

Development of a Facial Recognition System for Suspects for Integration into a Video Surveillance Security System with Direct Alerts Using a Mavic 2 Pro Drone: Case of Burkina Faso

Yanogo Kiswendsida Jean Hermann^{1*}, Djieni Abdoul Aziz Guiyaassou², Kahoun Zita Philippe², Tounwendyam Frédéric Ouedraogo³

¹Institute of Computer Engineering and Telecommunication Polytechnic, Ouagadougou, Burkina Faso

²Faculty of Science and Engineering, Aube Nouvelle University, Ouagadougou, Burkina Faso

³Faculty of Science and Engineering, Norbert Zongo University, Koudougou, Burkina Faso

Email: *yanogohermann27@gmail.com

How to cite this paper: Hermann, Y.K.J., Guiyaassou, D.A.A., Philippe, K.Z. and Ouedraogo, T.F. (2026) Development of a Facial Recognition System for Suspects for Integration into a Video Surveillance Security System with Direct Alerts Using a Mavic 2 Pro Drone: Case of Burkina Faso. *Engineering*, 18, 356-366.
<https://doi.org/10.4236/eng.2026.189020>

Received: April 25, 2026

Accepted: September 20, 2026

Published: September 23, 2026

Copyright © 2026 by author(s) and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

This research presents a facial recognition system based on cameras installed in public spaces in Burkina Faso, which sends real-time alerts with the location and time of detection of the suspect. The objective of this research is to develop a security system solution capable of helping Burkina Faso combat terrorism. The system is developed and tested using a drone operating at medium altitude to simulate its operation before integration with the cameras. The system uses ArcFace, an open-source model compatible with facial differentiation constraints. The results achieved a detection rate of 92% and an accuracy of 90% under ideal conditions, while maintaining good performance in a dynamic environment. This research is important because Burkina Faso is experiencing periods of insecurity linked to terrorism. To help Burkina Faso, we are implementing this scalable solution to strengthen security and enable the monitoring of suspects, which will contribute to the strategy for combating terrorism in Burkina Faso. In this research, we design the system architecture, build a Python script and function, and test the system's effectiveness. The result shows a perfect detection of the suspect.

Keywords

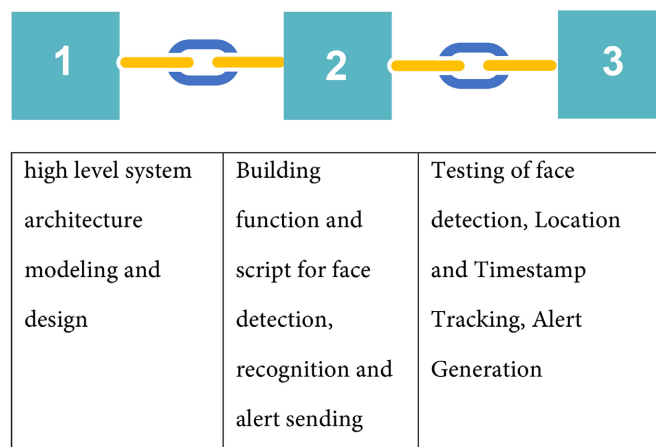
Facial Recognition, Security System, Drone, Monitoring of Suspects, Surveillance Camera

1. Introduction

Burkina Faso faces a serious security threat primarily related to terrorism, with numerous recorded deaths. Despite the ongoing efforts of the national army to restore peace, terrorists are increasingly adopting deceptive tactics, such as disguises, to infiltrate public spaces at any time, undetected. This increases the public safety stakes and underscores the need for intelligent surveillance solutions that support security forces while respecting ethical boundaries. It is therefore important to implement automated systems that provide real-time alerts with precise, timestamped location data to counter dynamic terrorist threats. This research proposes a facial recognition system for integration with video surveillance, tested with Mavic 2 Pro drones from Da-Jiang Innovations (DJI) to simulate the perspectives of roadside cameras, given restricted access to the Smart-Burkina network, thereby ensuring infrastructure monitoring. It is important to notice that with the development of drone technology, that is, unmanned aerial vehicles (UAVs) with advanced imaging capabilities, the field of aerial photography has changed dramatically. Indeed, Drones offer a much more flexible, economical, and effective answer than traditional aerial photography techniques [1].

