­

Title

By

Name

to

The School of Computing

for the BSc in Computer Science

Thesis Supervisor

Plageras Andreas

Module Professor

Tasos Stylianou

University of Derby

Mediterranean College

2022

**Acknowledgements**

I would first like to thank my professors Andreas Plageras and Stylianou Anastasios, for the support that they have provided me throughout this amazing journey. It has been extremely interesting and profitable.

**Abstract** |

KEYWORDS | Drones, Safety, artificial intelligence, OpenCV, Computer vision

**Table of Contents**

[1. Introduction 5](#_Toc103263031)

[1.1 Current Study Overview 5](#_Toc103263032)

[1.2 Scientific Contribution 5](#_Toc103263033)

[1.3 Outline 5](#_Toc103263034)

[2. Related Work 6](#_Toc103263035)

[3. Applications of drones and potential 8](#_Toc103263036)

[3.1 Drones in agriculture 8](#_Toc103263037)

[3.2 Drones in health care 9](#_Toc103263038)

[4. Proposed Idea… 10](#_Toc103263039)

[Why haven’t we see existing many applications on law enforcement implemented yet? 10](#_Toc103263040)

[Idea 10](#_Toc103263041)

[Application architecture 11](#_Toc103263042)

[Database 11](#_Toc103263043)

[5. Experimental Results (Testing/Evaluation) 16](#_Toc103263044)

[6. Conclusion 17](#_Toc103263045)

[7. Future Work and Directions 18](#_Toc103263046)

[8. Bibliography 19](#_Toc103263047)

**List of figures**

|  |  |
| --- | --- |
| **Figure Number** | **Page** |
| **(1)** | **11** |
| **(2)** | **13** |
| **(3)** | **25** |
| **(4)** | **25** |

**List of tables**

|  |  |
| --- | --- |
| **Table Number** | **Page** |
| **(1)** | **14** |
| **(2)** | **20** |

**List of Abbreviations**

|  |  |
| --- | --- |
| **Abbreviation** | **Text** |
| BD | Big Data |
| CC | Cloud Computing |
| IoT | Internet of Things |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# Introduction

Εδώ θα μπεί πρώτα το κομμάτι 2 του Ethical Form…

Έπειτα θα πρέπει αφού εισάγουμε τον αναγνώστη στο θέμα και τη σημαντικότητα της εργασίας να αναφέρουμε τον σκοπό και τους στόχους της εργασίας (κομμάτι 1 του Ethical Form)…

## Current Study Overview

## Scientific Contribution

## 1.3 Outline

Εδώ θα μπεί το κομμάτι 3 του Ethical Form….

# 2. Related Work

The internet of things is a very sophisticated network of devices that are capable to connect with each other and provide a very useful service. These devices, can either gather useful information via sensors, interact with the world through various actuators or do both. Throughout the years Drones evolved significantly and became an essential part of the fascinating internet of things and have provided a wide range of services through various applications. It is no secret that drones will control the low aerial space in years to come. With the increasing number of drones into the low aerial space traffic management has become challenging, also due to very sensitive data transfer via drones security concerns have raised that need to be accessed.

Labib, N.S., Brust, M.R., Danoy, G. and Bouvry, P. (2021). The Rise of Drones in Internet of Things: A Survey on the Evolution, Prospects and Challenges of Unmanned Aerial Vehicles. *IEEE Access*, 9, pp.115466–115487.

The Internet of drones (IoD) is the layered network architecture that drones utilize to coordinate navigate to the lower aerial space. It is conducted into layers:

Lin, C., He, D., Kumar, N., Choo, K.-K.R., Vinel, A. and Huang, X. (2018). Security and Privacy for the Internet of Drones: Challenges and Solutions. *IEEE Communications Magazine*, 56(1), pp.64–69.

In recent years, UAV applications for civil and military purposes have been increased significantly. This is because drones are very cost efficient to maintain and can provide very useful information with a bird-eye-view and deliver packages in hard-to-reach locations. Some fields that drones are used are courier services, search and rescue operations, security surveillance etc. Artificial intelligence and more specifically machine learning has a very solid contribution into the development of more complex and utilitarian applications. Therefore, this broad range of applications have caused security threats that need to be managed. Drones handle very sensitive data with a form of audio video or image through communication channels such as WIFI which is not the most secure protocol for data transfer. The biggest security concerns are spoofing, false data injection, jamming etc. The most common way to counterattack these threats is via data encryption during transform.

Shafique, A., Mehmood, A. and Elhadef, M. (2021). Survey of Security Protocols and Vulnerabilities in Unmanned Aerial Vehicles. *IEEE Access*, 9, pp.46927–46948.

The number of drones that are been used for a variety of applications is increasing rapidly day by day. The latter has raised security safety, and privacy concerns. Drones are not always designed with safety in mind and that can lead to unpleasant events such as physical accidents for example (Drones falling and injuring civilians). Furthermore, research have shown that drones can be vulnerable to spoofing, malware infection, data interference and injection, Wi-Fi jamming etc. Additionally, a big issue is the possibility of violation of personal space, drones can reach places and record video or take photos of people without their consent. Based on data that was collected by the Canadian Public safety such incidents have caused a lot of trouble and had led to blackmailing and other unpleasant events. The concerns that were mentioned above can be counter-attacked through data encryption, multi-factor authentication protocols, anti-malware software, and strict legislations according to flight protocols for UAVs from governments.

Yaacoub, J.-P. and Salman, O. (2020). Security Analysis of Drones Systems: Attacks, Limitations, and Recommendations. *Internet of Things*, [online] p.100218. Available at: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7206421/.

# Applications of drones and potential

## Drones in agriculture

Agriculture is a sector that UAVs have been used extensively. One area that drones are utilized in agriculture is on monitoring and assessing crops with remote sensing. Furthermore, drones are also used for precision distribution of agricultural chemicals and biological control agents. Additionally, they can be used to monitor the health of animals and retrieve samples remotely.

Remote sensing via a camera is an old invention and its first application dates way back in the mid-1800s. In World War 2 soldiers captures infrared photos from hot aired balloons and airplanes and utilized them to detect camouflaged military equipment and facilities. Nowadays, infrared photos that are captured from drones are used to assess agricultural development process and ensure its integrity. Therefore, by measuring the reflectance between visible and near infrared light, vegetation health and growth potential can be assessed and that can be very useful to farmers.

