

ภาคผนวก ก

โค้ดโปรแกรมสำหรับการสร้างระบบตรวจการณัจฉริยะสำหรับหอพัก
โรงเรียนสุรวิวัฒน์ โดยใช้อากาศยานไร้คนขับ

ก.1 โปรแกรมประมวลผลภาพ คนบุกรุกป็นรั้ว (intrusion)

```

from ultralytics import YOLO
import cv2
import cv2 as cv
#import serial
#import time
import requests
import time
import numpy as np
url = 'https://notify-api.line.me/api/notify'
token = 'RiEABMX5yL43KeFtKX0V0jaAH5aXr2QUu8Kx3pLUKE7'
headers = {'content-type':'application/x-www-form-urlencoded','Authorization':
f'Bearer {token}'}
data = {'message': 'มีผู้บุกรุก_intrustion'} #, 'imageThumbnail':'img_url',
'imageFullsize':'img_url'
#img_url = 'https://drive.google.com/file/d/1dSHL-
CCks5P3ydcLaWiYzmu6heaD4Ho1/view?usp=drive_link'
#def draw_boundary():
def changeRes(width,height):
    cap.set(3,width)
    cap.set(4,height)
cap = cv2.VideoCapture("rtmp://live.sut.ac.th/live/dji")
model = YOLO("best.pt")
conf_score = []
cid = []
lasttime = 0
class_path = "coco.names"
with open (class_path,"r") as f:
    class_name = f.read().strip().split("\n")
while (True): #เมื่อมีภาพให้แสดงทั้งหมด
    ret, frame = cap.read( ) #อ่านภาพแสดงผลทีละเฟรม

```

```

#res = model.predict(source = frame, show = True) #แสดงภาพต่อเนื่อง
#cv2.imshow("frame", frame)
#requests.post(url=url, headers=headers, data=data)

if ret != True:
    break
timenow = time.time()
results = model(frame)

for result in results:
    for data in result.bboxes.data.tolist():
        x1,y1,x2,y2,confidence,class_id = data
        x1,y1,x2,y2,class_id = int(x1),int(y1),int(x2),int(y2),int(class_id)
        class_id = int(class_id)
        names = class_name[class_id] #

        if names == "person":
            #flie_name = "img_%s.jpg"%(np.random.randint(0,100))
            #flie_name = "line/img_%s.jpg"%(np.random.randint(0,10))
            #cv2.imwrite(flue_name,frame[y1:y2,x1:x2])
            cv2.rectangle(frame,(x1,y1),(x2,y2),(0,255,0),3)
            cv2.putText(frame,f"{names}",(x1,y1),cv2.FONT_HERSHEY_SIMPLEX,1,(0,0,0),2)

        if names == "intrusion":
            flie_name = "line/img_%s.jpg"%(np.random.randint(0,1)) #
            cv2.imwrite(flue_name,frame[y1:y2,x1:x2])
            cv2.rectangle(frame,(x1,y1),(x2,y2),(0,0,255),3)
            cv2.putText(frame,f"{names}",(x1,y1),cv2.FONT_HERSHEY_SIMPLEX,1,(0,0,0),2)
            #cv2.imshow("imag", frame)
            flie_name = "intrusion/img_%s.jpg"%(np.random.randint(0,1))
            cv2.imwrite(flue_name,frame)

```

```

#r=requests.post(url=url, headers=headers, data=data)
#cv2.waitKey(0)

if cv2.waitKey(1) & 0xFF == ord('q'): #กด q ปิดการแสดงผลภาพทุกเฟรม
    break #break or pass
cv2.imshow("frame",frame) #แสดงผลภาพต่อเนื่อง
cap.release() #ทำการหยุดและคืนทรัพยากร
cv2.destroyAllWindows() #หยุดการทำงานทั้งหมด