2. Methodology Used

The methodology used in this research is the following:



The methodology is the following:

- High-level system architecture modeling and design.
- Building a function and script for face detection, recognition, and alert sending.
- Testing of face detection, location and timestamp tracking, and alert generation.

3. The Concepts of CCTV Surveillance and Facial Recognition

In the context of urban security, video surveillance is a major approach to security through monitoring and allows for many hours of surveillance. However, in Burkina Faso, identifying suspects is difficult because it currently relies on inadequate databases, and privacy issues arise. Facial recognition leverages deep learning to pro-

vide automatic identification with cutting-edge accuracy. The InsightFace framework is defined for face detection, and ArcFace for feature extraction, based on the Insight framing, is considered the most effective. The conventional drones used for demonstration in our research are suitable for the test environment, giving us the advantage of observing the system's performance at high altitude. It should be noted that Burkina Faso has already deployed video surveillance in several areas, and our research focuses on facial identification and alerts issued based on our database to detect terrorists and criminals. It is important to notice that Facial scan systems through CCTV have software solutions that process images existing in CCTV cameras and processing systems with facial recognition technology. Human face detection is in the domain of computer vision applications focused on the recognition and location of the image of a human face in the picture, regardless of the size, position, and condition [2].

4. The Importance of Privacy in CCTV Systems

Privacy is a major concern in the context of video surveillance, particularly in Burkina Faso, where public distrust of Smart Burkina cameras is evident. The ability to visually track people present in the scene is essential for any surveillance system. However, the widespread deployment and increased advancement of video surveillance systems have raised awareness of privacy to the public [3]. Privacy protection technologies, such as temporary data storage and encryption, are essential for reconciling security and individual freedoms. Unlike drones, which process data in real time, video surveillance systems often store recordings indefinitely, increasing the risk of misuse. This research proposes protocols to ensure that only suspicious data is retained, in accordance with international privacy standards, and to promote public acceptance.

5. High-Level System Architecture Modeling and Design

Design and Modelling constitute a structured approach to planning and visually representing the architecture, behavior, and components of the system. In this research, these processes ensure that the facial recognition pipeline from data acquisition to alert generation is logically organized, modular, and aligned with operational requirements. Architecture definition, which is central to system design, is one of the two most used technical processes in the practice of model-based systems engineering [4].

5.1. Requirement Analysis

Functional requirements include real-time detection, email alert delivery, and privacy compliance. Technical requirements encompass CCTV compatibility, 1080p imaging, and secure data handling.

5.2. System Architecture

The system is composed of three main components: User Interface (UI), Logic Layer,

and Database.

Component Breakdown:

- User Interface (UI): Manages all interactions with Ministry operators, displaying real-time alerts, interactive maps, and system status.
- Logic Layer: Processes images from drones or CCTV cameras, performs facial recognition using InsightFace, and generates alerts.
- Database: Stores suspects metadata and ensures privacy-compliant data handling.