In agriculture, the conditions under the data are collected are essential. Light reflects differently depending on the type of surface. Therefore, several different types of drones are used with different kind of sensors for different purposes. The two main types of sensors that are used in agriculture drones are passive and active sensors. Active sensors have the ability to emit energy and detect the reflection of that emitted energy, on the other hand passive sensors can only measure the emitted or reflected energy from a scene. Passive sensors are more cost-efficient than active sensors, but they do not work well when ambient conditions have a strong presence in the scene. Meanwhile, active sensors are heavier and less cost-efficient, but they are capable of producing reliable data on variated ambient conditions.

Some examples of existing applications are:

* **Seedling emergence assessment.** There is a need for constant observation of crops during the first stages of crop development. This is because if something goes wrong in the early stages of the development process due to environmental factors, farmers have the ability to replant and potentially save a damaged set of crops. The latter is achieved through emergent mapping and measurement of seedlings. Drones can collect very high-resolution images and provide information on whether germination is unsuccessful in a specific area.
* **Crop damage assessment.** During the crop developmentprocess, it is very likely to have damages. Damages can be caused by unwanted weather conditions, insects, etc. Drones can calculate the area and the size of the damage and provide very useful information.
* **Water management.** One of the biggest challenges in agriculture is water management. One common way that farmers use to manage water supply is soil moisture sensors. This is not the most efficient way to manage this issue because results can be extracted from these sensors only when severe damage on crops have already taken place. Drones can have a very important role into gathering useful information that can be used to manage faster and more efficiently water distribution.
* **Livestock applications**. Recently agriculture have seen an increase usage of livestock drone applications. This because drones are a very cost efficient and at the same time can provide high quality video. Livestock monitoring can provide services such: observing animals behaviors or infrastructure that is responsible to keep animals in one place like fences and gates. Also, thermal sensors in drones are used to detect if animal have fever and therefore aid into preventing diseases via early diagnosis (Merwe et al., 2020).

## Drones in health care

Another sector that drones are used is health care. Below ways that drones are in health care are analyzed.

* **Medical transportation.** In many cases across the globe medical transportation is a problem due to geographical issues. The latter can be overcome because drones can move in an incredibly high speed of 40-60 miles per hour and cover great distance in very short time. Therefore, drones have been used for all short of medical transportations Mexico has many regions that are tough to reach, recently a company named Aidronix in Mexico began the development of an innovative application that its goal is to distribute medical supplies such as medications, vaccines antibiotics etc. to rural regions of Mexico(Wulfovich, Rivas and Matabuena, 2018). Moreover, blood is very important and often lifesaving, but many times there is shortage, especially in African countries. The most common way to deliver blood in most African countries is via an ambulance or car. Because Africa does not have a good road network blood deliverance is often delayed. In 2016, San Francisco Bay–based Zipline began a drone delivery operation in Rwanda, that increased the speed and deliverance of blood. More specifically, between 2016 and 2019 over 4000 missions have completed, that delivered 7000 units of blood, 1/3 of these missions were lifesaving situations (Ling, G. & Draghic, N. ,2019).
* **Emerging cardiac care.** In the USA, cardiac arrests that happen outside the hospital are one of the most common mortality reasons. This type of drone application is emerging and currently under development. It is likely that drones can be automatically deployed and transfer AED (automated external defibrillators) with the help of GIS systems. The latter can save valuable time, aid significantly in a potentially crucial situation, and save a human life. Various studies showed that AED transformation was able to reach faster its destination than traditional ways (Zègre-Hemsey et al., 2018).

## 3.2 Drones in law-enforcement

In many different cases, drones have played a significant role in providing not only a basic form of security but also been involved in many, different law enforcement scenarios. In detail, and by their first appearance in 1936 by Lt. Cmdr. Fahrney, who was developing an Aircraft Project, managed to introduce the term of the drone in his report. Soon drones have been introduced as military devices that were described as Unmanned Aerial Vehicles, specifically in 1946 in the United States. Most importantly, the name “drone” was taken from the male bee word from the Old English term and the systems that divide them were named as Unmanned Aerial System (UAS), Small Unmanned Aerial System (sUAS), Micro Aerial Vehicle (MAV), Unmanned Aerial Vehicle (UAV) and finally the Remotely Operated Aircraft (ROA), who all manage to make the technological world take over the public and private service of our community (Jones, 2019). During the last years, drones, as told above, have played a very important role in military operations and especially in the Iraq and Afghanistan wars. Moreover, drones have played a significant role in public maintenance such as on surveying roads, equipment, and pipelines, specifically in the Prudhoe Bay area of Alaska. As we may also have seen, drones have also managed to be included within the transportation sector and even the delivery of products, with many large-scaled companies to be a part of that new norm. With drones to be a huge part of the present and especially the upcoming future, farmers have also maintained to keep these changes for themselves and worked with drones in order to count and check their cattle and monitor their fence conditions throughout their lives and during their work (Jones, 2019). Although drones are an important asset in the everyday lives of people during their work, they are also taking over the marketplace and in sometimes even replacing smartphones as gifts. In last years, there were people who wished to have a smartphone on their holidays, but instead these days there are even children that wish to have a drone just for themselves, with the latter showing how important they have been lately. In terms of law enforcement again, currently 910 public safety agencies own drones and the law enforcement purchases have managed to be increased by two thirds of the last.

However, despite how important and helpful drones can be and will be soon even more, governments had to decide on what should be better for the safety of citizens and police officers to be included within the basic work and functionalities. Specifically, the military, in terms of war, was testing the required tools to make the trade. These tools of the trade were including both the tactics and the equipment of the drones themselves. Specifically, history has shown that in different circumstances the military and law enforcement in general have made a transition from military application to tactical equipment (Jones, 2019).

As history also suggests and shows, the use of drones in the military and in every military mission that is available, will be also available for the law enforcement. This means that their efficiency, effectiveness, and economical value are now tested in order for them to be used in more dangerous or important cases that may require saving human lives. In addition, the use of drones in the private and public sectors has been increased largely over the last decade and this shows us how important they are also in terms of the aircrafts. Specifically, in 2012, the FAA (Federal Aviation Administration) Modernizations and Reform Act was using drones in order to safely guide other aircrafts into their domestic airspace and thus guide them to their desired goal (Jones, 2019). With this way, they were tasked to develop their regulations and make them relevant for the use of UAS for public spaces. Regarding the FAA, they require for the free use of drones in the law enforcement guidelines that also persist of the Fourth amendment considerations that use force and arming the drones themselves and thus making them lethal. This law has been enforced by many states and the legislation was pertaining the use of drones themselves only by the law enforcement. However, many states also provide inconsistent information for the legislation, which also makes it difficult to find a middle line between the requirements for the drone policies and the actual purpose of the legislation for the use in law enforcement.