```

ก.2 โปรแกรมส่งข้อความและภาพคนบุกรุกป็นรั้ว (intrusion)

```

import requests

def linenotify(message):
    url = 'https://notify-api.line.me/api/notify'
    token = 'RiEABMX5yL43KeFtKX0V0jaAH5aXr2QUu8Kx3pLUKE7' # Line Notify Token
    img = {'imageFile': open('intrusion/img_0.jpg','rb')} #Local picture File
    data = {'message': message}
    headers = {'Authorization': 'Bearer ' + token}
    session = requests.Session()
    session_post = session.post(url, headers=headers, files=img, data =data)
    print(session_post.text)

message = 'มีผู้บุกรุก_intrusion' #Set your message here!
linenotify(message)

```

ก.3 โปรแกรมฝึกสอนโมเดล

```

!nvidia-smi
import os
HOME = os.getcwd()

```

```

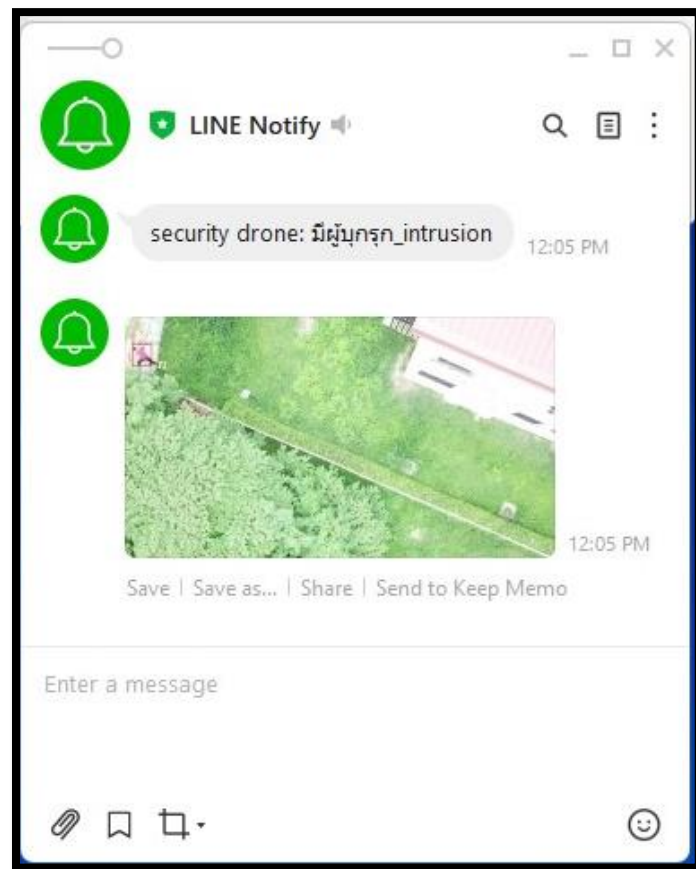
print(HOME)
# Pip install method (recommended)
!pip install ultralytics==8.0.196
from IPython import display
display.clear_output()
import ultralytics
ultralytics.checks()
from ultralytics import YOLO
from IPython.display import display, Image
!mkdir {HOME}/datasets
%cd {HOME}/datasets
!pip install roboflow --quiet
!pip install roboflow
from roboflow import Roboflow
rf = Roboflow(api_key="ZS5OCn8BO6vuPGCdPxJx")
project = rf.workspace("pamorn").project("security-drone")
version = project.version(50)
dataset = version.download("yolov8")
%cd {HOME}
!yolo task=detect mode=train model=yolov8s.pt data={dataset.location}/data.yaml
epochs=50 imgsz=800 plots=True
!ls {HOME}/runs/detect/train/
%cd {HOME}
Image(filename=f'{HOME}/runs/detect/train/confusion_matrix.png', width=600)
%cd {HOME}
Image(filename=f'{HOME}/runs/detect/train/results.png', width=600)
%cd {HOME}
Image(filename=f'{HOME}/runs/detect/train/val_batch0_pred.jpg', width=600)
%cd {HOME}
!yolo task=detect mode=val model={HOME}/runs/detect/train/weights/best.pt
data={dataset.location}/data.yaml

```

```
%cd {HOME}
!yolo task=detect mode=predict model={HOME}/runs/detect/train/weights/best.pt
conf=0.25 source={dataset.location}/test/images save=True
import glob
from IPython.display import Image, display
for image_path in glob.glob(f'{HOME}/runs/detect/predict3/*.jpg')[:3]:
    display(Image(filename=image_path, width=600))
    print("\n")
project.version(dataset.version).deploy(model_type="yolov8",
model_path=f'{HOME}/runs/detect/train/')
```