5.3. Class Diagram

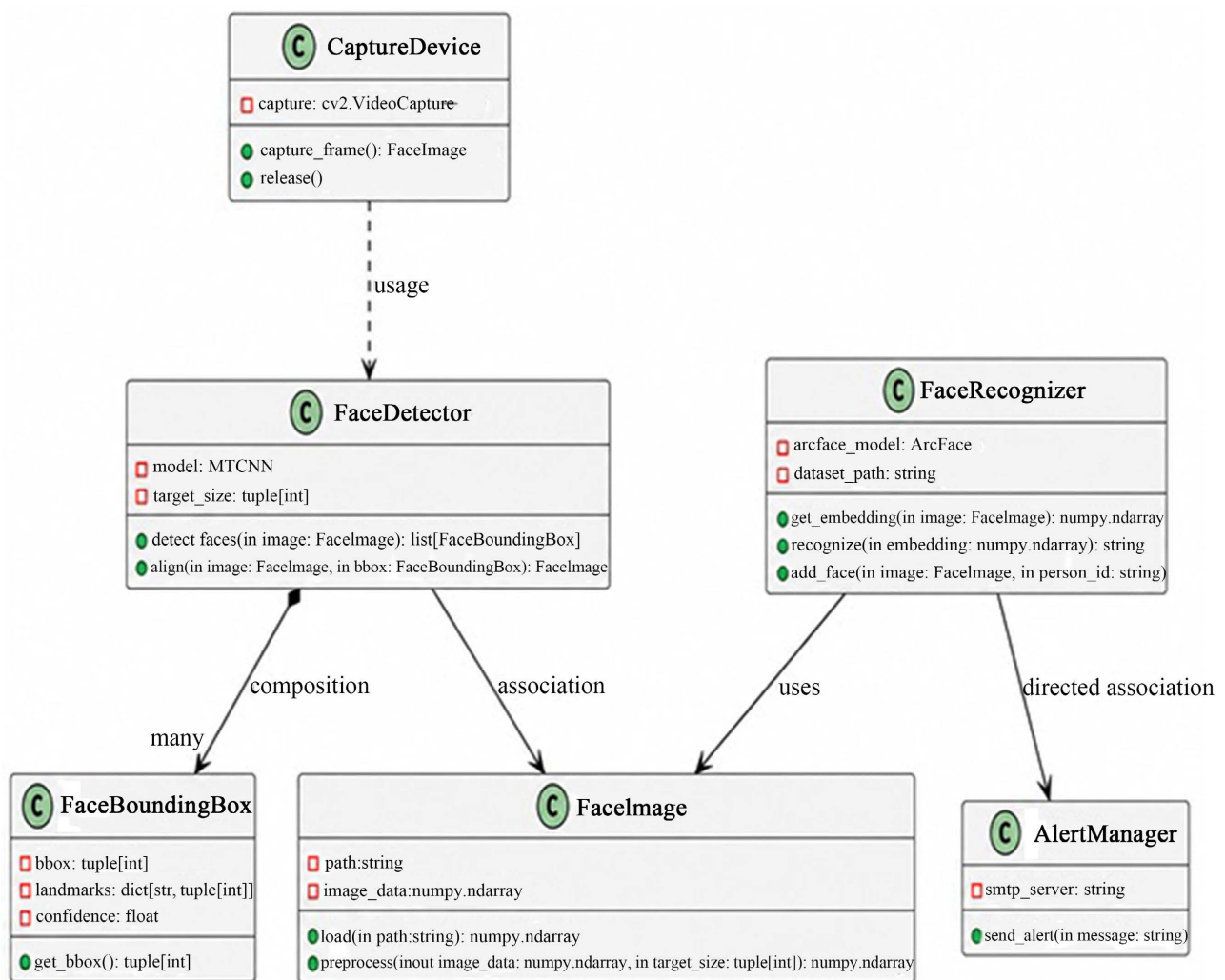


Figure 1. Class diagram.

A class diagram is a type of UML (Unified Modeling Language) diagram that depicts the static structure of a system by showing its classes, their attributes, methods, and the relationships among objects. It is widely used in object-oriented design to model the blueprint of a software system. Class Diagrams are a fundamental element in the software development process, providing an organized visual

representation of software components and their relationships. Therefore, ensuring the quality of these diagrams is important to maintain consistency, design integrity, and project success [5].

Here are the use cases of a class diagram:

- They help visualize and organize the components of the system, serving as a blueprint for implementation.
- They encourage discussions about the system's design, promoting a shared understanding among team members.
- Many development tools support code generation from class diagrams, enabling developers to create code skeletons efficiently.
- This capability minimizes manual coding efforts and helps ensure that the implementation aligns with the design.

The class diagram is the following:

Figure 1 shows the corresponding class diagram for our system. It shows the different relation, the methods, and their attributes.

5.4. Sequence Diagram

The automation of Unified Modeling Language (UML) sequence diagram generation has posed a persistent challenge in software engineering [6]. Indeed, a sequence diagram is a type of UML (Unified Modeling Language) diagram that visually represents the interactions between objects or components in a system over time. It focuses on the order and timing of messages or events exchanged between different system elements. The diagram captures how objects communicate with each other through a series of messages, providing a clear view of the sequence of operations or processes.

Figure 2 shows the sequence diagram of our system.

