Weapon Detection using Artificial Intelligence and Deep Learning for Security Applications

A

Project Report

Submitted in partial fulfilment of the
Requirements for the award of the Degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE & ENGINEERING

By

M AARTHI (1602-19-733-309)

Under the guidance of

Dr. Bhargavi Peddi Reddy

ASSOCIATE PROFESSOR

Department of Computer Science & Engineering

Vasavi College of Engineering (Autonomous)

(Affiliated to Osmania University)

Ibrahimbagh, Hyderabad-31

2023

Vasavi College of Engineering (Autonomous)

(Affiliated to Osmania University)

Hyderabad-500 031

Department of Computer Science & Engineering

DECLARATION BY THE CANDIDATE

We, **M AARTHI** bearing hall ticket number, **1602-19-733-309** hereby declare that the project report entitled **Weapon Detection using Artificial Intelligence and Deep Learning for Security Applications** under the guidance of

Dr. Bhargavi Peddi Reddy, ASSOCIATE PROFESSOR, Department of Computer Science & Engineering, VCE, Hyderabad, is submitted in partial fulfilment of the requirement for the award of the degree of **Bachelor of Engineering** in **Computer Science & Engineering**.

This is a record of bonafide work carried out by me and the results embodied in this project report have not been submitted to any other university or institute for the award of any other degree or diploma.

M AARTHI,

1602-19-733-309

Vasavi College of Engineering (Autonomous)

(Affiliated to Osmania University)

Hyderabad-500 031

Department of Computer Science & Engineering

BONAFIDE CERTIFICATE

This is to certify that the project entitled **Weapon Detection using Artificial Intelligence and Deep Learning for Security Applications** being submitted by **M AARTHI** and bearing **1602-19-733-309** in partial fulfilment of the requirements for the award of the degree of Bachelor of Engineering in Computer Science & Engineering is a record of bonafide work carried out by him/her under my guidance.

Dr. Bhargavi Peddi Reddy	**Dr.T. Adilakshmi,**
Associate Professor,	**Professor & HEAD,**
Internal Guide	**Dept. of CSE**

ACKNOWLEDGEMENT

The satisfaction that accompanies that the successful completion of the project seminar would not have been possible without the kind support and help of many individuals. We would like to extend our sincere thanks to all of them.

It is with immense pleasure that we would like to take the opportunity to express our humble gratitude to **Dr. Bhargavi Peddi Reddy ,Associate Professor, Computer Science and Engineering** under whom we executed this project.We are also grateful to Dr.M.Sunitha for her guidance. Their constant guidance and willingness to share their vast knowledge made us understand this project and its manifestations in great depths and helped us to complete the assigned tasks.

We are very much thankful to **Dr.T.Adilakshmi, Professor and HOD, Dept. of CSE,** for his kind support and for providing necessary facilities to carry out the work.

We wish to convey our special thanks to **Dr. S.V.Ramana, Principal** of **Vasavi College of Engineering** for giving the required information in doing my project work. Not to forget, we thank all other faculty and non-teaching staff, and my friends who had directly or indirectly helped and supported me in completing my project in time.

We also express our sincere thanks to the Management for providing excellent facilities. Finally, we wish to convey our gratitude to our family who fostered all the requirements and facilities that we need.

M AARTHI

1602-19-733-309

ABSTRACT

Due to an increase in crime in crowded events or unsettling lonely areas, security is always of the utmost significance in any field. Computer vision is extensively used in abnormal discovery and monitoring to break colorful issues. Due to the adding need to defend particular safety, security, and property, videotape surveillance systems that can identify and decrypt scene and anomaly circumstances are essential for intelligence monitoring. In this study, the SSD and Faster RCNN algorithms, which are grounded on complication neural networks(CNNs), are used to develop automatic gun(or armament) discovery. The suggested perpetration employs two different datasets. One dataset had photos that were formerly labelled, and the other contained images that demanded to be manually labelled. When the results are tallied, both algorithms do well.

TABLE OF CONTENTS

LIST OF TABLES……………………………………….

1. INTRODUCTION……………………………………

   1.1 MOTIVATION…………………………………

   1.2 PROBLEM DEFINATION…………………….

   1.3 OBJECTIVE……………………………………

   1.4 SCOPE OF THE PROJECT…………………….

   1.5 LITERATURE SURVEY

2. SYSTEM ANALYSIS/OVERVIEW OF PROPOSED SYSTEM………………

2.1 TRADITIONAL SYSTEM…………………………………………………………

2.2 SYSTEM REQUIREMENTS………………………………………………

2.3 SYSTEM STUDY……………………………………………………….

3. SYSTEM DESIGN…………………………………………………………..

3.1 SYSTEM ARCHITECTURE…………………………………………….

3.2 OBJECTIVE OF PROPOSED MODEL…………………………………

3.3 ALOGRITHMS USED FOR PROPOSED MODEL………………………

3.4 DESIGNING…………………………………………………………….

3.5 STEPWISE IMPLEMENTATION…………………………………………

4. PSEUDO ALGORITHM/IMPLEMENTATION…………………………………….

5. RESULTS…………………………………………………………………...

6. TESTING…………………………………………………………………

7. CONCLUSION & FUTURE WORK…………………………………………

8. REFERENCES…………………………………………………………..

List of Figures

FIGURE NO.	TITLE	PAGE NO.
2.1.1	Luggage Checking Machine	4
2.1.2	Walk-Through Metal Detector	7
2.1.3	Implementation of Walk-Through Metal detector	8
3.1.1	System Architecture for Proposed Model	24
3.3.1	Basic Diagram of SSD Algorithm	26
3.3.2	Basic Diagram of Faster RCNN Algorithm	28
3.4.1	Describes the Methodology	33

3.4.2	Detection & Tracking	34
3.4.3	Block Diagram	35
5.1.1	Step 1 during execution	41
5.1.2	Step 2 in execution	42
5.1.3	Step 3 in execution if detected	43
5.1.4	Step 4 in execution if not detected	44
5.1.5	Step 5 in execution if detected	45
5.1.6	Step 6 in execution if not detected	46
5.1.7	Step 7 in execution if detected	47

1. INTRODUCTION

1.1 MOTIVATION

In traditional videotape surveillance, security agents have to visually descry the presence of munitions in covered scenes by watching security footage and snappily make opinions grounded on it. AI vision styles are used to fete shanks and ordnance with the cctv discovery to reduce crimes and increase public safety and security. Weapon or Anomaly discovery is the identification of irregular, unanticipated, changeable, unusual events or particulars, which isn't considered as a typically being event or a regular item in a pattern or particulars present in a dataset and therefore different from being patterns. An anomaly is a pattern that occurs else from a set of standard patterns. thus, anomalies depend on the miracle of interest. Object discovery uses point birth and literacy algorithms or models to fete cases of colorful order of objects. Proposed perpetration focuses on accurate gun discovery and bracket. Also concerned with delicacy, since a false alarm could affect in adverse responses. Choosing the right approach needed to make a proper trade- off between delicacy and speed. The use of AI and deep literacy for armament discovery has several advantages over traditional styles similar as homemade webbing or essence sensors. It's briskly, more accurate, and can descry a wider range of munitions, including those made ofnon-metallic accoutrements. It also reduces the need for mortal intervention, allowing security labour force to concentrate on further critical tasks. The methodology of munitions discovery using deep literacy. Frames are uprooted from the input videotape. Frame differencing algorithm is applied and bounding box created before the discovery of object. Dataset is created, trained and fed to object discovery algorithm. Grounded on operation suitable discovery algorithm (SSD or fast RCNN) chosen for gun discovery. The approach addresses a problem of discovery using colorful machine literacy models like Region Convolutional Neural Network (RCNN), Single Shot Detection (SSD).

1.2 PROBLEM DEFINITION

To create and install a reliable System to identity Weapons nearby

Input: Image containing weapon.

The procedure involves using the CNN algorithm to identify each individual component portion of the weapon.

If any weapons are discovered after identification, they will be found.

Output: when a weapon is discovered, show the weapon type.

1.3 OBJECTIVE

Security Operations influence AI and deep literacy ways for armament discovery. These vision-grounded algorithms can identify and comprehend scenes from videotape from surveillance systems. Chancing one or further effective targets from still images or videotape data is the primary thing of object discovery. It completely incorporates a wide range of pivotal styles, including image processing, pattern recognition, artificial intelligence, and machine literacy.

1.4 SCOPE OF THE PROJECT

These days, keeping an eye out for illicit activity involves ongoing human observation.

Handheld weapons, particularly the pistol and gun, are the principal cause of the majority of these activities.

Weapons like knives and firearms have been detected using object detection algorithms. Due to frequent background cluttering, occlusion, and changes in viewpoint in a scene, handgun and knife detection is one of the most difficult tasks. The algorithms that have been utilised to detect knives and handguns were evaluated and categorised in this research along with their advantages and disadvantages. The algorithms used to identify knives and pistols are reviewed in this study. Since this model only recognises the name of the weapon, it can be enhanced in the future by include more features, such the number of weapons, provided we supply more information.

1.5 LITERATURE SURVEY

Abstract:

In these modern days the world's population is increasing day by day and also the technology is increasing day by day. The security of the human being should be updated up to date. In our daily life we were very familiar that terror activities are very high. By these activities many innocent people were lost their beautiful life. By considering these types of activities Our goal is to create a system that can quickly and accurately identify potential threats and alert security personnel, allowing them to take appropriate action to prevent harm to individuals and property. The use of AI and deep learning in weapon detection can significantly improve the accuracy and reliability of security systems, reducing the risk of false alarms and enhancing overall public safety. Due to growing demand in the protection of safety, security and particular parcels, requirements and deployment of videotape surveillance systems can fete and interpret the scene and anomaly events play a vital role in intelligence monitoring. This paper implements the detection of various weapons like hand guns machine guns knifes explosives etc., through closed circuit television (CCTV) automatically by applying the CNN based algorithms called SSD & Faster RCNN algorithms. We compare both the results in the form of accuracy and speed.

2. SYSTEM ANALYSIS

2.1 TRADITIONAL SYSTEM:

2.1.1 Luggage Checking Machine:

It uses x-ray vision to scan the language and displays the weapon to the monetarized person on the screen. This aids in locating criminals and is primarily utilised in airports and metro stations. Airport RFID readers read the information about the baggage from the tag and send it to the control station. Moving the luggage from one stage to the next and so forth. This RFID tag maintains constant communication with the RFID reader and provides the user with luggage details via a message sent via GSM modem, as well as transferring the same information to the database. An emerging breakthrough called Radio Frequency Identification (RFID) is rapidly taking over commerce and industry. The carrier sector is one of the many businesses that.

2.1.1 Luggage Checking Machine

Disadvantages:

There is currently no security X-ray technology on the market that can decide on its own if a danger is acceptable and dependable or not. All remain heavily.
rely on operators who are humans to see and interpret the photos. You must be aware of both the drawbacks and benefits of using X-rays as a security measure if you want to minimize any issues and be able to solve them.

2.1.2 Luggage Checking Implementation

X-Ray Image Enhancement

The notion enhancing digital images in the spatial sphere is grounded on applying some fine pollutants on the image matrix. improvement styles in the spatial sphere are astronomically divided into three orders point processing styles, histogram- grounded processing styles, and mask processing styles. In this work histogram equalization and un-sharp masking are used. Where Pr(rk) is the probability occurrence of intensity position rk in the digital image, MN is the total no of pixels in the image, nk is the no of pixels that have intensity rk. The computation of new intensity value of each intensity position is done using the metamorphosis(mapping)
$sk = T(rk) = (L - 1) \sum Pr(rj)$

Ammos Detection:

The process of image improvement involves converting the enhanced X-ray image into a double image using Otsu thresholding. This is followed by removing noise from the double image using morphological drivers. Next, any connected objects on the borders of the double image are eliminated. Finally, morphological drivers are used to remove holes of specific sizes. In phase 2 of the process, the Otsu thresholding system is utilized to identify the optimal threshold that minimizes the intra-class friction. This friction is defined as a weighted sum of the dissonances of the two classes.

$$\sigma w (t) = w (t) \sigma (t) + w (t) \sigma (t)$$

The study involved improving X-ray images using various image processing techniques such as histogram equalization, un-sharp masking, and Otsu's thresholding to enable better detection and classification of ammunition. Connected-component labelling was utilized to identify connected regions in binary digital images, with the 8-connected component labelling algorithm selected for its ability to check diagonal neighbours' as well as immediate neighbour's. The algorithm works by scanning the image in a row-wise manner, identifying points with intensity value V={1}, and examining their surrounding neighbours' to assign a label. Depending on the state of the neighbours', either a new label is allocated, or an existing label is assigned to the point. Any equivalences between labels are noted, sorted into classes, and assigned an exclusive label using the Floyd-Warshall Algorithm. Finally, the image is scanned a second time to replace each label with the label assigned to its equivalence class.

2.1.2 Walk-Through Metal Detector:

The walk-through metal detector is a machine that detects the weapons from any person. It is usually seen in the hotels, malls government offices and sports events etc., It works under the principle of electromagnetic induction which states that when an electric pulse is transferred through a coil of line, it creates a glamorous field. When the field hits a essence object, it reflects back and can be detected using another coil of line. The size and timing of the detected palpitation is used to define the size and position of the object.

