



WEAPON DETECTION SYSTEM

Powered by: **OpenCV**, **Imutiles**, **Numpy**

A project by:

SYED MUHAMMAD HAMZA ALI

A project of **Artificial Intelligence**

Introduction:

Overview: by the help of OpenCV library installed and Imutiles library the prototype detects various images of the gun by subtracting the background of the image, and using the classifier of cv2 on HAAR cascade file (.xml) we are able to the gun from images regardless of the background.

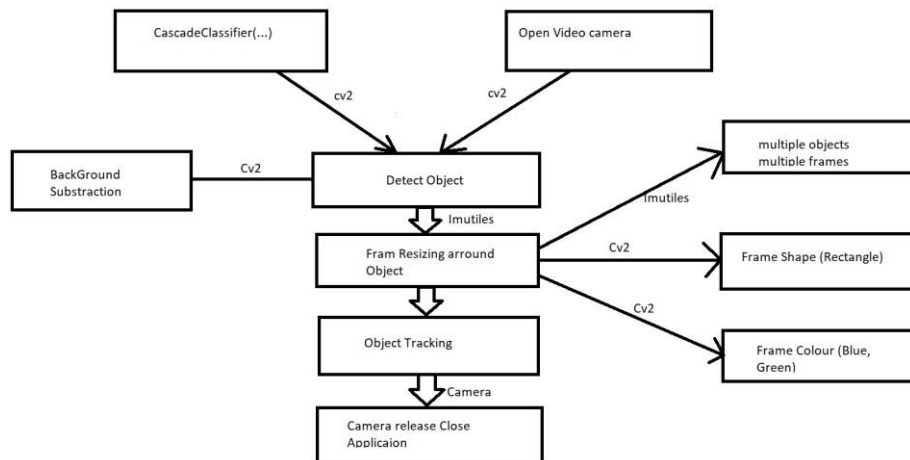
Objective: the objective of the project is to enhance the security systems. Being able to detect the gun using videocam in runtime from an image this project can further be improved for the development security purposes.

Scope and limitations: the scope of the project is the purpose of enhancing the security issues, by installing it in home-cams, offices-cam, and public places. As being the project for the security of public can develop the interest of authorities for the project. The limitation of this project righ now is that it detect the gun from an image. And only hand guns can be detected, as per respect to the shape for which it is trained.

Application-areas: there are various application areas of this project such as home-cams, public places cameras, banks cameras, offices cameras.

Methodology

Flow Diagram:



Data set: the Dataset being used is cascade file (.xml) that is classified by the help of (OpenCV) cv2 cascade classifier, that use machine learning, a cascade function is trained using a large number of both positive and negative images. The next step is to use it to find or detect objects in other pictures.

Libraries used: The libraries used in the project include:

NumPy: For the numerical computation used in this project to mention up the respective scale values in the format of (X,Y) coordinates weight and height.

OpenCV (Cv2): Cv2 library working on this OpenCV project (computer vision project). Through this library we were will be able to detect the gun in the respective image.

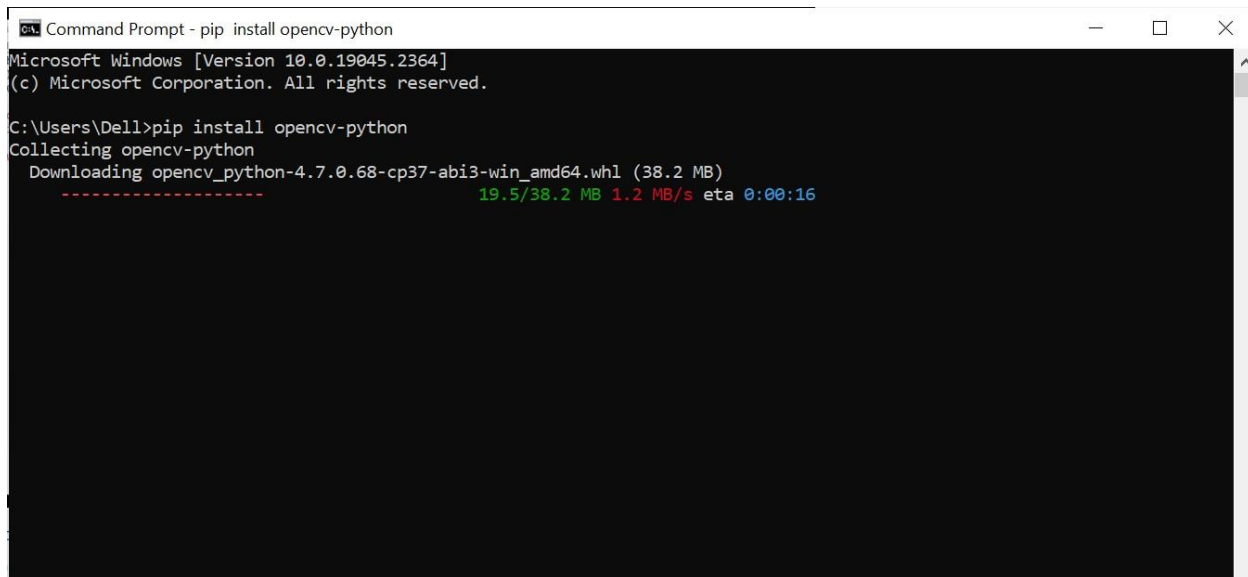
Imutils: this library is used to carry out series of image processing functions such as rotation, resizing. Imutils is used in this project for resizing out the frame and specifying the width of frame tat is 50.