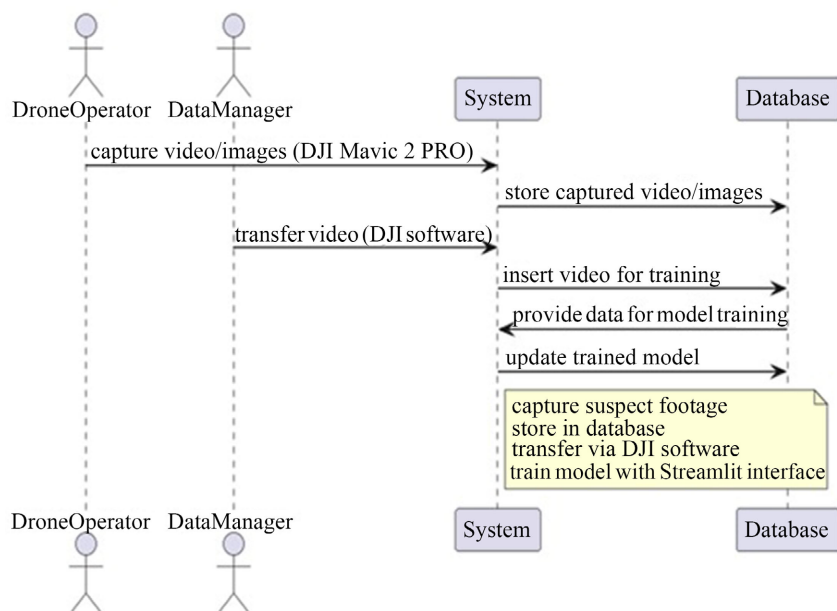


Figure 2. Sequence diagram.

5.5. Use Case Diagram

A Use Case Diagram is a behavioral diagram in the Unified Modeling Language (UML) that visually represents the interactions between users and a system through various functional scenarios. Use case diagram, an example of UML diagram, is a very powerful tool to model the requirements specified by the users while also helping the development teams understand the functionalities and interactions between users and the system [7]. It illustrates what the system is supposed to do by depicting all the external entities that interact with it and the specific services or functions the system provides to those entities. Use case diagrams help clarify the system's functionality from the user's perspective and are particularly useful during the requirements-gathering phase, serving as a clear and concise tool for communication among stakeholders, analysts, and developers. The use case diagram is one of the important models to be used to design the system requirements, which focuses on the system functionality. The design of the system is part of the System Development Life Cycle (SDLC) phase, which becomes a guideline to develop any system [8].

Figure 3 shows the use case diagram of our system.

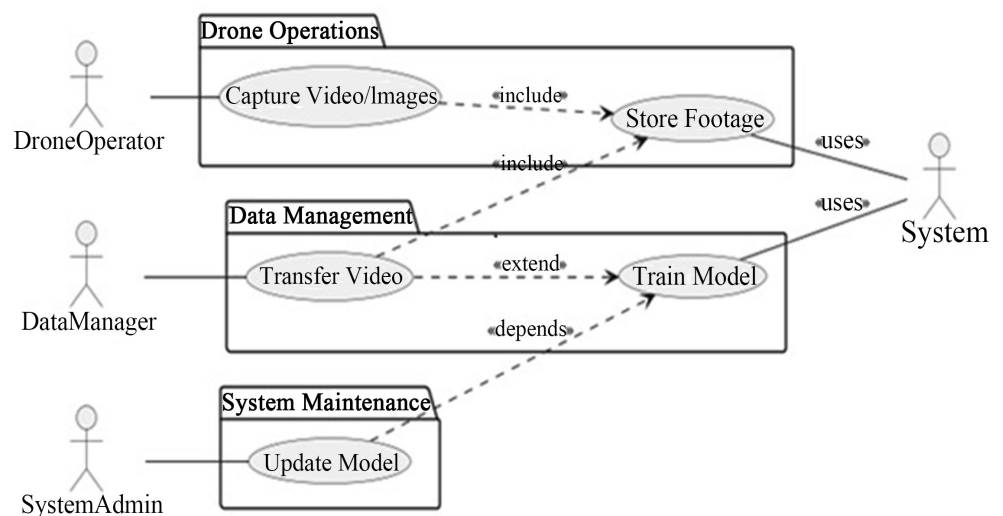


Figure 3. Use case diagram.