ภาคผนวก ข

แผนภาพการเชื่อมต่อระบบตรวจการณ์อัจฉริยะสำหรับหอพัก
โรงเรียนสุรวิวัฒน์ โดยใช้อากาศยานไร้คนขับ การตรวจจับภาพคนบุกรุก
และการแจ้งเตือน



รูปที่ ข.3 ข้อความและภาพถ่ายทางอากาศ security drone : มีผู้บุกรุก_intrusion

ภาคผนวก ค

บทความที่ได้รับการตีพิมพ์เผยแพร่

รายชื่อบทความทางวิชาการที่ได้ตีพิมพ์เผยแพร่ในระหว่างศึกษา

Pamorn, K & Suradet, T., (2024). INTELLIGENT SURVEILLANCE SYSTEM FOR SURAWIWAT SCHOOL DOMITORY USING UNMANNED AERIAL VEHICLES. The 19th International Joint Symposium on Artificial Intelligence and Natural Language Processing, iSAL-NLP 2024, 11-15 November 2024, Pattaya, Thailand

Intelligent surveillance system for Surawiwat school dormitory using unmanned aerial vehicles

Pamorn Khueankham
School of Mechatronics Engineering
Institute of Engineering
Suranaree University of Technology
Nakhon Ratchasima, Thailand
244017@sut.ac.th

Suradet Tantrairatn
School of Mechanical Engineering
Institute of Engineering
Suranaree University of Technology
Nakhon Ratchasima, Thailand
suradetj@sut.ac.th

Abstract— This research presents an innovative approach to dormitory surveillance at Surawiwat School by employing an unmanned aerial vehicle (UAV) for autonomous monitoring. The UAV is used for aerial reconnaissance, allowing for efficient surveillance of the dormitory's perimeter to enhance security. By capturing high-resolution aerial images, the system aims to identify and track potential intruders, specifically those attempting to climb the dormitory fence. The images captured by the UAV are processed using advanced machine learning techniques, with a focus on object detection through deep learning. The system is built around a Convolutional Neural Network (CNN) and leverages the YOLOv8 (You Only Look Once) algorithm. YOLOv8 is recognized for its high accuracy and real-time processing capabilities, making it an ideal choice for real-time surveillance and detection tasks. The CNN-based model is trained to accurately detect human figures and identify unusual activities within the captured images. When the system detects an intruder, it sends an immediate alert, along with the captured aerial image, through the Line application to designated personnel. This instant notification system enhances response times, allowing school security to address potential threats proactively. Overall, this research demonstrates a sophisticated, AI-driven surveillance solution that combines UAV capabilities with state-of-the-art object detection, contributing to enhanced safety and security for school dormitories.

Keywords— Machine learning, Deep learning, Image processing, Object detection.

I. INTRODUCTION

This research addresses the security challenges faced by Suranaree University of Technology, which functions as a "university town" with a range of facilities, including learning and research centers, agricultural learning farms, schools, hospitals, recreational and exercise areas, and residential housing like the Suraniwat and Surawiwat Dormitory Buildings. Given the vast campus and the increasing security demands, exacerbated by a reduction in security personnel, there is a critical need for enhanced surveillance tools, particularly for the expansive and hard-to-monitor Surawiwat School dormitory area. The dormitory's perimeter is adjacent to Surasawat Land Village, with natural entry points that pose security risks, making it vulnerable to unauthorized access.

This research, therefore, explores the development of an intelligent surveillance system that utilizes an unmanned aerial vehicle (UAV) to autonomously patrol the dormitory's

perimeter. The UAV is designed to capture aerial images along the fence line, particularly focusing on identifying potential intrusions, such as individuals attempting to climb the fence. By employing machine learning [11-12] and image processing techniques [1], the system processes these images to detect intrusions in real-time [6].

