International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

WEAPON DETECTION USING ARTIFICIAL INTELLEGENCE AND DEEP LEARNING FOR SECURITY APPLICATIONS

Divya Bhargavi Medi[1], G. Naga Pratyusha[2], R. Aishwarya Raj[3]

Department of Computer Science and Engineering, Stanley College of Engineering and Technology for Women, Telangana, India

Abstract. Security is always a main concern in every domain, due to a rise in crime rate in a crowded event or suspicious lonely areas. In This project we implement automatic gun (or) weapon detection using a Convolution Neural Network (CNN) based SSD and Faster RCNN algorithms. Proposed implementation uses two types of datasets. One dataset, which had pre-labelled images and the other one is a set of images, which were labelled manually.

Keywords:

Hand tracking, Computer Vision, OpenCV, Air Writing, Media Pipe, Color Tracking, Color Detection, Mask, Eraser, Volume Gesture, Finger Counting.

1. Introduction

1.1 About Project

Weapon or Anamoly detection is the identification of irregular, unexpected, unpredictable, unusual events or items, which is not considered as a normally occurring event or a regular item in a pattern or items present in a dataset and thus different from existing patterns. An anomaly is a pattern that occurs differently from a set of standard patterns. Therefore, anomalies depend on the phenomenon of interest [3] [4]. Object detection uses feature extraction and learning algorithms or models to recognize instances of various category of objects [6]. Proposed implementation focuses on accurate gun detection and classification. Also concerned with accuracy, since a false alarm could result in adverse responses [11] [12]. Choosing the right approach required to make a proper trade-off between accuracy and speed. Figure 1 shows the methodology of weapons detection using deep learning. Frames are extracted from the input video. Frame differencing algorithm is applied and bounding box created before the detection of object [7] [8] [14]. Fig.1.Methodology Fig.2. Detection and T racking The flow of object detection and tracking as shown in figure 2. Dataset is created, trained and fed to object detection algorithm. Based on application suitable detection

algorithm (SSD or fast RCNN) chosen for gun detection. The approach addresses a problem of detection using various machine-learning models like Region Convolutional Neural Network (RCNN), Single Shot Detection (SSD).

1.2 Objectives of the Project

In the particular case of firearm detection, while the results obtained with these novel methods are promising, there are still substantial limitations when they are applied in new scenarios different to those used for training, especially an unacceptable number of false positives Furthermore, most of the proposed methods for automatic detection of firearms are based only on the appearance of the weapon in the image, without taking into account additional information that may help provide a more robust and accurate detection. This work proposes the use of the human pose as complementary information to improve the performance of current handgun detectors based on deep learning.

1.3 Scope of the Project

Security is always a main concern in every domain, due to a rise in crime rate in a crowded event or suspicious lonely areas. This Weapon Detection is very much needed to identify any abnormal situations. Due to growing demand in the protection of safety, security and personal properties deployment of video surveillance systems can recognize and interpret the scene and anomaly events and play a vital role in intelligence monitoring.

2. Literature Survey

Ruben J Franklin et.al., "Anomaly Detection in Videos for Video Surveillance Applications Using Neural Networks," International Conference on Inventive Systems and Control,2020.Deep learning has gained a tremendous influence on how the world is adapting to Artificial Intelligence since past few years. Some of the popular object detection algorithms are Region-based Convolutional Neural Networks (RCNN), Faster RCNN, Single Shot Detector (SSD) and You Only Look Once (YOLO). Amongst these, Faster-RCNN and SSD have better accuracy, while YOLO performs better when speed is given preference over accuracy. Deep learning combines SSD and Mobile Nets to perform efficient implementation of detection and tracking. This algorithm performs efficient object detection while not compromising on the performance. Keywords — Mobile Nets, Single Shot Detector, COCO. I. INTRODUCTION Since Alex Net has stormed the research world in 2012 ImageNet on a large scale visual recognition challenge, for detection in-depth learning, far exceeding the most traditional methods of artificial vision used in literature. In artificial vision, the neural convolution networks are distinguished in the classification of images. Fig. 1. Basic block diagram of detection and Tracking Fig. 1 shows the basic block diagram of detection and tracking. In this paper, an SSD and Mobile Nets based algorithms are implemented for detection and tracking in python environment. Object detection involves detecting region of interest of object from given class of image. Different methods are –Frame differencing, Optical flow, Background subtraction. This is a method of detecting and locating an object which is in motion with the help of a camera. Detection and tracking algorithms are described by extracting the features of image and video for security applications . Features are extracted using CNN and deep learning [9]. Classifiers are used for image classification and counting . YOLO based algorithm with GMM model by using the concepts of deep learning will give good accuracy for feature extraction and classification . Section II describes SSD and Mobile Nets