5.6. Conceptual Approach

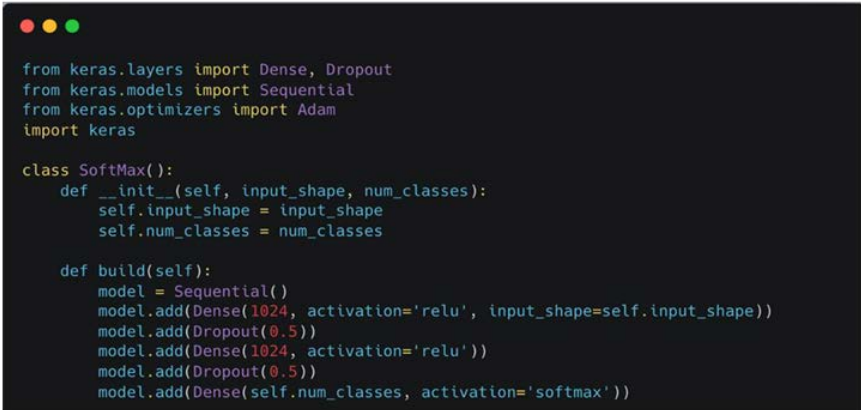
The system development follows a structured and technology-driven conceptual approach, integrating modern technologies, including Streamlit for the frontend development and SQLite for the back-end database management. The app is designed with a clean architecture that separates the user interface, business logic, and data management layers. The architecture of Streamlit follows a reactive, script-based paradigm that is designed to simplify the development of interactive web applications in Python. At its core, Streamlit operates on a client-server model where the Python script runs on the server-side, and the user interface is rendered in the browser through a lightweight frontend built with modern web technologies. **Table 1** shows the functional and non-functional requirements of our system.

Table 1. Functional and non-functional requirements.

Requirement Type	Description
Functional Requirements	• The system captures high-quality images for facial recognition. • Privacy protocols such as JWT authentication, AES 256 encryption are included. • The system processes images in real-time for face detection recognition. • The system generates real-time alerts with a location timestamp. • The system integrates with the InsightFace framework.
Non-Functional Requirements	• The user interface is intuitive and user-friendly. • Data are securely stored and encrypted. • The system is scalable to accommodate a growing user base and increasing amounts of data. • The system is responsive across different devices.

6. Building a Function and Script for Face Detection, Recognition and Alert Sending

In our research, we use a Python script to build and train a neural network for multi-class classification tasks. This type of model is commonly applied in areas such as image recognition, natural language processing, and datasets where the output needs to be categorized into multiple classes. The following code that we use defines a custom SoftMax class that builds a neural network using Keras. The network consists of two hidden layers with 1024 units each, using the Rectified Linear Unit (ReLU) activation and 50% dropout for regularization, followed by an output layer with as many units as the number of classes and a softmax activation. It uses the Adam optimizer with a learning rate of 0.001 and categorical cross-entropy loss, optimized with Adaptive Moment Estimation with Long-Term Memory to train the model for classification tasks. The model is compiled with accuracy as a metric. With the integration of Python and PHP scripting programs, the developed system is now simple and easy [9].



```
from keras.layers import Dense, Dropout
from keras.models import Sequential
from keras.optimizers import Adam
import keras

class SoftMax():
    def __init__(self, input_shape, num_classes):
        self.input_shape = input_shape
        self.num_classes = num_classes

    def build(self):
        model = Sequential()
        model.add(Dense(1024, activation='relu', input_shape=self.input_shape))
        model.add(Dropout(0.5))
        model.add(Dense(1024, activation='relu'))
        model.add(Dropout(0.5))
        model.add(Dense(self.num_classes, activation='softmax'))
```

Figure 4. Softmax code.

Figure 4 shows the softmax code, a Python script used to build and train a multi-

class neural network. The softmax function is a ubiquitous function, frequently used as a probabilistic link function for unordered categorical data, in different kinds of models, such as regression, artificial neural networks, or probabilistic cognitive models [10].

6.1. Face Detection and Landmark Extraction Code

We use a script that uses OpenCV and MTCNN (Multi-task Cascaded Convolutional Networks) to detect faces in a video stream. It captures video input, detects faces, and extracts facial landmarks such as the left and right eyes, nose, and mouth. Detected faces are saved as images with bounding boxes and landmarks, and the process continues until the video ends or a key is pressed.