The process begins with the design and deployment of the UAV for continuous patrols. Collected images are processed using deep learning, specifically with a convolutional neural network (CNN), to facilitate accurate detection of intruders. The YOLOv8 (You Only Look Once) algorithm, known for its high-speed, one-stage object detection capabilities, is applied here to recognize and alert security personnel of any suspicious activity. Upon detecting an intruder, the system sends an alert and image via the Line application to the designated response team, allowing for immediate action.

Overall, this research demonstrates the potential for UAV-based, AI-driven surveillance systems [3] to significantly enhance security at large facilities like Surawiwat School's dormitory, providing a proactive solution to the growing demand for campus safety.



Fig. 1. Person climb the dormitory fence

II. MATERIALS AND METHODS

A. Step 3.1 Machine Learning

Machine learning [11-12] of 7 steps, gathering data, preparing that data, choosing a model, training model, evaluation, hyperparameter tuning, Prediction.

Step 1 gathering data : The first step involves collecting a dataset of relevant images, including scenarios with individuals near or attempting to climb the dormitory fence. Data is collected from aerial images captured by the UAV, along with additional labeled images as necessary for training.

Step 2 preparing that data : Collected data undergoes preprocessing to ensure it's suitable for the machine learning model. This includes tasks like resizing images, labeling objects (e.g., individuals, fence), and augmenting the data (e.g., rotating, flipping) to increase dataset diversity and improve the model's robustness.

Step 3 choosing a model : Based on the task requirements, the YOLOv8 model is selected due to its high performance in real-time object detection. This one-stage object detection [1, 3, 4, 5, 6] model is specifically effective in surveillance [3] tasks requiring quick and accurate identification of objects (such as intruders) in images.



Fig. 2. Object detection YOLOv8

Step 4 training model : The model is trained on the prepared dataset, learning to identify individuals and detect unusual behavior such as fence climbing. During training, the model optimizes its internal parameters to improve detection accuracy, learning to differentiate between normal and suspicious activities.

Step 5 evaluation : Once trained, the model is evaluated using test data that it hasn't seen before. This step checks the model's performance in detecting intruders with metrics like mAP (mean Average Precision) to ensure it meets the accuracy and precision requirements for real-time [6] surveillance.

Step 6 hypermeter tuning : Based on the evaluation results, the model's hyperparameters (e.g., learning rate, batch size) are adjusted to enhance its performance. Fine-tuning helps achieve optimal results, improving detection accuracy and reducing false positives.

Step 7 Prediction : After final tuning, the model is deployed for real-world predictions. It processes live data from the UAV, identifying and sending alerts when it detects an intruder, enabling real-time [6] monitoring of the dormitory perimeter.

This seven-step approach ensures a structured and effective implementation of machine learning [11-12] for an intelligent surveillance system, allowing for precise, real-time [6] detection and alerting capabilities.

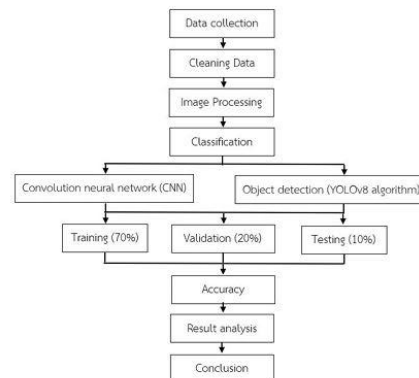


Fig. 3. Flowchart

The UAV captures images at three specific heights : 20, 25, and 30 m. These heights provide varying perspectives, which improve the model's ability to detect individuals across different scales and angles. The camera is positioned at a 90° angle (directly downward), which ensures consistent top-down imagery of the dormitory perimeter for uniform data collection. Images are captured in both RGB and CMYK color systems, allowing flexibility in color representation and potentially improving the model's capacity to handle color variations in different lighting conditions. Each image is initially captured at a high resolution of 3,840 x 2,160 pixels, which captures fine details necessary for detecting individuals and their actions. Downsizing for Model Input : Images are then resized to 640 x 640 pixels to fit the input requirements for YOLOv8, reducing computational load while retaining enough detail for effective object detection. Total Dataset: Approximately 10,000 images make up the dataset, providing substantial data for model training and validation. Person : Comprising 49.56% of the dataset, Person climbing the fence. This category represents 50.44% of the dataset, with images of individuals attempting to climb the fence. This balanced class distribution ensures the model can effectively learn both regular and suspicious behaviors. This approach leverages diverse image data captured at multiple heights, optimized image sizes, and color variations, ensuring robust training for accurate detection of intruders in real-time [6]. The balanced dataset with equal representation of regular individuals and fence climbers also enhances the model's ability to differentiate between normal and potentially dangerous activities, contributing to a reliable UAV-based surveillance system.