2.1.2 Walk Through Metal Detector

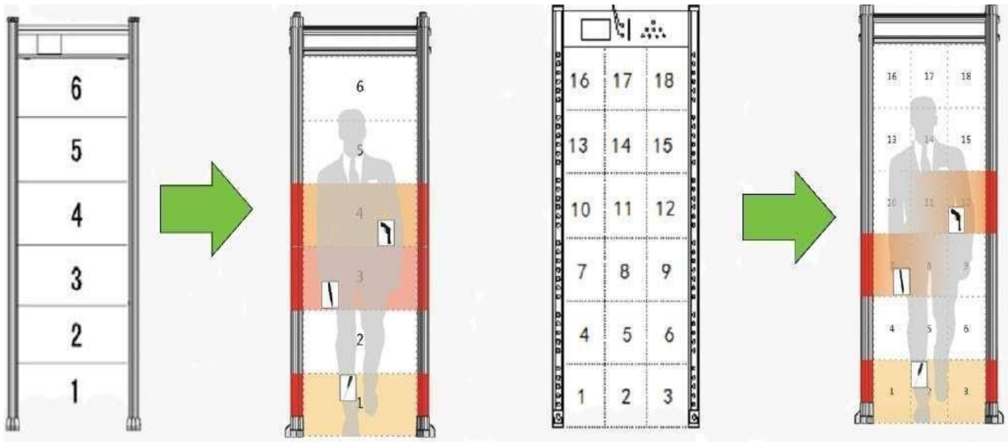
Disadvantages:

Electromagnetic fields are produced by metal detectors. Electronic devices that a person may own may experience electrical interference as a result. This also applies to medical equipment like pacemakers. Some metal detectors have a strong electromagnetic field that can damage pacemakers. It is extremely advised that they avoid passing past such detectors in this situation because they may prove fatal for those individuals.

2.1.3 Walk- Through Metal Detector Implementation

Ultramodern walk- through essence sensors use PI, or palpitation induction technology. PI systems send out short and important bursts of currents through a coil of line and each palpitation generates a short glamorous field. When a piece of essence passes through the glamorous field, a reflected glamorous field is created. A typical PI- grounded essence sensor sends about 100 beats per second, but the number can vary grounded on the manufacturer and model, ranging from about 25 beats per second to over 1,000. Walk- through essence sensors use glamorous fields to descry metal that passes through them. The wisdom behind how they work is grounded on

Maxwell’s equations. $\oint \to B \cdot d \to \ell = \mu 0(I \, ddt(\varepsilon 0 \int \to E \cdot d \to A))$ this gives the total glamorous force around a circuit.



2.1.3. Implementation of Walk-Through Metal Detector

2.1.4 Key issues for Existing System

- **Laguagge Checking:**

Privacy: Bag checking can infringe on people's privacy, especially if the searches are invasive or conducted without cause. It is important to balance security needs with the right to privacy.

Discrimination: Bag checking can be used to unfairly target certain groups of people, such as those based on their race, religion, or ethnicity. It is important to ensure that bag checks are conducted fairly and without discrimination.

Inefficiency: Bag checking can be time-consuming and can cause long lines or delays, especially if the procedures are poorly designed or if the staff is not adequately trained.

False positives: Bag checking technology is not always accurate, and innocent people may be flagged for further inspection. It is important to have protocols in place to handle false positives and minimize the inconvenience to travellers.

Legal issues: Bag checking conducted by law enforcement officials must comply with constitutional protections against unreasonable searches and seizures. It is important to understand the legal framework and ensure that searches are conducted in accordance with the law.

Safety risks: Bag checking can create safety risks if the procedures are not conducted safely or if the staff is not adequately trained. It is important to prioritize safety and minimize the risk of injury or harm to travellers and staff.

- **Walk Through Metal Detector:**

False positives: One of the biggest issues with walk-through metal detectors is that they can often generate false positives, meaning that they may detect metal objects that are not actually weapons or other prohibited items. This can lead to delays and inconvenience for individuals passing through the detector.

Sensitivity settings: Walk-through metal detectors can be adjusted to different levels of sensitivity, which can impact their ability to detect different types of metal objects. Setting the sensitivity too high can result in more false positives, while setting it too low can lead to missed detections.

Calibration: Walk-through metal detectors require regular calibration to ensure that they are functioning properly and accurately detecting metal objects. Failure to properly calibrate the detectors can result in missed detections or false positives.

Privacy concerns: Some individuals may be uncomfortable with the idea of passing through a walk-through metal detector, particularly if they feel that their privacy is being violated. This can be particularly true if the detectors are being used in non-security-related settings, such as at concerts or other public events.

Health concerns: Some individuals may be concerned about the potential health risks associated with passing through a walk-through metal detector, particularly if they are pregnant or have medical implants that could be affected by the detector's electromagnetic fields. However, the amount of radiation emitted by these detectors is generally considered to be minimal and safe for most individuals.

Accessibility: Some individuals with mobility or other disabilities may have difficulty passing through a walk-through metal detector, particularly if they are unable to stand or walk through the detector on their own. In such cases, alternative screening methods may be necessary to accommodate these individuals.

2.1.5 Technologies used for Existing System

- **Luggage Checking:**

X-ray machines: X-ray machines are commonly used to scan bags and detect any potential threats. These machines use high-energy radiation to create images of the contents of the bags.

Explosive Trace Detection (ETD): ETD machines are used to detect traces of explosive materials on bags. These machines can analyze swabs taken from bags or other items and determine if there are any explosive particles present.

Computed Tomography (CT) scanners: CT scanners use advanced imaging technology to create 3D images of bags and their contents. These scanners can detect even small amounts of potential threats, including weapons and explosive materials.

Automated Target Recognition (ATR) software: ATR software uses artificial intelligence and machine learning algorithms to analyze X-ray images of bags and automatically identify potential threats. This technology can help to reduce human error and improve the efficiency of the bag checking process.

- **Walk Through Metal Detector:**

Induction Balance: This technology uses a coil of wire to create a magnetic field. When a metal object passes through the field, it disrupts the magnetic field, which triggers an alarm.

Pulse Induction: This technology sends a pulse of electricity through a coil of wire, which creates a magnetic field. When a metal object passes through the field, it reflects the pulse, which is detected by the detector.

Very Low Frequency (VLF): This technology uses two coils of wire to create a magnetic field. One coil is a transmitter, and the other is a receiver. When a metal object passes through the field, it causes a change in the magnetic field, which is detected by the receiver.

Microwave: This technology uses electromagnetic waves to detect metal objects. When a metal object passes through the microwave field, it reflects the waves, which are detected by the detector.

.2.2 SYSTEM REQUIREMENTS:

HARDWARE REQUIREMENTS:

- System: i3 Processor 2.4 GHz.

- Hard Disk : 500GB.
- RAM : 8GB.

SOFTWARE REQUIREMENTS:

- **Operating System:** Windows
- **Coding Language**: Python 3.7

2.3 SYSTEM STUDY

A feasibility study for a armament discovery system would involve assessing the specialized and profitable viability of developing and enforcing such a system.

1. Specialized feasibility: The first step in a feasibility study would be to determine whether it's technically doable to develop an armament discovery system that can directly identify and detect arms and other munitions. This would bear assessing the available technology for detecting munitions, similar as essence sensors, X-ray scanners, and advanced imaging ways.

2. Cost: Developing a armament discovery system can be precious, so it's important to determine whether the design is economically feasible. This would involve conducting a detailed cost analysis of the development, manufacturing, installation, and conservation of the system, as well as any fresh costs similar as staff training and functional charges.

3. Legal and ethical considerations: Weapon discovery systems raise a number of legal and ethical issues, similar as sequestration enterprises and the eventuality for false cons. It would be important to estimate the legal and ethical counteraccusations of enforcing such a system, and to insure that it complies with all applicable laws and regulations.

4. Performance and delicacy: The effectiveness of a armament discovery system would depend on its performance and delicacy in detecting and locating arms and other munitions. It would be important to test the system under colorful conditions to determine its trustability and effectiveness.

5. Functional considerations: Eventually, it would be important to consider the practicalities of operating a armament discovery system, similar as staffing conditions, training requirements, and conservation schedules. Overall, a feasibility study for an armament discovery system would involve assessing a range of factors, from specialized and profitable feasibility to legal and ethical considerations. By conducting a comprehensive analysis of these factors, it would be possible to determine whether such a system is feasible and, if so, what way would be needed to develop and apply it effectively.

WHAT IS PYTHON?

Python is known for its simple, quick to use, and readability, making it a great language for newcomers to learn programming. It has a large standard library and supports multiple programming paradigms, including object- acquainted, imperative, and functional programming.

Advantages of Python:

Python has several advantages, some of which are

1. Easy to Learn Python is designed to be easy to read and learn, which makes it a great choice for newcomers. Its syntax is simple and easy to understand.

2. Versatility Python is a protean language that can be used for a wide range of operations, from web development to data analysis, scientific computing, machine literacy, and artificial intelligence.

3. Large Standard Library Python has a large standard library that provides a wide range of functions and modules for tasks similar as train running, networking, and data manipulation. This means that inventors don't have to write law from scrape for common tasks, which saves time and trouble.

4. Interpreted Language Python is an interpreted language, which means that inventors can write and run law without the need for compendium. This makes it easy to write and test law snappily.

5. Open- source Python is open- source, which means that the source law is freely available and can be modified and distributed by anyone. This has led to a large and active community of inventors who contribute to the language and produce libraries and fabrics that extend its functionality.

6. Cross-Platform comity Python is compatible with multiple platforms, including Windows, Linux, and macOS. This means that inventors can write law on one platform and run it on another without having to make any major changes.

Overall, Python's simplicity, versatility, large standard library, and active community make it an excellent choice for a wide range of operations.

Disadvantages of Python:

While Python has many advantages, there are also some disadvantages, which include:

1. Slow Performance: Compared to other programming languages such as C++, Java, and Go, Python's interpreted nature can result in slower execution speeds.

2. Global Interpreter Lock: Python has a Global Interpreter Lock (GIL), which prevents multiple threads from executing Python code simultaneously. This can limit the scalability of Python applications and make it harder to take advantage of multiple cores or processors

3. Limited Memory Management Control: Python's automatic memory management can make it easier for developers to write code, but it can also lead to inefficiencies and limitations in certain use cases, such as developing large-scale applications that require precise memory management.

4. Design Restrictions: Python's design philosophy can sometimes be a disadvantage. For example, the emphasis on code readability and simplicity can result in limitations that make it more difficult to write complex code.

Overall, Python's disadvantages are relatively minor and are often outweighed by its advantages. However, it is important to consider these limitations when deciding whether to use Python for a particular application.

Advantages of Python Over Other Languages:

1. lower Coding: Python also has a miraculous standard library support, so you don't have to search for any third- party libraries to get your job done

2. Affordable: Python is free thus individualities, small companies or big associations can work the free available coffers to make operations. Python is popular and extensively used so it gives you better community support. The 2019 Github periodic check showed us that Python has overhauled Java in the most popular programming language order.

3. Python is for Everyone: Python law can run on any machine whether it's Linux, Mac or Windows. and also make games and important visualizations. It's an each-rounder programming language.

Install Python Step- by- Step in Windows and Mac

Installing Python on your Windows or Mac computer is a simple and straightforward process. also's a step- by- step companion to help you get started

1. First, navigate to the sanctioned Python website and download the bottommost interpretation of Python. insure that you download the applicable interpretation for your operating system

2. Windows, handpick the" Add Python to PATH" option during the installation process.

3. This will insure that Python is added to your computer's system PATH and can be entered from any directory You should see the Python interpretation number displayed on the screen.

4. still, it's recommended that you install the Anaconda distribution of Python, which comes with popular packages analogous as NumPy and Pandas pre-installed, If you plan on using Python for

data analysis or scientific computing.

5. To install Anaconda, download the applicable interpretation for your operating system from the sanctioned website and run the installer. Follow the prompts and handpick the options that you bear during the installation process.

6. Once installation is complete, open a command prompt or terminal window and type" jupyter tablet" to launch the Jupyter Tablet interface, which is a popular tool for data analysis and visualization.

7. Congratulations! You have now successfully installed Python and are ready to start rendering. Flash back, Python has a vast online community with cornucopia of resources available to help you learn and grow as a programmer

How to Install Python on Windows and Mac

There have been several updates in the Python interpretation over the times. The question is how to install Python? It might be confusing for the beginner who is willing to start learning Python but this tutorial will break your query. The rearmost or the newest interpretation of Python is interpretation3.7.4 or in other words, it's Python 3. Note The python interpretation3.7.4 can't be used on Windows XP or earlier bias. So the way below are to install python interpretation3.7.4 on Windows 7 device or to install Python 3. Download the Python Cheat distance then. The way on how to install Python on Windows, corridors to help understand better. Download the Correct interpretation into the system

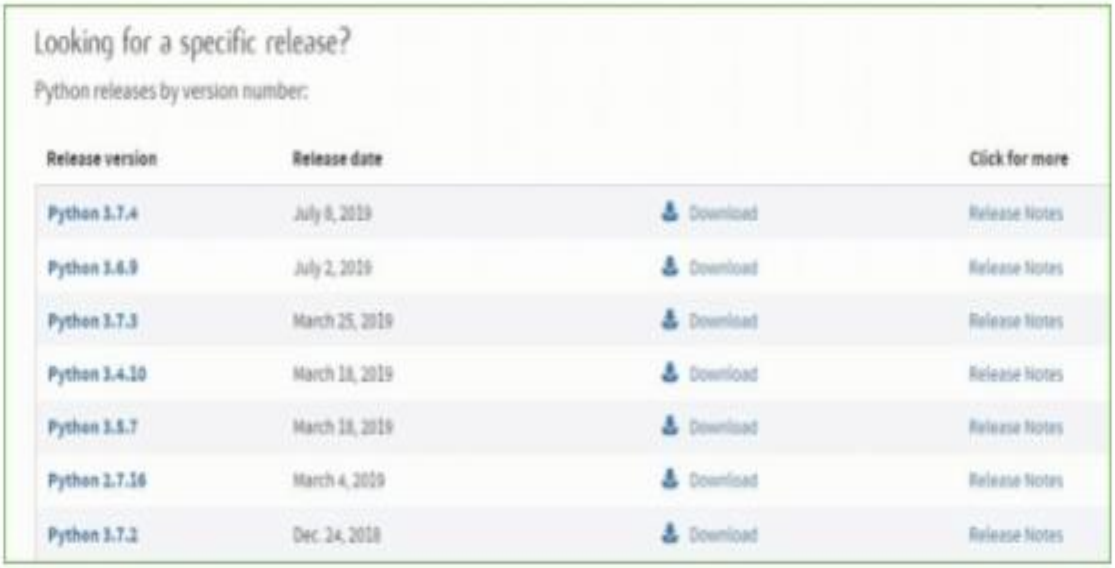
Step 1 First we need to open Google Chrome or any other web cyber cybersurfer and install python. OR Click on the following link https//www.python.org Now, check for the streamlined and the accurate interpretation for your operating system.