Datetime: this library is used for giving the current date and time .

Languages: The Language used to built the project is python and the IDE used for it is PyCharm and IDLE Python 3.11 (64-bit).

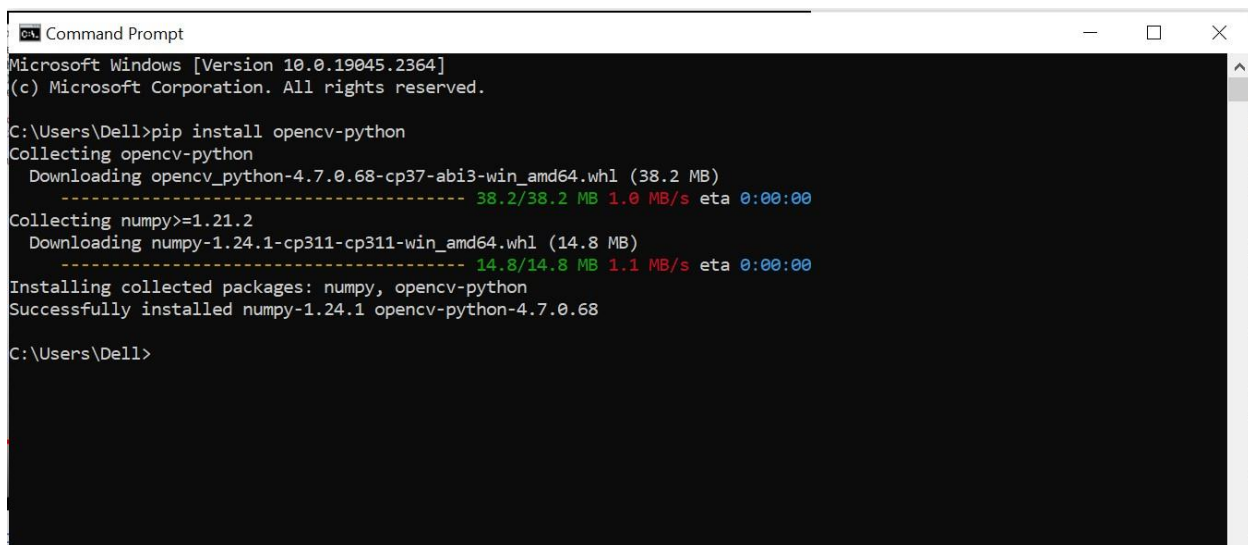
Packages: the PyCharm 3.11 come with pre-installed packages but the OpenCv for importing cv2 and Imutiles packages need to be installed to run image processing techniques in the project. The scree shots of installation is given.