The model's performance during training is shown through various evaluation metrics and functions, as depicted in Fig. 4 and 5. Here's a breakdown of the results and what they indicate about the model's effectiveness. Reduction over time. The object loss function shows a downward trend, indicating that the model is effectively learning to detect objects with increasing accuracy as training progresses. A decrease in object loss typically suggests that the model is minimizing errors in identifying relevant objects (in this case, individuals and fence-climbing attempts). Gradual Decrease: Additional loss functions, such as classification and localization loss, also

decrease gradually. This steady decrease suggests that the model's overall ability to classify and localize objects improves as training continues. Model Performance 5% mAP: The mean Average Precision at a 5% threshold has increased from 82.2% to 85.1%. This improvement reflects better performance at lower IoU thresholds, suggesting that the model is becoming more accurate in detecting objects, even in scenarios where bounding boxes might not perfectly overlap. 95% mAP: The mAP at a higher threshold (95%) increased from 55.4% to 61.5%. This metric highlights the model's improvement in precise object localization and detection, as it requires a closer overlap between predicted and actual bounding boxes. Recall increased slightly, from 76.2% to 77.7%. This indicates that the model has become better at identifying a greater proportion of actual positive instances (i.e., individuals or intrusions) in the test data. Precision improved from 92.3% to 95.0%, showing a reduction in false positives. This means the model more accurately identifies only relevant objects, resulting in fewer incorrect detections. Training over 10, 20, 40, 60, 80 to 100 epochs contributed to the improvements in these metrics, as additional epochs allow the model to learn better feature representations and further refine its detection capabilities. In summary, the model shows significant improvements in detection accuracy, precision, and recall. The increase in 5% and 95% mAP scores reflects enhanced object detection capability across both lenient and strict IoU thresholds, making the model well-suited for the real-time, high-precision demands of UAV-based surveillance [3].

TABLE I. RESULTS INDEX FROM YOLOV8

Metric	10 epoch	20 epoch	40 epoch	60 epoch	80 epoch	100 epoch
Precision	92.3%	93.0%	93.6%	94.3%	94.6%	95.0%
Recall	76.2%	76.6%	77.1%	77.6%	77.6%	77.7%
5% mAP	82.2%	83.4%	84.1%	85.1%	85.1%	85.1%
95% mAP	55.4%	58.1%	59.6%	61.1%	61.3%	61.5%

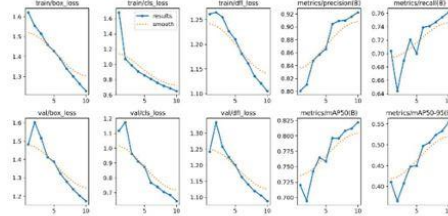


Fig. 4. Results index from YOLOV8 @ 10 epochs

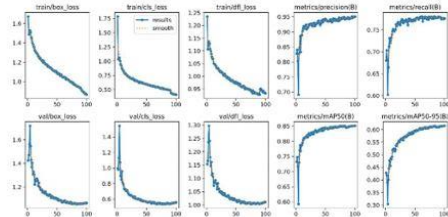


Fig. 5. Results index from YOLOV8 @ 100 epochs

B. Step 3.2 Design a surveillance system

The consists of Unmanned Aerial Vehicle (UAV) [1-5], and control equipment application, live broadcast system (SUT Data Center), Control Room for analyzing aerial photographs.