algorithm, section III explains method of implementation, and section IV describes simulation results and analysis. II. OBJECT DETECTION AND TRACKING ALGORITHMS A. Single Shot Detector (SSD) algorithm SSD is a popular object detection algorithm that was developed in Google Inc. It is based on the VGG-16 architecture. Hence SSD is simple and easier to implement. Fig. 2. VGG-16 SSD Model. Fig. 2 shows VGG 16 SSD model. A set of default boxes is made to pass over several feature maps in a convolutional manner. If an object detected is one among the object classifiers during prediction, then a score is generated. The object shape is adjusted to match the localization box. For each box, shape offsets and confidence level are predicted. During training, default boxes are matched to the ground truth boxes. The fully connected layers are discarded by SSD architecture. The model loss is computed as a weighted sum of confidence loss and localization loss. Measure of the deviation of the predicted box from the ground truth box is localization loss. Confidence is a measure of in which manner confidence the system is that a predicted object is the actual object. Proceedings of the International Conference on Inventive Research in Computing Applications (ICIRCA 2018) IEEE Xplore Compliant Part Number:CFP18N67-ART; ISBN:978-1-5386-2456-2 978-1-5386-2456-2/18/$31.00 ©2018 IEEE 1305 Elimination of feature resampling and encapsulation of all computation in a single network by SSD makes it simple to train with Mobile Nets. Compared to YOLO, SSD is faster and a method it performs explicit region proposals and pooling (including Faster R-CNN). B. Mobile Nets algorithm Mobile Nets uses depth wise separable convolutions that helps in building deep neural networks. The Mobile Nets model is more appropriate for portable and embedded vision based applications where there is absence of process control.

2.1 Existing System

Security is always a main concern in every domain, due to a rise in crime rate in a crowded event or suspicious lonely areas. Abnormal detection and monitoring have major applications of computer vision to tackle various problems. Due to growing demand in the protection of safety, security and personal properties, needs and deployment of video surveillance systems can recognize and interpret the scene and anomaly events play a vital role in intelligence monitoring.

2.2 Proposed System

This paper implements automatic gun (or) weapon detection using a convolution neural network (CNN) based SSD and Faster RCNN algorithms. Proposed implementation uses two types of datasets. One dataset, which had pre-labelled images and the other one is a set of images, which were labelled manually. Results are tabulated, both algorithms achieve good accuracy, but their application in real situations can be based on the trade-off between speed and accuracy

The **Systems Development Life Cycle (SDLC)**, or Software Development Life Cycle in systems engineering, information systems and software engineering, is the process of creating or altering systems, and the models and

methodologies that people use to developthesesystem

In software engineering the SDLC concept underpins many kinds of software development methodologies. These methodologies form the framework for planning and controlling the creation of an information system the software development process

3. Proposed Architecture

3.1 SYSTEM ARCHITECTURE

The purpose of the design phase is to arrange an answer of the matter such as by the necessity document. This part is that the opening moves in moving the matter domain to the answer domain. The design phase satisfies the requirements of the system. The design of a system is probably the foremost crucial issue warm heartedness the standard of the software package. It's a serious impacton the later part, notably testing and maintenance.

The output of this part is that the style of the document. This document is analogous to a blueprint of answer and is employed later throughout implementation, testing and maintenance. The design activity is commonly divided into 2 separate phases System Design and Detailed Design.

System Design conjointly referred to as top-ranking style aims to spot the modules that ought to bewithin the system, the specifications of those modules, and the way them move with one another tosupply the specified results.

At the top of the system style all the main knowledge structures, file formats, output formats, and also the major modules within the system and their specifications square measure set. System designis that the method or art of process the design, components, modules, interfaces, and knowledge fora system to satisfy such as needs. Users will read it because the application of systems theory to development.

Detailed Design, the inner logic of every of the modules laid out in system design is determined. Throughout this part, the small print of the info of a module square measure sometimes laid out in a high-level style description language that is freelance of the target language within which the software package can eventually be enforced.

In system design the main target is on distinguishing the modules, whereas throughout careful stylethe main target is on planning the logic for every of the modules.

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

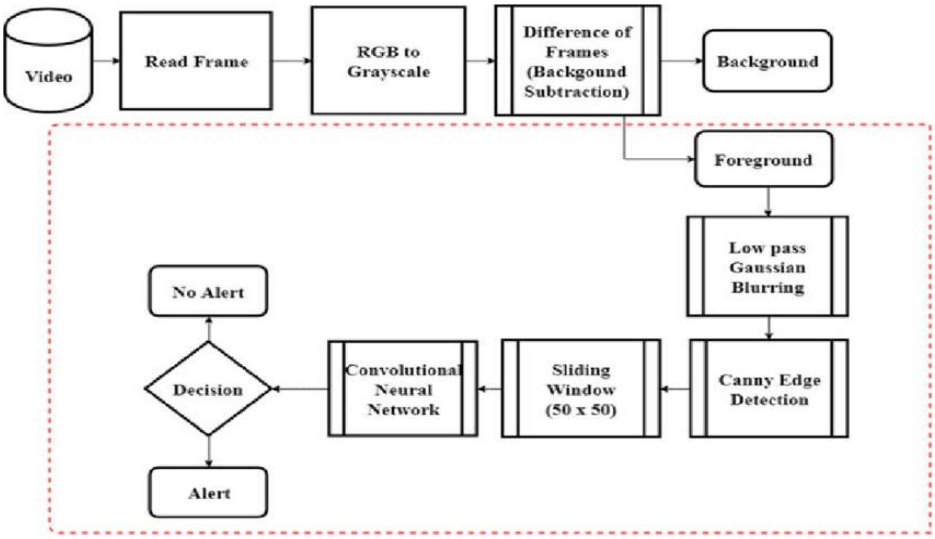


Fig: 1 Architecture

Here first we collect the data sets and process the data and we remove if there are any impurities in the data sets. Next the data is normalized if needed like it can be converted to smaller volume of data.Next the data is converted to supporting format. And then it is stored in the databases. Next the required method is applied. Now we get the final results.

3.2 INPUT AND OUTPUT DESIGN

3.2.1 INPUT DESIGN

Input design is a part of overall system design. The main objective during the input design is as givenbelow:

- To produce a cost-effective method of input.
- To achieve the highest possible level of accuracy.
- To ensure that the input is acceptable and understood by the user.