Installing OpenCV Library



```
Command Prompt - pip install opencv-python
Microsoft Windows [Version 10.0.19045.2364]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Dell>pip install opencv-python
Collecting opencv-python
  Downloading opencv_python-4.7.0.68-cp37-abi3-win_amd64.whl (38.2 MB)
    ----- 19.5/38.2 MB 1.2 MB/s eta 0:00:16
```



```
Command Prompt
Microsoft Windows [Version 10.0.19045.2364]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Dell>pip install opencv-python
Collecting opencv-python
  Downloading opencv_python-4.7.0.68-cp37-abi3-win_amd64.whl (38.2 MB)
    ----- 38.2/38.2 MB 1.0 MB/s eta 0:00:00
Collecting numpy>=1.21.2
  Downloading numpy-1.24.1-cp311-cp311-win_amd64.whl (14.8 MB)
    ----- 14.8/14.8 MB 1.1 MB/s eta 0:00:00
Installing collected packages: numpy, opencv-python
Successfully installed numpy-1.24.1 opencv-python-4.7.0.68

C:\Users\Dell>
```

```
*IDLE Shell 3.11.1*
File Edit Shell Debug Options Window Help
Python 3.11.1 (tags/v3.11.1:a7a450f, Dec 6 2022, 19:58:39) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> import cv2
Traceback (most recent call last):
  File "<pyshell#0>", line 1, in <module>
    import cv2
ModuleNotFoundError: No module named 'cv2'
>>> import cv2
>>> print(cv2.__version__)
4.7.0
>>>
```

Installing Imutiles Library

```
Command Prompt
(a)bort.
--trusted-host <hostname> Mark this host or host:port pair as trusted, even though it does not have valid or any
                        HTTPS.
--cert <path> Path to PEM-encoded CA certificate bundle. If provided, overrides the default. See 'SSL
Certificate Verification' in pip documentation for more information.
--client-cert <path> Path to SSL client certificate, a single file containing the private key and the
certificate in PEM format.
--cache-dir <dir> Store the cache data in <dir>.
--no-cache-dir Disable the cache.
--disable-pip-version-check Don't periodically check PyPI to determine whether a new version of pip is available for
download. Implied with --no-index.
--no-color Suppress colored output.
--no-python-version-warning Silence deprecation warnings for upcoming unsupported Python versions.
--use-feature <feature> Enable new functionality, that may be backward incompatible.
--use-deprecated <feature> Enable deprecated functionality, that will be removed in the future.

C:\Users\Dell>pip3 install imutls
Collecting imutls
  Downloading imutls-0.5.4.tar.gz (17 kB)
  Preparing metadata (setup.py) ... done
Installing collected packages: imutls
  DEPRECATION: imutls is being installed using the legacy 'setup.py install' method, because it does not have a 'pyproj
ect.toml' and the 'wheel' package is not installed. pip 23.1 will enforce this behaviour change. A possible replacement
is to enable the '--use-pep517' option. Discussion can be found at https://github.com/pypa/pip/issues/8559
  Running setup.py install for imutls ... done
Successfully installed imutls-0.5.4

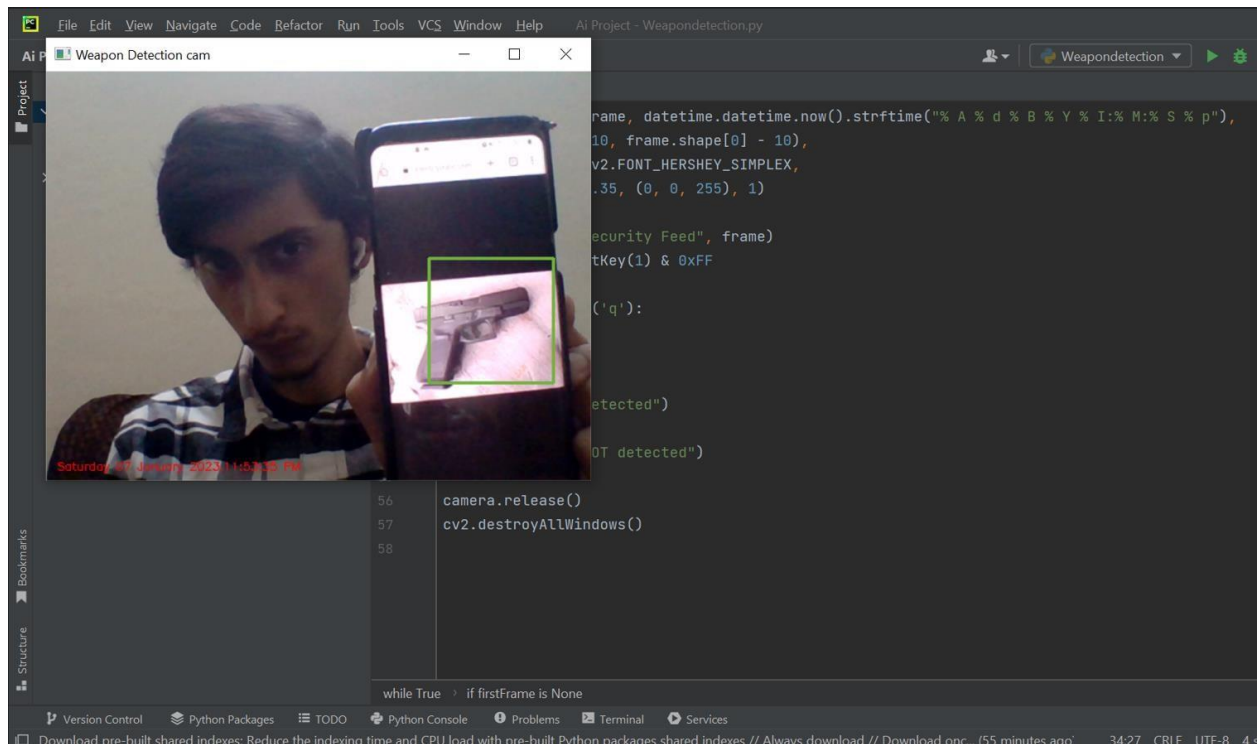
C:\Users\Dell>
```