To develop this intelligent surveillance system using drone-captured images, the following process is established for detecting and reporting fence-climbing intruders in real-time. Drone-Camera Setup : A camera mounted on a drone captures images or streams video of the dormitory perimeter. The camera provides high-resolution imagery of the area around the fence, focusing on detecting individuals near or attempting to climb the fence. Image Processing Principles: Using basic image processing techniques in OpenCV, each frame is analyzed to identify relevant pixel areas and detect shapes or movements that resemble a person climbing. This step involves filtering, edge detection, and segmentation to isolate the areas of interest (i.e., person-like shapes). Pixel Area Classification: The segmented image is analyzed to identify regions of pixels that correspond to a person's size and shape. By classifying areas based on pixel density and configuration, the system can filter out irrelevant objects and focus on likely instances of individuals. Detected regions are classified by their pixel area, allowing the system to differentiate between typical objects in the scene and the specific shape of a person climbing the fence. Grouping and Labeling Data: Once potential intrusions are identified by pixel area, these images are grouped, labeled, and prepared for further processing. They serve as input for the machine learning algorithms, where labeled data distinguishes between regular and suspicious activities. Machine Learning Techniques. This prepared dataset is fed into a convolutional neural network (CNN) and the YOLOv8 model, both of which are trained to analyze images for accurate object detection and classification. Object Detection and Classification with YOLOv8 and CNN. The CNN model is used to extract features from the images, focusing on distinguishing characteristics of a person and the act of climbing. YOLOv8, as a one-stage object detection model, performs real-time analysis on incoming frames. It detects and classifies any instances of fence climbing based on the learned model parameters, sending an alert if an intrusion is detected. RTMP Streaming with Python and OpenCV : The live video feed from the drone's camera is streamed in real-time using the RTMP protocol, facilitated by a Python script running in Visual Studio Code. OpenCV captures and processes each frame, sending the video stream to a central computer for monitoring and analysis. Intrusion Detection and Notification. Intruder Alert: If YOLOv8 or the CNN model detects a person climbing the fence, an alert is triggered. Line Notify Alert: The system sends an immediate alert via the Line Notify application, including a brief message ("Person climbing the fence intrusion") along with an image of the detected individual. This enables real-time notification for quick response from security personnel. This setup combines image processing, deep learning, and real-time streaming to provide a robust and responsive security solution. The drone-based surveillance system not only detects intrusions but also instantly notifies security staff, enhancing the safety of the dormitory area through effective and automated monitoring.

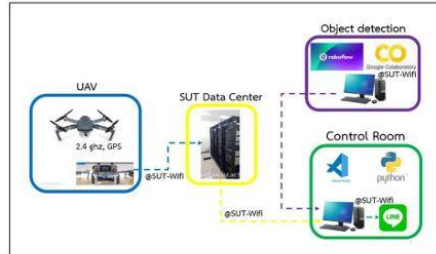


Fig. 6. Structure diagram Intelligent surveillance system for Surawiwat School dormitory using UAV



Fig. 7. Control room

C. Step 3.3 Set flight route

To establish an effective UAV-based surveillance strategy for the Surawiwat School Dormitory, the following flight plan and operational parameters are designed for regular inspections and patrols. Surawiwat School Dormitory Layout. The dormitory occupies a rectangular area in Fig. 8 and 9, necessitating a systematic approach for aerial surveillance to cover the entire perimeter efficiently. The drone takes off and lands at a designated area in front of the dormitory, providing a safe and controlled environment for operations. The UAV will cover an approximate flight distance of 310 m. at varying speeds of 2.5 m/s, 3.5 m/s, and 4.5 m/s. The drone will operate at three different altitudes: 20 m, 25 m, and 30 m. The camera is positioned at a 90° angle (directly downward) to ensure a clear and unobstructed view of the ground below. All inspections are conducted during daylight hours to ensure optimal visibility and effective image capturing. The UAV performs inspections every hour, providing regular updates on the security status of the dormitory perimeter. This frequency allows for timely detection of any unauthorized activity or potential security breaches. This structured flight plan ensures that the UAV conducts thorough and regular inspections of the Surawiwat School Dormitory, enhancing security through proactive monitoring and timely detection of potential threats. By adhering to this operational strategy, the drone surveillance system can effectively safeguard the dormitory area while optimizing its flight capabilities.



