الجمـــهوريـــة الجــزائريـــة الديـمقراطيـــــــة الشـعبيـــــة
PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA
وزارة التعـليـم الـعالـي و الـبحـث الـعلـمي
MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH
جـامعـة عمّـــار ثـليـــجـي بالأغـــواط
UNIVERSITY OF AMAR TELIDJI LAGHOUAT
كـليـــة الـعلــــوم

FACULTY OF SCIENCES

COMPUTER SCIENCE DEPARTMENT

MASTER THESIS

Domain : Mathematics and Computer Science
Field : Computer Science
Option : Networks, Systems, and Distributed Applications

by:
ZAOUI Meriem & GUERROUDJE Asma

THEME

Weapon Detection from Camera Footages Using YOLO and SSD models

Thesis defended publicly on *25-06-2022*

Jury Members

Dr. GUELLOUMA Younes	*Associate Professor*	President
Dr. GUIBADJ Fatna	*Associate Professor*	Examiner
Dr. BENAROUS Leila	*Associate Professor*	Supervisor

N ……… University Year 2021/202

بسم الله الرحمن الرحيم

Dedication

I dedicate this work, the fruit of many years of study to :

To my dear parents for their patience,

My dear mother, Thank you for your advice, your sacrifices, your support, and your encouragement

Papa my gratitude is not enough to express what you deserve for all your sacrifices since my birth, during my childhood, and even in adulthood

To my sisters Asma, Fatima, Charaf, Rahmouna and Khadidja

To all my friends, who supported me in the accomplishment of this humble work

To all my teachers and all those who have committed in these modest works

To all my Zaoui family,

And to all who have helped me from near or far for the realization of this work.

Zaoui Meriem

Dedication

I dedicate this work, the fruit of many years of study to:

To my dear parents for their patience,

My dear mother, Thank you for your advice, your sacrifices, your support, and your encouragement.

Papa my gratitude is not enough to express what you deserve for all your sacrifices since

my birth, during my childhood, and even in adulthood
To my sisters Habiba, Fatima, and Mafaz

To all my friends, who supported me in the accomplishment of this humble work
To all my teachers and all those who have committed in these modest works

To all my Guerroudje family,

And to all who have helped me from near or far for the realization of this work.

Guerroudje Asma

Acknowledgments

First, we thank Allah the great who gave us the courage and the will to accomplish this humble job.

Words cannot express our gratitude to our supervisor Dr. Leila BENAROUS for her invaluable patience and feedback.

I would also like to thank the honorable jury members

Who have agreed to judge our work.

Finally, we would like to express our deep gratitude to

All our Professors in Computer Science Department at Laghouat University

who generously and earnestly contributed to our training and educations

الملخص

في بيئة اليوم، الأمن هو قضية أساسية. تعتمد القوة الاقتصادية لأي بلد على قدرته على توفير بيئة آمنة للمستثمرين والسياح. على الرغم من استخدام كاميرات المراقبة لمتابعة الجرائم مثل السرقة، إلا أنها لا تستطيع منعها ولا يزال الإشراف والتدخل البشري مطلوبين. في عصرنا سريع ، نحن في حاجة ماسة إلى التكنولوجيا التي يمكنها اكتشاف الاختلالات و الجرائم تلقائيًا وبدقة مثل المواقف التي تهدد الحياة. يمثل مجال رؤية الحاسوبية الذكية التي تدمج رؤية الحاسوبية مع تقنيات التعلم العميق حلاً واعدًا لهذه المشكلة. التعلم العميق عبارة عن مجموعة من خوارزميات التعلم الآلي المستخدمة لتدريب الشبكات العصبية، وتتألف من طبقات داخلية متعددة. لقد مكنت هذه التقنيات من تحقيق تقدم كبير وسريع في مجال تحليل إشارات الصوت أو الفيديو.

في مشروعنا هذا، ركزنا على اكتشاف الأسلحة مثل استخدام المسدسات والسكاكين من لقطات الكاميرا باستخدام تقنيات الرؤية الحاسوبية الذكية. قمنا بتنفيذ الكشف التلقائي عن الأسلحة باستخدام الشبكة العصبية التلافيفية (CNN) القائمة على خوارزمية SSD Mobile Net، YOLOV3 وYOLOV4. كان الغرض من استخدام ثلاثة نماذج هو المقارنة بين دقتها والتحقيق في استخدامها المحتمل ومدى ملاءمتها في بيئة المراقبة المباشرة. كانت نتائج النماذج الثلاثة جيدة. ومع ذلك، من حيث الدقة، أظهر YOLOV4 نتائج واعدة أكثر يليها نموذج SSD ثم YOLOV3 .على الرغم من أن الدقة ليست المعيار الوحيد الذي يجب مراعاته في تطبيقات المراقبة المباشرة التي تتطلب الدقة والسرعة العالية. لذلك، فإن أخذ هذه المقايضة بين الدقة والسرعة فأن نموذج YOLOV4 هو الأنسب.

الكلمات المفتاحية: الكشف عن الأسلحة، الشبكات العصبية التلافيفية (CNN)، YOLOV3، YOLOV4، SSD Mobile Net
، كاميرات المراقبة

Abstract:

In today's environment, security is a core issue. The economic strength of a country depends on its ability to provide a safe environment for investors and tourists. Although CCTV cameras are used to monitor acts such as theft, they cannot prevent them and human supervision and intervention are still required. In our fast-paced era, we are in dire need of technology that can automatically and accurately detect anomalies such as life-threatening situations. Smart computer vision field that merges the computer vision with the deep learning techniques present a promising solution to this issue. Deep learning is a set of machine learning algorithms used to train neural networks consisting of multiple internal layers. These technologies have enabled significant and rapid advancements in the field of audio or video signal analysis.

In our project, we concentrated on detecting weapons as guns and knives use from camera footages using the smart computer vision techniques. We implemented an automatic weapon detection using a convolutional neural network (CNN) based SSD mobile network YOLOV3 and the YOLOV4 algorithm. The purpose behind using three models was to compare between their accuracy and investigate their potential use and suitability in real-time environment. The results for the three models were good. However, in term of accuracy, YOLOV4 showed more promising results followed by the SSD model then theYOLOV3. Even though, the accuracy is not the only criterion to consider in real-world applications requiring short latency and high speed. Therefore, taking this tradeoff between accuracy and speed the YOLOV4 model seems to be the most suitable.

Keywords: Weapon detection, Convolutional Neural Networks (CNN), YOLOV3, YOLOV4, SSD mobile net, CCTV.

Résume

Actuellement, la sécurité est une des concernes fondamentaux. La force économique d'un pays dépend de son aptitude d'assurer un environnement sûr et sécurisé aux investisseurs et aux touristes. Bien que les caméras de surveillance soient utilisées pour contrôler les actes criminels aux endroits publiques comme le vol, elles ne peuvent pas les empêcher et l'intervention humaine est toujours nécessaire. À notre ère trépidante, nous avons définitivement besoin d'une technologie capable de détecter automatiquement et avec précision des anomalies comme des situations mettant la vie en danger. Le champ de vision intelligente par ordinateur qui fusionne la vision par ordinateur avec les techniques d'apprentissage profond présente une solution prometteuse à ce problème. L'apprentissage profond est un ensemble d'algorithmes d'apprentissage automatique utilisés pour former des réseaux neuronaux composés de multiples couches internes. Ces technologies ont permis des avancées importantes et rapides dans le domaine de l'analyse des signaux audio ou vidéo.

Dans notre projet, nous avons concentrés sur la détection des pistolets et des couteaux à partir de supports de caméra à l'aide des techniques intelligentes de vision par ordinateur. Nous avons mis en œuvre un système de détection automatique des armes à l'aide d'un réseau de neurones convolutifs (CNN) basé sur l'algorithme de SSD Mobile Net ,YOLOV3 et YOLOV4. Le but de l'utilisation de trois modèles était de comparer leurs exactitudes et d'étudier leurs utilisations potentielles et leurs adéquations dans l' environnement à temps réel. Les résultats pour les trois modèles étaient bons. Cependant, en termes de précision, YOLOV4 a montré des résultats plus prometteurs, suivi par le modèle SSD puis YOLOV3. Même si la précision n'est pas le seul critère à prendre en compte dans les applications du monde réel nécessitant une latence courte et une grande vitesse. Par conséquent, en prenant ce compromis entre précision et vitesse le modèle YOLOV4 semble être le plus approprié.

Mots-clés : détection d'armes, réseaux de neurones convolutifs (CNN),YOLOV3, YOLOV4, SSD Mobile Net, CCTV.

Table of contents :

List of figures :

List of tables :

LIST OF ACRONYMS

AI: Artificial Intelligence

CCTV: Closed-circuit television.

CNN: Convolutional Neural Network.

DL: Deep Learning.

GUI: Graphical User Interface.

HCI: Human-Computer Interaction.

ML: Machine Learning

MPEG-7: Moving Picture Experts Group.

NN: Neural Network.

RCNN: Region-Based Convolutional Neural Network.

R-FCN: Region-based fully convolutional network.

RS: Region shape descriptor

SSD: Single Shot Detector.

SVM: support vector machine.

Tk GUI: Toolkit Graphical User Interface

YOLO: You only look once.

General introduction

Due to an increase in crime rates in crowded or suspiciously lonely regions, security is always a top priority in every domain. Gun violence has become a global human rights issue in recent years. Gun violence puts our most basic human right, the right to life in danger. Weapon violence is a daily tragedy that has an impact on people all around the world. The availability of firearms has always been the major cause behind the rise of crime and disorder.

The widespread use of hand-held weapons in violent acts made crime rates soaring-high around the world. A country's security situation must be under control to keep advancing and prospering. Whether we want to attract investors or make money from tourism, we need to ensure a calm and a safe atmosphere. Gun crime rates are very high in many parts of the world. It mainly includes countries where gun ownership is legal. The world has become a global village and everything we say and write affects others. Even if the information they receive is false, it can cause damage as it spreads across the globe in a matter of hours, thanks to the media, especially social media. Today, people are more frustrated, unable to control their anger, and hate speech drives them crazy.

CCTV cameras have played an important role in solving this problem and are considered to be one of the most important security needs, but despite the installation of surveillance cameras, their use for security purposes in previous years did not succeed in eradicating crimes. In order to monitor the screen, someone must be present at all times. For hours, CCTV operators have to look at tens of screens. They must observe, spot and report dangerous situations affecting people and things. People's ability to pay attention to each detail decreases significantly as the number of screens and hours of work increase.

Installing surveillance cameras that can automatically detect firearms and generate alerts to notify the operators or security guards is a solution to the above-mentioned problems. Thanks to the availability of large data sets, faster GPUs, improved machine learning algorithms, and superior computing power, we can now efficiently prepare PCs and build computer-based automated systems to differentiate and identify different objects with high accuracy. Machine learning and advanced image processing algorithms appear to play an important role in intelligent surveillance and security systems.

The goal of this project is to create a system that can identify the use of weapons, specifically guns, and knives. For this, we used some of the deep learning (DL) methods trained using images to identify a weapon. Recent advances in the fields of machine learning and deep learning, particularly convolutional neural networks, have shown significant success in the domains of picture object detection and recognition. Object detection and categorization are crucial for additional object tracking duties in any video surveillance application. We used the SSD Mobile Net, YOLOV3, and YOLO v4 algorithms to achieve this.

Our thesis is divided into three chapters:

Chapter1: This chapter focuses on reviewing the various efforts made to prevent the crimes committed by weapons as well as a critical analysis of existing weapon detection systems using both classical and deep learning approaches

Chapter 2: This chapter provides a detailed overview of recent advances and achievements in object detection using different deep learning techniques. We then explain our intelligent computer vision system, which is designed to identify weapons in the video, with the aim of notifying the police as quickly as possible on how to find these weapons (knives and guns) in order to prevent crime.

Chapter 3: Presents all the implementation phases, the test of the app, and the results of our experiments.

Lastly, we recap the thesis with a general conclusion and future perspectives.

Chapter 01: Literature Review of Existing Solutions

I.1. Introduction

Recent technological advances have changed the way people live and work in society. Tasks that were previously considered impossible appear to be made possible by the advancement of new technologies. Today, technology has become an integral part of society and seems to be a natural component of personal life. Nevertheless, along with these technological advances, criminal acts relying on these technologies are also evolving. Many statistics indicate that the rate of violence using guns and harmful weapons is rising every year, making it difficult for law enforcement organizations to deal with the problem on time. Many regions, especially in places where there are no gun control laws, have a high rate of crime caused by firearms or knives. Despite, the known fact that early discovery of violent crimes is critical to prevent their occurrence and ensure the safety of individuals. The use of advanced technologies to automate crime identification and prevention blooming late. Although, the spread of surveillance cameras was highly recommended at an early stage. This chapter will focus on reviewing the various efforts made to prevent the crimes committed by weapons as well as a critical analysis of existing weapon detection systems using both classical and deep learning approaches.