INPUT STAGES

The main input stages can be listed as below:

- Data recording
- Data transcription

- Data conversion
- Data verification
- Data control
- Data transmission
- Data validation
- Data correction

INPUT TYPES

It is necessary to determine the various types of inputs. Inputs can be categorized as follows:

- External inputs, which are prime inputs for the system.
- Internal inputs, which are user communications with the system.
- Operational, which are computer department's communications to the system?
- Interactive, which are inputs entered during a dialogue.

INPUT MEDIA

At this stage choice has to be made about the input media. To conclude about the input mediaconsideration has to be given to;

- Type of input
- Flexibility of format
- Speed
- Accuracy
- Verification methods
- Rejection rates

- Ease of correction
- Storage and handling requirements
- Security
- Easy to use
- Portability

Keeping in view the above description of the input types and input media, it can be said thatmost of the inputs are of the form of internal and interactive. As

Input data is to be the directly keyed in by the user, the keyboard can be considered to be the mostsuitable input device.

3.2.2 OUTPUT DESIGN

Outputs from computer systems are required primarily to communicate the results of processing to users. They are also used to provide a permanent copy of the results for later consultation. The varioustypes of outputs in general are:

- External Outputs, whose destination is outside the organization
- Internal Outputs whose destination is within organization and they are the
- User's main interface with the computer.
- Operational outputs whose use is purely within the computer department.
- Interface outputs, which involve the user in communicating directly.

OUTPUT DEFINITION:

The outputs should be defined in terms of the following points:

- Type of the output
- Content of the output
- Format of the output

- Location of the output
- Frequency of the output
- Volume of the output
- Sequence of the output

It is not always desirable to print or display data as it is held on a computer. It should bedecided as which form of the output is the most suitable.

3.3 SOFTWARE REQUIREMENT SPECIFICATION

- Collection of Images or Videos
- Removal of unwanted data and organizing Images
- Predicting the weapons like Guns in a efficient way

NON FUNCTIONAL REQUIREMENTS

NON-FUNCTIONAL REQUIREMENT (NFR) specifies the quality attribute of a software system. They judge the software system based on Responsiveness, Usability, Security, Portability and other non-functional standards that are critical to the success of the software system. Example of nonfunctional requirement, *"how fast does the website load?"* Failing to meet non-functional requirements can result in systems that fail to satisfy user needs. Non- functional Requirements allows you to impose constraints or restrictions on the design of the system across the various agile backlogs. Example, the site should load in 3 seconds when the number of simultaneous users are > 10000. Description of non-functional requirements is just as critical as a functional requirement.

- The images or videos collection is taken from crowded and suspected areas.
- Efficient Hardware Resources like GPU will be required

3.4 UML CONCEPTS

Data Flow Diagram can also be termed as bubble chart. It is a pictorial or graphical form, which can be applied to represent the input data to a system and multiple functions carried out on thedata and the generated output by the system.

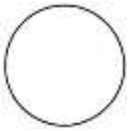
A graphical tool accustomed describe and analyze the instant of knowledge through a systemmanual or

automatic together with the method, stores of knowledge, and delays within the system. The transformation of knowledge from input to output, through processes, is also delineate logicallyand severally of the physical elements related to the system. The DFD is also known as a data flow graph or a bubble chart.The BasicNotation used to create a DFD's are as follows:

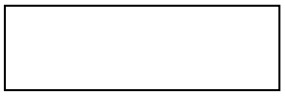
- Dataflow:

- **Process:**

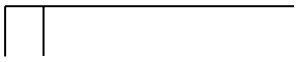
.



- Source:



- **Data Store:**



- **Rhombus**: decision

UML DIAGRAMS

The Unified Modeling Language allows the software engineer to express an analysis model using themodeling notation that is governed by a set of syntactic semantic and pragmatic rules.

A UML system is represented using five different views that describe the system from distinctlydifferent perspective. Each view is defined by a set of diagram, which is as follows.

User Model View

This view represents the system from the users perspective. The analysis representation describes ausage scenario from the end-users perspective.

Structural Model view

In this model the data and functionality are arrived from inside the system. This model view modelsthe static structures.

Behavioral Model View

It represents the dynamic of behavioral as parts of the system, depicting the interactions of collection between various structural elements described in the user model and structural model view.

Implementation Model View

In this the structural and behavioral as parts of the system are represented as they are to be built.

3.4.1 USE CASE DIAGRAM

A use case diagram at its simplest is a representation of a user's interaction with the system and depicting the specifications of a use case. A use case diagram can portray the different types of users of a system and the various ways that they interact with the system. This type of diagram is typically used in conjunction with the textual use case and will often be accompanied by other typesof diagrams as well.

