation Computer System 140 (2023), p. 36-52.
11. Villegas, I. D., Camargo, J. R., Perdomo Ch., C. A.: Recognition and Characteristics EEG Signals for Flight Control of a Drone. In: IFAC PapersOnLine 54(4) (2021), 50-55.
12. Xiaoyan Guo, Jie Yang, Xinyu Jia, Chuanyan Zang, Yan Xu & Zhaoyang Chen: A novel dual-pooling attention module for UAV vehicle re-identification. In: Scientific Reports 14 (2024), 2027.
13. Yaacoub, J. P., Hassan, N., Ola, S., Ali, C.: Security analysis of drones systems: Attacks, limitations, and recommendations. In: Internet of Things 11 (2020), p. 1-39.
14. Yanmaz, E., Yahyanejad, S., Rinner, B., Hellwagner, H., Bettstetter, C.: Drone networks: Communications, coordination and sensing. In: Ad Hoc Networks 68 (2018), 1-15.
15. Zandieh, F., Ghannadpour, S. F., Mazdeh, M. M.: Integrated ground vehicle and drone routing with simultaneous surveillance coverage for evading intentional disruption. In: Transportation Research Part E: Logistics and Transportation Review 178 (2023), 103266.
16. Zandieh, F., Ghannadpour, S. F., Mazdeh, M. M.: New integrated routing and surveillance model with drones and changing station considerations. In: European Journal of Operation Research 313 (2024), 527-547.