I.2. Crimes and their rates

A violent crime occurs when the perpetrator employs or threatens to use harmful force on the victim. Violent crimes include homicide, assault, battery, sexual assault, rape, kidnapping, murder, manslaughter, robbery, and other crimes involving the use of force, as well as offenses using weapons. The exact list differs by depending on nations' laws [1].

The rate of violent crime varies substantially between countries. Despite the fact that there is rarely a clear explanation why crimes are done, crime rates are known to be influenced by a number of factors. Countries with high crime rates often have high poverty levels and limited job opportunities, forcing people to resort to riskier, more desperate, and ethically dubious options to obtain money to survive [2]. According to a survey conducted by United Nations Office of Drugs and Crime's in 2019 [3], countries with high firearm rates have higher rates of deliberate homicide. Noting that Europe has some of the lowest rates of violent crime. The rate of violent crime in several European countries is less than 1 in 100,000 because gun prohibitions are tougher in these countries.

In the other hand, Even with the strengthening of the security establishment's efforts and the raising of budgets allocated to it to equip it, according to an official report published by the Algerian police, the country's crime rate increased alarmingly by 14% compared to last year's rate, especially in the capital and major cities. The General Inspector of National Security, Arzkni Haj Saeed, confirmed in a press conference on Thursday that the data showed a 14.71 % increase in the crime rate in the past year 2021 compared to the year before 2020, which is a large percentage compared to previous years, as it had not increased in 2019 compared to 2018, only 3 %.[4].

This highly emphasizes the importance of automated early detection and prevention of crimes made by weapons in public spaces to ensure the citizen's safety. In the next section, we talk about efforts made to reduce crime rates worldwide.

I.3. Crime reduction efforts

In this section, we review the efforts made worldwide to reduce the crime rates, their effects, and their evolution.

I.3.1. Streetlights

Most people believe there is a straightforward and obvious link between illumination and crime. Criminals who profit from hiding in the dark will be prevented by better illumination. In a well-lighted environment, anyone who witnesses a crime can intervene or contact the police. They can also recognize the criminal and describe them to the police. This fear halted criminals from acting stray for a while. However, according to Professor Ken Pease, a crime prevention specialist better lighting demonstrated a crime reduction rate in a variety of ways. It also occasionally increased their rates since the police will also be visible, which allows the criminals to plan their escape roots and execute them as soon as the police are spotted or their sirens are heard. He also emphasized that the lighting alone cannot guarantee safety, if so, crimes during daylight would not have been recorded. Moreover, he mentioned that although the streets may be lighted, viewers may not necessarily intervene nor call the police, they also may refuse to give testimonies about the crimes they witness. Furthermore, the lighted street may give the safety vibes and thus it may attract people who enjoy the nightlife, this lead to the spread of robberies targeting empty houses and parked cars. Lastly, lighted streets became the attraction of youngsters and the center of their noisy night activities which disturbs the people living in those areas.[5].

I.3.2. Surveillance Camera usage

Closed-circuit television(CCTV) cameras play a major role in resolving this issue and are regarded one of the most important security requirements [6].

CCTV cameras are now put in every public location and are primarily utilized for safety, crime investigation, and other security measures. In court, the most crucial evidence is CCTV footage. When a crime is committed, law enforcement authorities respond to the scene and take the video recording with them [7]. When comparing the surveillance systems of other countries throughout the world, the United Kingdom has approximately 4.5 million surveillance cameras. Around the year 2010, Sweden had roughly 50000 cameras installed. By putting just 450 cameras in the city of Poznan, the Polish government was able to reduce drug cases by 60% and street fights by 40% [8]. China boasts the world's largest surveillance system, with 170 million cameras installed around the country, with an additional 400 million in 2020. Using their powerful CCTV cameras network and facial recognition technology, Chinese officials found and apprehended BBC correspondent John Sudworth in just seven minutes and put him behind bars [9]. In many nations, proving crimes has been one of the most difficult tasks for judges, thus, many criminals go unpunished due to lack of evidences, motivating new crimes. The camera has been demonstrated in performance testing to reduce robbery and general crime in casinos and car parks in the United Kingdom by more than 51% [10].

However, several factors conspire against CCTV cameras operators in their efforts to proactively detect crime. They have a set of limitations that may reduce their effectiveness. First, although the cameras record everything, the monitoring video surveillance is done by humans, this means that it is prune to errors because person's ability to spot suspicious behavior is fallible. Moreover, the monitoring and/or rewinding the videos to collect evidences of crimes can be both tedious and time consuming. Second, factor is that CCTV spread in public spaces although is used to ensure safety, it can violate the peoples’ privacy especially toward some individuals with past criminal records, they may be continuously watched or monitored which is considered as a violation of their public rights. Thirdly, with the availability of the latest technology in the market, trespassers and thieves are becoming trained on discovering the cameras blind zones and moving undetectably under these cameras. Lastly, in most of case, CCTV cameras are used to record videos which may include crimes, although these videos are helpful in court, but they are used post the crime. Unfortunately, they cannot halt in progress the crime nor prevent them [10].

The most widely utilized solution is CCTV due to its ability to display videos in a monitoring environment. However, CCTV is a luxury technology in its use since installation costs are high. Moreover, CCTV operations consume a lot of resources, including energy, storage, and video output supervisors. With today's technical advancements, it is now feasible to develop CCTV systems and fix some of their limitations. One of the methods is the use of Internet of Things (IoT) technologies to connect environmental systems. IoT technology allows systems to be aware of their surroundings, such as in-home buildings, aquaculture, and health care. IoT is a notion in which an item may interact with other things without the need for human intervention. As a result of this benefit, an IoT integrated surveillance system is referred to as a smart item or smart surveillance system.

M. H. Udin et al [11] explain the implementation of IoT as a surveillance system. The system has 3 types of components. The gateway which is the core/controller of other components. Sensor nodes to detect movement and inform about it. Camera nodes to capture images/videos from the environment. On the software side, there is a web server that is able to store data sent by the gateway. In addition, there is an android application as the interface of the system. With the support of web servers that are on the internet network, it allows the system to be accessible from anywhere and at any time. The term smart monitoring is used when a surveillance system has awareness of its surrounding in several previous types of research. The purpose of their study was to illustrate how an IoT-based surveillance monitoring system may be implemented. The results of the experiment revealed that the sensor node can detect human movement in any position up to a 3-meter range. The ideal location for the sensor node is 2 meters above ground level with a 45-degree angle. The camera node, on the other hand, can take a picture in an average of 0.7047 seconds. The local system operates in near-real-time, with a response time of 1.6 seconds. While the entire system (including the internet area) takes 6.905 seconds on average. The experiment was carried out on numerous aspects of the system in order to see if it could be used as a surveillance system. Although, the results showed that this purpose was fulfilled by their system, however, further studies are required to evaluate its capabilities in real-time surveillance environment.

I.3.3. Computer-based monitoring and detection

The Haar cascade classifier is a method for identifying objects in images, videos, and live streaming. Paul Viola and Michael Jones introduced this approach in 2001. This method needs a large number of photographs, both positive and negative. Haar cascade employs positive and negative pictures for training and then extracts features from them[12]. Positive pictures include the desired object, in this case, paper firearms (machine guns, assault rifles, or pistols). Negative pictures, on the other hand, are images that are devoid of things. So that the categorization is simple and accurate feature extraction is possible.

In January 2016, Michał Grega et al .[13] Propose an automatic detection of guns and knives in CCTV images It's an algorithm that alerts a human operator when a gun or knife appears in an image. They focused on limiting the number of false positives to using the system. The specificity and sensitivity of the knife assay are significantly better than other recently published methods. They also managed to come up with a version of the gun detection algorithm that had a near-zero false positive rate. We have demonstrated that a system can be created to provide early warning of dangerous situations, resulting in faster response times and a reduced number of potential victims. They designed a tool detection algorithm based on visual descriptors and machine learning. The first step is to select image candidates from the input as crop parts. they select candidates using an improved sliding window technique. Unlike the original sliding window, we only look for knives near silhouettes, and if at least one silhouette appears in the image, the next step is to convert the image to its digital representation. We use a sliding window mechanism to find parts of the image that contain tool features. This allows us to determine the approximate location of the knife in the image. We don't need to detect knife edges, which is not trivial when looking at images with variable and non-uniform backgrounds. The current literature describes many different visual descriptors and their advantages and disadvantages.

They decided to use visual descriptors from the MPEG-7 standard [14]. Due to a large number of different knife types, they decided to use similarity-based descriptors instead of keypoint matching or precise shape-based descriptors. The digital representation of the descriptor is stored as a binary vector for faster access time and easier processing. Eigenvectors are used in the decision-making part of the system. The extracted feature vectors are input to a support vector machine (SVM)[15]. We use a ν-SVM with a given decision function. On the other hand, they conducted a series of proof-of-concept experiments to evaluate different approaches to the firearm detection problem. After the first experiment, they decided to focus on a single type of firearm - the pistol. First, run a simple background subtraction algorithm. It is based on image differences between consecutive frames. Image differences can leave multiple artifacts due to image flickering and lighting changes, so we support this with two simple operations: erosion and dilation. These two operations allow us to remove these artifacts and focus further steps of the algorithm on the foreground part of the image. The algorithm was chosen for its simplicity, low computational requirements, and good performance. They experimented indoors so they didn't have to deal with interference from small moving components Next, the Canny edge detection algorithm[16] is implemented to convert the image into a set of edges.

The algorithm only works on foreground regions detected in the previous step to save computing power. The next step is to sample the image using a sliding window technique. Images are analyzed multiple times as the sliding window size increases. In the next step, the samples obtained through the sliding window are scaled to a common size of 40 × 30 pixels, creating a vector with 1200 values. We removed all samples with a low number of edges (less than 11%) because they were not informative. In the next step, we use the MPEG-7 Region shape descriptor (RS) [17], [34, 40, 41] to compare the shapes found in the candidate regions selected by the NN with a generic gun descriptor constructed from positive examples in the training set.

In 2019, Mohammad Zahirul Islam and al. [18] have focused their efforts. The following research on various types of advanced criminal activity and possible defense mechanisms for those harmful activities is presented in theire paper. The system has been developed with a particular focus on criminal activity such as hostage situations. An all-terrain tracked robot with a track arm has been developed to learn about the inside situation of the hostage building or location. This robot is capable of traversing any type of rough terrain as well as climbing stairs [19]. This system also includes a long-distance communication system. The system includes multiple camera visions for live stream and image analysis. Through image processing, the system developed an algorithm for detecting stairs, human movement, and various types of guns using a custom Haar Cascade Classifier [20][21]. After detecting stairs, the system can use two ultrasonic sonar sensors to measure the distance and calculate the height of the stair steps and then adjust the track arm's movement autonomously to climb the stairs. Outside of the hostage area, detection of human movement and guns aids the system in distinguishing between criminal and victim. We discovered that the detection of all three required objects is successful after a large amount of real-world testing.

In 2020, J. Aarchi et al. [22]described a real-time gun detection system with model and type recognition. The approach employs real-time video as an input and the Haar Cascade Classifier from the Open-CV package as the method. This strategy is primarily intended for the administration of security and safety. This approach was created to recognize and categorize firearms in a video, and it has a high accuracy rate of 95% in submachine guns. Because the framework's essential software is easily available, it turns out to be a more affordable framework.

I.3.4. Deep learning approaches

The percentages of crime caused by firearms are alarming in many areas throughout the world, particularly in countries where gun ownership is legal or was legal for a time. As a result, the number of CCTV spread in public and private areas, such as roads, cinemas and malls, has increased rapidly and remarkably. Consequently, number and size of their recordings amplified as well making it difficult for a human administrator to examine and assess if a potentially dangerous situation will arise [23]. Weapon identification from recorded videos is becoming critical for developing security systems that can ensure public safety by preventing crimes from occurring.

Due to object detection's close relationship with video analysis and image understanding, it has attracted much research attention in recent years. Traditional object detection methods are built on handcrafted features and shallow trainable architectures. With the rapid development in deep learning, more powerful tools, which can learn semantic, high-level, deeper features, are introduced to address the problems existing in traditional architectures.

In this subsection, we provide a review of deep learning-based object detection frameworks which dependent on different methods.