Now, check for the updated and the accurate version for your operating system.

Step 2: Click on the Download Tab.

Step 3: You can either handpick the Download Python for windows3.7.4 button in Yellow Color or you can scroll further down and click on download with separate to their

interpretation. Then, we're downloading the most recent python interpretation for windows3.7.4



Looking for a specific release?

Python releases by version number:

Release version	Release date		Click for more
Python 3.7.4	July 8, 2019	Download	Release Notes
Python 3.6.9	July 2, 2019	Download	Release Notes
Python 3.7.3	March 25, 2019	Download	Release Notes
Python 3.4.10	March 18, 2019	Download	Release Notes
Python 3.5.7	March 18, 2019	Download	Release Notes
Python 2.7.16	March 4, 2019	Download	Release Notes
Python 3.7.2	Dec. 24, 2018	Download	Release Notes

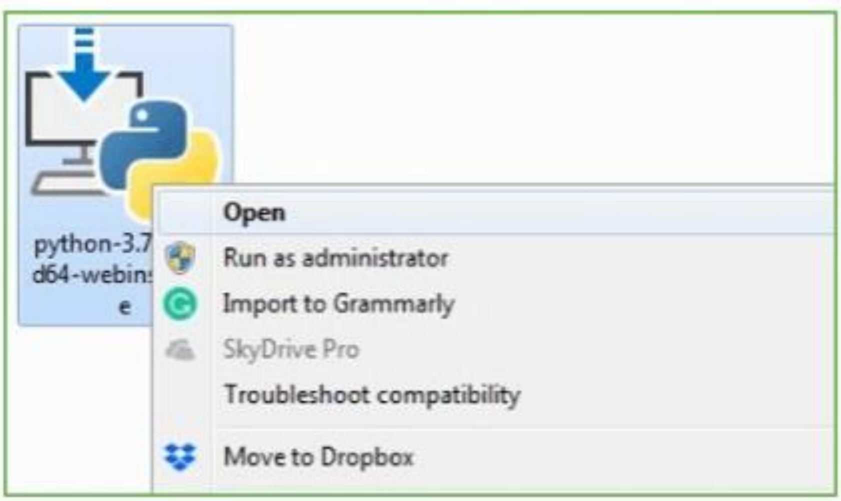
Files

Version	Operating System	Description	MD5 Sum	File Size	GPG
Gzipped source tarball	Source release		68111671e5b2db4aef7b9ab01bf0f9be	23017663	SIG
XZ compressed source tarball	Source release		d33e4aae66097051c2eca45ee3604803	17131432	SIG
macOS 64-bit/32-bit installer	Mac OS X	for Mac OS X 10.6 and later	6428b4fa7583daff1a442cba8cee08e6	34898416	SIG
macOS 64-bit installer	Mac OS X	for OS X 10.9 and later	5dd605c38217a45773bf5e4a936b241f	28082845	SIG
Windows help file	Windows		d63999573a2c06b2ac56cade6b4f7cd2	8131761	SIG
Windows x86-64 embeddable zip file	Windows	for AMD64/EM64T/x64	9b00c8cf6d9ec0bfabe83184a40729a2	7504391	SIG
Windows x86-64 executable installer	Windows	for AMD64/EM64T/x64	a702b4b0ad76debdb3043a583e563400	26680368	SIG
Windows x86-64 web-based installer	Windows	for AMD64/EM64T/x64	28cb1c608bbd73ae8e53a3bd351b4bd2	1362904	SIG
Windows x86 embeddable zip file	Windows		9fab3b81f8841879fda94133574139d8	6741626	SIG
Windows x86 executable installer	Windows		33cc602942a54446a3d6451476394789	25663848	SIG
Windows x86 web-based installer	Windows		1b670cfa5d317df82c30983ea371d87c	1324608	SIG

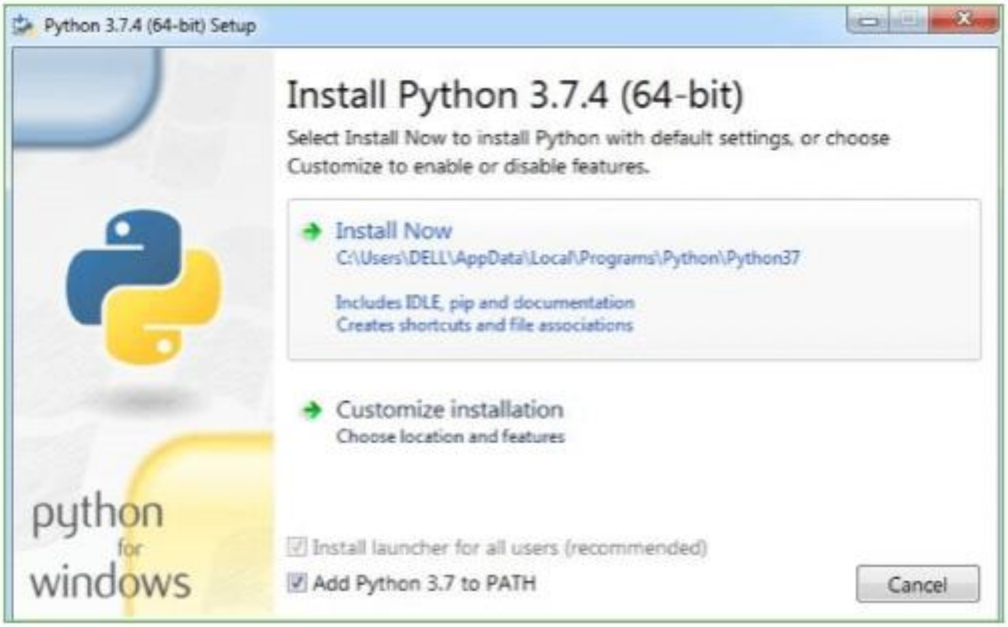
Note: To know the revision or updates that are made in the interpretation you can click on the Release Note Option.

Installation of Python

Step 1: Go to Download tab and open the downloaded click on python interpretation to carry out the installation process.



Step 2: Before you click on Install Now, Make sure to put a click on Add Python 3.7 to PATH.



Step 3: Click on Install Now and After the installation is successful. Click on Close.

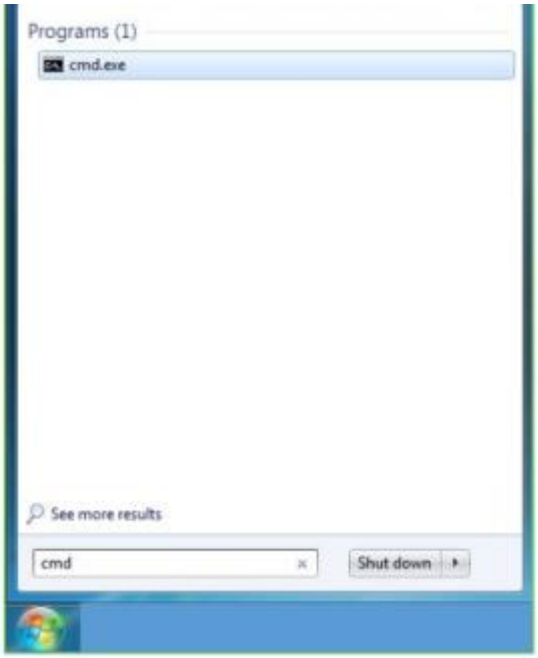
With these above three steps on python installation, you have successfully and rightly installed Python. Now is the time to check the installation is working duly or not.

Note: The installation process might take a few nanoseconds of time.

Verify the Python Installation

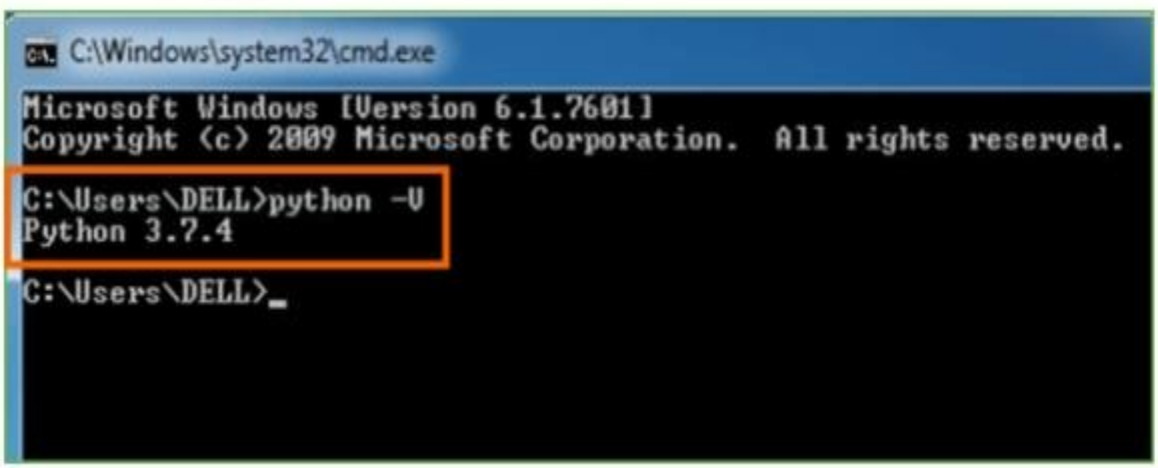
Step 1: Click on Start

Step 2: In the Windows Run Command, type “cmd”.



Step 3: Open the Command advisement.

Step 4: Let us check whether the python is rightly installed. Type **python –V** and press Enter.



Step 5: You can now see on windows like python 3.7.4

Note: If you have any of the before performances of Python formerly installed. You must first uninstall the earlier interpretation and also install the new one.

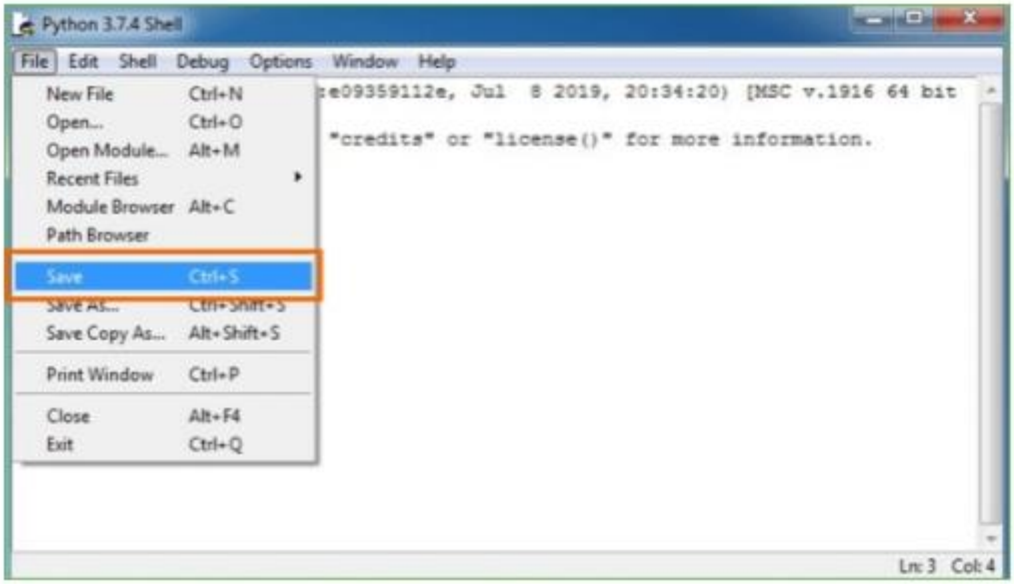
Check how the Python IDLE works

Step 1: Click on Start

Step 2: In the Windows Run command, type “python idle”.

Step 3: Click on IDLE (Python 3.7 64-bit) and launch the program

Step 4: To go ahead with working in IDLE you must first save the file. **Click on File > Click on Save**

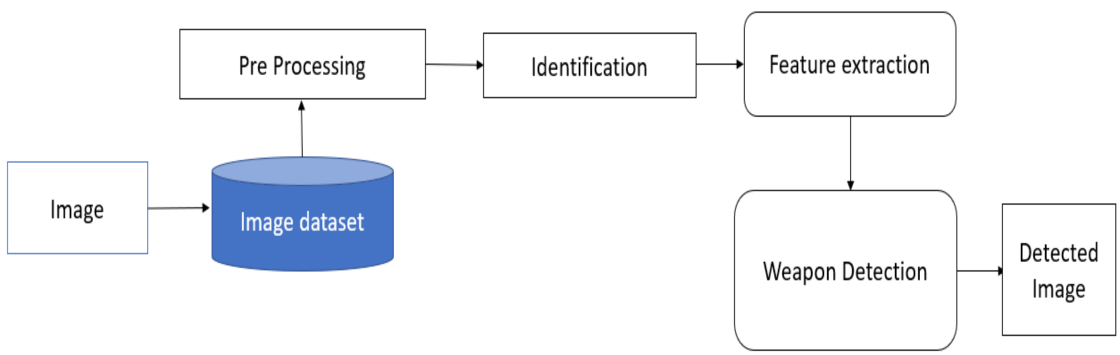


Step 5: Name the file and save as type should be Python files. Click on SAVE. Here I have named the files as Hey World.

Step 6: Now for e.g., **enter print**

3. SYSTEM DESIGN

3.1 SYSTEM ARCHITECTURE:



3.2 OBJECTIVE OF PROPOSED MODEL

The objective of our project is to develop a system that can automatically identify the presence of weapons in real-time using video or image analysis. The system should be able to detect various types of weapons, including firearms, knives, and other dangerous items.