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

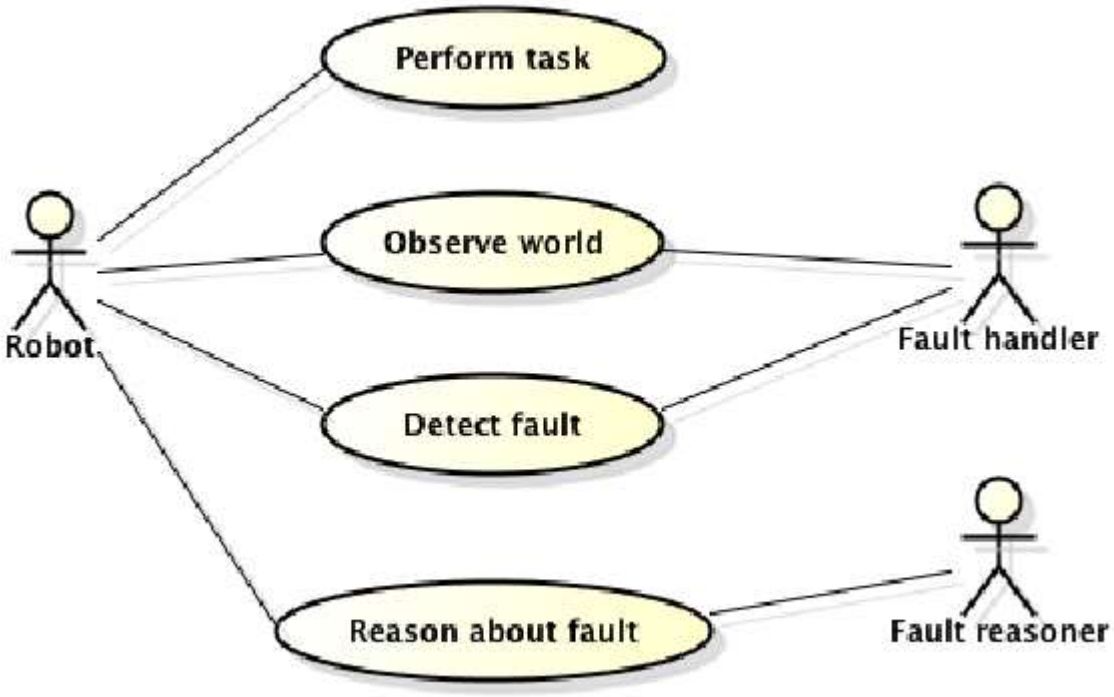


Fig 3.4.1 Use Case Diagram

3.4.2 CLASS DIAGRAM

The class diagram is the main building block of object oriented modeling. It is used both for general conceptual modeling of the systematic of the application, and for detailed modeling translating the models into programming code. Class diagrams can also be used for data modeling. The classes in aclass diagram represent both the main objects, interactions in the application and the classes to be programmed. A class with three sections, in the diagram, classes is represented with boxes which contain three parts:

The upper part holds the name of the class

The middle part contains the attributes of the class

The bottom part gives the methods or operations the class can take or undertake.