The first work in addressing weapon detection in videos using CNNs was by Olmos et al.[24]. Their work focused on pistols detection and it was evaluated on videos of movies from the nineties. The works related to the use of deep learning for weapon detection continued to flourish with multiple works papers published ever since. Although, an extensive listing may not be possible, we will mention few of the recently published works. J. Y. Lim et al.[25] published in 2019 a large-scale dataset of handguns filmed by a CCTV camera. This dataset was composed of 5500 photos extracted from 250 CCTV-recorded videos significantly improving the handgun identification based on deep learning. J. Harsh Jain et al. [26] used convolutional neural networks (CNN)-based SSD and Faster RCNN methods which is object detection methods used to detect things in images and videos, to accomplish automated weapon identification. Two datasets are used in their proposed implementation. One dataset contains photographs that have been pre-labeled, while the other contains images that have been manually labeled. The results are tabulated, and both algorithms attain high accuracy, but their use in real-world scenarios depends on the trade-off between speed and accuracy. In terms of performance, the SSD algorithm is faster, at 0.736 frames per second. Faster RCNN, on the other hand, achieves a frame rate of 1.606s/frame, which is slow when compared to SSD. Faster RCNN performs better in terms of accuracy, with an accuracy of 84.6 percent. SSD, on the other hand, has an accuracy of 73.8 %, which is low when compared to the speedier RCNN. Due to its higher speed, SSD provided real-time detection, but faster RCNN gave greater accuracy. In May 2021, N. Geetha et al. [27] suggested employing the YOLOv3 algorithm to identify firearms. The YOLO models are end-to-end deep learning models that are popular due to their speed and accuracy in detecting threats. They first constructed a dataset including three types of weapons: handguns, knives, and heavy guns. This training dataset was passed to YOLOv3 (You Only Look Once)[28] for weapon classification. Once the training finishes, the system was used to identify the type of weapon present in real-time input video from surveillance cameras, the identification confidence score was calculated. The system sends an alarm to the authorities (police) once a weapon is spotted. For further analysis, they employ photos in the jpeg, jpg, and tiff formats. They reported that the accuracy of the identified weapons from a video frame for the three types is 95% for handguns, 87% for knives, and 84 % for heavy guns.

E. Arif et al. [29] implemented and compared faster approaches such as Region (R-CNN) and Region Fully Convolutional Networks (R-FCN) with feature extractor Visual Geometry Group (VGG) and ResNet, respectively. They also developed and tested a hybrid CNN model that combines R-CNN and R-FCN to decrease false positives in weapon identification. Their experiments revealed that their model for the weapon detection system produced good results. In general, the hybrid R-FCN model outperformed the other models. In comparison to quicker R-CNN and R-FCN, hybrid R-FCN produces higher outcomes in terms of accuracy, recall rate, precision, and F1-score.

I.4. Critical analysis

Safety is always a major concern in any field, as crime rates increase at crowded events or in suspiciously out-of-the-way areas. Due to the ever-increasing demands for security and personal property protection, the need for foundational solutions has become a necessity. In our research, we outline different approaches to crime reduction, first we focus on the classic approach to technical solutions, then we talk about the use of computer vision in crime detection and prevention, and finally we talk about intelligent computer vision (Deep Learning + Computer Vision). Table I.1 describes the different properties of the methods.

I.5. Conclusion

Security and safety are big issues in today's world. For a country to be economically strong, it must provide a safe and secure environment for investors and tourists. That is why most countries are investing in public safety insurance means. Among which are the closed-circuit television cameras. Although the CCTV are used to monitor and record criminal activities, but, this monitoring is human-based and in most of cases, the use of CCTVs does not prevent the crimes from occurring. The CCTVs may be more helpful in tracking criminals post committing the crimes and they provide an irrefutable solid proof to court. Therefore, the need for systems that can automatically detect these illegal activities on the spot and prevent them is mostly needed. Works relying on the combining of the deep learning and computer vision techniques are flourishing. The introduced smart computer vision may provide the solution to real-time accurate weapon usage detection in public spaces, helping in the reduction of weapon-based crimes. In the next chapter, we explain our smart computer vision systems proposed to detect weapons in real-time videos with the intent to inform the police once these weapons (knives and guns) are spotted in order to prevent crimes from occurring.

Table I. 1:the comparison between methods

Crime reduction method		Weapons	Crime Reduction Rate / Accuracy	Pros	Cons
Technological solution	Light use	-	36%[30]	clear surveillance	facilitate undesirable behavior
	CCTV use	-	51%[10]	recording evidence	cost /long-term work
	Sensor Use	-	detect up to a 3-meter range response time of 1.6 sec	No need for human intervention	Can't specify the object detected (only movement)
Computer vision	Haar Classifiers	Submachine Guns	95%	more affordable framework	bigger data set, slow in training
		Machine Guns	85%		
		Assault Rifles	87.5%.		
		pistol	80%		
		knife	81.18		
Smart computer vision	CNN-based SSD	gun	73.8	Low accurate	Faster in detection
	R-CNN	gun	84.6%	High accurate	Slow in detection
	Yolo	handgun	95%	Faster in detection	Lower accuracy / large number of frames per second
		knife	87%		
		Heavy gun	84 %		
	R-FCN	Different weapons	89%	Good accuracy	Slower in detection
	Hybrid R-FCN	Different weapons	91%	Higher accuracy, recall rate, precision, and F1-score	-

Chapter 2: Deep Learning-based weapon detection System

II.1. Introduction

One application area of computer vision has seen remarkable progress in recent years: object recognition. One of the most challenging and fundamental problems in object detection is locating a specific object from multiple objects present in a scene. Early traditional detection methods detect objects by introducing convolutional neural networks. As of 2012, feature extraction technology based on deep learning has been applied , and significant breakthroughs have been made in this field. [31]. This chapter provides a detailed overview of recent advances and achievements in object detection using different deep learning techniques. We will then explain our intelligent computer vision system, which is designed to identify weapons in the reel-video, aiming of notifying the police as soon as the use of knives or guns is spotted in public places covered by CCTV in order to prevent crime.

II.2. Deep Learning:

Deep learning is an artificial intelligence (AI) function that mimics the way the human brain works in processing data and building decision-making models. Deep learning is a subset of machine learning in artificial intelligence characterized by networks capable of unsupervised learning from unstructured or unlabeled data. Also known as Deep Neural Learning or Deep NeuralNetworks.[32], the **Figure II.1** shows The relationship between artificial intelligence

,Machine learning and Deep learning .

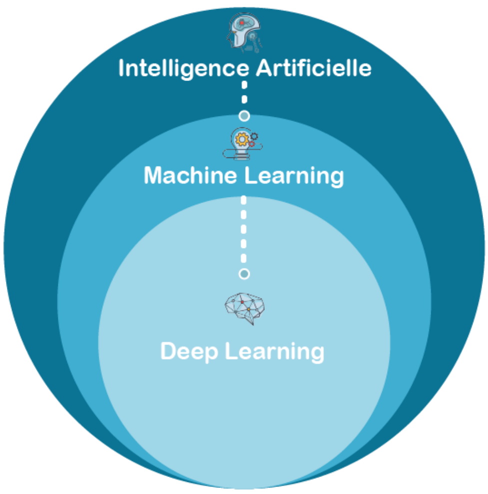

Figure II. 1:*The relationship between artificial intelligence ,Machine learning and Deep learning*

II.2.1.Deep Neural Network Architectures

Deep Learning is a rapidly growing field, and new alternative architectures to the algorithms appear every week [15]. In the next section, we will present a brief overview of the common structures found in many deep networks.

II.2.2.Convolutional neural networks

Convolutional Neural Networks, also known as CNNs or ConvNets, are a class of neural networks that specialize in processing data with a lattice topology, such as photos. Digital images are binary representations of visual data. It contains a series of pixels arranged in a grid, containing pixel values to indicate how bright each pixel should be and what color it should be. As soon as we see an image, the human brain processes a lot of information.

Each neuron works in its own receptive field and connects to other neurons in a way that covers the entire field of view. Just as each neuron responds only to stimuli within a limited field of view known as the receptive field in biological visual systems, each neuron in a CNN only processes data in its receptive field. The layers are arranged in such a way that they first identify simpler patterns and then more complex patterns. By using a CNN, it is possible for the computer to see. [34]

- ***Layers of convolutional neural networks:***

A CNN typically has three layers: a convolutional layer, a pooling layer, and a fully connected layer. The **Figure II. 2** shows the architecture of the CNN and the layers of it .

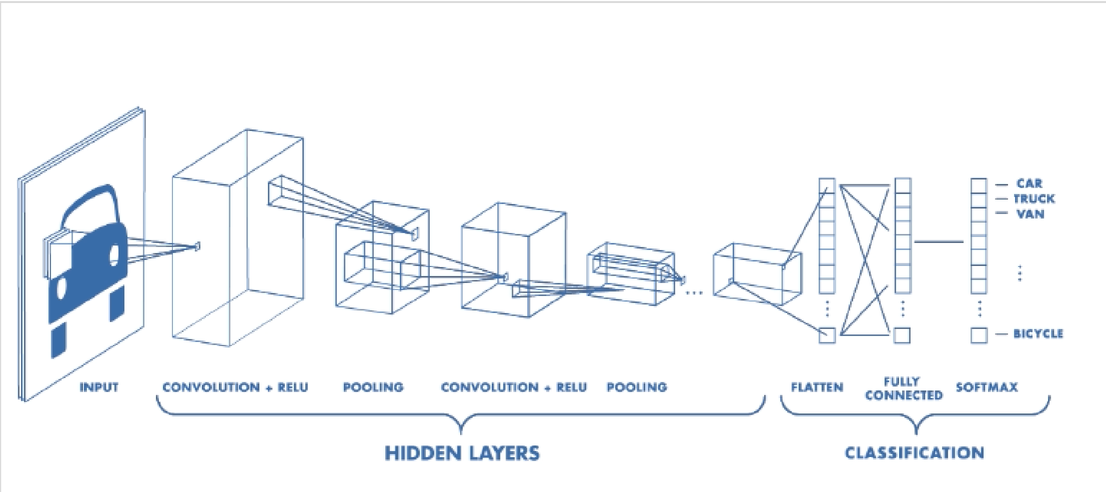

Figure II. 2*:Architecture of a CNN*

II.3. Basic Block Diagram of Object Detection

The fundamental goal of generic object detection is to see if an image contains an object in the image from the defined classes (e.g., animals, cars, and people), but if so, to report the spatial location and extent of a single object (by bounding box). In real-time videos or still images, the goal is to recognize real-world object instances. It paves the way for object recognition, localization, and detection of single/multiple objects within a video frame or an image, resulting in a much better overall interpretation of the image. When performing object detection, difficult challenges such as occlusion and irregular lighting conditions should be handled with extreme care [31].

Object detection was used to solve extremely difficult vision-related problems such as scene understanding, image captioning, instance segmentation, semantic segmentation, object recognition, and tracking. Object detection applications include intelligent military surveillance systems, security, self-driving cars, robot vision, human-computer interface (HCI), consumer electronics, and others [31].

Deep learning methods have recently emerged as the most effective techniques for automatically learning features from raw data. Deep learning methods, in particular, have made significant progress in object detection, a problem that has piqued the interest of many researchers in the last decade. Video surveillance is one of the most difficult and fundamental areas of security systems because it is entirely dependent on object detection and tracking. It observes people's behavior in public in order to detect any strange activities. [31]

II.4. Object detection methods based on deep learning:

In the deep learning era, object detection can be grouped into two genres: “two-stage detection” and “one-stage detection”, where the former frames the detection as a “coarse-to-fine” process while the latter frames it as “complete in one step”:

II.4.1.Two-stage detectors

Such as Faster R-CNN, which divides the detection, process into two steps. The first step uses a Region Proposal Network to generate regions of interest that have a high probability of being an object. The second step then performs the final classification and bounding-box regression of objects by taking these regions as input. These two steps are named the Region Proposal Step and the Object Detection Step respectively. Such models reach the highest accuracy rates but are typically slow.[35]

II.4.2.One-stage detectors

Such as YOLO and SSD, which treat object detection as a simple regression problem by taking an input image and learning the class probabilities and bounding box coordinates. The approach is simple and elegant because it eliminates region proposal generation, encapsulating all computation in a single network. Such models reach lower accuracy rates, but are much faster than two-stage object detectors and show higher memory efficiency[35].

In the next section, we will show some of the methods we use in order to create our system of weapon detection and the steps of creation.