Many research methods have indicated that due to legislation and regulatory analysis, which eventually leads to uncertainty, many law enforcement agencies have not yet developed their acceptable policies for the right use of drones, and maybe not even drones themselves within their premises. This shows that they were all designed to showcase the procedures for the armed drones and also display a proactive approach to the armed drone policy development for law enforcement (Jones, 2019). As more and more legislations have come to the surface and make themselves visible, since the application of the Fourth Amendment, in most cases law enforcement have to obtain a warrant in order to search a property or an individual themselves and assume their respected working scenarios except in specific cases, where these cases will later be examined by the court. However, most of the states have managed to enact legislation which governed the use of drones for the law enforcement, and all of that happened by the end of 2017. Although, everything was functioning well, these laws have been extremely restrictive for drones and caused certain confusion that only can be addressed when an incident occurs, and the actual results are decided in the court of law. Specifically, any unlawful action of drones that assume that surveillance of an individual on their owned property or residence, could cause a violation of their Fourth amendment and lead right to privacy concerns on the new technology (Jones, 2019).

## Computer Vision ([LINK](https://www.tutorialspoint.com/artificial_intelligence_with_python/artificial_intelligence_with_python_computer_vision.htm))

Computer Vision is simply explained to be a form of modelling and replicating vision that applies to human abilities and is also used in computer software and hardware. Specifically, it is a discipline that includes a variety of ways to rebuild, understand and interrupt a 3D scene from the 2D images that represent the scene and adjust its properties and the structure of the scene. There is a variety of hierarchical values for computer vision and mostly these values are divided into three categories that can be based upon certain things and circumstances. In detail, the low-level vision, includes basic processing of images with the sole purpose of extracting the features for the final results. Furthermore, the intermediate-level vision is able to include some basic object recognition and the final interpretation of a 3D scene. Finally, for the whole categories purpose in the computer vision hierarchy, the high-level vision, as a final step, includes a basic and conceptual description representing an activity, behavior and intention of a scene.

## Computer Vision and Image Processing

Specific processing studies about images are considered to be the main aspect of image processing. Specifically, the input and output of the image processing situation are the actual both images that are about to be processed. Computer vision is explicitly constructing meaningful descriptions of certain objects from their image that are physical and the final output of the computer vision process is the actual description or the interpretation of the structures in a 3D scene (Tutorialspoint, 2022).

In terms of the final applications and how computer vision finds them inside specific fields, first the robotics field specializes in specific sectors. In detail, with robotics, Computer Vision manages to accomplish localization and determine whether a robot is located automatically and the navigation of the technology. Furthermore, it is able to avoid specific obstacles and also in most situations assembly, by peg-in-hole, welding or painting. Also, it achieves manipulation, for example with PUMA robot manipulator and finally the Human Robot Interaction (HRI) which achieves more intelligent robots that are also able to interact and serve people (Tutorialspoint, 2022).

For medicine, the classification and the final detection of Computer Vision can be achieved with lesion or cells classification and in generic cases even tumor detection. Moreover, 2D and 3D segmentation with also 3D reconstruction of human organs with the help of famous features and capabilities, such as MRI or ultrasound. Finally for medicine, technology has also evolved in the surgery part of the section, which also includes vision-guided robotics surgery (Tutorialspoint, 2022).

Security has also played a significant role within the technology sector of Computer Vision, which also requires biometrics, such as fingerprint scanners, iris detectors and face recognition systems and finally surveillance-detecting activities and behaviors (Tutorialspoint, 2022).

For transportation purposes, autonomous vehicles and safety issues often dissolved with driver vigilance monitoring for safer distances and driving security (Tutorialspoint, 2022).

For the final category, industrial automation applications can lead to industrial inspections with defect detection techniques and assembly just like the robotics category. Also, the barcode and package label reading all include the industrial automation application. Finally, object sorting and OCR for document understanding techniques can also include the final results for this category (Tutorialspoint, 2022).

## Useful Computer Vision Packages

Regarding some specific packages for the Computer Vision technique, there are certain features that also help provide best behavior for the packages that are able to be installed. In specific, for computer vision functioning with Python, there is a very popular library that is called OpenCV (Open Source Computer Vision). This is a library that makes use of specific programming functions that fully aim to have a real-time computer vision. The library is written in C++ and it is the primary interface for the language overall. For this specific scenario, the most available and used feature is the code method below:

*pip install opencv\_python-X.X-cp36-cp36m-winX.whl*

The value X represents the actual Python version of the machine that it is installed into and of course the win32 or 64 bit system that it is running on the specific computer. For anaconda environments in specific, the following code can be used to install the OpenCV environment:

*conda install -c conda-forge opencv*

For the specific reading, writing and displaying each image separately, most of the CV applications can easily get images and input and of course produce the actual desired images as output. For instance, specific function can be used in terms of how crucial it is to follow functions for the sole purpose of making the images function properly and are divided in three different sections. Firstly, the imread() function, is able to read the image and it is used solely for this specific purpose. OpenCV imread() is able to read and support different kinds of images and their formats, such as PNG, JPEG, JPG, TIFF and many more. Secondly, the imshow() function, is able to function with the sole purpose of showing the image to the window and the window then itself will automatically fit the image itself to the desired image size, finally allowing the program to support various image types, such as PNG, JPG, JPEG, TIFF and many more, just like the first step. Finally, for the third step, the inwrite() function, manages to write the image, which allows the OpenCV imwrite() function to support various image types, just like the three other categories mentioned before, which include, PNG, JPG, JPEG, TIFF and many others (Tutorialspoint, 2022).

### OpenCV

As previously mentioned for the OpenCV functions and their specificity when it comes to imread, imwrite and imshow functions, there are specific examples that can be described and analysed for the final purpose of the essay. Specifically, the following example is able to show in Python code, how an image is being read in one format and showed in a window and finally written in another format with the same type of the image. At first, we are able to import the general OpenCV package by importing the cv2 package itself inside the code:

*import cv2*

After doing the basic installation of the package, we can now read a particular and specific image with the help of the above mentioned imread() function which is clearly shown below in a self-declared image variable:

*image = cv2.imread('image\_flower.jpg')*

With the sole purpose of finally showing the image to be displayed, we use the imshow() function with the name of the window to show the image with the name **image\_flower** and finally providing us with the result from the below taken code:

*cv2.imshow('image\_flower',image)*

*cv2.destroyAllwindows()*



Figure : The final result of a taken image (Tutorialspoint, 2022)

With the below taken code we can clearly observe that the same image can be altered to another type by providing another name for it with the imwrite() function:

*cv2.imwrite('image\_flower.png',image)*

The actual output of the above code will soon provide us with the true value, as this represents an actual PNG icon, and the final wish of deleting the actual created window with the flower is achieved with the end method of destroyallWindows().