6.2. Face Recognition and Age-Gender Prediction

The provided Python code is used to recognize faces and predict attributes like age and gender from images. This can be applied in social media filtering, personalized marketing, security authentication, or demographic analysis based on facial features. This script processes face images using MTCNN for detection and a pre-trained model for recognition. It aligns detected faces, extracts features, and predicts age and gender using a deep learning model. The processed images are saved, and the script outputs the predicted age and gender. It leverages MTCNN and a pre-trained deep learning model and is an invaluable tool for a focused study on facial recognition and demographic attribute prediction. Face recognition is important because it detects, extracts features, and predicts age and gender, which is important for decision-making in terms of security research.

Figure 5 shows the code for the face recognition and age-gender recognition.

```
from __future__ import absolute_import
from __future__ import division
from __future__ import print_function

from scipy import misc
import sys
import os
import argparse
import tensorflow as tf
import numpy as np
import mxnet as mx
import random
import cv2
import sklearn
from sklearn.decomposition import PCA
from time import sleep
from easydict import EasyDict as edict
from mtcnn_detector import MtcnnDetector
sys.path.append(os.path.join(os.path.dirname(__file__), '..', 'src', 'common'))
import face_image
import face_preprocess

def do_flip(data):
    for idx in xrange(data.shape[0]):
        data[idx, :, :] = np.fliplr(data[idx, :, :])

def get_model(ctx, image_size, model_str, layer):
    _vec = model_str.split(',')
    assert len(_vec) == 2
    prefix = _vec[0]
    epoch = int(_vec[1])
    print('loading', prefix, epoch)
    sym, arg_params, aux_params = mx.model.load_checkpoint(prefix, epoch)
    all_layers = sym.get_internals()
    sym = all_layers[layer + '_output']
    model = mx.mod.Module(symbol=sym, context=ctx, label_names = None)
    model.bind(data_shapes=[('data', (args.batch_size, 3, image_size[0], image_size[1]))], label_shapes=[('softmax_label', (args.batch_size, 1))])
    model.bind(data_shapes=[('data', (1, 3, image_size[0], image_size[1]))])
    model.set_params(arg_params, aux_params)
    return model
```

Figure 5. Face recognition and age-gender prediction code.

7. Results

After designing the system architecture and building the Python script, we found that the system provides real-time suspect detection and location alerts, which can significantly improve response time and situational awareness for defense and security forces, enhancing national safety.

Figure 6 shows an experimental suspect used for system testing. **Figure 7** shows the suspect's name, which is Stephane Zoungrana. The timestamp is 2025-05-30 01:12:16. The location is Burkina Institute of Technology. The alert sender using Gmail typically refers to a system or application that sends automated alert messages (emails) through a Gmail account. The alert location is made based on camera metadata.



Figure 6. Experimental suspect 1.

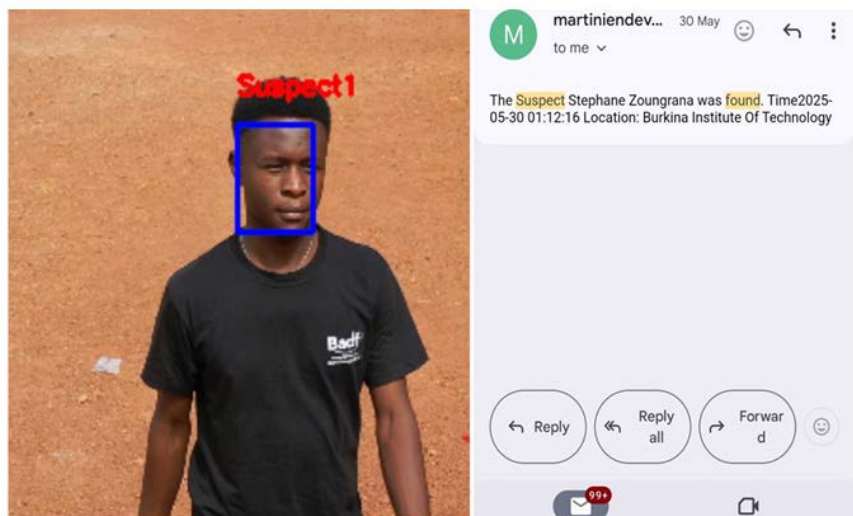


Figure 7. Alert sender abstract.