II.5. The proposed methods

II.5.1.The first method

In our first proposed system, we used the process indicated in Figure II.3 to detect weapons using the YOLO algorithm. Initially, we use a dataset that consists of two classes of weapons – Handgun and Knife. This dataset is trained for the classification of weapons using the YOLO (You Only Look Once) algorithm using the V3 and V4. Once the data is trained, the system can classify the type of weapon present in the real-time input video from the surveillance cameras along with the confidence score of each weapon. An alert will be sent to the authorities if the weapon is detected. The **Figure II. 3** shows the architecture of the system using the Yolo algorithm.

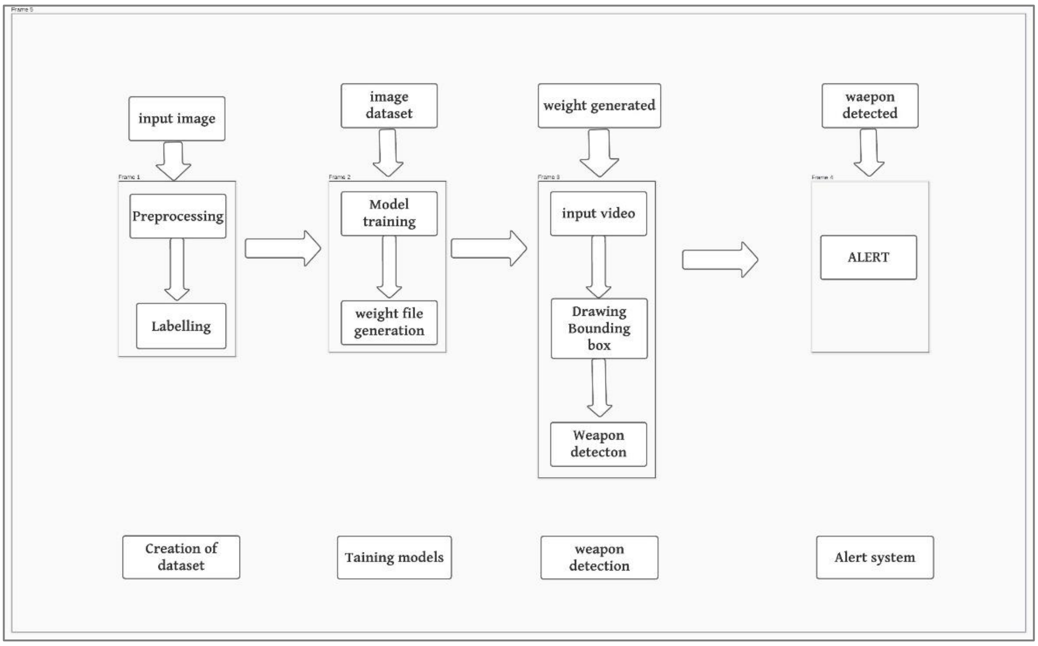

Figure II. 3*:application design using Yolo algorithm*

- **Yolo algorithm working :**

The YOLO algorithm works by dividing the image into N grids, each of which has an equal-dimensional area of SxS. Each of these N grids is responsible for detecting and locating the objects it contains. Thus, these grids predict the coordinates of B's bounding box relative to its cell coordinates, as well as object labels and the probability of the object appearing in the cell [36].

- **Yolov3 algorithm**

You Only Look Once, Version 3 (YOLOv3) is a real-time object detection system that recognizes specific things in films, live feeds, and photos. Previous approaches, such as region-based convolutional neural networks (R-CNN), required thousands of network evaluations to create predictions for a single image, making optimization time-consuming and uncomfortable. Object localization and feature extraction were combined into a single monolithic block in YOLOv3. Their single-stage design results in a very quick inference time. It predicts the bounding box coordinates and class probabilities for these boxes using the full image as a single instance.

The most significant benefit of adopting YOLO is its incredible speed, it can process 45 frames per second. Unlike previous algorithms that scan images with a sliding window, YOLO passes the entire image through a convolutional neural network in one run and predicts the output [27].

- **Yolov4 algorithm**

You Only Look Once, Version 4 (YOLOv4) is a real-time object detection system that recognizes specific things in films, live feeds, and photos. YoloV4 is an important improvement of YoloV3, the implementation of a new architecture in the Backbone and the modifications in the Neck have improved the mAP (mean Average Precision) by 10% and the number of FPS(Frame per Second) by 12%. In addition, it has become easier to train this neural network on a single GPU[38].

- ***Architecture of yolov4***

In YOLOv4, Bag-of-Freebies (BoF) and Bag-Of-Specials (BoS) are used for the backbone of the model, and the efficiency of the model is improved by CSPDarknet53(Cross-StagePartial Darknet53). BoF is used to increase only the training cost while keeping the inference cost low by using data augmentation, and BoS is used to increase the inference cost by a small amount while significantly improving object detection accuracy.

A modified SPP (Spatial Pyramid Pooling), which generates a fixed-size output regardless of the input size, and transmits it to a fully connected network, and a modified PaNet (Path Aggregation Network), which aims to better propagate information about textures and the mode information used by the layer is used to move to the next layer.[39]

- **Dataset**

The dataset has been cloned from a github repository ,[41] which has the format of jpg and jpeg, the images are mostly had high resolution and different sizes. Its description is given in **Table II.1**.

<table>
<tr><th colspan="9">Table II. 1 :Dataset description</th></tr>
<tr><th colspan="2"></th><th>Up</th><th>Down</th><th>Left</th><th>Right</th><th>Bullet ejection</th><th>Facing forward</th><th>Back sign</th></tr>
<tr><td colspan="2">Knives</td><td>178</td><td>151</td><td>350</td><td>321</td><td>/</td><td>/</td><td>/</td></tr>
<tr><td rowspan="5">Gun</td><td>With hand</td><td>96</td><td>41</td><td>142</td><td>183</td><td>83</td><td>36</td><td>33</td></tr>
<tr><td>Just pistol</td><td>15</td><td>22</td><td>82</td><td>129</td><td>/</td><td>/</td><td>/</td></tr>
<tr><td>Cartoon</td><td>6</td><td>8</td><td>39</td><td>27</td><td>/</td><td>/</td><td>/</td></tr>
<tr><td>Using CCTV</td><td>3</td><td>9</td><td>17</td><td>19</td><td>/</td><td>/</td><td>/</td></tr>
<tr><td>On the waist</td><td colspan="7">10</td></tr>
</table>

Once the dataset of weapon images is collected, it is annotated using the tool Labelimg toolbox [42]. **LabelImg** is a graphical image annotation tool that labels object bounding boxes in images. It is a free, open-source tool It's written in Python and uses QT for its graphical interface. The position of the weapons was marked in the images based on the two classes of weapons – handgun, and knife. The coordinates for these boxes were generated for each image and stored in a text file. The classes for which the images were marked were stored in a label file. This is used as the training dataset. *Table II. 2:Dataset description*

- ***Training the model***

For this project, the training sessions were conducted using Google Colab. YOLOV3 and YOLOv4 are trained to detect harmful weapons. The training dataset is stored in Google Drive, where Google Colab can retrieve the dataset. In the background, Alexey Darknet53 [27] uses a pretrained CNN network for image classification tasks to use Yolov3, and Yolov4 for object recognition. We use a transfer learning approach to add our layers to the previously trained model. Then, we download the darknet53.conv.74 pretrained weights. Therefore, our custom model will be trained using these pretrained weights instead of randomly initialized weights, saving us a lot of time and computation. Before the training session starts, some parameters must be defined, such as batch size, subdivision, max batch, number of classes, width and height in the yolov3.cfg + yolov4.cfg (Figures II.4, II.5) file .

The training takes place for 4000 iterations. Once the training is completed the training, weights files are generated which can be used for weapon detection.

```
[ ]  # Change lines in yolov3.cfg file
     !sed -i 's/batch=1/batch=64/' cfg/yolov3_training.cfg
     !sed -i 's/subdivisions=1/subdivisions=16/' cfg/yolov3_training.cfg
     !sed -i 's/max_batches = 500200/max_batches = 4000/' cfg/yolov3_training.cfg
     !sed -i '610 s@classes=80@classes=2@' cfg/yolov3_training.cfg
     !sed -i '696 s@classes=80@classes=2@' cfg/yolov3_training.cfg
     !sed -i '783 s@classes=80@classes=2@' cfg/yolov3_training.cfg
     !sed -i '603 s@filters=255@filters=21@' cfg/yolov3_training.cfg
     !sed -i '689 s@filters=255@filters=21@' cfg/yolov3_training.cfg
     !sed -i '776 s@filters=255@filters=21@' cfg/yolov3_training.cfg
```

Figure II. 4: Changing the parameters of yolov3.cfg to train 2 classes of images

```
batch=64
subdivisions=16
width=416
height=416
channels=3
momentum=0.949
decay=0.0005
angle=0
saturation = 1.5
exposure = 1.5
hue=.1

learning_rate=0.001
burn_in=1000
max_batches = 8000
policy=steps
steps=4800,5400
scales=.1,.1
```

Figure II. 5 : Changing the parameters of yolov4.cfg to train 2 classes of images

II.5.2.The second method

In our second system, we used the process depicted in Figure II.6 to detect weapons using the SSD Mobile Net algorithm. Initially, we use the same dataset from before and train it and fed to an object detection algorithm. Based on the application suitable detection algorithm(SSD) was chosen for weapon detection., Once the data is trained, the system can classify the type of weapon present in the real-time input video from the surveillance cameras along with the confidence score of each weapon. An alert will be sent to the authorities if the weapon is detected. The **Figure II.6** shows the application design using the SSD Mobile Net

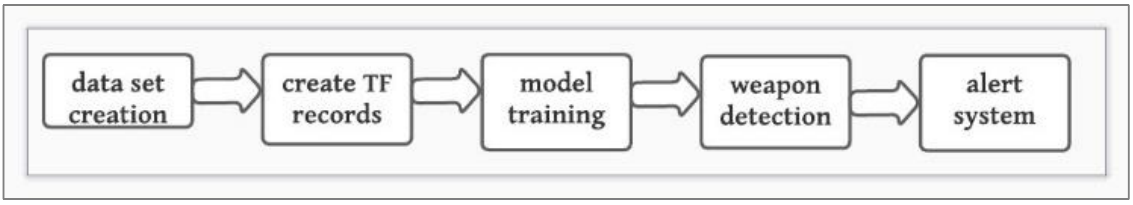

Figure II. 6 : application design using SSD mobile net

- ***SSD (Single Shot Detector)***

SSD Mobilenet V2 is a one-stage object detection model that has gained popularity due to its thin network and novel depthwise separable convolutions. It is a commonly used model on devices with lower computing power such as mobile devices (hence the name Mobilenet) and has high accuracy performance. In this section, we walk through the entire process of training a custom image or dataset using the SSD Mobilenet V2 architecture. The entire training process takes place on Colab, which has GPU capabilities to speed up training. [43] [44], The SSD architecture is a single convolutional network that learns to predict bounding box locations and classifies those locations in one go. Therefore, SSD can be trained end-to-end. The SSD network consists of a basic architecture (MobileNet in this case) and several convolutional layers [45]. Figure II.7 shows its architecture.

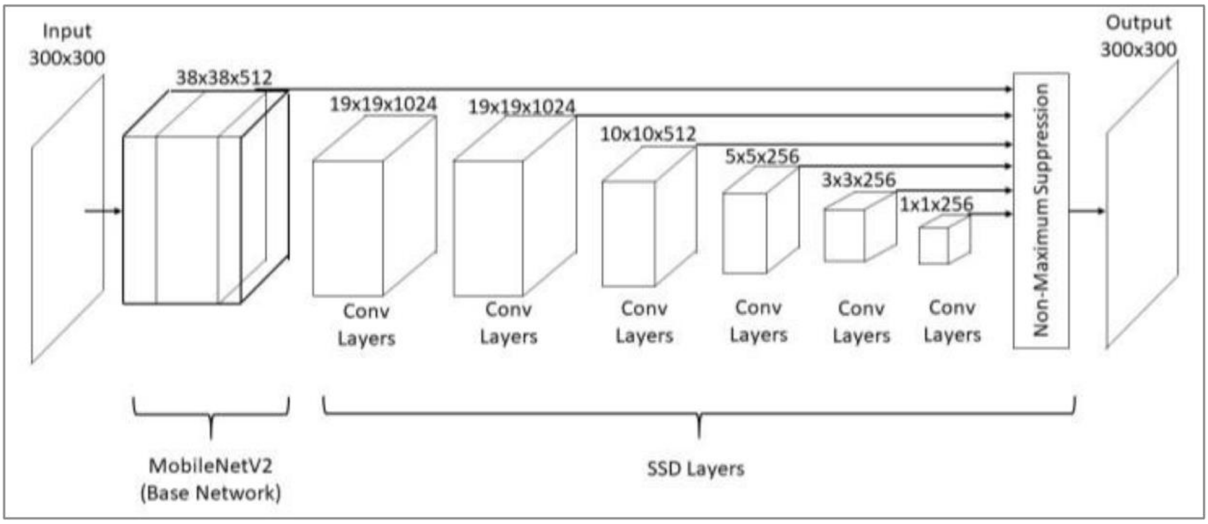

Figure II. 7:SSD Mobile Net Architecture[46].

