TRACING PERSON USING MACHINE LEARNING TECHNIQUE IN CLOUD ENVIRONMENT

Shourya Singh, Vivek Kumar Maddeshiya, Pranav Goel, Sai Venkat Raparthi

Abstract— Every day, a large number of people go missing around the world, including children, teenagers, mentally challenged people, elderly people with Alzheimer's disease, and so on. The majority of them have gone unnoticed. This paper proposes a system that would benefit both the police and the general public by speeding up the search process by using face recognition (using Ultralight Model or Yolo face). In this project, we are going to trace the subject using machine learning techniques in the cloud environment. Model that we are going to build is using machine learning techniques like KNN/CNN. Model will be trained based on given pictures of the subject and the trained model will be deployed on cloud, where it will take input from different CCTV cameras and will check each frame of footage of different cameras using multithreading. As Model will run on multimode at a time so it is going to use a lot of system resources as of now it is not possible to deploy model from Any Personal Computers so a server based on cloud will be useful to run the task smoothly.

Index Terms— Alzheimer, KNN, CNN, Multithreading, Ultralight model, Yolo face, Face recognition, Cloud environment, Multimode,ONNX

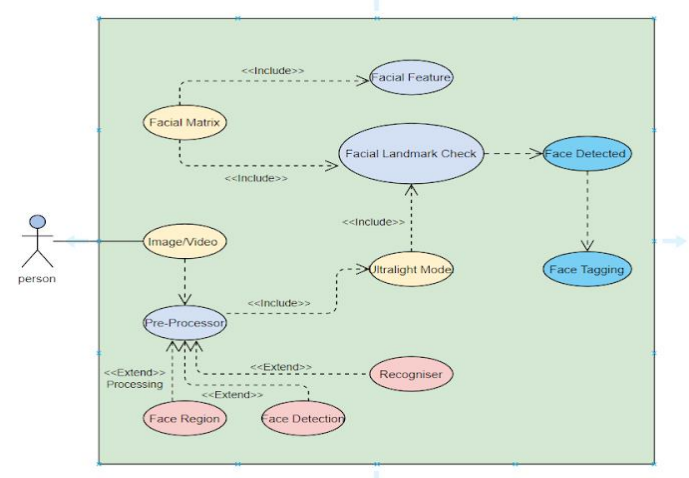
1 INTRODUCTION

Facial recognition is an area of research and development which deals with giving machines the ability to recognize and verify human faces.

Generally, there are three steps taken to recognize a face [1]:

1) Data pre-processing
2) Facial feature extraction
3) Comparison of features between the target face and face from the database.

This project mainly focuses on tracing the person by searching for the provided input face dataset into each frame of CCTV footage by using UltraLight model of face recognition in the cloud environment. The provided dataWe would be trying to extend this system further by connecting our system to public cameras and detecting faces real-time. The frames will be continuously sent by the public cameras to our system where our system will be continually monitoring the frames. When a lost person is identified in any of the frames, it will be notified to the concerned authorities. The data set of the lost person will be used for model training and by processing each frame of the CCTV footage, it will search for the person. This project is able to process multiple streams of input by using threading, class and object concept.



Objective of this project is to maximize the efficiency of the police department by Finding/Tracing the subject using processing of live CCTV footage [5] and reducing the workload of officers as they will not have to search for personnel on their own. The Trained model and designed system will automatically examine each frame of footage and will return the position of cameras where it has found that person.

This model can be implemented in Smart cities for making it more secure and it will help to reduce the crime scenes too. The frames will be continuously sent by the public cameras to our system where our system will be continually monitoring the frames. When a person is identified in any of the frames, it will be notified to the concerned authorities.

Test results of multiple face detection systems have been used to find the best suitable algorithmic model for this case, where in fraction of time we have to detect the maximum number of faces and have to process to match the face from the dataset.