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

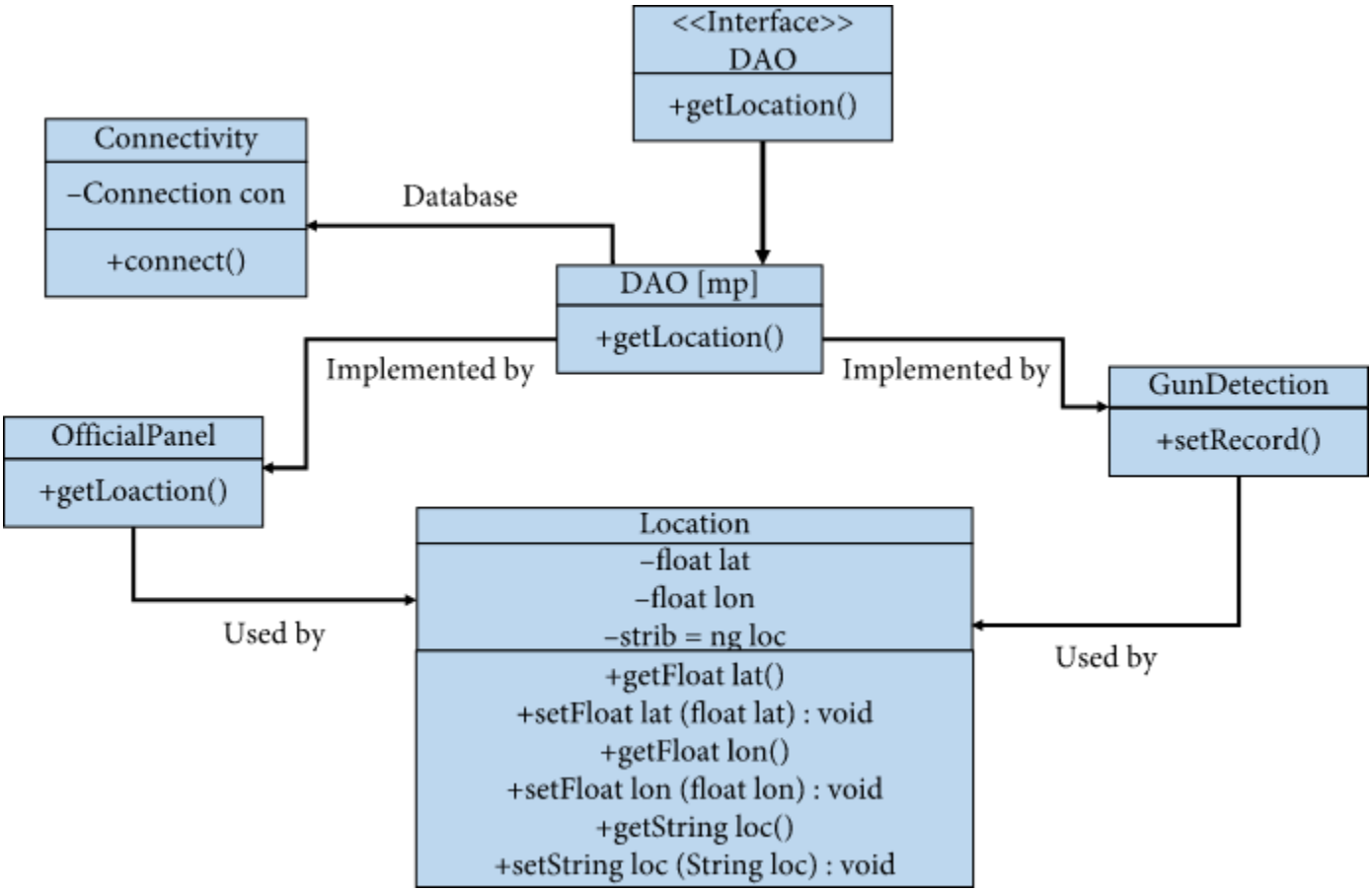


Fig 3.4.2: Class Diagram

3.4.3 SEQUENCEDIAGRAM

A sequence diagram is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

development. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

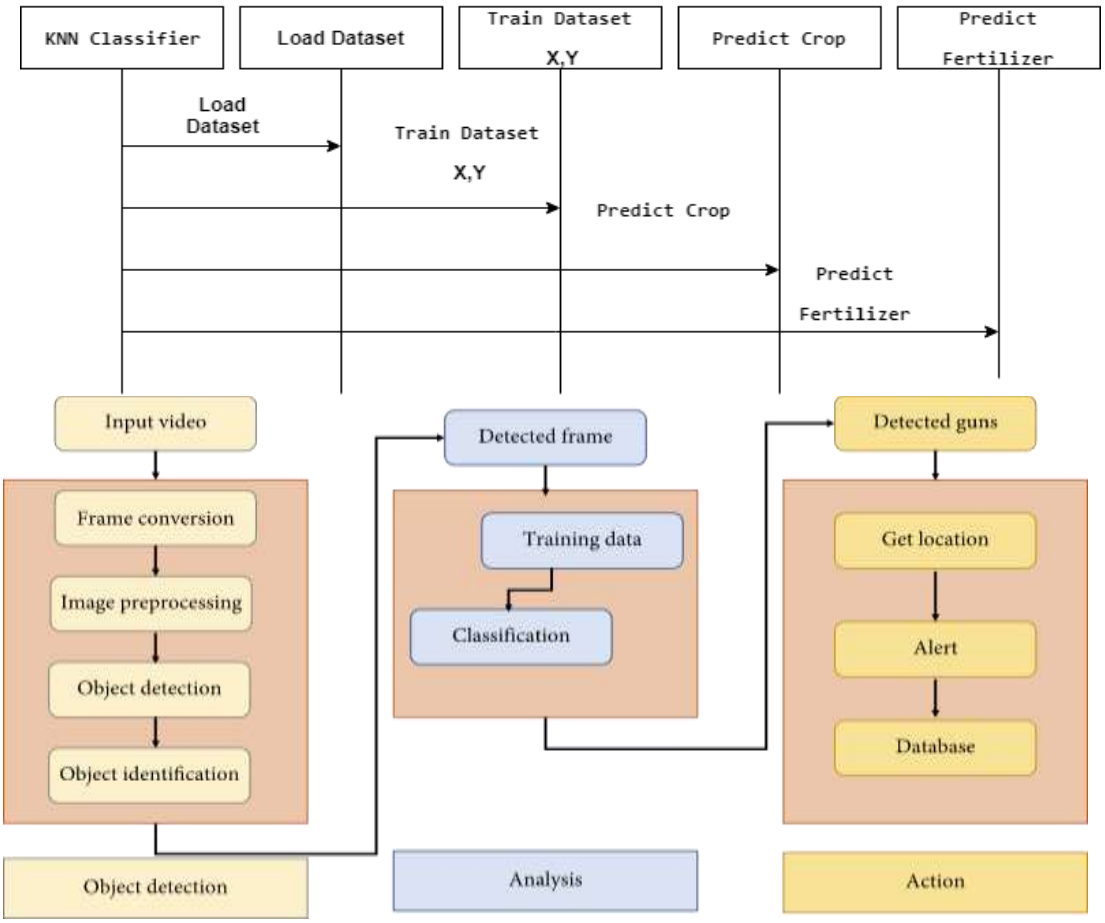


Fig 3.4.3: Sequence Diagram

3.1.4 DIAGRAM

Activity diagrams are graphical representations of workflows of stepwise activities and actions withsupport for choice, iteration and concurrency. In the Unified Modeling Language, activity diagramscan be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of component

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

3.4.5 COMPONENT DIAGRAM

Component diagram is a special kind of diagram in UML. The purpose is also different from all other diagrams discussed so far. It does not describe the functionality of the system but it describesthe components used to make those function