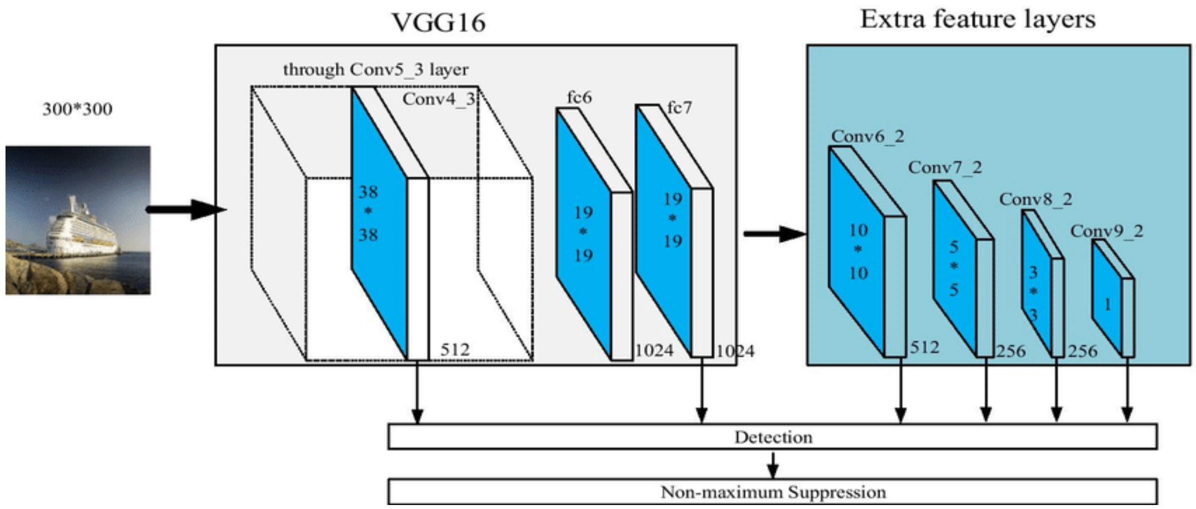
The use of artificial intelligence and deep learning algorithms enables the system to learn and improve over time, making it more accurate and efficient in identifying weapons. The system should also be able to work in different environments and lighting conditions, making it suitable for use in various security applications, such as in airports, schools, and other public areas.

The ultimate goal of this objective is to enhance public safety by preventing the entry of weapons into sensitive areas, and alerting security personnel in real-time when a weapon is detected, so that appropriate measures can be taken to prevent potential harm.

3.3 ALGORITHMS USED FOR PROPOSED MODEL

Single Shot Detection:

Single Shot Detection (SSD) is a notorious object discovery algorithm used in computer vision. It's a real- time approach for detecting objects in images, meaning that it can dissect an image and descry objects within it with high speed and delicacy. SSD workshop by dividing an input image into a grid of cells, where each cell predicts a fixed set of objects with different aspect rates and scales. For each cell, SSD predicts a set of bounding boxes and corresponding class chances. These bounding boxes represent the position of objects within the cell. In SSD, the network is divided into two corridor a backbone network and a vaticination network. The backbone network is a deep convolutional neural network that excerpts feature charts from the input image. The vaticination network also takes these point maps as input and predicts the bounding boxes and class chances for each cell. One of the advantages of SSD is that it can handle objects of different sizes and aspect rates, as it predicts bounding boxes with different scales and aspect rates for each cell. It also uses a single neural network to perform both object localization and bracket, making it effective and easy to apply. Overall, SSD is a important algorithm for object discovery that can descry objects in real- time with high delicacy and effectiveness. It has been used in a wide range of operations, including independent driving, robotics, and surveillance systems



3.3.1 Architecture of Single Shot Algorithm

Advantages:

SSD (Single Shot Detector) is a famous object detection algorithm that has several advantages over other object detection algorithms:

1. Speed: SSD is a fast algorithm that can detect objects in real life scenario. It can achieve high detection speed without sacrificing accuracy, making it suitable for real-world applications such as video surveillance and self-driving cars.

2. Accuracy: SSD is highly accurate in detecting objects of various sizes and shapes. It achieves this by using a multi-scale feature map that detects objects at different scales, ensuring that even small objects are detected accurately.

3. Efficiency: SSD is an efficient algorithm that requires fewer computations than other object detection algorithms.

4. Simplicity: The SSD algorithm is simple and quick to understand, making it quickly to implement and modify compared to other object detection algorithms. This makes it a popular choice for developers or finders who want to experiment with object detection.

5. Flexibility: SSD is a versatile algorithm that can detect multiple object classes simultaneously. This makes it suitable for a wide range of applications, including face detection, vehicle detection, and pedestrian detection.

SSD offers a balance of speed, accuracy, efficiency, simplicity, and flexibility that makes it a popular choice for object detection applications.

Disadvantages:

While SSD (Single Shot Detector) has several advantages over other object detection algorithms, there are also some disadvantages to consider:

1. Low localization accuracy: Compared to other object detection algorithms such as Faster R-CNN, SSD has lower localization accuracy. This means that it may have difficulty accurately identifying the exact location of an object in an image, which can be problematic in some applications.

2. Limited ability to detect small objects: While SSD is capable of detecting objects of various sizes, it may struggle to detect very small objects. This is because small objects may not have enough information in the image to be accurately detected.

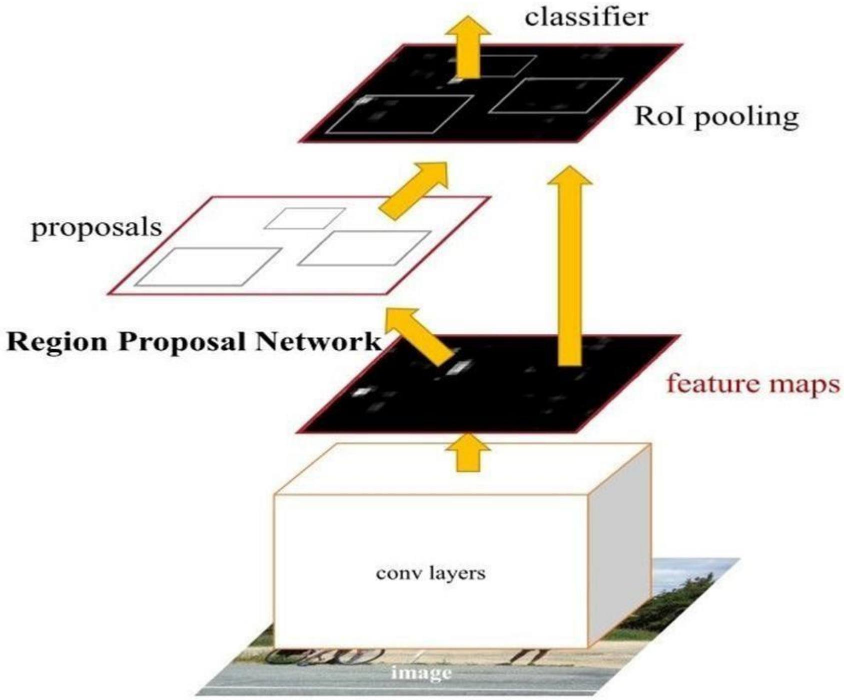
3. Difficulty in detecting objects with complex shapes: SSD may struggle to detect objects with complex shapes or irregular boundaries. This is because it uses a simple bounding box model, which may not be able to accurately capture the shape of certain objects.

4. Limited ability to handle occlusions: Like other object detection algorithms, SSD may have difficulty detecting objects that are partially occluded or obscured by other objects in the image.

5. Requires large amounts of training data: To achieve high accuracy, SSD requires a large amount of training data. This can be difficult to obtain in some applications, particularly those that involve rare or unusual objects.

while SSD has many advantages, it may struggle in certain situations, such as when dealing with small objects or objects with complex shapes. Additionally, it requires a large amount of training data to achieve high accuracy.

Faster RCNN: Faster R-CNN (Region-based Convolutional Neural Network) is a famous object detection algorithm. It is an improvement over the earlier R-CNN and Fast R-CNN algorithms and is known for its high accuracy and faster processing speed. The key idea behind Faster R-CNN is the use of a Region Proposal Network (RPN) that generates object proposals in an efficient manner. The RPN is a fully convolutional network that takes an image as input and outputs a set of object proposals, along with their objectness scores, which indicate the likelihood of an object being present in that region of the image. Once the object proposals are generated, they are passed on to a Region of Interest (RoI) pooling layer, which extracts features from each proposal and transforms them into a fixed-size feature map. These features are then fed into a classifier network, which predicts the object class and refines the object's bounding box coordinates. The Faster R-CNN algorithm is trained end-to-end using a combination of a classification loss and a regression loss. The classification loss measures the difference between the predicted class probabilities and the ground-truth class labels, while the regression loss measures the difference between the predicted bounding box coordinates and the ground-truth bounding box coordinates. Faster R-CNN has been shown to achieve state-of-the-art performance on a number of benchmark datasets, such as COCO (Common Objects in Context) and PASCAL VOC (Visual Object Classes). It is widely used in various applications, including self-driving cars, robotics, and surveillance.



3.3.2 Architecture of Faster RCNN Algorithm

Advantages:

1. High accuracy: Faster R-CNN achieves state-of-the-art performance on a range of object detection benchmarks, making it one of the most accurate object detection algorithms available.

2. Accurate object localization: Faster R-CNN uses a two-stage detection process that accurately localizes objects in an image. This makes it particularly effective for applications that require precise object detection, such as medical imaging.

3. Flexibility: Faster R-CNN is a flexible algorithm that can be adapted to a wide range of object detection tasks. It can detect objects of various sizes, shapes, and orientations, and can be customized to detect specific types of objects.

4. Robustness: Faster R-CNN is robust to occlusions, background clutter, and other challenging conditions that can make object detection difficult. This makes it suitable for a range of real-time applications, such as self-driving cars.

5. Ease of use: Faster R-CNN is relatively easy to implement and use, making it a choice for finders and developers who want to do experiments with object detection.

Disadvantages:

1. Slower detection speed: Faster R-CNN is a two-stage detection process that can be slower than other object detection algorithms such as SSD. This can make it unsuitable for real-time applications that require fast detection speeds.

2. High computational requirements: Faster R-CNN requires significant computing ability, including powerful GPUs and large amounts of memory, to achieve high accuracy. This can make it expensive to implement and deploy in some applications.

3. Complexity: Faster R-CNN is a complex algorithm that can be difficult to understand and implement, particularly for developers who are new to object detection. This can make it challenging to modify and customize for specific applications.

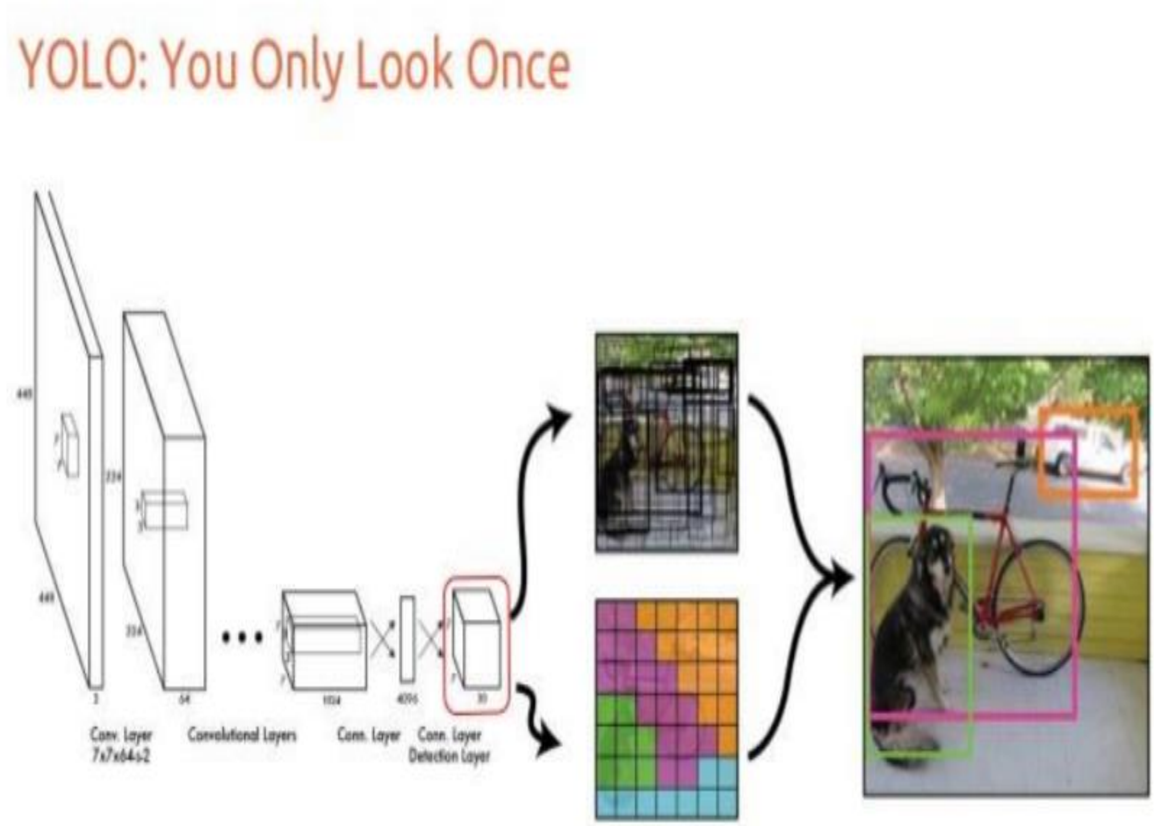
4. Limited ability to handle small objects: Faster R-CNN may struggle to accurately detect very small objects, as they may not have enough information in the image to be accurately detected.