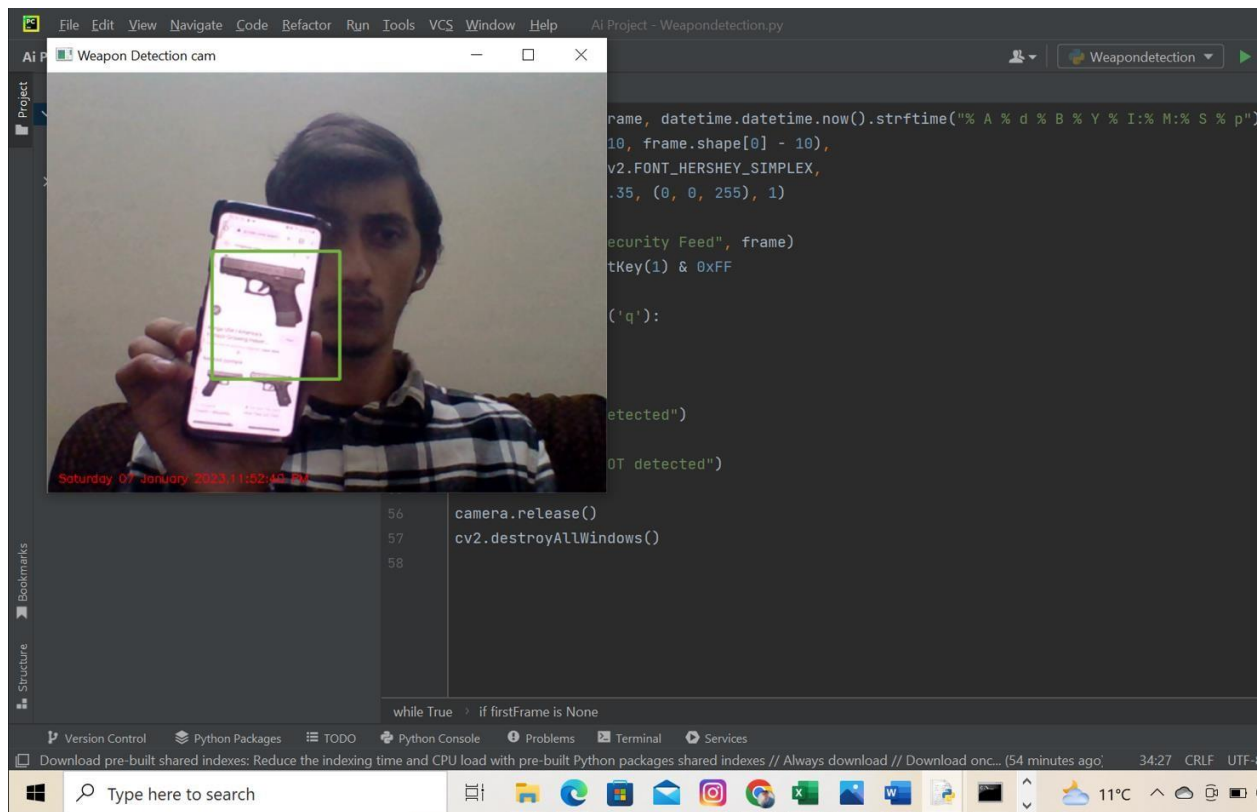
Conclusion: the project is able to detect the hand gun through the videocam of the laptop in real time with the help of object imported of libraries and conclude whether the gun is detected or not. The project can be improved as for now it detects the hand gun from the images in future it can be updated to detect all

types of guns regardless of the environment, a special type of application will be available that can be installed with in the security cams of the houses, offices, banks that will enhance the security level of that area / place.