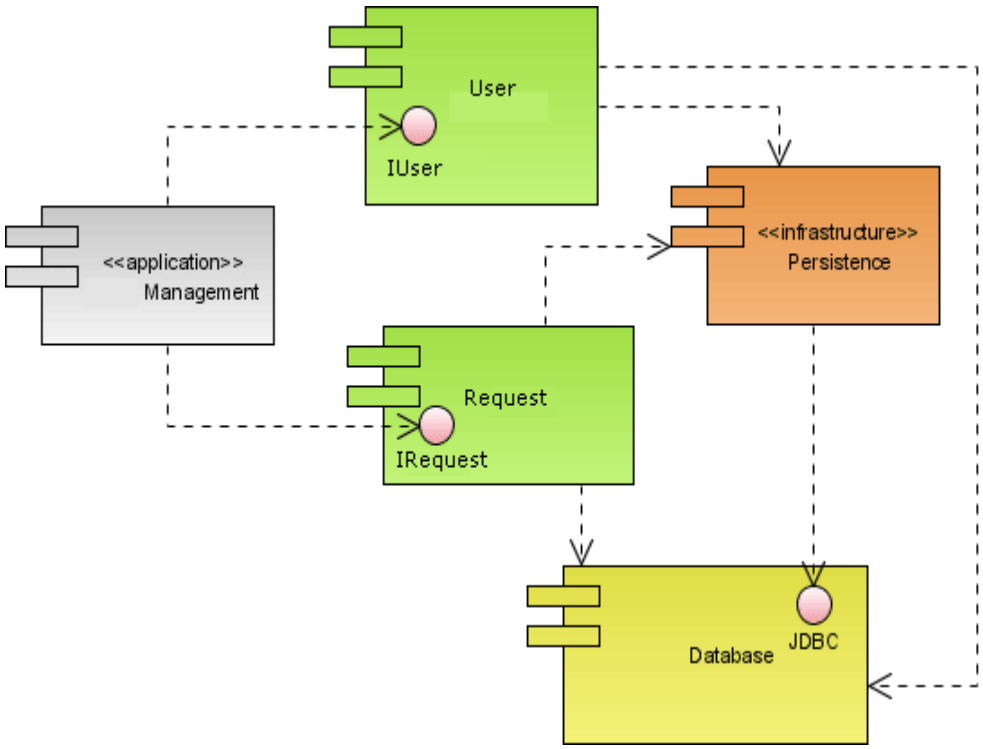


Fig 3.4.5: Component Diagram

International Journal for Innovati
Engineering and Management Research
A Peer Revieved Open Access International Journal
www.

3.4.6 DEPLOYMENT DIAGRAM

Deployment diagram shows the configuration of run time processing nodes and the components ofthe application. It is a kind of structure diagram used in modeling the physical aspects of an objectoriented system.

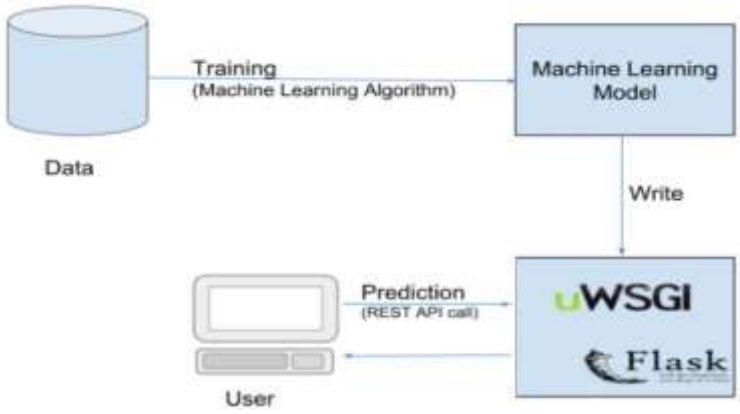


Figure 3.4.6: Deployment diagram

4.1 Algorithm

- Considering the input video frames are extracted.

- To demonstrate how to build a convolution neural network based Image classifier,we shall build a 7 layer neural network that will identify and separate one image from another.

- In This network we shall build is a very small network that we can run on a cpu as well. Traditional neural networks that a very good at doing image classification have many more parameters and take a lot of time if trained on normal cpu

- SSD and Faster RCNN algorithm are simulated for pre labeled and self created image dataset for weapon (gun)detection these algorithm are efficient and give good results.

- At the end the weapon is detected efficiently.

4.2 Code Implementation

STEP 1: Import CV2
STEP 2: Import nummpy as np, glob, random
STEP 3: Import base 64
STEP 4: Import real time object detection weights: yolo3_training_last.weights
STEP 5: Import configuration source file: testing cfg
STEP 6: Load set of images that want to detect whether there is gun or not
STEP 7: Open visual studio and run the python code
STEP 8: Give absolute path of yolo3, weight file
STEP 9: Give source path of config.cfg file
STEP 10: Give source path of .jpg files
STEP 11: CV2 will read the weight and config
STEP 12: CV2 will search for similar kind of images in yolo weight data files using deep learning
STEP 13:Input image will be converted to base64, which can be scanned pixel by pixel of a .jpg frame
STEP 14: From the input .jpg file , system will look for parameters like, depth, width and height and values stored in a array
STEP 15: CV2 will compare the result data of "yolo.weight" file with the base64 data of a input jpg frame given by user
STEP 16: If data of guns from yolov3. Weight file and uses input .jpg file data got matched , system will send response saying 'GUN' , highlighting gun in given input .jpg file
STEP 17: If there is not gun system will skip the response

```
import cv2

import numpy as np

import glob

import random

import os

import base64


print(base64.b64decode(b'Cl9fICAgICAgICBfX19fX18gICAgICBf
X19fICAKXCBcICAgICAgLyAvIF9fX3xfXyBffCAgXyBcIAogX
CBcIC9cIC8gLyB8ICAgLyBfYCB8IHxfKSB8CiAgXCBWICBWI
C98IHxfX3wgKF98IHwgIF8gXCAgXF8vICBcICBcX1
xfXyxffF98IFxfXAo=').decode("UTF-8"))

print("\n\nGIVE ABSOLUTE PATH FOR THE FILES
REQUIRED\nEXAMPLE OF WEIGHT FILE PATH :
E:\\data\\yolov3_trained.weights\n\n")


# Paths

weight_path = str(input("\n\nEnter the absolute path of the Weight
file : "))

config_path = str(input("Enter the absolute path of the Test Config
file : "))

image_path = str(input("Enter the absolute path of the image(s) to
test [*.jpg for all] : "))



# Load Yolo

net = cv2.dnn.readNet(weight_path, config_path)


# Name custom object

classes = ["Gun"]
```