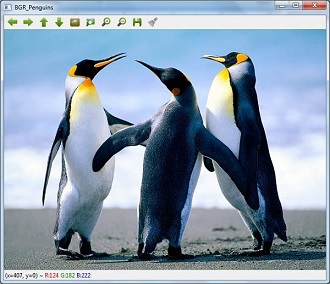
In other words, the color space conversion for the OpenCV language includes a variety of other different technologies and paradigms that are often included within such categories. For the OpenCV color space conversion, the images are not stored within using the conventional RGB color. However, they are stored within the reverse order which often is the BGR order. This includes and sums how the default color code for reading each image is BGR and also the cvtColor() function for the conversion is converting the image from the first color to another Tutorialspoint, 2022.

For example, if we’d like to consider converting an image from BGR to grayscale, we would use a certain and specific method, by first importing the library and then applying the imread() function in order to final show it with imshow() just as the code below shows.

*import cv2*

*image = cv2.imread('image\_flower.jpg')*

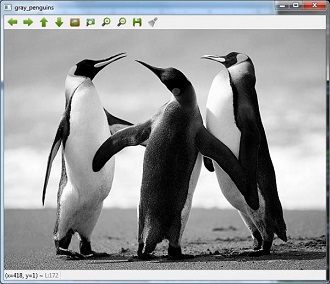
*cv2.imshow('BGR\_Penguins',image)*



In order to provide the best method in order to show the above image into grayscale we just need to confirm the previous steps and also use imshow() again in order to apply the new image as grayscale, just like it is shown below.

*image = cv2.cvtColor(image,cv2.COLOR\_BGR2GRAY)*

*cv2.imshow('gray\_penguins',image)*



OpenCV can also be displayed in edge detection which achieves further details and advice can be given also for the final result. Edge detection is allowing humans to see through a rough sketch and can easily recognize some object types and their poses which they have. The latter is achieved because the edge can play a significant role for humans and in the applications of computer vision overall. For image and edge detection OpenCV provides a mostly simple and useful function which is called Canny() and the sole purpose of it is to detect the edges of the image. For example, after importing the main Computer Vision library and the numpy as np for the main further library methods, we can clearly read the following wishing image to be edited and then imread function should be used properly as a declared variable. Additionally, with the Canny() function, we can detect the edges of the image we would like to edit and after doing that, with the help of imwrite() function we can clearly detect the edges. Finally, and as shown below, we use imshow() to show the edges of the image and imshow() provides us with the final result.

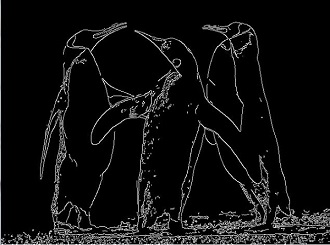
*import cv2*

*import numpy as np*

*image = cv2.imread('Penguins.jpg')*

*cv2.imwrite(‘edges\_Penguins.jpg’,cv2.Canny(image,200,300))*

*cv2.imshow(‘edges’, cv2.imread(‘‘edges\_Penguins.jpg’))*



Except from the main and basic methods of image detection shown above, there are two other techniques that require face and eyes detection. Firstly, for face detection, it represents a fascinating application of computer vision that also makes more realistic and futuristic display. With the OpenCV built-in functionality the library is able to perform the face detection method and with the Haar classifier for face detection, it can finally be done (Tutorialspoint, 2022).

At first, with the Haar cascade classifier data we can find the actual dataset in the OpenCV package and import them directly into the code. For the actual code and the finaly face detection system to be working properly, we can define a variable with a base name of “face\_detection” and provide the code with the necessary path in order to read the xml data directly. Then, with the imread() function the image is read and finally grayscaled with the color functions. Finally, the image takes a detection testing for faces and draws a rectangle in order to provide the user with the detected final face from the image.

*import cv2*

*import numpy as np*

*face\_detection=*

*cv2.CascadeClassifier('D:/ProgramData/cascadeclassifier/*

*haarcascade\_frontalface\_default.xml')*

*img = cv2.imread('AB.jpg')*

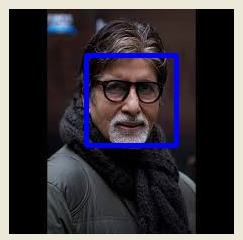
*gray = cv2.cvtColor(img, cv2.COLOR\_BGR2GRAY)*

*faces = face\_detection.detectMultiScale(gray, 1.3, 5)*

*for (x,y,w,h) in faces:*

*img = cv2.rectangle(img,(x,y),(x+w, y+h),(255,0,0),3)*

*cv2.imwrite('Face\_AB.jpg',img)*



Finally, except from the Face Detection application on computers with the help of OpenCV, the system provides us with another well-structured and organized application for eye detection. Eye detection is another application for Computer Vision, which is able to make the appliance of images more realistic and futuristic at the end. OpenCV has a functionality that is able to detect eyes from an image and the Haar cascade is used also in this part (Tutorialspoint, 2022). Specifically, in order to define and finally detect the eyes of a person within an image we should be able to make the first configurations within the code and import the basic libraries for cv2 and numpy as np for the code, just like the code snippet below.

*import cv2*

*import numpy as np*

After that, using the Haar cascade classifier for the detection of the face we declare a basic variable just like the other previous methods, and import the actual classifier within our code to finally be able to read the image with the well-known imread() function.

*eye\_cascade = cv2.CascadeClassifier('D:/ProgramData/cascadeclassifier/haarcascade\_eye.xml')*

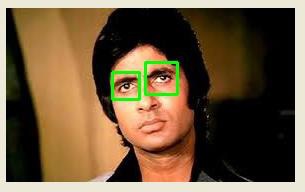
*img = cv2.imread('AB\_Eye.jpg')*

As the first configurations for the final detection of the face directly from the image have been taken, we should now be able to grayscale the image itself and then detect the multiscale for finally performing the actual face detection. With this way, we can import the detectMultiScale method and grayscale the image itself as shown below.