Fig. 8. Set flight routes for 4 points

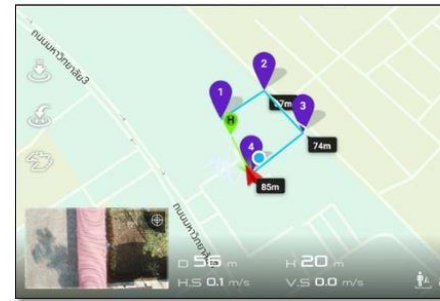


Fig. 9. Flight route plan (Path) points 1 to 4

III. EXPERIMENTAL RESULTS

The results from the intelligent surveillance system implemented for the Surawiwat School Dormitory reflect its effectiveness in detecting potential intruders. Here's a detailed breakdown of the testing outcomes:

Testing Overview system was subjected to 72 test scenarios, designed to assess its ability to accurately detect individuals attempting to climb the dormitory fence. Correct Detections: Out of the 72 tests, the system successfully detected intruders in 70 tests. failed Detections: The system failed to detect intruders in 2 tests, indicating some limitations in its detection capability under certain conditions. he system achieved a success rate of 97.22%, demonstrating its high effectiveness.

- To calculate the overall success rate of the detection system, the formula used is:

$$\text{Success Rate} = \left(\frac{\text{Number of Correct Detections}}{\text{Total Tests}} \right) \times 100$$
- Applying the numbers:

$$\text{Success Rate} = \left(\frac{70}{72} \right) \times 100 = 97.22\%$$

Fig. 10. Success rate calculation

A. Object detection

The intelligent surveillance system's ability to detect intruders based on their clothing color has been tested effectively, yielding valuable results. Here's a detailed summary of the findings related to the detection of individuals climbing the dormitory fence, specifically focusing on the color variations in shirts. The system analyzes detected images of potential intruders wearing shirts of different colors, utilizing both the RGB and CMYK color models. The system was evaluated through 72 tests focused on detecting individuals attempting to climb the fence. system successfully detected intruders in 70 tests, failures in 2 tests.

B. Flight height for image detection

- Height 20m, detected correctly 23 test, failures 1 test.
- Height 25m, detected correctly 24 test.
- Height 30m, detected correctly 23 test, failures 1 test.

C. Flight speed for image detection

- Speed 2.5m/s, detected correctly 23 test, failures 1 test.
- Speed 3.5m/s, detected correctly 24 test.
- Speed 4.5m/s, detected correctly 23 test, failures 1 test.



Fig. 11. UAV image aerial photos at height 25 m.



Fig. 12. System successfully detected intruders.

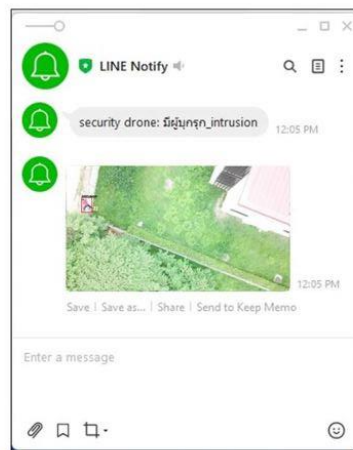


Fig. 13. Message and aerial photos. "Security drone : Person climbing the fence_intrusion"

IV. CONCLUSION

This research presents an innovative approach to enhancing security at the Surawiwat School dormitory through an intelligent surveillance system utilizing drones. The findings and recommendations outlined below summarize the key aspects of the system's performance, the impact of flight parameters on detection accuracy, and suggestions for future improvements. Autonomous Flight : The drone operates autonomously, enabling it to conduct various tasks including rescue missions, surveillance, inspections, area patrols, target searches, and specifically monitoring for individuals attempting to climb the dormitory fence. Mission Success Rate. The system has demonstrated an impressive mission completion rate of 97.22%, reflecting its reliability and effectiveness in detecting potential security breaches. Flight Parameters and Their Impact. Higher Altitudes. While flying at greater heights allows for a wider field of view, it also results in smaller images of objects on the ground. This can hinder the drone's ability to accurately identify and classify intruders. Optimal Height Selection: Finding an optimal flight altitude is crucial for balancing coverage and image detail. Lower altitudes may enhance image clarity but could reduce the area covered in a single flight. Flight Speed higher flight speeds can lead to motion blur, resulting in unclear images. This blurriness can significantly impact the detection performance of the system, as fine details necessary for accurate identification may be lost. Adjusting the speed during flights to ensure clarity in image capture is essential. Slower speeds might be necessary during critical inspection phases to obtain sharp, actionable imagery. Recommendations for Improvement. Data Collection from Varied Environments to enhance the model's decision-making capabilities, it is recommended to gather and incorporate data from diverse environments. This exposure will enable the machine learning model to generalize better and improve its accuracy across different scenarios. Reducing Noise in Data Transmission improving the efficiency of image