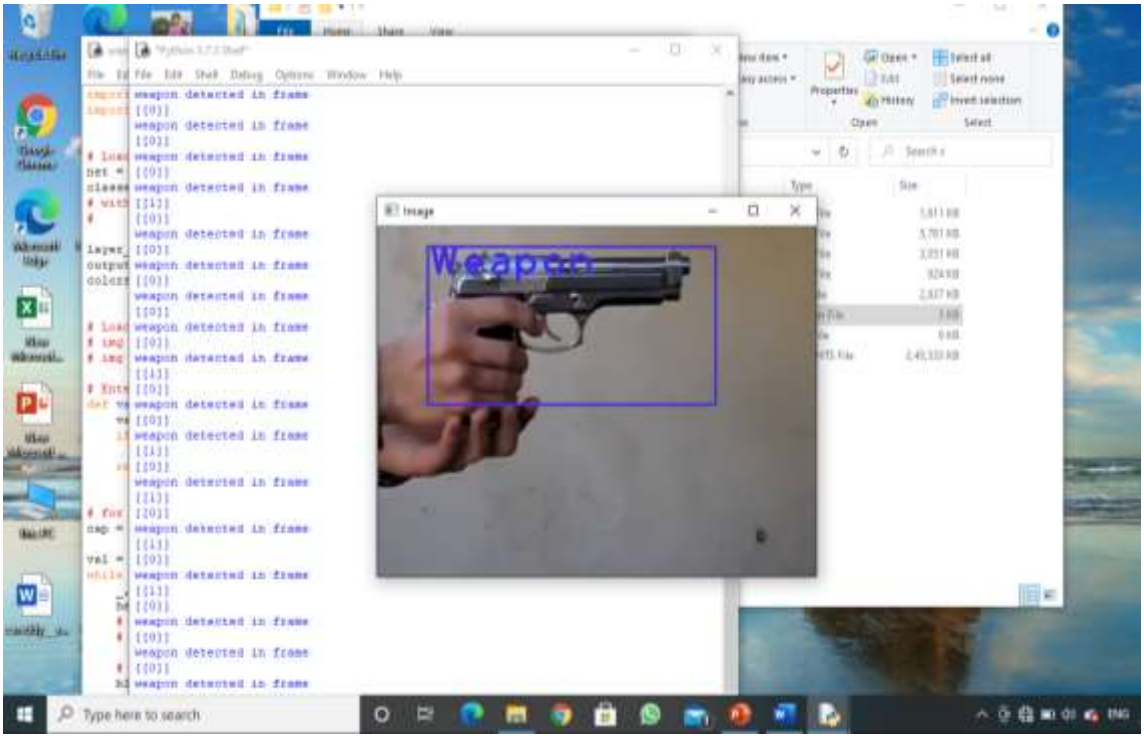
```
# Images path
images_path = glob.glob(image_path)
#                    Path for the image to test


layer_names = net.getLayerNames()
output_layers = [layer_names[i - 1] for i in
net.getUnconnectedOutLayers()]
colors = np.random.uniform(0, 255, size=(len(classes), 3))


# Insert here the path of your images
random.shuffle(images_path)
# loop through all the images
for img_path in images_path:
    # Loading image
    img = cv2.imread(img_path)
    img = cv2.resize(img, None, fx=0.4, fy=0.4)
    height, width, channels = img.shape


    # 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 informations 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 > 0.3:
            # Object detected
            print(class_id)
            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)
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, 2)


cv2.imshow("Image", img)
key = cv2.waitKey(0)
```