*gray = cv2.cvtColor(img, cv2.COLOR\_BGR2GRAY)*

*eyes = eye\_cascade.detectMultiScale(gray, 1.03, 5)*

With the drawing of the grayscale method and the final cuts for the result on face detection, we can now provide the algorithm with the loop that draws the rectangle for each eye and can also detect the face directly from the eyes of the individual shown within the image (Tutorialspoint, 2022).



# 4. Proposed Idea…

Drone applications have a vast range of applications on numerous fields some of which are mentioned above. The field that intrigued and inspired me to propose and implement an application is the field of law enforcement. So far, we have seen many applications of drones related to law enforcement, but many countries have not implemented them yet. The latter is mainly because strict legislations and level of complexity I will analyzed and present these reasons in detail below. My proposed idea and implementation are divided into key areas of which I will analyze and present separately.

### Why haven’t we see existing many applications on law enforcement implemented yet?

Although many existing implementataions related to the field of law enforcement have already been developed and tested, they have not been integrated into the system and that is because of two main reasons.

The first one is because of very strict legislation regarding UAVs. A UAV has very identical legislation to a regular aircraft. The latter means that there must be an infrastructure that has to be followed for every application and that included a ground control unit, well-trained personnel, and a very robust infrastructure of networking. These requirements are not easy to have in every law enforcement unit because of cost inefficiency and sometimes complicated bureaucracy. Also, the fact that police drones are dealing with civilian data raises privacy concerns that are hard to deal with and perplexes bureaucracy issues even more. Finally, considering the fact that these drones will fly in the low aerial space of civilian regions safety concerns are becoming an extra issue (for example a drone falls off the sky and injures a civilian).

The second reason is technical challenges. More specifically, drones especially inexpensive drones have a short flight duration and that is because most of them use lithium batteries that do not meet the requirements of the desired power consumption. Considering the latter and given the fact that police patrols often last many hours, a time issue arises. Furthermore, police drones have mostly been used for surveillance. A surveillance operation requires a pilot to operate the drone and an officer that is occupied with the surveillance part of the given purpose. It makes sense to try to automate some of the tasks to reduce the number of personnel required for surveillance. However, if for example, a pilot is replaced with an automated flight system security concerns will arise because a drone will fly differently in different weather conditions and that can lead to unwanted accidents.

### Idea

Many times, law enforcement units patrol specific regions to detect criminals that already exist in the database of police departments. The latter often requires long hours of patrolling and investigation that can be time-consuming, cost-inefficient, and sometimes produce no results. So, it makes sense to try to automate and enhance the procedure of criminal detection by any means. In my opinion, drones can strongly contribute to this area through various methods. Some very high-quality drones have integrated ultra-high-resolution cameras that have the ability to record or live stream video directly to a server. The latter can become extremely beneficial and strongly contribute to criminal detection if it is integrated with artificial intelligence.

Artificial Intelligence is a field of computer science that has seen rapid development in recent years and has strongly contributed to the solution of difficult and multiplex problems in the real world. In recent years one area of artificial intelligence that has seen a massive improvement is computer vision.(mia phgh na ekshgei). Computer vision is the field of computer science whose purpose is to obtain data consisted of images or videos and apply machine learning or deep learning algorithms to extract useful information that will aid various fields. For example, many smartphones today use computer vision to unlock the screen of the phone. The latter is achieved after the image processing and recognition of the owner’s face are completed with the front camera. Computer vision has been used for many other reasons such as object detection object tracking etc.

The general idea is to use these drones with super useful cameras and automatically detect the faces of criminals that have already been inserted into the databases of law enforcement units. The application aims to function in real-time, so it has to be as light as possible. This is not a simple task, it requires a very robust software architecture to reduce the processing power that is necessary to achieve these results, I will analyze this architecture in detail in the next chapter. Also, the drones must be directly operated from the ground control unit so there is a need for an input system to control the drone which I will also present in the following chapters. Furthermore, to identify the faces of specific criminals there is a need to create an artificial intelligence model that is essential to perform face recognition. In order to achieve the latter in the next chapters, I will explain the data preprocessing techniques as well as the training methods for the model. Additionally, given the fact that the datasets of criminals are huge, separate datasets will be created for every zone to reduce the processing power and provide efficient results. Finally, a database schema will be proposed that implements a strong infrastructure that aims to improve the data retrieval speed.

### Technologies

Many modern technologies were used for the implementation. Below I will analyze the technologies as well as programming libraries that were included during the development process.

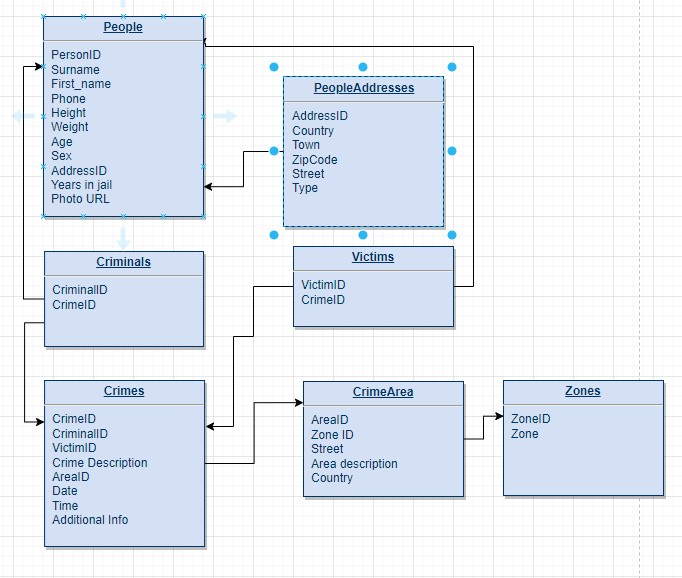
First of all, python is the programming language that the application was written. The latter is because python had already many useful libraries related to computer vision and drone programming. The drone programming library that is used for this application is called tello and it was published by Gabriel Heinzer (<https://github.com/gheinzer/python-tello>). This library is specifically designed to program the tello drone which is manufactured by ryze robotics. Furthermore, for the artificial intelligence part of the application I used opencv. Finally, for the database I used XAAMP and MySQL combined.

### Application architecture

The application consists of four essential parts the database schema, the artificial intelligence algorithm, the training method, and the input system.