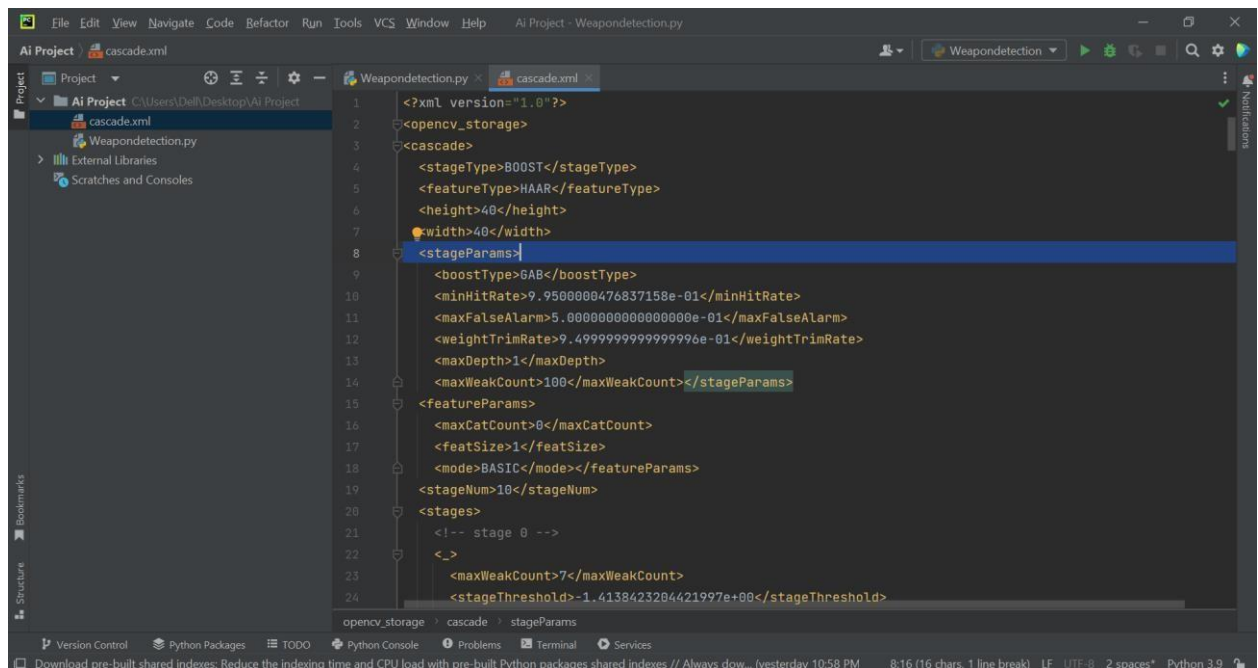
The Actual Output:

The Guns can be detected on different Displays of devices and regardless of a same picture.





Cascade File:



Actual Code:

```
import numpy as np
import cv2
import imutils
import datetime

gun_cascade = cv2.CascadeClassifier('cascade.xml')
camera = cv2.VideoCapture(0)
firstFrame = None
gun_exist = False
while True:

    ret, frame = camera.read()
    frame = imutils.resize(frame, width=500)
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    gun = gun_cascade.detectMultiScale(gray,
    1.3, 5,
    minSize=(100, 100))

    if len(gun) > 0:
        gun_exist = True
        for (x, y, w, h) in gun:
            frame = cv2.rectangle(frame,
            (x, y),
            (x + w, y + h),
            (255, 0, 0), 2)
            roi_gray = gray[y:y + h,
            x:x + w]
            roi_color = frame[y:y + h, x:x + w]

            if firstFrame is None:
                firstFrame = gray
                continue

            # print(datetime.date(2019))
            # draw the text and timestamp on the frame
            cv2.putText(frame, datetime.datetime.now().strftime("% A % d % B % Y % I:% M:% S % p"),
            (10, frame.shape[0] - 10),
            cv2.FONT_HERSHEY_SIMPLEX,
            0.35, (0, 0, 255), 1)
            cv2.imshow("Weapon Detection cam", frame)
            key = cv2.waitKey(1) & 0xFF
            if key == ord('q'):
                break

            if gun_exist:
                print("guns detected")
```



```
else:  
    print("guns NOT detected")  
  
camera.release()  
cv2.destroyAllWindows()
```