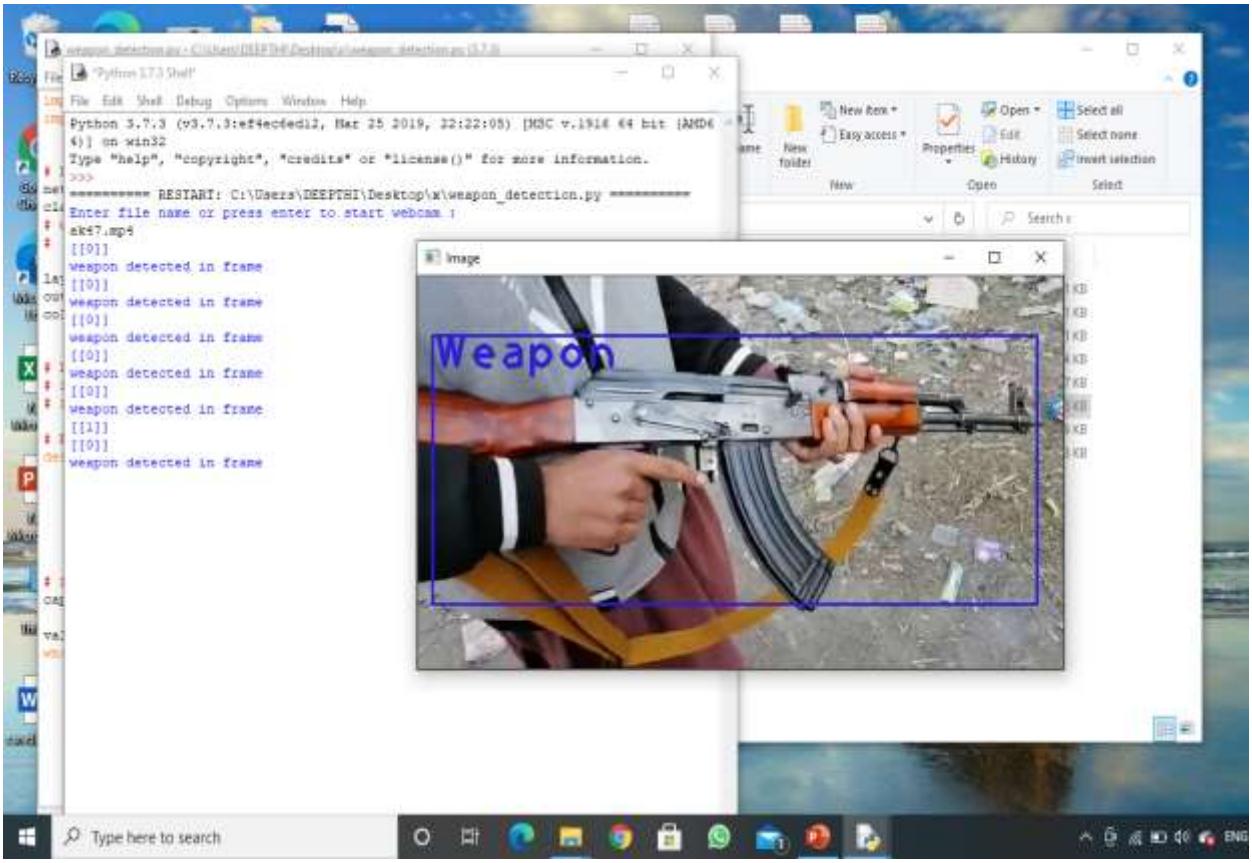
International Journal for Innovati
Engineering and Management Research
A Peer Revieved Open Access International Journal
www.

5. Result



From the above image we an see that the weapon is detected correctly based on the input.

International Journal for Innovati
Engineering and Management Research
A Peer Revieved Open Access International Journal
www.



6. Conclusion

SSD and Faster RCNN algorithms are simulated for pre labeled and self-created image dataset for weapon (gun) detection. Both the algorithms are efficient and give good results but their application in real time is based on a tradeoff between speed and accuracy. In terms of speed, SSD algorithm gives better speed with 0.736 s/frame. Whereas Faster RCNN gives speed 1.606s/frame, which is poor compared to SSD. With respect to accuracy, Faster RCNN gives better accuracy of 84.6%. Whereas SSD gives an accuracy of 73.8%, which is poor compared to faster RCNN.SSD provided real time detection due to faster speed but Faster RCNN provided superior accuracy.

7. Future Scope

Further, it is enforced for larger datasets by coaching victimization GPUs and high-end DSP (Digital Signal Processors) and FPGA (Field-programmable gate array) kits. In order that we are able to deliver the goods speed in detection of weapons kind several range of varieties and models.

8. References

1. [1] Wei Liu et al., “SSD: Single Shot MultiBox Detector”, European Conference on Conputer Vision, Volume 169, pp 20-31 Sep. 2017.

2. [2] D. Erhan et al., “Scalable Object Detection Using Deep Neural Networks,” IEEE Conference on Computer Vision and Pattern Recognition(CVPR),2014.

3. [3] Ruben J Franklin et.al., “Anomaly Detection in Videos for Video Surveillance Applications Using Neural Networks,” International Conference on Inventive Systems and Control,2020.

4. [4] H R Rohit et.al., “A Review of Artificial Intelligence Methods for Data Science and Data Analytics: Applications and Research Challenges,”2018 2nd International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud), 2018.

5. [5] Abhiraj Biswas et. al., “Classification of Objects in Video Records using Neural Network Framework,” International conference on Smart Systems and Inventive Technology,2018.

6. [6] Pallavi Raj et. al.,“Simulation and Performance Analysis of Feature Extraction and Matching Algorithms for Image Processing Applications” IEEE International Conference on Intelligent Sustainable Systems,2019. [7] Mohana et.al., “Simulation of Object

Detection Algorithms for Video Survillance Applications", International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud),2018.

7. [8] Yojan Chitkara et. al.,"Background Modelling techniques for foreground detection and Tracking using Gaussian Mixture model" International Conference on Computing Methodologies and Communication,2019.
8. [9] Rubner et.al, "A metric for distributions with applications to image databases", International Conference on Computer Vision,2016.
9. [10] N. Jain et.al., "Performance Analysis of Object Detection and Tracking Algorithms for Traffic Surveillance Applications using Neural Networks," 2019 Third
10. International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud), 2019.
11. [11] A. Glowacz et.al., "Visual Detection of Knives in Security Applications using Active Appearance Model",Multimedia Tools Applications, 2015.
12. [12] S. Pankanti et.al.,"Robust abandoned object detection using regionlevel analysis,"International Conference on Image Processing,2011. [13] Ayush Jain et.al.,"Survey on Edge Computing - Key Technology in Retail Industry" International Conference on Intelligent Computing and Control Systems,2019.
13. [14] Mohana et.al., Performance Evaluation of Background Modeling Methods for Object Detection and Tracking," International Conference on Inventive Systems and Control,2020.