### Database

### Database schema

The database that is used for this application is MySQL. The database schema has in total 7 tables which are: people, people addresses, criminals, victims, zones, crimes, crime area, and zones. Below I will analyze the table fields and the relationships between the fields.

To understand better database schema below a brief description of each table and each fields:

Crimes: Crimes is a table that records a crime that has been committed and stores the necessary information.

Crimes fields:

* CrimeID: This field is the primary key of the table and uniquely identifies each record.
* CriminalID: This field Is foreign key to Criminals table.
* VictimID: This field Is foreign key to victims table, and it uniquely identifies each victim.
* CrimeDescription: This field describes the crime and is not null.
* AreaID: This field is a foreign key to table CrimeArea it can take null values.
* Date: This field stores the date that the crime was committed.
* Time: This field stores the time that the crime was committed.
* AdditionalInfo: This is an optional field, and it stores any additional information related to the crime.

CrimeArea: CrimeArea is the table that stores all the information needed for the location of the crime.

CrimeArea fields:

* AreaID: This is the primary key of the table and it is also a foreign key to the table Crimes.
* ZoneID: This field is a foreign key to table Zones.
* Street: This field stores the street where the crime was committed.
* AreaDescriprion: This field describes the area that the crime was committed.
* Country: is the field that stores the country that the crime was committed (All the data in this dataset have the same value “Greece”).

Zones:

This table stores the zone in which the crime was committed. Each zone refers to specific regions within a country.

Zones fields:

* ZoneId: This field is the primary key of the table.
* Zone: This field stores the regions in which each zone is stored. An example value is: “Epirus, Thrace, Makedonia, Thessaly.”

Criminals:

This is the table that stores the information needed to identify a criminal.

Criminals fields:

* CriminalId: This field is the primary key of the table, it is also a foreign key in table People.
* CrimeID: This field is a foreign key to table Crimes and it basically stores the which were committed by this criminal.

Victims:

This is the table that stores the information needed to identify a victims

Victims fields:

* VictimsId: This is the primary key of the table, it is also a foreign key in table People.
* CrimeID: This field is a foreign key to table Crimes and it basically stores the which were committed by this criminal.

People:

This table stores all the necessary information about both criminals and victims.

People fields:

PersonID: This field stores is the primary key of the table, it is also a foreign key in tables criminals and victims.

* Surname: This field stores the person’s surname.
* First\_name: This field stores the person’s first name.
* Phone: This field stores the person’s phone, it is not a mandatory field.
* Height: This field stores the person’s height, it is not a mandatory field.
* Weight: This field stores the person’s height, it is not a mandatory field.
* Age: This field stores the person’s age it is, not a mandatory field.
* Sex: This field stores the person’s gender it, is not a mandatory field.
* AddressId: This field is a foreign key to the table people address.
* Years in jail: This field refers only to records where the person is a criminal if the person is a victim most likely will be filled with “-”. It stores the total years if any a criminal has spent in prison.
* Photo URL: The photo URL is the field that holds the value of the image path that is stored in the server. The photo that is stored in the URL is showing a clear photo of the criminal or victim. This phot is used in the graphical user interface when a criminal is recognized.

People addresses:

This table stores the criminal’s and victim’s address details.

People addresses fields:

* AddressId: This is the primary key for the table
* Country: This field stores the person’s country of origin.
* ZipCode: This field stores the person’s zip code of residence.
* Street: This field stores the person’s street of residence.
* Type: This field stores the type of person’s residence. (For example. A person can have both work and a permanent address).

### Database relations

To make the database more effective relationships have been created between tables, below a brief description of the database relations.

**People table**: This table has three relations. The two relations are with the same field personId and the type is many to many with the crimes criminal id field and victims victimId accordingly. The third is with people address table and it is a one to many relationship.

**Crimes:** This table has also three relationships. The first relationship is with table criminals on field crimes.crimeID -> criminals.crimeID. The second relationship is with table victims and it is on field crimes.crimeID -> victims.crimeID. The third relationship is with table crimes area and it is with field crimes.AreaID -> crimesArea.AreaId. All three relationships are of type many to many.

**CrimeArea:** This table has one relationship with table zones on-field CrimeArea.zoneID -> Zones.zoneID. It is a many-to-many type of relationship.

### Database requests

The user can choose two database requests for retrieving criminal depending on zones with two different functions: The first function is “local\_known\_criminals” and it is used to get the dataset related to the criminals that have committed crimes within the zone. The second function is: “remaining\_known\_faces” and it used to get the datasets of criminals that have committed crimes outside the operating zone. Each function will be called according to the desired functionality, but most of the times it makes sense to call the first function.

The datasets of criminals are spitted into zones, in my implementation I have three zones that covered all the land and islands of Greece. More specifically, the zones are:

* Zone 1: Epirus, Thrace, Macedonia, Thessaly.
* Zone 2: Central Greece, Peloponnese, Ionian islands.
* Zone 3: Crete, Aegean islands.

The distribution of areas in zones aims to improve the efficiency of the machine learning algorithm. If for example, the dataset was the same for all zones it would make the algorithm very dysfunctional due to its enormously large size. By splitting the datasets into zones, the algorithm’s efficiency increases significantly because the datasets are much smaller. It makes sense to create even more zones, but this is totally dependent on the total amount of data that exists in each dataset.

Finally, if a criminal is detected the user retrieves all the necessary information needed to display to the screen with function “GetCriminalDataById” that returns a dictionary.

### Input system

To make the application directly usable from the ground control unit I provided an input system that can be utilized by any standard keyboard. It is worth mentioning that this is not something mandatory to the general functionality of the software. Nevertheless, below I will analyze the keys and the functionality of each key:

The keys used for the application are: “Q”, “E”, “P”, “W”,” A”,” S”,” D”, “LEFT ARROW”, “RIGHT ARROW”, “UP ARROW”, “DOWN ARROW”.

* Q: This key is used for drone take-off.
* E: This key is used to land the drone.
* P: This key is used take a photo if needed. (This functionality is not in the main part of the software).
* W: This key is used to move the drone forward.
* A: This key is used to move the drone backwards.
* S: This key is used to move the drone left.
* D: This key is used to move the drone right.
* RIGHT ARROW: This key is used to rotate the drone clockwise.
* LEFT ARROW: This key is used to rotate the drone anticlockwise.
* UP ARROW: This key is used to raise the altitude of the drone.
* DOWN ARROW: This key is used to lower the altitude of the drone.