By using SSD, we only need a single shot to detect multiple objects in an image, while region proposal network (RPN) based methods such as the R-CNN series require two shots, one for generating region proposals and another for topic proposal detecting each object. Therefore, SSD is much faster compared to two-layer RPN based methods.

- **SSD working :**

SSD divides the image using a grid, making each grid cell responsible for detecting objects in that area of the image. Discovering objects simply means predicting the class and location of objects in that area. If there is no object, we treat it as a background class and ignore the position.[47]

- ***Training of SSD***

Object detection is a computer vision task that has recently been influenced by the progress made in Machine Learning. In the past, creating a custom object detector looked like a time-consuming and challenging task. Now, with tools like TensorFlow Object Detection API, we can create reliable models quickly and with ease. In this part, we use the second generation of the TensorFlow Object Detection API, which:

- Allows employing state-of-art model architectures for object detection,
- Gives a simple way to configure models.

The Figure II. 8 shows the steps of the training using Tensorflow object detecting :

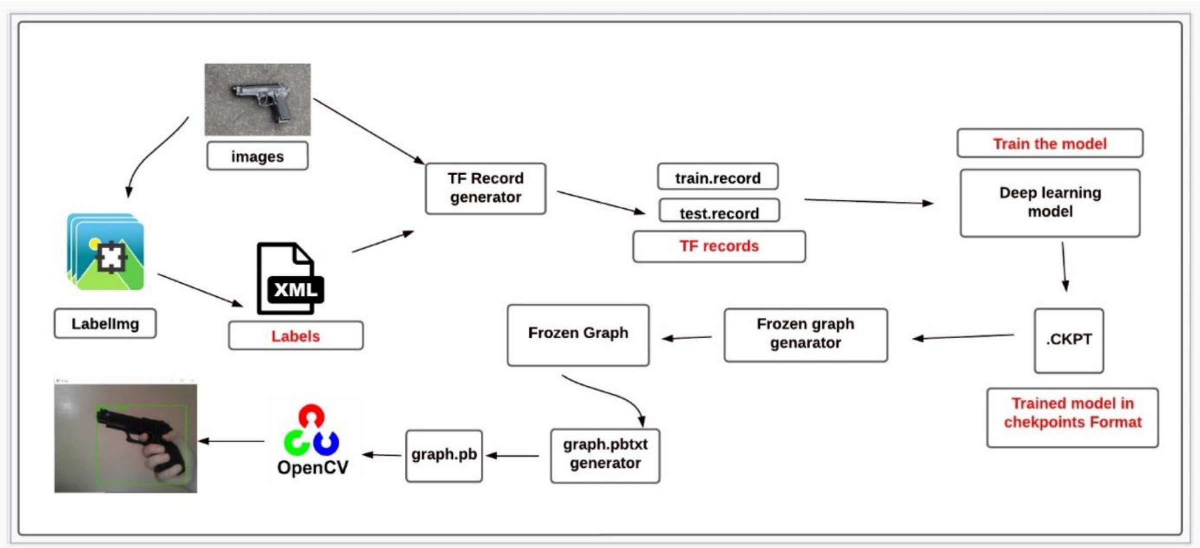

Figure II. 8:the training using Tensorflow object detection

In the training of this model, we use the pre-trained model that was cloned from the website of TensorFlow model ZOO [48] (ssd_mobilenet_v2_fpnlite_320x320_coco17_tpu-8), using the google colab for the training in free GPU. And the neural network is trained for our weapon data that contains 2000 images (1800 for training and 200 for testing), the annotations of the images will create XML files.Models based on the TensorFlow object detection API need a special format for all input data, called TFRecord. That will create by executing this command:

```
!python 'TF_RECORD_SCRIPT']}-x 'IMAGE_PATH'-l 'LABELMAP']} -'train.record'
!python 'TF_RECORD_SCRIPT']}-x 'IMAGE_PATH'-l 'LABELMAP']} –'test.record'
```

The training takes place for more 500000 steps. Once the training is completed the training. weights file is generated which can be used for weapon detection.

Update Configuration file For Transfer Learning :

- Change ***num_classes*** to number of your classes.
- Change ***test.record*** path, ***train.record*** path & ***labelmap*** path to the paths where you have created these files.
- Change ***fine_tune_checkpoint*** to the path of the directory where the downloaded checkpoint.
- Change ***fine_tune_checkpoint_type*** with value **classification** or **detection** depending on the type.
- Change ***batch_size*** to any multiple of 8 depending upon the capability of your GPU. (eg:- 24,128,...,512).Mine is set to 64.
- Change ***num_steps*** to number of steps you want the detector to train.

II.6. Conclusion

Object recognition is one of the most active research areas in computer vision, and it includes both object classification, which is the classification of each object in an image, and object localization, which is locating each object by drawing a bounding box around the object. Today, with the increasing use of object detection in several interesting applications such as video surveillance, robotics, self-driving cars, etc., it becomes necessary to develop more accurate and faster systems. In the next chapter, we explain the implementation phases of our proposed system for weapon detection in images videos, and real-time videos and our experimental results in terms of accuracy in both methods.

Chapter 3: Implementation and analysis

III.1.Introduction

After the previous chapter introduced the theory of deep learning and how we implement its methods (image labeling, training). This part is dedicated to implementing the weapon detection system and its results.

Public safety is the function of government to ensure that citizens, the people, organizations and institutions in their territories are protected from threats to their well-being and the prosperity of their communities. To address growing public safety challenges, responsible public agencies and organizations can use their own intelligence to successfully address potential threats ahead of time [49]. which can be achieved through the use of advanced crime or incident reduction techniques, the Figure III.1 shows some of the ways used to help in this:

Figure III. 1:detection systems for public safety

In our project, we are interested in weapon detection system surveillance cameras,

Figure III .2 shows the target system and the parts we will implement on it.

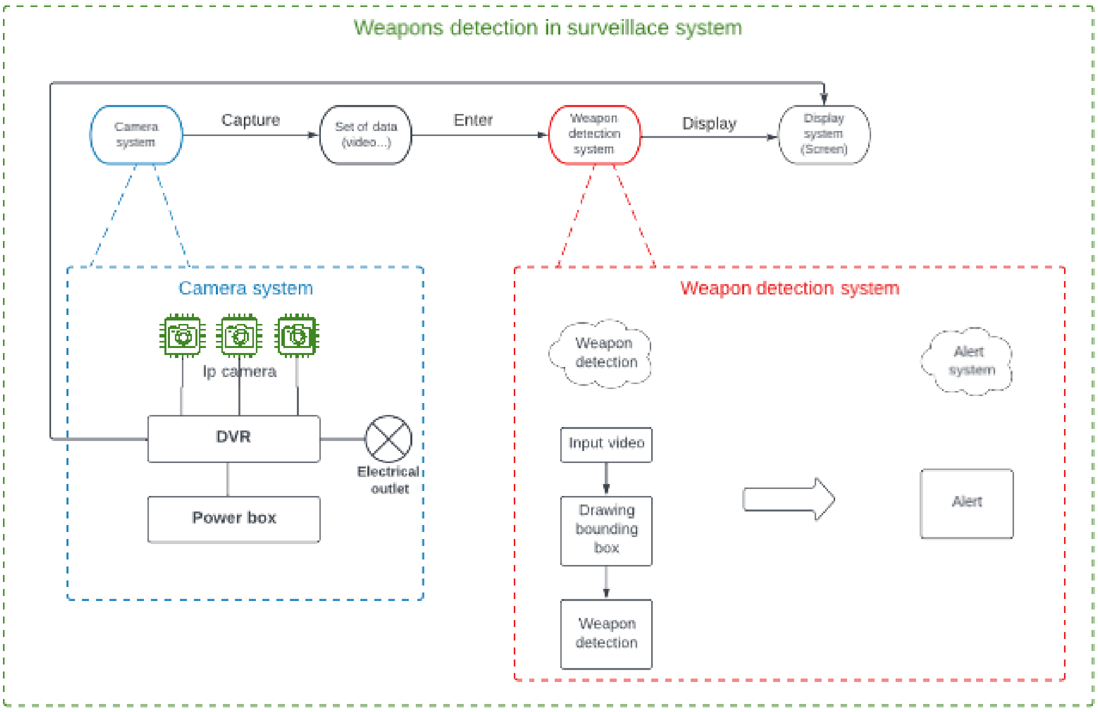

Figure III. 2 :the architecture of the weapon detection system

III.2.Implementation phases

In this section, we explain the different implementation phases and all what we use in the development.

III.2.1. Font-end

In the font end of the project we create a desktop application to help us in the testing phase to test the models we already trained.For that We use the Tkinter library to create the application's interface . We use the Python programming language because it is easy to learn and very helpful for developing applications using deep learning and computer vision. We use the Tensorflow platform to train and test our model,we test the model in 3 different input (images ,video, real-time).

III.2.2. Back-end

The Figure III. 3 shows the project flow diagram. First, the process starts by reading the input image, video files, and real-time webcam frame by frame. Then the model starts to detect an object on the input frame. The detected object is bounded by the bounding box where the bounding box has a threshold value to be achieved. In this system, the threshold value is set to a minimum of 0.5, therefore when the threshold is above 0.5, only then it will display a bounding box. Or else, it will just ignore the predicted bounding box and continue to read frames.we chose to the value 0.5 for the threshold because when we use it with a lower one the number of the detection of FP was high .

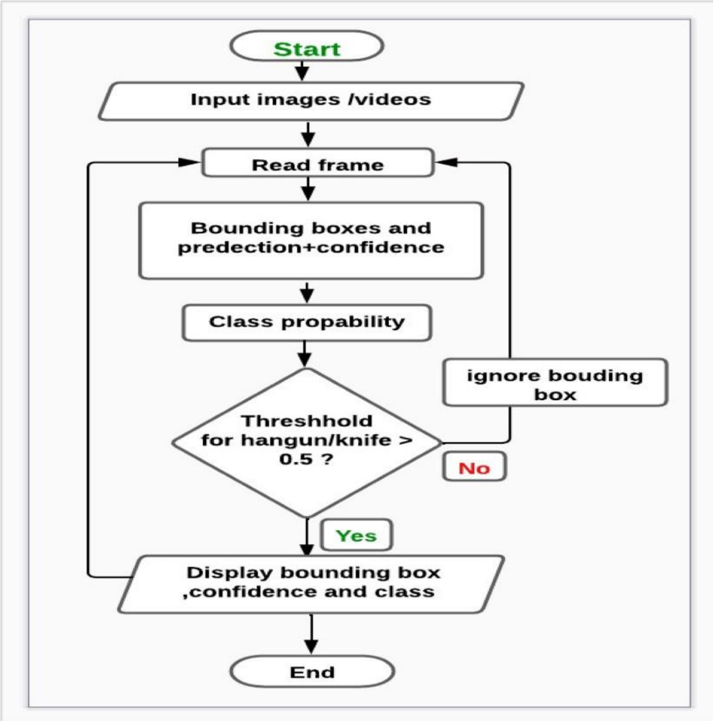

Figure III. 3: *Project flow diagram*

III.3.Implementation environment

In this section, we will present the hardware and software used in our work. Table III.1 shows all the resources and libraries that we use in all the phases of the development. First, we do the training of our model using the google colab because it provide free GPU that we can do the training faster than CPU. We use openCV library to test our models in images, video and real-time because it help as to access them easily. We use Tkinter library to create the interface of our application. We use the python programing language because it is easy to learn and very helpful in the development of the application using deep learning and computer visions. We use Tensorflow platform to train and test our model.