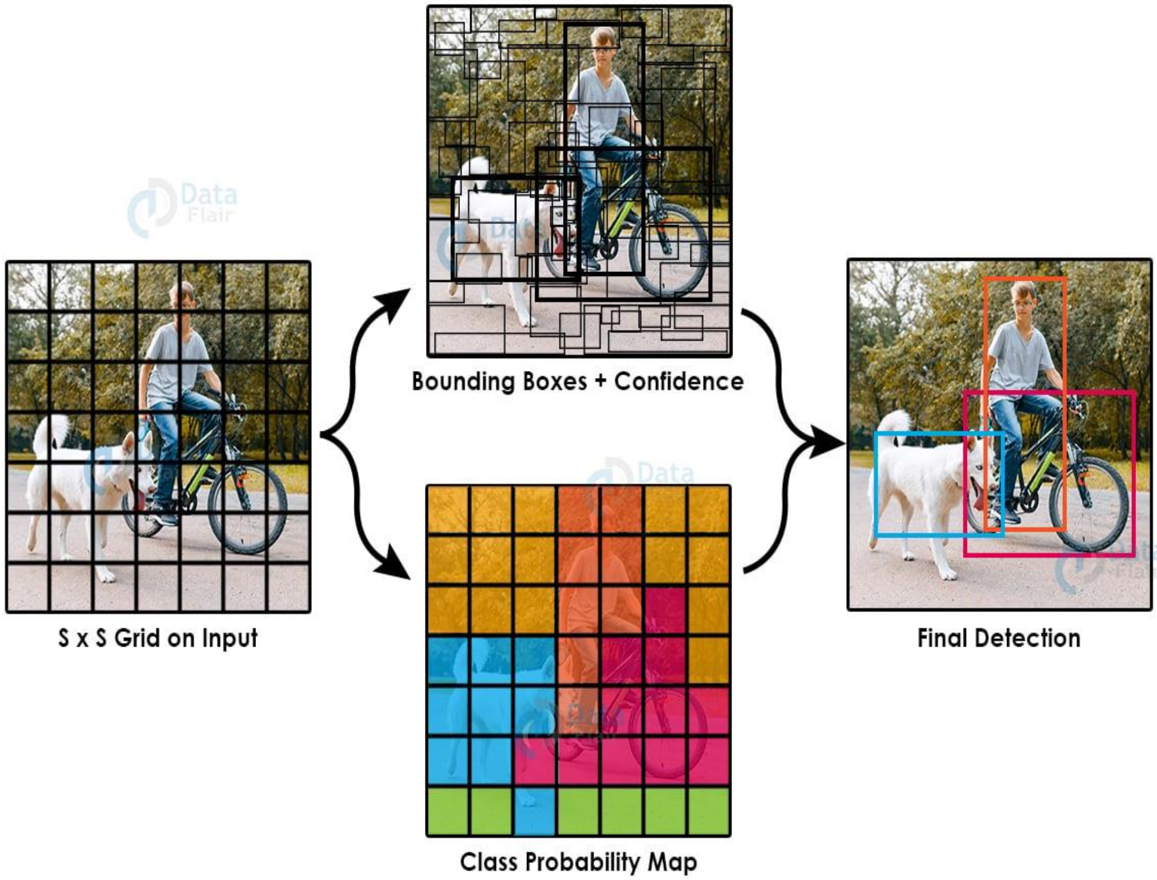
5. Difficulty in handling occlusions: Like other object detection algorithms, Faster R-CNN may have difficulty detecting objects that are partially occluded or obscured by other objects in the image.

while Faster R-CNN offers high accuracy, precise object localization, and flexibility, it may have limitations in terms of detection speed, computational requirements, complexity, and handling small objects or occlusions. These factors should be considered when choosing an object detection algorithm for a particular application.

YOLOv3: A real-time object detection system called YOLOv3 (You Only Look formerly, Version 3) can recognise certain things in videos, live streams, or photos. A deep convolutional neural network is used by the YOLO machine literacy method to learn characteristics that are used to describe objects. Joseph Redmon and Ali Farhadi devised the first three iterations of YOLO, with the third iteration being a more accurate approximation of the original machine learning method. The initial YOLO interpretation was made in 2016, and the three subsequent interpretations—of which this piece makes extensive use—were made in 2018—two times each. Advanced versions of YOLO and YOLOv2 are interpreted in YOLOv3. Keras or OpenCV's deep literacy libraries are used to enforce the YOLO principle.

We have used yolov3 for training and testing the model and COCO dataset.





Advantages:

YOLOv3 is a prominent object detection algorithm that can be used for training weapon detection systems. Here are some of the advantages of using YOLOv3 for weapon detection:

1. Speed: YOLOv3 is known for its rapidly detection speed, which makes it suitable for real-life applications such as weapon detection. This is due to its single-shot detection architecture, which processes the entire image in one pass.

2. High accuracy: YOLOv3 has achieved state-of-the-art accuracy on several object detection benchmarks, which makes it a reliable algorithm for weapon detection. It also has a relatively low false positive rate, which reduces the chances of false alarms.

3. Ability to detect small objects: YOLOv3 is capable of detecting small objects, which is important for weapon detection systems. It can also detect objects of various sizes and orientations, making it versatile for a range of weapon detection tasks.

4. Easy to implement and customize: YOLOv3 is relatively easy to implement and customize, which makes it accessible to researchers and developers who want to experiment with weapon detection. It also has a large community of developers who have contributed to its development and improvement.

5. Large training data set: YOLOv3 has been trained on a large and diverse dataset, which includes weapons and other objects. This makes it a suitable algorithm for weapon detection systems, as it has already learned to recognize weapons and distinguish them from other objects.

YOLOv3 is a suitable algorithm for training weapon detection systems, thanks to its speed, high accuracy, ability to detect small objects, ease of implementation and customization, and large training data set.

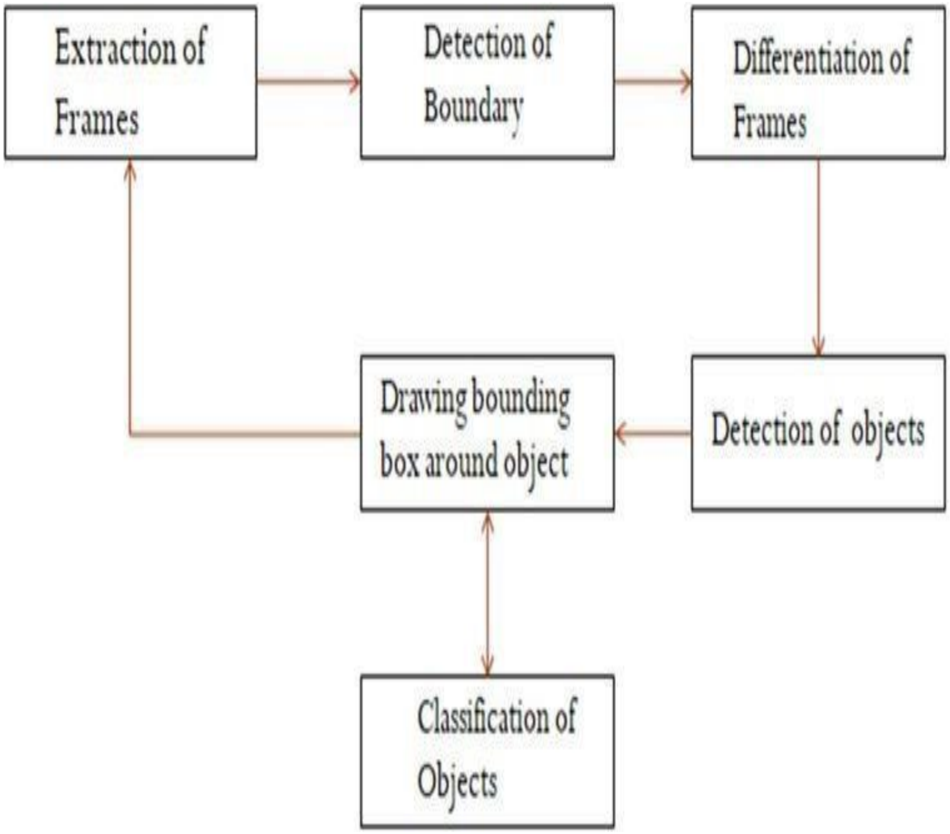
Disadvantages:

1. Lower localization accuracy: YOLOv3's single-shot detection architecture can result in lower localization accuracy compared to other object detection algorithms, such as Faster R-CNN. This means that the algorithm may not be as precise in identifying the exact location of objects in an image.

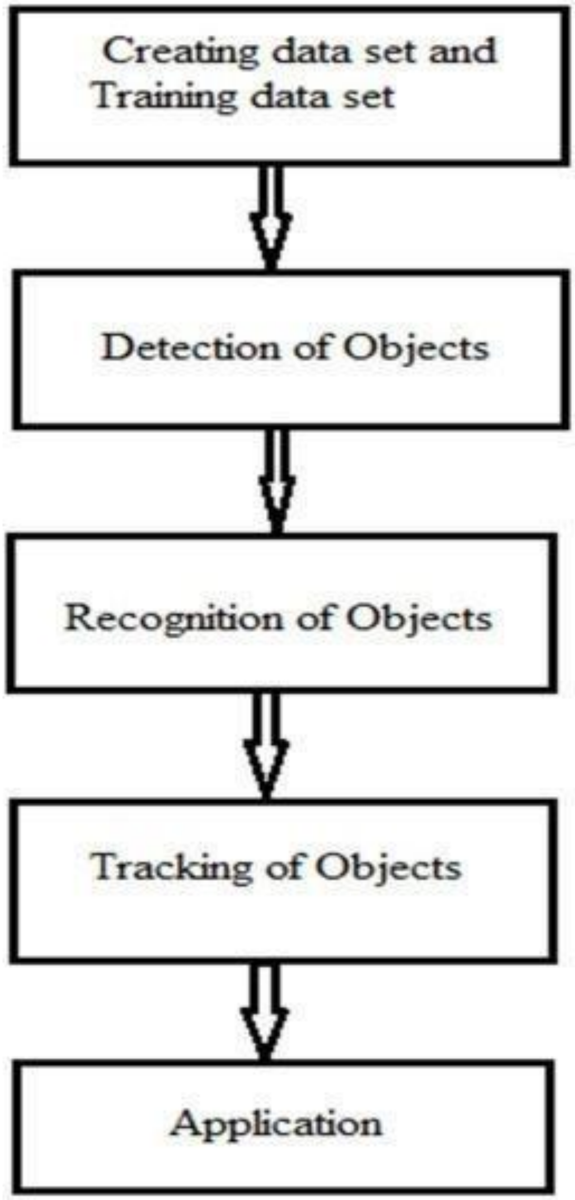
2. Sensitivity to object size: YOLOv3 may have difficulty detecting small objects in images, as they may not have enough information to be accurately detected. Conversely, YOLOv3 may also struggle to detect very large objects.

3. Training complexity: While YOLOv3 is relatively easy to implement and customize, training the algorithm can be complex and computationally intensive. This requires access to high-performance computing resources, such as GPUs, and can be time-consuming.

3.4 DESIGNING

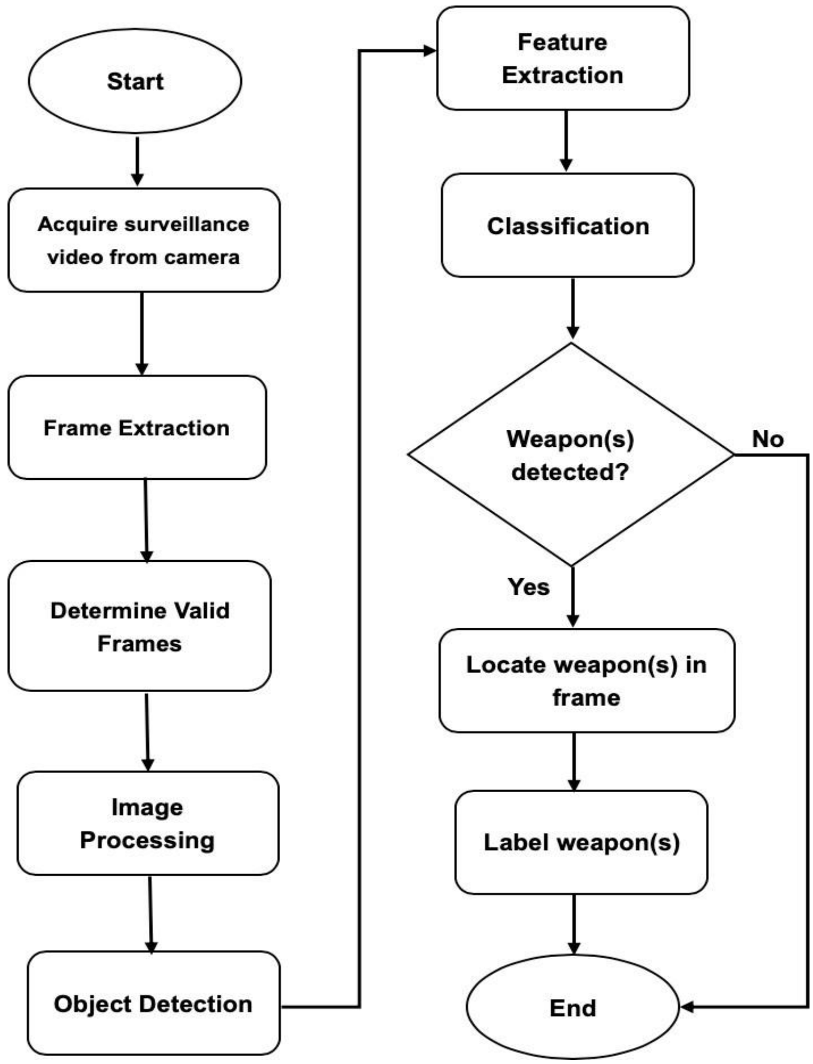


3.4.1 Describes the Methodology



3.4.2 Detection and Tracking

BLOCK DIAGRAM



3.4.3 Block Diagram

3.5 STEPWISE IMPLEMENTATION:

The assumed system involves the following stages:

Data Collection: The first step involves collecting data in the form of images and videos from various sources such as cameras, CCTVs, drones, and other sensors that capture the area of interest.

Pre-processing: In the case of weapon detection, pre-processing typically involves several steps, including image resizing, image normalization, and data augmentation. For example, in a video surveillance system, the images may need to be resized to a standard size to ensure consistency across the data set. Image normalization techniques such as contrast stretching and histogram equalization can be used to enhance the quality of the images.