To get the user input I used “pygame” which is a very famous python library for game development with python. In this application to the get the input the user has to click inside a black window that will pop pup automatically when the application is initialized. To prompt that window I used the script in the image below:Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

Moving forward, to take the input user every frame I created a python function that returns a dictionary with the input from the user.Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

Finally, to convert the user input into drone command I used the predefined function from the tello python library:



The instructions variable is the dictionary that is been returned from GetKeyboardPressed function.

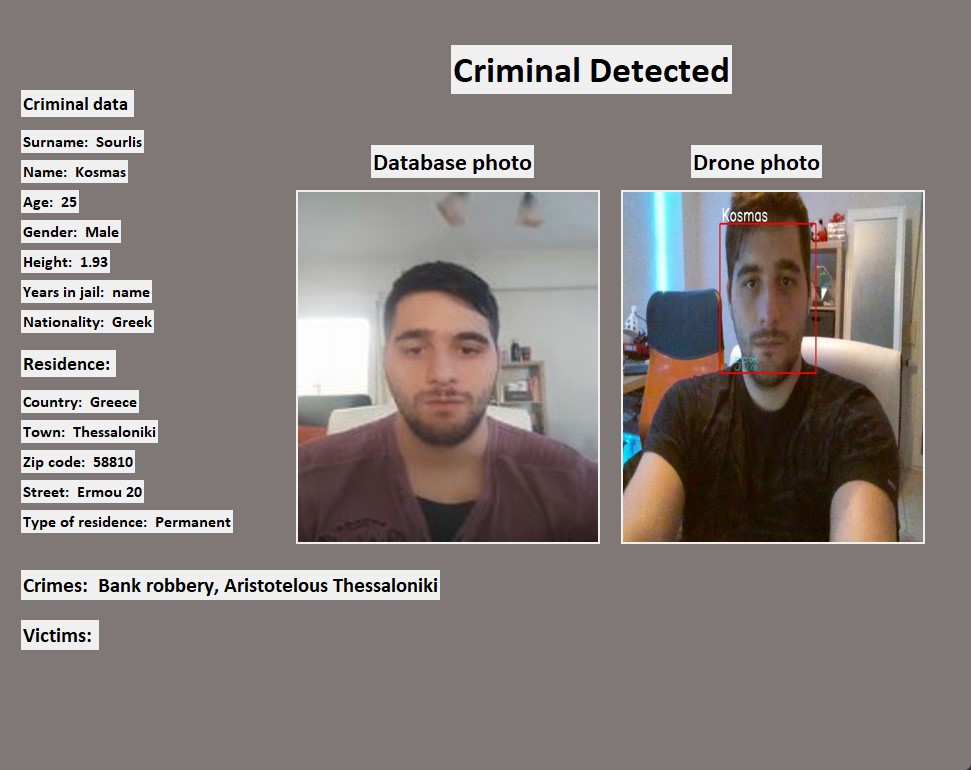
### Graphical user interface

When the drone detects a face that has a match equal or greater to 55% with a criminal face, a graphical user interface is prompt into the screen with all the criminal’s known data and crimes. The libraries used for this GUI can be seen in the image bellow.

Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

Bellow you can see a demo detection image. All the data displayed are entirely fictional the face is mine. This image aims only to demonstrate how the app will work if a criminal is detected.



### Data preprocessing

Data preprocessing is something every data scientist must do before applying the artificial intelligence algorithm. For this application the data preprocessing is done with two different techniques. The first acts on an existing dataset, potentially 50 or 60 photos of a criminal face. The second is by recording a video and taking one photo per frame. Nevertheless, below I will analyze and present each way of data preprocessing.

Existing Dataset technique:

This technique requires an already existing dataset of criminal photos that have no other faces in it. The script below is used to crop each photo and modify the color of the picture.

Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

In the first three lines the libraries need for these operations are imported. Moving forward, the variable photos store the preprocessed images of the criminal. The line with the variable face\_detector uses the cv2 algorithm which recognizes the faces of the photo. Finally, the two nested loops iterate through all the preprocessed images, crop all the faces, and store them on the directory “user\_userID\_images” in grey color. If there is only one face on the photo the second loop will run only once, it should be only one face in the photo, if there are more than one faces in the photo the algorithm will be confused.

Video technique:

This technique can function on live video streaming or with an already recorded video. Nevertheless, the script below runs on live video streaming.

Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

This function is called every frame from a different part of the application so there is only one for loop. Similar to the first technique the face detector variables store faces from each frame with the usage of “cv2.CascadeClassifier” function. The color of which each photo is stored is grey. The name of each photo that is stored is random and therefore it is saved with the milliseconds of current time. The total photo of each user is 65 and that is restricted with the if statement: “if (length < 65)”. The length variable stores the number of photos stored for each user.

### Training the model

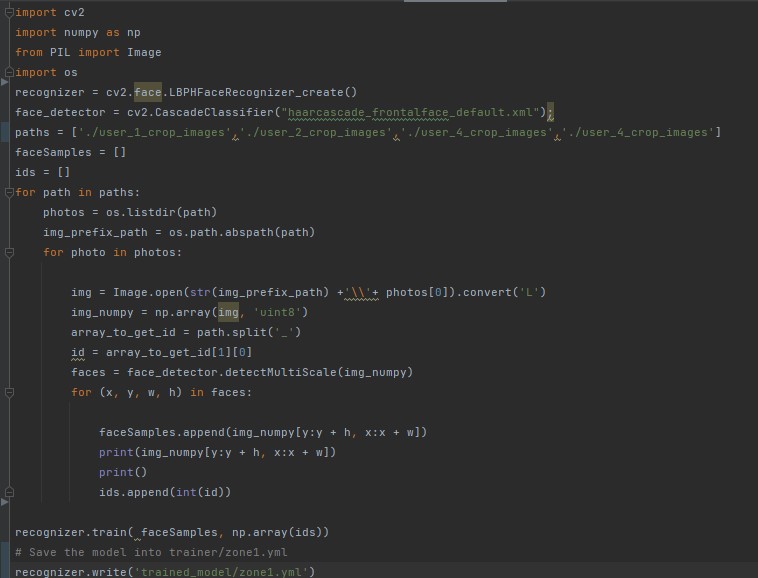
After preprocessing the data, the desired output of each photograph should be:

Εικόνα που περιέχει άτομο, άνδρας, εσωτερικό, τοίχος

Περιγραφή που δημιουργήθηκε αυτόματα

Figure 2. This is a sample photo that is ideal for the training algorithm

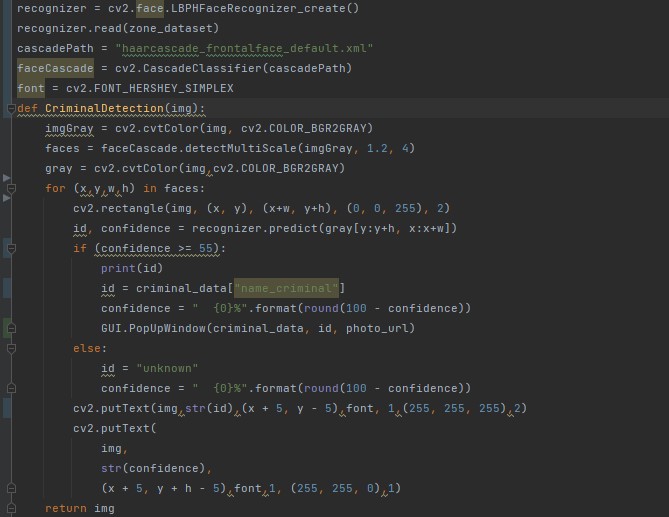
The algorithm that is used for training and therefore creating a dataset is LBPH which stands for (local binary pattern histogram). More specifically, in this application an implementation of OpenCV of this particular algorithm. This algorithm is very famous for providing extremely efficient results on frontal and side face recognition. In this application the algorithm is only trained for front face recognition. Also this algorithm works better when there al lot of light is the photo.

Bellow the script which I used to create to train the datasets

In the first for lines the libraries needed for training the model are imported. Moving forward variable recognizer stores the artificial intelligence algorithm that is needed to train the model, face\_detector variable stores the functionality that is required to detect the faces on the photograph, paths variable stores all the paths of photographs that that will be trained for each zone dataset, face samples and ids variables store the photos and ids of each criminal. There are three nested for loops the first one loops through each path of criminal photos, the second loop iterates through all photos of each criminals and the last loop must run only once because it must run for only one face. In the last 2 lines : “recognizer.train(faceSamples, ,nparray(ids))”, and “recognizer.write(“trained\_model/zone1.yml”)” the model is trained and the dataset is stored. Notice that each face is stored with a unique Id, this Id is the same to the database criminal Id.

### Recognizing the face in real time

After training the models and separate dataset have been create for each zone. Depending on the zone the user wants to detect faces different dataset will be implemented for the function. In the script below you can see the function thar recognizes the criminals faces.



In the first two lines the variable recognizer stores the algorithm that is able to provide that functionality needed to recognize a face from the desired dataset, faceCascade variable helps to identify any face from the video. The function CriminalDetection is called every frame and it detects any face present on the frame, The for loop will run only if a face is present on the image frame. If faces exist in the image frame, then it will iterate through all the faces, predict whether the face exist in the dataset and display a match percentage as well as the name of the criminal. The latter is been achieved with the command “recognizer.predict” and the popup screen only appears if the match percentage is greater or equal to 55%. If the face is not present in the dataset then the displayed name will be unknown and the match percentage will either be0% or a value below 10%.

# 5. Experimental Results (Testing/Evaluation)

In my implementation I tested two different data preprocessing techniques and concluded that the second with live streaming video was more efficient that the first one. The latter is because the first technique that used an existing dataset of photos had a wide variety of photos, and the lighting was different as well as the angle of the face, therefore it provided less efficient results compared to the live video streaming technique which recorder 65 images, each one with very similar angle and the same lighting. Also, I found out with both techniques the best number of images per face was around 50-70 if I trained the algorithm with more or less images per face then the results were significantly less efficient. Furthermore, I noticed that if the person were glasses or a mask it the success rate was substantially reduced. Moreover, I tested the application with ryze tech Tello drone which is a drone for mainly education purposes and realized that the drone only recognized faces from a close distance if I moved the drone further away the camera was not able to recognize the face and even if it did the match percentage was really low approximately 20-30%. Therefore, it makes sense to use a drone with a much better camera that can detect faces from greater distances and be more efficient.

Throughout the development process I tried to implement the face recognition part of the application with many different algorithms. The problem with these different algorithms was that almost no algorithm could provide satisfactory match percentages, and none was able to run on real time. Many produced better results but not on live stream. More specifically, keras\_vggface has many useful features but could not run-on real time.

# 6. Conclusion

# 7. Future Work and Directions

In my opinion my implementation is a great combination of emerging technologies and it produced very strong results. Nevertheless, I strongly believe that there are some feature works and improvements that can be made that would significantly improve the application. To begin with, the facial recognition algorithm implemented covers only the front of the face, in the future it can be improved so it can cover the side face also, Additionally, the algorithm can expand and also recognize a criminal from his movement (walk, run, etc) but that can be extremely difficult due to high level of complexity. Furthermore, when detecting a criminal an option can be given to the user whether he wants to automatically follow the criminal and track his movement. The latter will require an autonomous flight system and that would raise numerous security concerns, but it would help greatly the operator of the drone. Also, the input system can be significantly improved, it would be better not to use the keyboard and build a dedicated controller to have better control of the drone, control the speed the altitude etc. Another thing that can be improved is the algorithm itself, in my implementation I managed to get up to 80 percent match on a face, but that worked when the scene was filled with very good lighting. Therefore, it would make sense to develop an algorithm that would work well on darker environments and produce better results. Moreover, the application can be modified to function on regular CCTV cameras, that would be quite tricky because CCTV cameras often do not have great resolution. Finally, the database schema must be modified to fit the standards of each law enforcement unit, the existing database schema is generic, and it designed to be flexible and easily convertible. In order to implement this kind of application in society many obstacles need to be overcome, solution must be found on bureaucracy and eliminate security and privacy concerns.

# 8. Bibliography

Merwe, D., Burchfield, D., Witt, T., Price, K. and Sharda, A., 2020. Drones in agriculture. *Advances in Agronomy*, pp.1-30.

Ling, G. and Draghic, N. (2019). Aerial drones for blood delivery. *Transfusion*, 59(S2), pp.1608–1611.

Wulfovich, S., Rivas, H. and Matabuena, P., 2018. Drones in Healthcare. *Health Informatics*, pp.159-168.

‌ Zègre-Hemsey, J., Bogle, B., Cunningham, C., Snyder, K. and Rosamond, W., 2018. Delivery of Automated External Defibrillators (AED) by Drones: Implications for Emergency Cardiac Care. *Current Cardiovascular Risk Reports*, 12(11).

‌