Table III. 1:*implementation environment*

Environment				**Description**
Dataset	**Training**	2000 images	Downloaded from [41]	See Tables II.1 and III.6
	Testing	100 images	Downloaded from google	
Training	**Hardware**	CPU	-	We trained the models using the google colab which is a web IDE for python, to enable Machine Learning with storage on the cloud using free resources [50].
		RAM	13GB	
		GPU	Tesla T4	
	Software	SSD algorithm		
		Yolov3 algorithm		
		Yolov4 algorithm		
Testing	**Hardware**	CPU	Intel® Core i3-6006U CPU 2.00GHz	-
		RAM	12GB	
		GPU	-	
	Software	OS	Windows 10	
Development	**Computer Vision**	OpenCV		OpenCV is an open-source Python library for computer vision [51]
	GUI	Tkinter		It is a Python binding to Tk GUI toolkit [52].
	Programing language	Python		Python is a high-level programming language [53]
	DL platform	Tensorflow		TensorFlow is an open-source library developed by Google [54]

III.4.Experiments and results

In this section, we will show all the results of the different phases of the implementation (training and testing).

III.4.1. Evaluation metrics

Let us now define the evaluation metrics and its relative terminology:

True positives (TP): The number of cases in which the application successfully detected a weapon where it actually exists.

True negatives (TN): The number of cases in which the application did not mistakenly identify the existence of a weapon where it actually is not.

False positives (FP): The number of cases in which the application identified weapons where they are not (mistake in identification). , it is calculated as illustrated in equation (1):

$$\frac{\text{FP}}{\text{total number of predictions}} \quad (1)$$

False negatives (FN): The number of cases in which the application failed to identify weapons while they existed. , it is calculated as illustrated in equation (2):

$$\frac{\text{FN}}{\text{total number of predictions}} \quad (2)$$

Accuracy: it defines, in general, the correctness of a model, it is calculated as illustrated in equation (3):

$$\frac{(\text{TP}+\text{TN})}{\text{total number of predictions}} \quad (3)$$

Error Rate: it defines, in general, the mistake ratio of a model, it is calculated as illustrated in equation (4)

$$\frac{(\text{FP}+\text{FN})}{\text{total number of predictions}} \quad (4)$$

The recall: it measures the model's ability to detect Positive samples. The higher the recall, the more positive samples detected. It is calculated as illustrated in equation (5)

$$\frac{\text{TP}}{(\text{TP}+\text{FN})} \quad (5)$$

III.4.2. Training

Table III.2 illustrates the results of testing the trained model using the training dataset. We note this accuracy as “training accuracy”. Overall, all models returned good accuracies for both the pistol and knife detection. Although, YOLOV4 presented more accurate weapon identifications with an overall accuracy of 95% compared to SSD which gave 92% , and the YOLOV3 which gave 93.3%. However, both YOLOV4 and YOLOV3 presented high accuracy in detecting guns, and the SSD model shows higher accuracy in detecting the knives. In the next phase, we tested our models on a new dataset to validate their performance and calculate the “testing accuracy”.

Table III. 2:the results of testing the models

Dataset	**Accuracy**			**Error Rate**								
	Yolov3	**Yolov4**	**SSD**	**Yolov3**			**Yolov4**			**SSD**		
				total	**FP**	**FN**	**total**	**FP**	**FN**	**total**	**FP**	**FN**
Guns	**95%**	**98%**	**89%**	**5%**	1%	4%	**2%**	0%	2%	**11%**	8%	3%
Knives	**92%**	**92%**	**95%**	**8%**	1%	7%	**8%**	2%	6%	**5%**	3%	2%

The Figure III.4 present the accuracy and the error rate of yolov3, yolov4 and SSD in the dataset of training of the gun

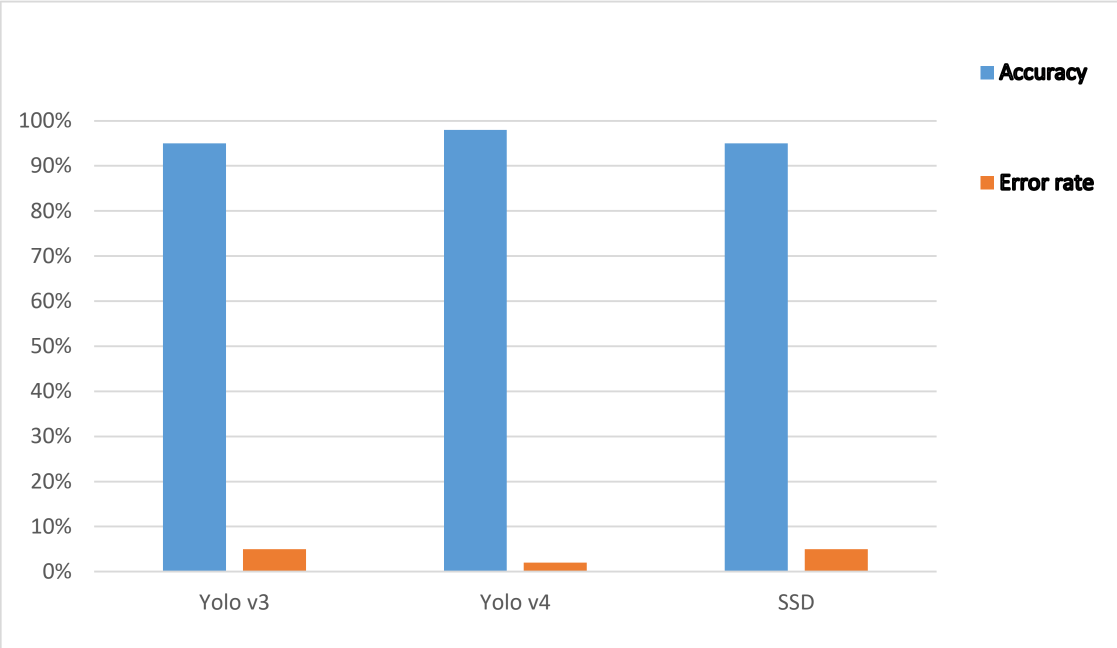

Figure III. 4:accuracy of yolov3 ,yolov4 and SSD in guns

The Figure III. 5 present the accuracy and the error rate of yolov3 ,yolov4 and SSD in the dataset of training of the knives.

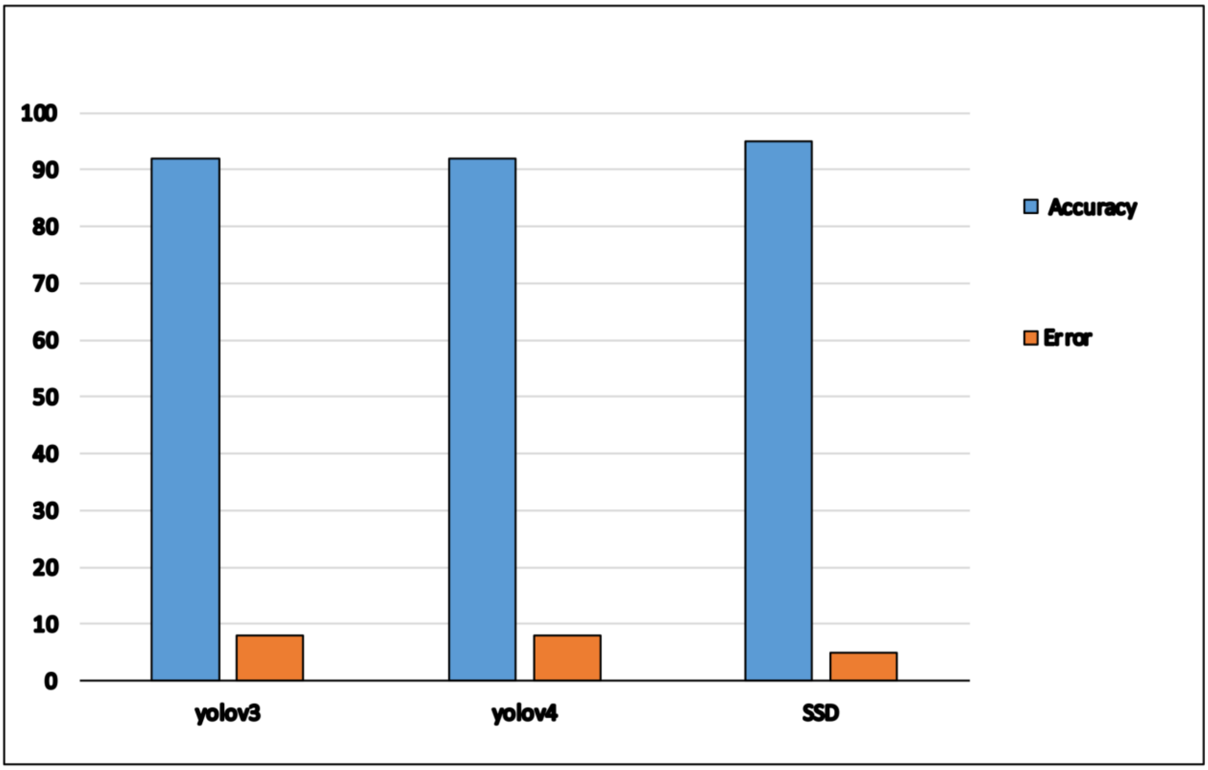

Figure III. 5 : Accuracy of yolov3 ,yolov4 and SSD in knives dataset

III.4.3. Testing

Table III.3 shows the results of testing the models on different new inputs (images, videos, real-time). We notice that the testing accuracy is different from the training accuracy. Its value has decreased and the error rate has increased for three models. In average,YOLOV3, YOLOV4 accuracy decreased to 74.3%, 84% respectively and SSD to 76.3%. Although the difference in accuracies between YOLOV3 and SSD is slight, YOLOV4 outbests both of them with around 12%. However, we notice that SSD gave the highest accuracy when processing real-time videos to detect weapons. YOLOV4, in the other hand, was the most accurate when the inputs were images and YOLOV3 gave the highest accuracy when the input is video.

The Figure III. 6 shows the accuracy rate of all the models (YOLOV3,YOLOV4 and SSD) after testing them with different input (images ,video ,real-time)

Table III. 3:the results of validation the models

Results / input	Accuracy			Error Rate								
	Yolov3	**Yolov4**	**SSD**	**Yolov3**			**Yolov4**			**SSD**		
Images	**74%**	**92%**	**77%**	**Total**	**FP**	**FN**	**Total**	**FP**	**FN**	**Total**	**FP**	**FN**
				26%	3%	23%	**8%**	2%	6%	**23%**	8%	15%
Video	**84%**	**72%**	**65%**	**16%**	7%	9%	**12%**	1%	11%	**34%**	21%	13%
Real-time	**65%**	**82%**	**87%**	**35%**	12%	23%	**18%**	2%	16%	**13%**	6%	7%

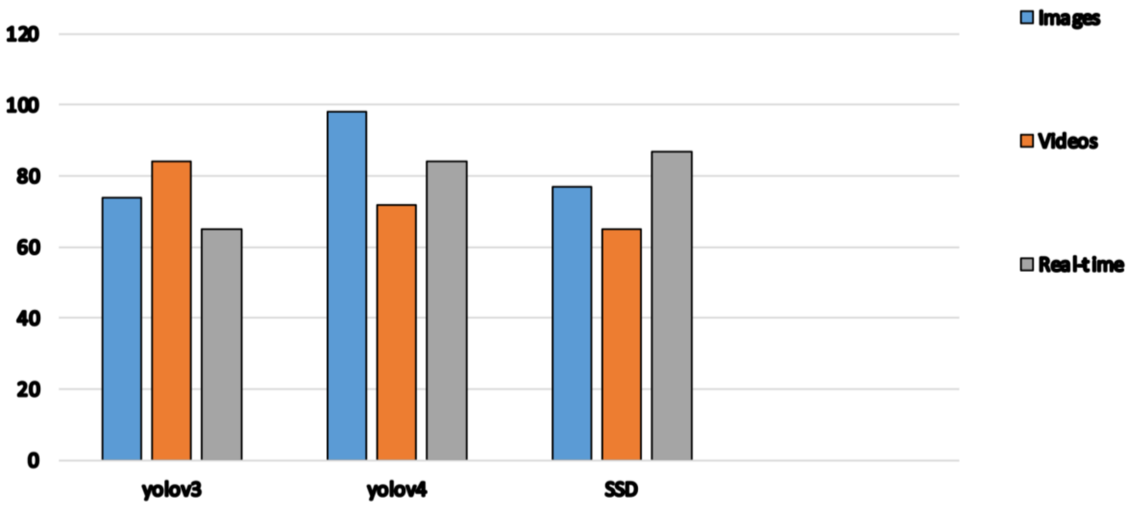

Figure III. 6 : Accuracy rate of all the models

The recall:

The Table III.4 illustrate the recall of the three models using 3 inputs, this table has been calculated using the Table III.3 of the testing ,we notice that the results of the recall were all good that means that all the models have the ability to detect Positive samples.