Identify applicable funding agency here. If none, delete this text box.

transmission between the drone and the receiving system is vital. Efforts should be made to minimize noise in the communication channels, which can affect the quality and reliability of transmitted images. Implementing advanced encoding techniques or using higher-bandwidth connections may enhance image quality during transmission.

The intelligent surveillance system for the Surawiwat School dormitory effectively utilizes drones for security monitoring, achieving a high level of performance. By considering the impact of flight height and speed on detection accuracy, and implementing the suggested improvements for data collection and transmission, the system can further enhance its effectiveness. Continued research and development in these areas will contribute to creating a more robust and reliable security solution for the dormitory and similar environments.

ACKNOWLEDGMENT

This research was supported by the Suranaree University of Technology (SUT)

REFERENCES

- [1] Aswini N and Uma S V, "Drone Object Detection Using Deep Learning Algorithms", 2021 Third International Conference on Inventive Research in Computing Applications (ICIRCA-2021). IEEE, 2021.
- [2] Aswini N and Uma S V, "Custom based Obstacle Detection Using Yolo v3 for Low Flying Drones", 2021 International Conference on Circuits, Controls and Communications (CCUBE). IEEE, 2021.
- [3] Rohit J, Rajesh P, Akshay D, S. M. Rathod and Ajay S, "Drone Based Object Detection using AI", 2022 International Conference on Signal and Information Processing (IconSIP). IEEE, 2022.
- [4] Oyku S and Sedat O, "YOLODrone : Improved YOLO Architecture for Object Detection in Drone Images Architectures on Object Detection of UAV Images" 2021 44th International Conference on Telecommunications and Signal Processing (TSP). IEEE, 2021.
- [5] Teddy S, Islam M, Mira K and Nanang I, "Performance Comparison of Various YOLO", 2022 IEEE 8th International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA). IEEE, 2022.
- [6] Mahdjoubi I and Abdelhafid B, "Real Time Object Detection With Drone Using Deep Learning Algorithm", 2022 International Conference of Advanced Technology in Electronic and Electrical Engineering (ICATEEE). IEEE, 2022.
- [7] Tasweer A, Marc C, Yutaka M and Helmut P, "Detecting Human Actions in Drone Images Using YoloV5 and Stochastic Gradient Boosting", Sensors 2022, 22, 7020.
- [8] Jen-Kai T, Chen-Chien H, Wei-Yen W and Shao-Kang H, "Deep Learning-Based Real-Time Multiple-Person Action Recognition System", Sensors 2020, 20, 4758.
- [9] M.E. Paoletti, J.M. Haut, J. Plaza, A. Plaza, "A new deep convolutional neural network for fast hyperspectral image Classification", 2018 ISPRS Journal of Photogrammetry and Remote Sensing 145 (2018) 120–147.
- [10] Rasha A, Prashanth R, Wei L, Mauro D, "Simultaneous extraction of roads and buildings in remote sensing imagery with convolutional neural networks", 2017 ISPRS Journal of Photogrammetry and Remote Sensing 130 (2017) 139–149.
- [11] Phan Rattanapoka. 2016. Introduction to Artificial Intelligence. King Mongkut's University of Technology North Bangkok. pp 318.
- [12] Athit Srikaew. 2009. Computational intelligence. 1st ed. Electrical Engineering Division, Faculty of Engineering, Suranaree University of Technology. pp 512 .