The model exhibits very high performance, with minimal loss and good accuracy on both the training and test datasets. This research directly supports Burkina Faso's broader efforts to combat terrorism through technological intelligence, potentially reducing civilian casualties and improving peacekeeping outcomes.

8. Conclusion

Burkina Faso is facing a critical security threat due to terrorism, with a range of fatalities recorded in 2023. Despite continuous efforts by the national army to restore peace, terrorists increasingly use deceptive tactics, such as disguises, to infiltrate public spaces. To address this, a facial recognition surveillance system was developed to support defense and security forces in automatically detecting terrorist suspects through public-area cameras. The system generates real-time alerts with precise locations and timestamps. A drone was used for evaluation due to limited access to public surveillance networks, with promising results 92% detection and 90% accuracy in ideal conditions. However, challenges remain, particularly regarding full integration with the infrastructure and the system's performance under variable conditions, such as poor lighting or dense crowds. Through careful requirements analysis, architectural planning, and the adoption of modern tools such as Streamlit, InsightFace, and SQL, the system was developed to perform real-time facial recognition and generate alerts. The use of UML diagrams ensured a structured and comprehensive design, while the integration of robust security protocols addressed data protection concerns.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

- [1] Khandelwal, N., Ganjihal, I., Karajgar, S. and Farakte, P. (2025) Capturing the Sky: Exploring Aerial Photography through Drone Technology. *International Journal of Scientific Research and Technology*, **2**, 85-100.
- [2] Lumaban, M.B. (2020) CCTV-Based Surveillance System with Face Recognition Feature. *International Journal of Advanced Trends in Computer Science and Engineering*, **9**, 349-355.
- [3] Wong, K.S., Tu, N.A., Maratkhan, A. and Demirci, M.F. (2020) A Privacy-Preserving Framework for Surveillance Systems. 2020 *The 10th International Conference on Communication and Network Security*, Tokyo, 27-29 November 2020, 91-98. <https://doi.org/10.1145/3442520.3442524>
- [4] Dickerson, C.E., Wilkinson, M., Hunsicker, E., Ji, S., Li, M., Bernard, Y., *et al.* (2020) Architecture Definition in Complex System Design Using Model Theory. *IEEE Systems Journal*, **15**, 1847-1860. <https://doi.org/10.1109/jsyst.2020.2975073>
- [5] Alsammak, O.R. and Abdulmajeed, A.A. (2025) A Modern Software Engineering Approach to UML Class Diagram Evaluation. In: *Communications in Computer and Information Science*, Springer, 175-190. https://doi.org/10.1007/978-3-032-07244-3_11
- [6] Xiao, C., Ståhl, D. and Bosch, J. (2025) UML Sequence Diagram Generation: A Multi-Model, Multi-Domain Evaluation. 2025 *IEEE/ACM 47th International Conference on*

- Software Engineering: Software Engineering in Practice (ICSE-SEIP)*, Ottawa, 27 April-3 May 2025, 272-283. <https://doi.org/10.1109/icse-seip66354.2025.00030>
- [7] Yildirim, U. and Campean, F. (2020) Functional Modelling of Complex Multi-Disciplinary Systems Using the Enhanced Sequence Diagram. *Research in Engineering Design*, **31**, 429-448. <https://doi.org/10.1007/s00163-020-00343-8>
- [8] Ab Rahman, S., Hashim, W.B. and Yusof, A. (2021) Designing a Use Case Diagram for Developing an Electricity Consumption (EC) System. 2021 *International Conference on Computer & Information Sciences (ICCOINS)*, Kuching, 13-15 July 2021, 282-285. <https://doi.org/10.1109/iccoins49721.2021.9497156>
- [9] Kamil, M.H.M., Zaini, N., Mazalan, L. and Ahamad, A.H. (2023) Online Attendance System Based on Facial Recognition with Face Mask Detection. *Multimedia Tools and Applications*, **82**, 34437-34457. <https://doi.org/10.1007/s11042-023-14842-y>
- [10] Khairuddin, M.H., Shahbudin, S. and Kassim, M. (2021) A Smart Building Security System with Intelligent Face Detection and Recognition. *IOP Conference Series: Materials Science and Engineering*, **1176**, Article 012030. <https://doi.org/10.1088/1757-899x/1176/1/012030>