***Table III. 4:**the recall of the models*

Input \ model	Yolov3	Yolo4	SSD
Images	76%	93%	83%
video	90%	88%	83%
Real-time	73%	83%	92%

III.4.4. Recapping the result

Table III.5 shows the average results obtained from the three models. We can see that the average testing accuracy rate is higher when using the YOLOV4 method with 84%. Moreover, and the false-positive rate is the highest when using the SSD Mobile net with 11.6%, and the false-negative rate is the highest when using the YOLOV3. Noting that YOLOV4 presented lower error ratio for both types FP and FN.

The high false-positive rate could cause problems for example: if the rate of the false positive is high it may wrongly accuse people of using weapons and send fake alarms to the authorities which make problems.

Moreover, in another hand, the high False-negative rate will make bigger problems as if the system may ignore some cases, when a real weapon appears in the screen and caused crimes because the alert does not send to the police to send help. The Figure III. 7 shows the average Rate of Accuracy and Error Rate of all the models used in the testing.

Table III. 5:the average of the results

The results \ The method	Accuracy	Total Error Rate	FP	FN
YOLOV3	74.3%	25.7%	7.4%	18.3%
YOLOV4	84%	12.6%	1.6%	11%
SSD	76.3%	23.3%	11.6%	11.6%

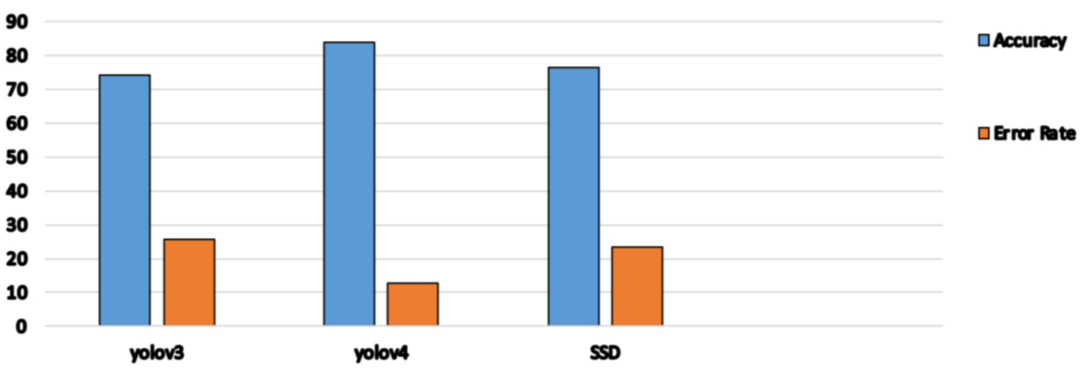

Figure III. 7 : The average Rate of Accuracy and Error Rate

III.4.5. Interpretation of test results

After calculating the average of the scales, we must note that the error rate is high, reaching 25.7%, 23.3% and 12.6% for YOLOV3, SSD and YOLOV4, respectively, and we propose several reasons for this increase.

Hypotheses:

- Record high resolution and mostly clear images.
- Limited training time.
- There are not enough images to cover all types of target weapons (knives and pistols).

To improve performance, we tried to handle the first hypothesis by adding more low-resolution images and collecting them using the PC camera. In this part of experiment, we did the same steps implementation like before (collected and labeling images, training, etc.). The description of the collected images is given in Table III.6.

Table III. 6:collected images description

	Up	Down	Left	Right	Facing forward	On the waist
Knives	264	175	276	284	/	/
Gun	227	182	205	239	58	91

Due the time limit, we test the new dataset only with the YOLOV3 and YOLOV4, the tables III.7 and III.8 resume the results. Table III.6 shows the results of testing the models on different new inputs (images, videos, real-time), this model was trained using the new dataset after adding more images to the previous data. We notice that the testing accuracy is different from the first dataset accuracy. Its value has increased and the error rate has decreased for both models. In average, YOLO V3, YOLOV4 accuracy increased to 77.6%, 89% respectively. The difference in accuracies between YOLOV3 and YOLOV4 is around 12%. YOLOV4 gave higher accuracy when processing images and videos to detect weapons and the accuracy of both models was equal when the inputs were real-time video.

The Figure III. 8 shows the results of validation the models after adding more images in the first dataset.

Table III. 7:the results of validation the models after adding more images

Results / input	Accuracy		Error Rate					
	Yolov3	Yolov4	Yolov3			Yolov4		
Images	76.3%	90%	Total	FP	FN	Total	FP	FN
			23.7%	7.7%	16%	10%	2%	8%
Video	74.6%	93%	25.4%	10%	15.4%	7%	6%	1%
Real-time	84%	84%	16%	12%	4%	16%	10%	6%

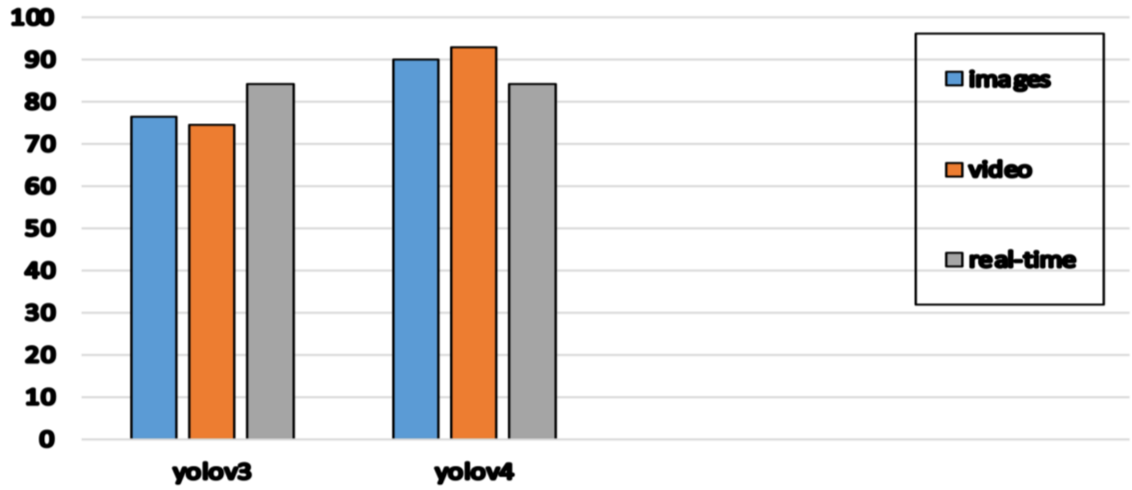

Figure III. 8 : The results of validation ofthe models after adding more images

Table III.8 recaps the average results between all the inputs in both methods, we can see that the testing accuracy rate is higher when using YOLOV4 method with 89%. Moreover, the Error Rate is higher when using YOLOV3 with 21.7%. The Figure III. 9 represent the average of Accuracy and Error rates in the second experiment.

Table III. 8:the average of the results after adding images to the dataset

The results / The method	Accuracy	Total Error Rate	FP	FN
Yolov3	77.6%	21.7%	9.9%	11.8%
Yolov4	89%	11%	6%	5%

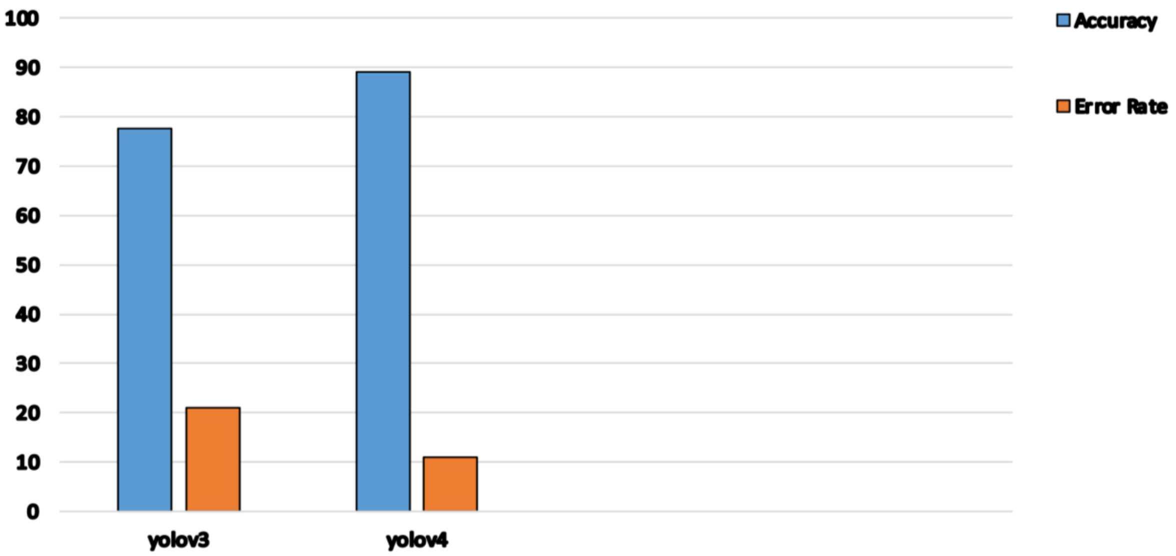

Figure III. 9 : The average Rate of Accuracy and Error

III.5.Conclusion

In this chapter, we explain the architecture of the application and how it works (front-end/backend), and show the implementation phases and environments in which we mentioned all working steps (training, testing, and development). In the implementation, we train our dataset in google colab using 3 models YOLOV3,YOLOV4 and the SSD mobile net. First, we test the trained model on the training dataset, the results are satisfactory.

Then we test the model with different inputs (image, video, real-time). We found a drop in results. Because of this, we decided to create another dataset from PC's webcam to acquire low-resolution images to increase real-time test results, but due to time constraints, we only use YOLOV3 and YOLOV4. The results we get are 77.6% and 89% for YOLOV3 and YOLOV4, respectively.

General Conclusion and Future perspectives

In today's conditions, criminal activities are increasing, it is important to detect the weapon usage from images taken by security cameras, in the most accurate, automatic and fast way. Weapon detection and recognition are important to prevent criminal activities before they occur and so that the appropriate parties can take necessary action. Most criminal activities are carried out using handheld weapons. Thus, it is necessary to determine the use of weapons from security cameras and to take the necessary precautions beforehand.

In our thesis, we used and compared between three methods of object detection models which are SSDMobileNetV2, YOLOV3 and YOLOV4. The performance is evaluated using the accuracy and error rate metrics.

Real-time weapons detection is a difficult task. Because our target object is small, recognizing it in an image in the presence of other objects, particularly those that present similarities with it, is hard. For the detection and classification task, deep learning models faced the following challenges:

- The first and main problem is the data through which CNN learn its features to be used later for classification and detection. There is no standard dataset available for weapons.
- For real-time detection scenarios, constructing the training dataset manually was a long and time-consuming process.
- As for real-time implementation, the detection systems works only when weapon is used explicitly, so weapon blocking or occlusion is also a problem that arises frequently and it could occur because of self, inter-object, or background blocking.

As a result of the comparison, in term of accuracy, YOLOV4 showed more promising results with 84% followed by the SSD model with 76.3 then the YOLOV3 with 74.3 that when we use the first dataset .

The real-time scenario was tested on YOLO models only after adding the self-constructed dataset, due the limit of time. We noted that the accuracy of both of them has increased, YOLOV4 to 89% and the YOLOV3 to 77.6%.

For future work, we intend to add different types of guns to the dataset to enrich it further. We also intend to explore the use of other neural network architectures with a focus towards small scale object detection in attempt to further improve the accuracy in detecting firearms and knives use from a surveillance video. Also, we intend to add the "send alarm" functionality to the system to send alerts to the police containing the criminal face, weapon used and location of the crime attempt.

The auto-evaluation grid

Task/objective	**State**	**Details and remarks**
☒: Achieved. ☐: Not achieved		
A historical review of the emergence and development of surveillance systems.	☒	
An overall study on crimes committed between individuals around the world.	☒	We mentioned the statistics in a non-extensive manner. Other existing may not have been mentioned.
Seeing the reasons for the failure of monitoring systems when they first appear.	☒	
Informing the reader of the importance of having a monitoring system in public places.	☒	We study some of the consequences of each method has been used before
Reviewing some public safety regulations and their importance.	☒	
Studying the history of object detection methods.	☒	We add only the previous methods related to our topic
Adding real-time alerts	☒	We only used audio alerts in the current version.
Studying some previous works on weapons detection using deep learning	☒	We add some examples of the existing works not all.
Study the basics of convolutional neural networks	☒	
Looking at object detection algorithms	☒	
Implementation of three object detection algorithms: Yolov3,Yolov4 and SSD mobile Net	☒	Due to the time limit, we chose only three to do the experiments.
Create a Graphical User Interface	☒	We create a simple interface to illustrate the results in real-time.
Weapon detection through the computer camera	☒	Due to the limited resources, we test the app in real-time from the pc camera, we may use a real CCTV camera in the future amelioration.
Analyzing the detection results of the three algorithms (Accuracy, false-positive, false-negative)	☒	
A comprehensive study and comparison of the three algorithms performance	☒	The comparative done using the metrics calculated.
Add more classes and different types of guns to the dataset	☐	Not added in the current version due to time limit, to be added in its amelioration.