Data augmentation: techniques such as flipping, rotation, and translation can also be applied to increase the size of the training dataset and reduce overfitting. In addition, pre-processing may involve the removal of noise and irrelevant information from the images to improve the accuracy of the weapon detection system.

Identification: It is essential for ensuring that the system accurately distinguishes between weapons and harmless objects. False positives can result in unnecessary alarm and response, while false negatives can lead to serious security breaches. Therefore, accurate identification is critical for the effectiveness of weapon detection. Object detection algorithms such as YOLO (You Only Look Once), SSD (Single Shot Detector), or Faster R-CNN (Region-based Convolutional Neural Network) are used to detect the presence of weapons in the pre-processed images or videos. These algorithms analyze the images to identify specific objects, such as guns or knives, by learning from large sets of labelled data.

Feature Extraction: This process is essential in weapon detection because it allows the system to identify the relevant information in the input data that is useful for detecting weapons. By extracting features that are indicative of weapons, the system can accurately classify objects in the input data as either weapons or non-weapons. Deep learning algorithms, such as convolutional neural networks (CNNs), are commonly used for feature extraction in weapon detection. These algorithms can automatically learn relevant features from large amounts of data, making them highly effective at detecting weapons in complex and dynamic environments. Once a weapon is detected in an image or video, the next step is to classify the weapon based on its type, such as a handgun or a knife. This step can be accomplished using machine learning classifiers such as Support Vector Machines (SVMs) or Convolutional Neural Networks (CNNs).

4. IMPLEMENTATION

```
import cv2
import numpy as np
 # Load Yolo
net = cv2.dnn.readNet("yolov3_training_2000.weights", "yolov3_testing.cfg")
classes = ["Weapon"]

# with open("coco.names", "r") as f:
# classes = [line.strip() for line in f.readlines()]
 layer_names = net.getLayerNames()
output layers = [layer_names[i[0] - 1] for i in net.getUnconnectedOutLayers()]
colors = np.random.uniform(0, 255, size=(len(classes), 3))

 # Loading image
# img = cv2.imread("room_ser.jpg")
# img = cv2.resize(img, None, fx=0.4, fy=0.4)

# Enter file name for example "ak47.mp4" or press "Enter" to start webcam
def value():
    val = input("Enter file name or press enter to start webcam : \n")
    if val == "":
        val = 0
```

```
    return val



# for video capture
cap = cv2.VideoCapture(value())

# val = cv2.VideoCapture()
while True:
    _, img = cap.read()
    height, width, channels = img.shape
    # width = 512
    # height = 512

    # Detecting objects
    blob = cv2.dnn.blobFromImage(img, 0.00392, (416, 416), (0, 0, 0), True, crop=False)
    net.setInput(blob)
    outs = net.forward(output_layers)

    # Showing information on the screen
    class_ids = []
    confidences = []
    boxes = []
    for out in outs:
        for detection in out:
```

```
        scores = detection[5:]
        class_id = np.argmax(scores)
        confidence = scores[class_id]
        if confidence > 0.5:
            # Object detected
            center_x = int(detection[0] * width)
            center_y = int(detection[1] * height)
            w = int(detection[2] * width)
            h = int(detection[3] * height)

            # Rectangle coordinates
            x = int(center_x - w / 2)
            y = int(center_y - h / 2)
            boxes.append([x, y, w, h])
            confidences.append(float(confidence))
            class_ids.append(class_id)

indexes = cv2.dnn.NMSBoxes(boxes, confidences, 0.5, 0.4)
print(indexes)
if indexes == 0: print("weapon detected in frame")
font = cv2.FONT_HERSHEY_PLAIN
for i in range(len(boxes)):
    if i in indexes:
        x, y, w, h = boxes[i]
```

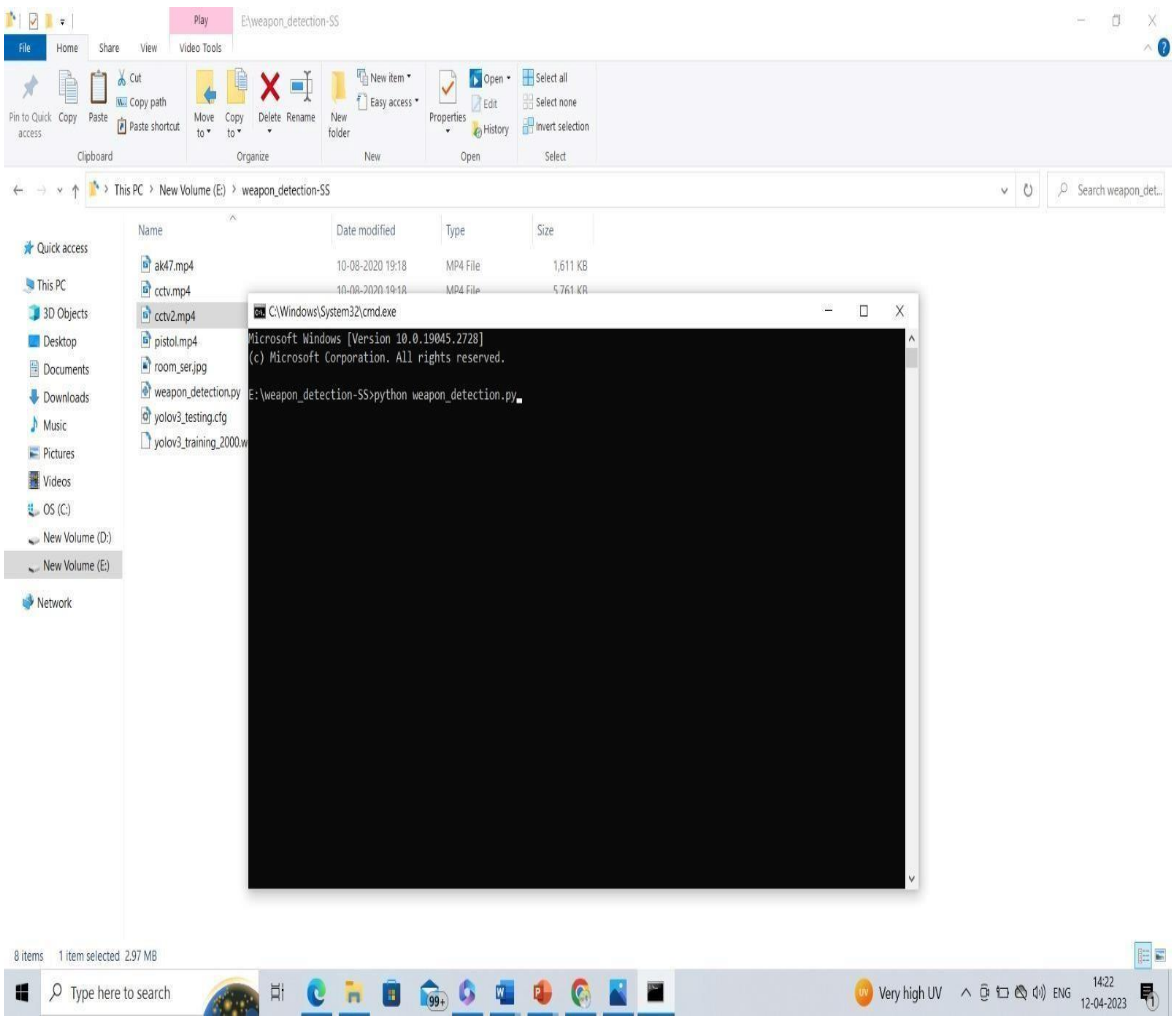
```
            label = str(classes[class_ids[i]])
            color = colors[class_ids[i]]
            cv2.rectangle(img, (x, y), (x + w, y + h), color, 2)
            cv2.putText(img, label, (x, y + 30), font, 3, color, 3)

    # frame = cv2.resize(img, (width, height), interpolation=cv2.INTER_AREA)
    cv2.imshow("Image", img)
    key = cv2.waitKey(1)
    if key == 27:
        break
cap.release()
cv2.destroyAllWindows()
```

5. RESULTS AND DISCUSSION

5.1 RESULTS:

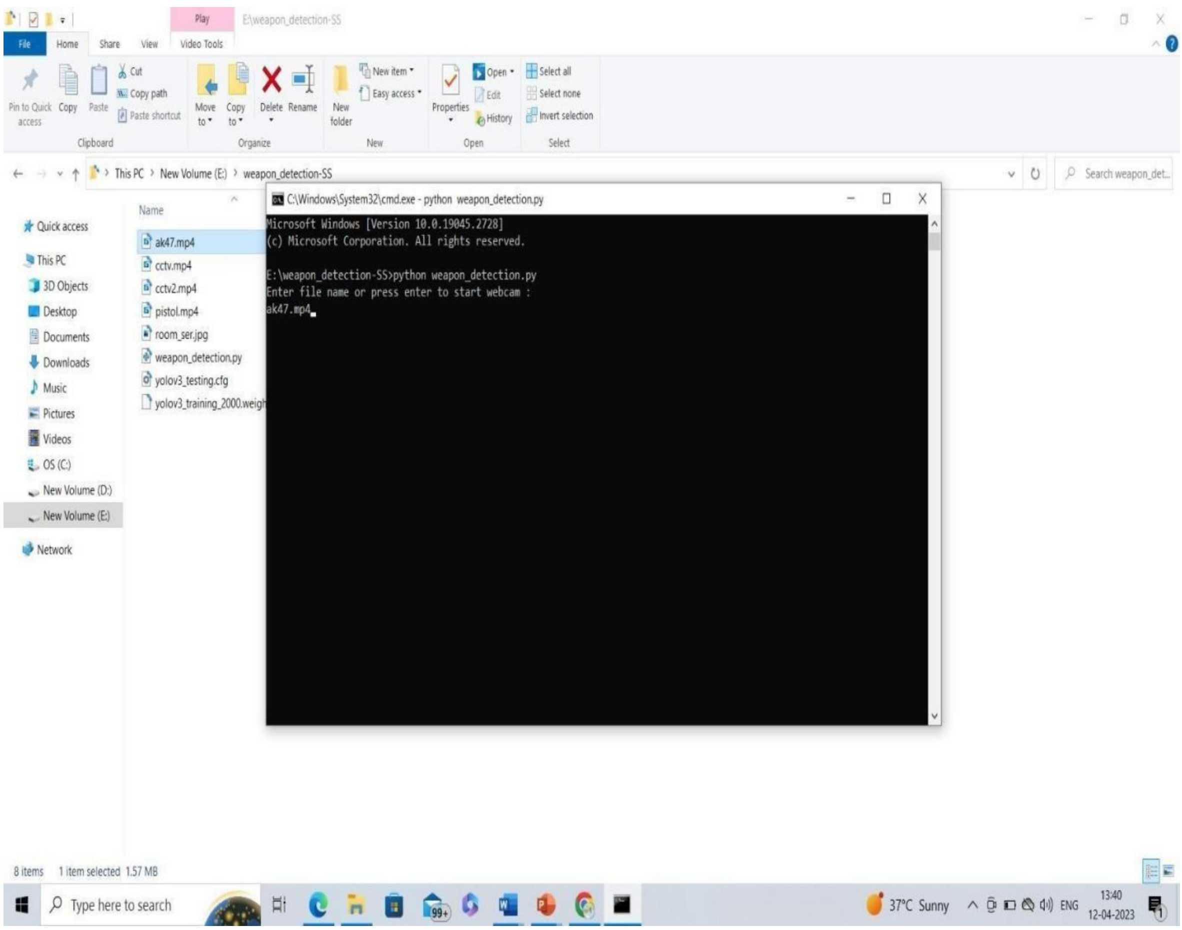
STEP 1: EXECUTING IN COMMAND PROMPT BY ENTERING python weapon_detection.py



5.1.1 Step 1 during execution

STEP 2:

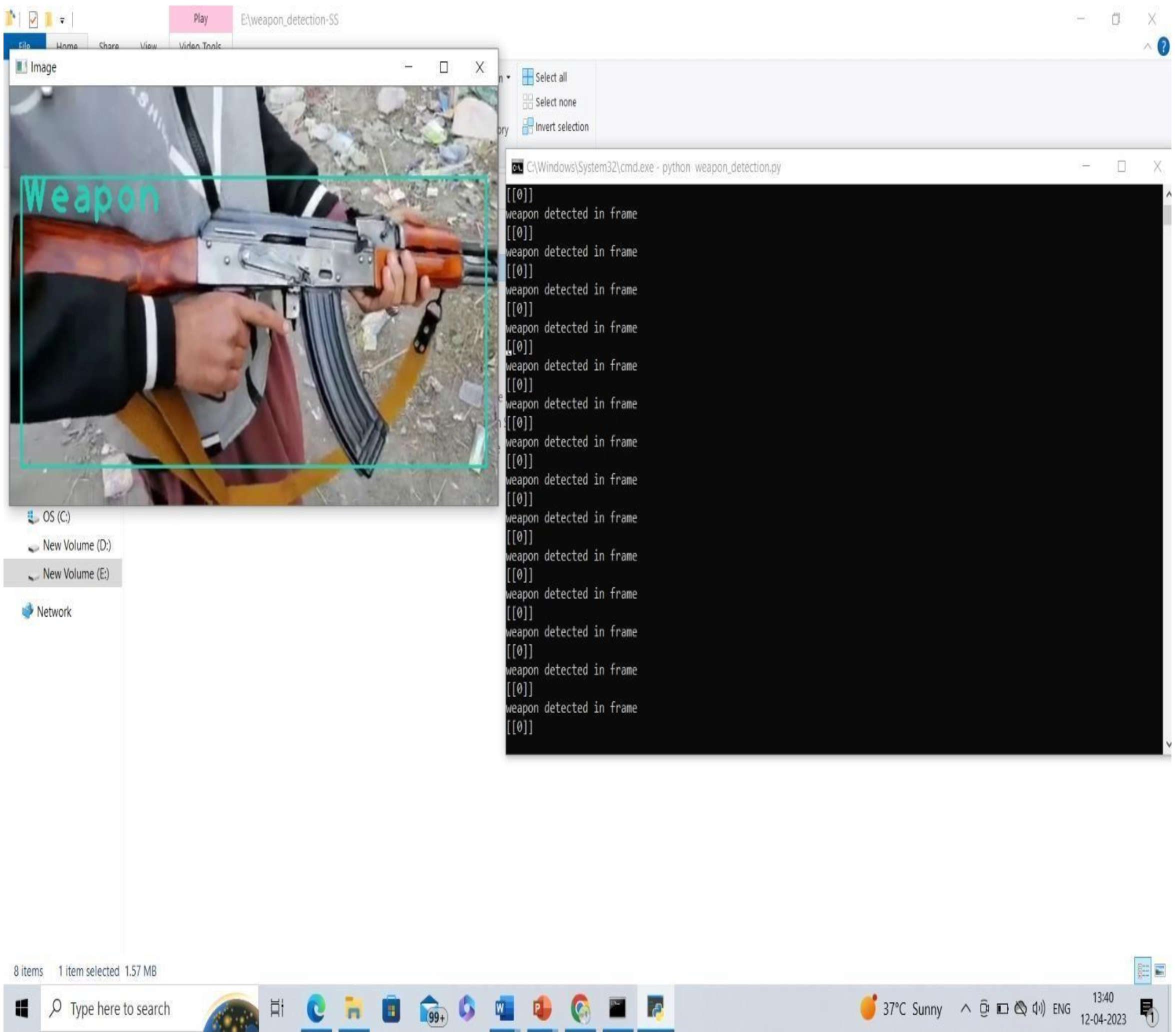
AFTER ENTERING COMMAND ENTER THE INPUT FILE NAME or BY PRESSING **ENTER** WE CAN START LIVE WEBCAM