BIBLIOGRAPHY

[1] World Population Review, “Violent Crime Rates by Country 2022,” *Violent Crime Rates by Country 2022*. 2022. [Online]. Available: https://worldpopulationreview.com/country-rankings/violent-crime-rates-by-country

[2] “Gun Deaths By Country 2020.” 2020. [Online]. Available: https://worldpopulationreview.com/countries/gun-deaths-by-country/

[3] “Stabbing Deaths by Country 2022.” [Online]. Available: https://worldpopulationreview.com/country-rankings/stabbing-deaths-by-country

[4] “29/6, 2021.” p. 2021, 2021. [Online]. Available: https://www.alaraby.co.uk/society/296-2021-ألف-جريمة-شهدتها-الجزائر-خلال

[5] R. V Clarke, *Problem-Oriented Guides for Police Improving Street Lighting to Reduce Crime in Residential Areas*, no. 8. 2008. [Online]. Available: http://cops.usdoj.gov/Publications/e1208-StreetLighting.pdf

[6] W. Deisman, *CCTV : Literature Review and Bibliography CCTV : Literature Review and Bibliography*, no. May. 2016.

[7] J. Ratcliffe, *Video Surveillance of Public Places*, no. 4.

[8] A. Matiolanski and P. Guzik, “Automated Detection of Firearms and Knives in a CCTV Image Automated Detection of Firearms and Knives in a CCTV Image,” no. December, 2015, doi: 10.3390/s16010047.

[9] J. Russell, “China’s CCTV surveillance network took just 7 minutes to capture BBC reporter | TechCrunch,” *Techcrunch.com*. 2018. [Online]. Available: https://techcrunch.com/2017/12/13/china-cctv-bbc-reporter/

[10] I. Darker, A. Gale, L. Ward, and A. Blechko, “Can CCTV reliably detect gun crime?,” *Proc. - Int. Carnahan Conf. Secur. Technol.*, pp. 264–271, 2007, doi: 10.1109/CCST.2007.4373499.

[11] W. T. Sung, S. C. Tasi, and S. J. Hsiao, “Aquarium Monitoring System Based on Internet of Things,” *Intell. Autom. Soft Comput.*, vol. 32, no. 3, pp. 1649–1666, 2022, doi: 10.32604/IASC.2022.022501.

[12] M. Siala, N. Khlifa, F. Bremond, and K. Hamrouni, “People detection in complex scene using a cascade of boosted classifiers based on Haar-like-features,” *IEEE Intell. Veh. Symp. Proc.*, no. July, pp. 83–87, 2009, doi: 10.1109/IVS.2009.5164257.

[13] M. Grega, A. Matiolański, P. Guzik, and M. Leszczuk, “Automated detection of firearms and knives in a CCTV image,” *Sensors (Switzerland)*, vol. 16, no. 1, 2016, doi: 10.3390/s16010047.

[14] T. Sikora, “The MPEG-7 visual standard for content description - An overview,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 6, pp. 696–702, 2001, doi: 10.1109/76.927422.

[15] B. Sch, R. C. Williamson, and P. L. Bartlett, “Schölkopf et al. - 2000 - New support vector algorithms.pdf,” vol. 1245, pp. 1207–1245, 2000.

[16] J. Canny, “A Computational Approach to Edge Detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, pp. 679–698, 1986, doi: 10.1109/TPAMI.1986.4767851.

[17] M. Bober, “MPEG-7 visual shape descriptors,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 6, pp. 716–719, 2001, doi: 10.1109/76.927426.

[18] A. Ahsan, “Gun and Human Movement Using Haar Cascade Classifier for Military Application,” no. Icnte, 2019.

[19] J. Guo, J. Shi, W. Zhu, and J. Wang, “Approach to autonomous stair climbing for tracked robot,” *Proc. 2017 IEEE Int. Conf. Unmanned Syst. ICUS 2017*, vol. 2018-Janua, pp. 182–186, 2018, doi: 10.1109/ICUS.2017.8278337.

[20] S. Reinius, “Object recognition using the OpenCV Haar cascade-classifier on the iOS platform,” *Institutionen för Informationsteknologi, Uppsala Univ.*, 2013, [Online]. Available: http://www.diva-portal.org/smash/record.jsf?pid=diva2:601707

[21] S. Soo, “Object detection using Haar-cascade Classifier Object detection using Haar-cascade Classifier”.

[22] A. Jain, Aishwarya, and G. Garg, “Gun detection with model and type recognition using haar cascade classifier,” *Proc. 3rd Int. Conf. Smart Syst. Inven. Technol. ICSSIT 2020*, no. Icssit, pp. 419–423, 2020, doi: 10.1109/ICSSIT48917.2020.9214211.

[23] C. Jyotsna and J. Amudha, “Detection from Surveillance Video,” *ICIMIA 2020 IEEE Int. Conf. Innov. Mech. Ind. Appl.*, no. Icimia, pp. 335–339, 2020.

[24] A. Castillo, S. Tabik, F. Pérez, R. Olmos, and F. Herrera, “Brightness guided preprocessing for automatic cold steel weapon detection in surveillance videos with deep learning,” *Neurocomputing*, vol. 330, pp. 151–161, 2019, doi: 10.1016/j.neucom.2018.10.076.

[25] J. Lim, M. I. Al Jobayer, V. M. Baskaran, J. M. Lim, K. Wong, and J. See, “Gun detection in surveillance videos using deep neural networks,” *2019 Asia-Pacific Signal Inf. Process. Assoc. Annu. Summit Conf. APSIPA ASC 2019*, no. November, pp. 1998–2002, 2019, doi: 10.1109/APSIPAASC47483.2019.9023182.

[26] H. Jain, A. Vikram, Mohana, A. Kashyap, and A. Jain, “Weapon Detection using Artificial Intelligence and Deep Learning for Security Applications,” *Proc. Int. Conf. Electron. Sustain. Commun. Syst. ICESC 2020*, no. Icesc, pp. 193–198, 2020, doi: 10.1109/ICESC48915.2020.9155832.

[27] N. Geetha, A. K. K. S, B. P. Akshita, and M. Arjun, “Weapon Detection in Surveillance System,” vol. 10, no. 05, pp. 122–127, 2021.

[28] Z. Q. Zhao, P. Zheng, S. T. Xu, and X. Wu, “Object Detection with Deep Learning: A Review,” *IEEE Trans. Neural Networks Learn. Syst.*, vol. 30, no. 11, pp. 3212–3232, 2019, doi: 10.1109/TNNLS.2018.2876865.

[29] E. Arif, S. K. Shahzad, R. Mustafa, M. A. Jaffar, and M. W. Iqbal, “Deep neural networks for gun detection in public surveillance,” *Intell. Autom. Soft Comput.*, vol. 32, no. 2, pp. 909–922, 2022, doi: 10.32604/iasc.2022.021061.

[30] A. Chalfin, B. Hansen, J. Lerner, and L. Parker, “Reducing Crime Through Environmental Design: Evidence from a Randomized Experiment of Street Lighting in

New York City," *J. Quant. Criminol.*, vol. 38, no. 1, pp. 127–157, 2022, doi: 10.1007/s10940-020-09490-6.

[31] C. B. Murthy, M. F. Hashmi, N. D. Bokde, and Z. W. Geem, "Investigations of object detection in images/videos using various deep learning techniques and embedded platforms-A comprehensive review," *Appl. Sci.*, vol. 10, no. 9, 2020, doi: 10.3390/app10093280.

[32] P. Minist *et al.*, "Thème : La détection de Covid-19 par l ' apprentissage profonde (Deep Learning)," 2021.

[33] "Comprendre le machine learning et le deep learning - Bial-R." [Online]. Available: https://www.bial-r.com/2019/05/22/comprendre-le-machine-learning-et-le-deep-learning/

[34] M. Mishra, "Convolutional Neural Networks, Explained | by Mayank Mishra | Towards Data Science." 2020. [Online]. Available: https://towardsdatascience.com/convolutional-neural-networks-explained-9cc5188c4939

[35] B. Yassine and G. Larbi, "An Overview of Deep Learning-Based Object Detection Methods," no. March, pp. 233–238, 2018, [Online]. Available: https://semanticscholar.org/paper/9e11c2a9b6e5aef371f10b2837230fd800cda678

[36] Hmrishav Bandyopadhyay, "YOLO: Real-Time Object Detection Explained," *V7*. p. 1, 2022.

[37] Y. Dai, W. Liu, H. Li, and L. Liu, "Efficient Foreign Object Detection between PSDs and Metro Doors via Deep Neural Networks," *IEEE Access*, vol. 8, pp. 46723–46734, 2020, doi: 10.1109/ACCESS.2020.2978912.

[38] Pierrick RUGERY, "Explanation of YOLO V4 a one stage detector | by Pierrick RUGERY | Becoming Human: Artificial Intelligence Magazine," *Becominghuman.Ai*. 2020. [Online]. Available: https://becominghuman.ai/explaining-yolov4-a-one-stage-detector-cdac0826cbd7

[39] A. Singh, T. Anand, S. Sharma, and P. Singh, "IoT Based Weapons Detection System for Surveillance and Security Using YOLOV4," *Proc. 6th Int. Conf. Commun. Electron. Syst. ICCES 2021*, pp. 488–493, 2021, doi: 10.1109/ICCES51350.2021.9489224.

[40] "YOLOv4 - Part 1_ Introduction _ VisionWizard." [Online]. Available: https://medium.com/visionwizard/yolov4-version-0-introduction-90514b413ccf

[41] "GitHub - ari-dasci_OD-WeaponDetection_ Datasets for weapon detection based on image classification and object detection tasks."

[42] KeitoTobi1, "tzutalin/labelImg: LabelImg is a graphical image annotation tool and label object bounding boxes in images," *GitHub*. 2017. [Online]. Available: https://github.com/tzutalin/labelImg

[43] "Object Detection using SSD Mobilenet V2 _ by Vidish Mehta _ Medium."

[44] W. Liu *et al.*, "SSD: Single shot multibox detector," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 9905 LNCS, pp. 21–37, 2016, doi: 10.1007/978-3-319-46448-0_2.

[45] M. Singhal, "Object Detection using SSD Mobilenet and Tensorflow Object Detection API : Can detect any single class from coco dataset. | by mayank singhal | Medium." p. 1, 2020. [Online]. Available: https://medium.com/@techmayank2000/object-detection-using-ssd-mobilenetv2-using-tensorflow-api-can-detect-any-single-class-from-31a31bbd0691

[46] T. J. Sheng *et al.*, "An Internet of Things Based Smart Waste Management System Using LoRa and Tensorflow Deep Learning Model," *IEEE Access*, vol. 8, pp. 148793–148811, 2020, doi: 10.1109/ACCESS.2020.3016255.

[47] "How single-shot detector (SSD) works_ _ ArcGIS API for Python."

[48] "models_tf2_detection_zoo." [Online]. Available: https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/tf2_detection_zoo.md

[49] "Technology and Innovation for Public Safety – Alan Blog."

[50] "How Google has crushed it with Colaboratory | by Sarah Quinn | Kainos Applied Innovation | Medium." [Online]. Available: https://medium.com/kainos-applied-innovation/how-google-has-crushed-it-with-colaboratory-5664b5fb5856

[51] "OpenCV Tutorial _ OpenCV using Python - javatpoint." [Online]. Available: https://www.javatpoint.com/opencv

[52] "List of Tkinter Widgets - with examples - CodersLegacy." [Online]. Available: https://coderslegacy.com/python/list-of-tkinter-widgets/

[53] P. Christensson, "Python Definition," *Techterms.com*. 2010. [Online]. Available: https://techterms.com

[54] Simplilearn, "What is Tensorflow: Deep Learning Libraries and Program Elements Explained," *Https://Www.Simplilearn.Com/Tutorials/Deep-Learning-Tutorial/What-Is-Tensorflow*. 2021.