5.1.2 Step 2 in execution

STEP 3:

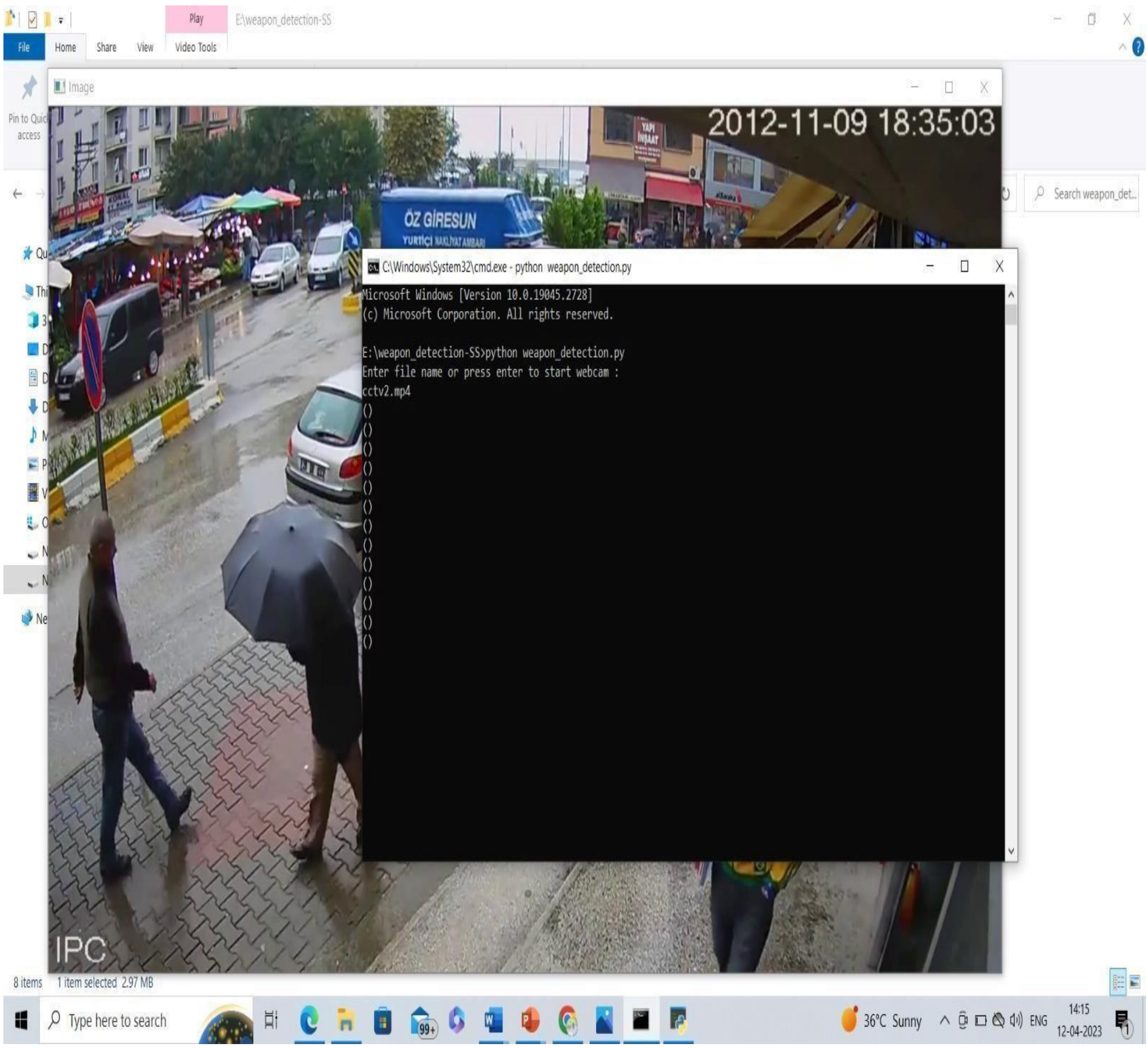
IF WEAPON DETECTS IT SHOWS **WEAPON DETECTED IN FRAME [0]**



5.1.3 Step 3 in execution

STEP 4:

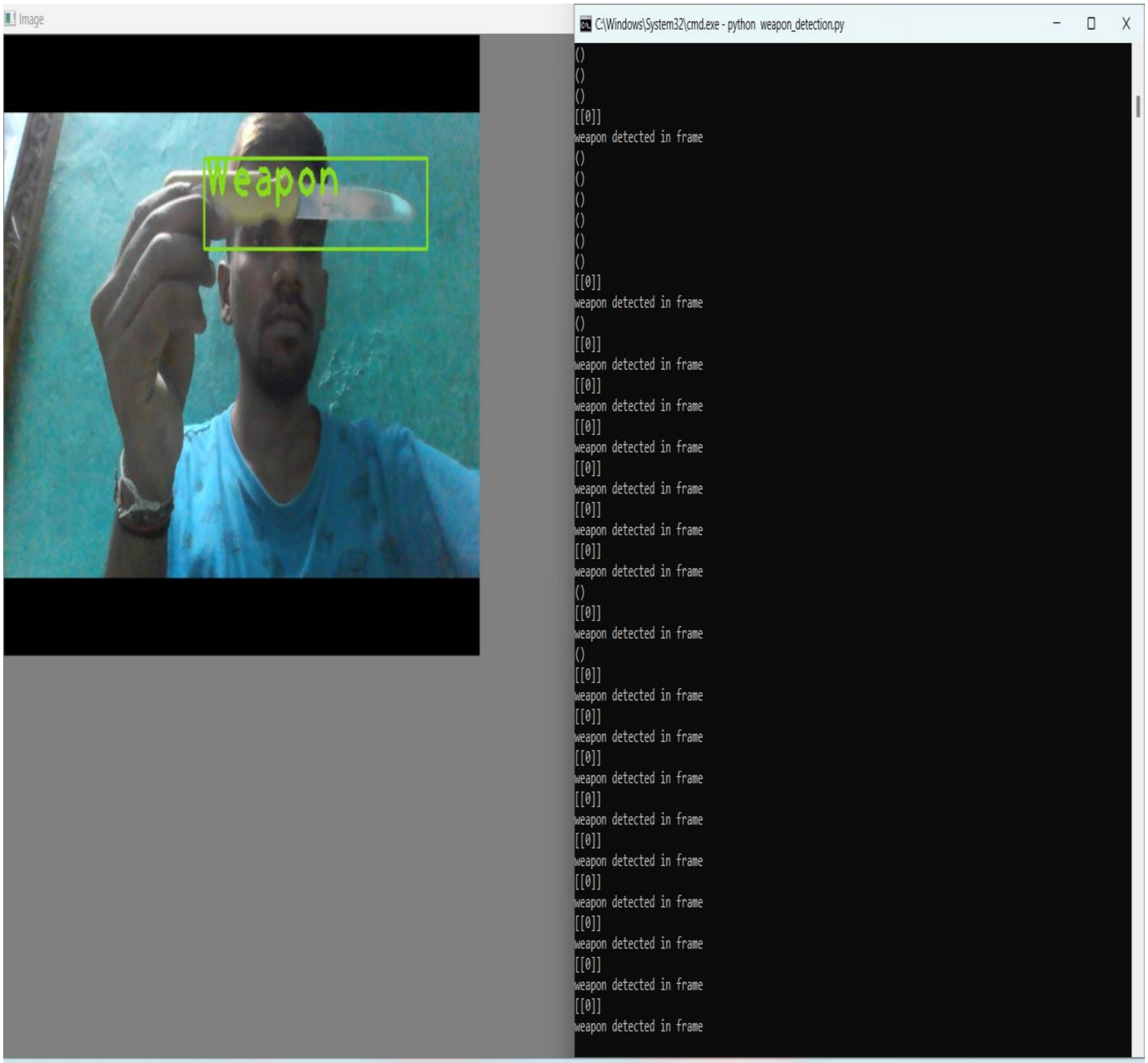
IF WEAPON IS NOT DETECTED IT SHOWS EMPTY BRACES **()**



5.1.4 If any weapon is not detected in frame it shows above result

STEP 5:

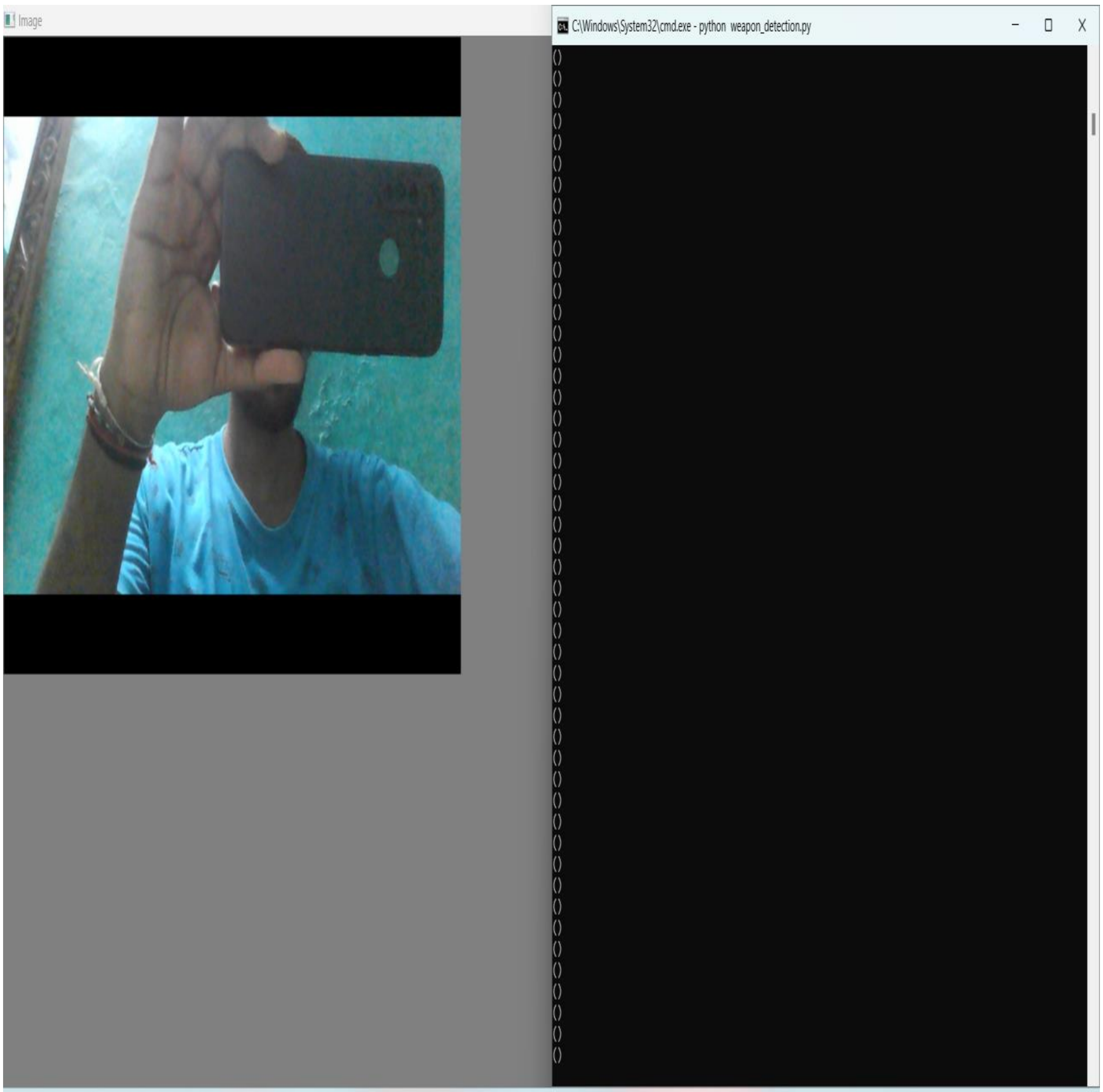
IF WEAPON DETECTS IT SHOWS **WEAPON DETECTED IN FRAME [0]**



5.1.5 It detecting the knife as weapon through webcam

STEP 6:

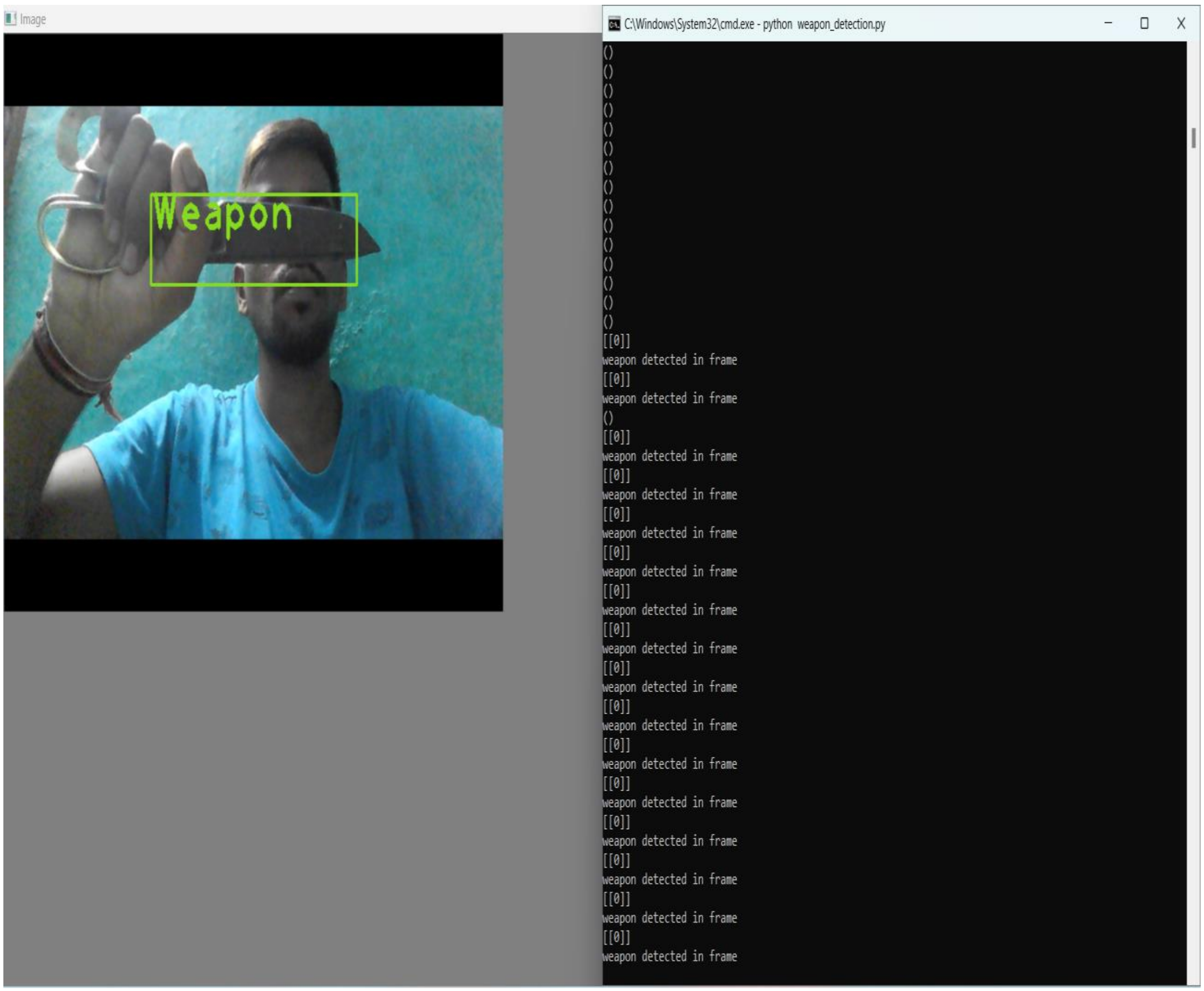
IF WEAPON IS NOT DETECTED IT SHOWS EMPTY BRACES **()**



5.1.6 It is not considering mobile as weapon.

STEP 7:

IF WEAPON DETECTS IT SHOWS **WEAPON DETECTED IN FRAME [0]**



5.1.7 It detects scissor as weapon.

5.2 Performance Metrics:

Analysis of SSD algorithm:

Gun Type	Average Accuracy	Speed (S)	Gun Detection	Correct Classification
AK-47	80%	0.67	Yes	Yes
Colt M1911	70%	0.89	Yes	Yes
S&W Model 10	66%	0.78	Yes	Yes
UZI Model	81%	0.61	Yes	No
Remington Model	72%	0.73	Yes	No

Analysis of Faster RCNN algorithm:

Gun Type	Average Accuracy	Speed (S)	Gun Detection	Correct Classification
AK-47	94%	1.28	Yes	Yes
Colt M1911	74%	1.63	Yes	Yes
S&W Model 10	91%	1.74	Yes	Yes
UZI Model	88%	1.49	Yes	No
Remington Model	76%	1.89	Yes	No

6.TESTING

Testing is an essential process in software development that aims to detect errors and ensure that the software system meets its requirements and user expectations. There are various types of tests that address specific testing needs:

1.Unit Testing: This involves designing test cases to validate that the internal program logic is performing correctly and that program inputs produce valid outputs. Unit testing is done after the completion of an individual unit before integration. It is a structural testing that relies on knowledge of the software's construction and is invasive.

A unit can be a single function, a method, a class, or a module, and it is tested in isolation from the rest of the application. The main goal of unit testing is to ensure that each unit of the software performs as expected and meets its requirements. Unit tests are typically automated and run frequently during the software development process.

Here are some general steps for performing unit testing:

1. Identify the units to be tested. This involves breaking down the application into smaller, testable units.

2. Write test cases for each unit.

3. Write code to implement each unit. The code should be designed to pass the unit tests.

4. Run the tests and evaluate the results. Any failures or errors should be addressed and fixed.

5. Refactor the code as needed based on the test results. 6. Repeat the process for each unit until all units have been tested.

Some benefits of unit testing include:

1. Catching errors early in the development cycle, when they are less expensive to fix.

2. Improving code quality by encouraging modular and loosely-coupled designs.

3. Providing a safety net for refactoring and changing code.

4. Increasing confidence in the correctness and stability of the application.

5. Reducing the need for manual testing and regression testing. Overall, unit testing is an essential part of software development that can help improve the quality, reliability, and maintainability of software applications.

2. Integration Testing: Integration testing is designed to test integrated software components to determine if they work together as expected. This type of testing exposes problems that arise from the combination of components.

3. Functional Testing: Functional testing is focused on testing the software's functions to ensure that they work as specified by the business and technical requirements. It includes testing valid and invalid inputs, functions, operations, systems, and procedures.

4. System Testing: This type of testing ensures that the entire integrated software system meets the requirements and produces known and predictable results.

5. White Box Testing: This is a testing method in which the tester has knowledge of the inner workings, structure, and language of the software. It is used to test areas that cannot be reached from a black box perspective.

6. Black Box Testing: This is a testing method in which the tester does not have any knowledge of the inner workings, structure, or language of the software. The software is treated as a "black box," and the test provides inputs and observes the outputs without considering how the software works.

Testing for weapon detection using YOLOv3 involves the following steps:

1. Collecting and preparing the dataset: The first step is to collect a dataset of images and videos that contain weapons. The dataset should have a good representation of the different types of weapons and the various angles and lighting conditions under which they can appear. Once the dataset is collected, it needs to be prepared for training the YOLOv3 model. This involves labeling the images and videos, dividing them into training and validation sets, and converting them into the required format for YOLOv3.

2. Training the YOLOv3 model: The next step is to train the YOLOv3 model on the prepared dataset. This involves configuring the model and setting the hyperparameters such as learning rate, batch size, and number of epochs. The model is then trained on the training set and validated on the validation set. The training process involves optimizing the model to detect weapons accurately and minimize false positives.

3. Evaluating the performance of the model: Once the model is trained, it needs to be evaluated to determine its accuracy and performance. This involves testing the model on a test set of images and videos that were not used during training. The performance metrics such as precision, recall, and F1 score are calculated to assess the model's performance.

4. Fine-tuning the model: Based on the evaluation results, the model may need to be fine-tuned to improve its performance. This involves adjusting the hyperparameters, adding more training data, or changing the model architecture.

5. Deploying the model: Once the model is trained and evaluated, it can be deployed for real-time weapon detection. This involves integrating the model into an application or system that can process live video feeds or images and generate alerts or notifications when weapons are detected.

6. Continuously monitoring and updating the model: Finally, it is important to continuously monitor the performance of the model and update it as needed. This involves periodically retraining the model with new data and adjusting the hyperparameters to maintain high accuracy and minimize false positives.

7. CONCLUSION AND FUTURE WORK

Conclusion

In conclusion, weapon detection using artificial intelligence (AI) and deep learning is a promising technology for enhancing security applications. With the use of advanced algorithms and deep learning techniques, AI-based weapon detection systems can accurately identify various types of weapons in real-time, enabling security personnel to respond quickly and prevent potential threats. AI-based weapon detection systems can be deployed in various environments, such as airports, public areas, schools, and government buildings, to enhance security and prevent the unauthorized possession of weapons. Moreover, these systems can be integrated with other security technologies, such as access control systems and surveillance cameras, to provide a comprehensive security solution. Overall, AI-based weapon detection systems have the potential to improve security in various applications, but it is important to carefully evaluate the technology and its limitations before deploying it inreal-world settings. It is crucial to balance the benefits of the technology with the ethical considerations and privacy concerns to ensure that the use of AI for weapon detection is safe and effective.

Future Work

Perpetration of this design in real world would be an evaluation in field of security and surveillance. Integration of being security systems like surveillance cameras with this software will be a game changer in area of security services. We could conceivably save numerous lives from pitfalls through immediate alert systems. In future we are going to extent this project so that it can detect hidden weapons.

8.REFERENCE

[1].https://kintronics.com/walk-metal-detectors-work/#:~:text=Walk%2Dthrough%20metal%20detectors%20typically,reflected%20magnetic%20field%20is% 20created

[2].https://www.ceia.net/security/index.aspx

[3].http://www.jtaer.com/apr2008/zhang_ouyang_he_p9.pdf.

[4].https://www.gujarindustries.com/blog/high-performance-baggage-scanner-machine[5].https://securityjkdc.com/news/advantages-of-using-x-ray-baggage-scanner.html

[6].https://www.houstonsystem.com/how-does-the-baggage-scanner-work-and-scan-hazardous-drugs

[7].https://www.researchgate.net/publication/352095753_Application_of_Deep_Learning_for_Weapons_Dete ction_in_Surveillance_Videos

[8].https://www.researchgate.net/publication/357492794_Deep_Neural_Networks_for_Gun_Detection_in_Public_Surveillance

[9].https://www.projectwale.com/lite/viewProject.php?id=786&projectname=Weapon-Detection-using-Artificial-Intelligence-and-Deep-Learning-for-Security-Applications

[10]. https://jonathan-hui.medium.com/ssd-object-detection-single-shot-multibox-detector-for-real-time-processing-9bd8deac0e06#:~:text=SSD%20is%20a%20single%2Dshot,offsets%20to%20default%20boundary%20boxes

[11].https://www.google.com/search?q=ssd+architecture&rlz=1C1DLBX_enIN877IN878 &sxsrf=APwXEddDo7s99wRvfrCEhB551Y1MzCeZtw:1681292257662&source=ln

ms&tbm=isch&sa=X&ved=2ahUKEwi99p_ehaT-AhUyTGwGHWOKDHsQ_AUoAXoECAEQAw&biw=1536&bih=722&dpr=1.25#imgrc

=w8DbJJ4QIk2luM

[12].https://www.google.com/search?q=faster+rcnn+architecture&rlz=1C1DLBX_enIN877IN878&sxsrf=APwXEdfTwXpGMGLyZYkEKn-APcCV-73zMg:1681292288689&source=lnms&tbm=isch&sa=X&ved=2ahUKEwisvIXthaT- AhUtSmwGHWTvBdsQ_AUoAXoECAEQAw&biw=1536&bih=722&dpr=1.25

[13].S. Saha, V. Sharma, and N. Sharma, "Deep learning based weapon detection using YOLOv3," 2019 IEEE 7th International Conference on Bioinformatics and Computational Biology (ICBCB), Hangzhou, China, 2019, pp. 39-43.

[14].L. Hu, W. Huang and W. Chen, "A Robust Deep Learning-Based Approach for Weapon Detection," 2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA,USA, 2019, pp. 2029-2032.

[15].K. S. Kaviya, S. Santhiya and V. K. Ananthashayana, "A Deep Learning Approach for Weapon Detection in Surveillance Videos," 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Bangalore, India, 2018, pp. 2022-2026.

[16].S. S. Zaveri, R. S. Bhatia and S. R. Desai, "Real-time Weapon Detection using Deep Learning Techniques," 2019 IEEE 5th International Conference for Convergence in Technology (I2CT), Pune, India, 2019, pp. 1-6.

[17].R. Vignesh, R. Kumar and M. Shanthi, "Deep Learning Based Weapon Detection and Classification in Surveillance Videos," 2019 International Conference on Smart Technologies for Smart Nation (SmartTechCon), Bangalore, India, 2019, pp. 917-922.

[18].S. Ali, A. Sharif, A. Iqbal and J. A. Shah, "A deep learning-based approach for weapon detection in security applications," Journal of Ambient Intelligence and Humanized Computing, vol. 10, no. 7, pp. 2625-2633, 2019.

[19].https://blog.paperspace.com/faster-r-cnn-explained-object-detection/

[20].https://towardsdatascience.com/ssd-single-shot-detector-for-object-detection-using-multibox-1818603644ca

[21].X. Liu, J. Jia, J. Guo and H. Ji, "Weapon detection in video surveillance using a deep learning approach," IEEE Access, vol. 7, pp. 97909-97916, 2019.

[22].T. Y. Sun, C. W. Huang and J. H. Liu, "A Deep Learning Approach for Weapon Detection in Real-Time Video Surveillance," 2019 2nd International Conference on Automation, Control and Robots (ICACR), Bali, Indonesia, 2019, pp. 60-64.

[23].K. Kanoun, N. N. Boufares and M. Abid, "A Deep Learning-Based Weapon Detection System for Security Applications," 2019 4th International Conference on Advanced Technologies for Signal and Image Processing (ATSIP), Sousse, Tunisia, 2019, pp. 1-5.