1.1 EXISTING SYSTEM

Pournami S. Chandran and his colleagues at the Centre for Development of Advanced Computing previously presented a work [9] that addressed a similar issue definition and goal. They present a system that uses Deep Learning-based Facial Feature Extraction and SVM matching (Support Vector Machine) The photographs of missing children are kept in a database. Faces are recognised in the photos, and a Convolutional Neural Network is used to learn features. A multi-class SVM classifier was trained using the learned features. This procedure was utilised to appropriately identify and label the child. and also in kenya mobile based face tracing system was proposed [2]

1.2 PROPOSED SYSTEM

Our proposed approach focuses on tracing any person who is missing or has been a criminal while being monitored by CCTV cameras across the city. Our system is a real-time application that will assist the police and investigation teams in quickly locating a person.

Our system is based on the Ultralight model-ONNX [4] method, which has a high efficiency of detecting Faces in a short

amount of time, i.e. it is able to detect roughly 80 photos in 0.102 seconds, which is a very high no. of face detected in such small span of time duration. This algorithm is one of the system's greatest assets, as it creates a very efficient working project.

1.3 Technical and Social Feasibility

The current technology allows for the construction and realisation of this project. The ultralight model in Python is one of the most efficient and dependable algorithms based on ONNX- on which our project is operating and getting the optimal needed results. Making the system for the end user, HTML/CSS is used to create a website as frontend of this system. Flask has been used to link the web frontend to our program, which is a fast approach to connect through the database and program. As we know there are a lot of pending cases of missing persons[7] and wanted criminals, and using old traditional methods it is not efficient to solve such cases in a short span of time and having a lot of dependencies, we have to rely on people to get information as authorized teams can't be present everywhere at the same time. So, this project is going to help by searching for a wanted or missing person in all CCTV footage and will respond back when it detects the person. Using this system we will be able to create a safe and secure environment for civilians, as this project is going to help finding lost people and also it can help in monitoring the activity of people having crime records in any investigation.

2 Module Description

Some forms or methods that are used to process this project "Tracing Person using Machine Learning in Cloud Environment" to make it a user end application.

- Ultralight Model
- Flask
- HTML/CSS/ PHP

2.1 Ultralight Model

The Ultralight model[4] is also known as LightWeight Face Detection Process. Despite the fact that mobile devices were not meant to run compute intensive AI models, AI-powered capabilities such as face identification, eye tracking, and speech recognition have been added to smartphones in recent years. Although much of the computation for such services is done in the cloud, these applications should be small enough to run on devices without an Internet connection in the perfect world.

Index	Method	No. of Faces	Time(s)
i	Face_recognition_cnn	70	108.499
ii	Face_recognition_hog	74	0.76
iii	Mtcnn	123	2.209
iv	Yolo face	129	1.053
V	**Ultra light**	**80**	**0.102**

2.2 Flask

Flask is a WSGI web application framework that is lightweight. It's built to make getting started simple and quick, with the flexibility to scale up to more sophisticated projects. It started off as a basic wrapper for Werkzeug and Jinja and has since grown into one of the most popular Python web application frameworks.

In this Project, Flask is used as a medium to connect the web interface to Python modules. Later this can be uploaded on Heroku- PaaS cloud service for public use.

2.3 HTML/CSS

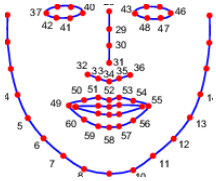
HTML stands for HyperText Markup Language. For designing web pages it is used. It is basically the combination of Hypertext Markup Language. It defines the link between the web pages. It helps to structure our website well. Knowledge of HTML is a must for the Web Development Domain. With the help of HTML we can build simple, static websites very easily.The term "hypertext" refers to links that connect online pages on sides of a single website or between websites. Links are an important part of the Internet. You become an active participant in the World Wide Web by uploading content to the Internet and linking it to other people's pages.

2.4 Working

The model we have used involves four main steps to perform face recognition.

STEP 1: Face detection – First, the Histogram of Oriented Gradients (HOG) algorithm is used to create face patterns.The photographs have been converted to black and white. This is where you'll find the part of the picture that looks the most like the original HOG face pattern. Finally, a bounding box is drawn around the detected face.

STEP 2: The face landmark[3] estimation algorithm is used to determine sixty-eight unique points (landmarks) that exist on every face. The OpenCV's affine transformation uses image transformations including scaling, shearing, and rotation to make the lips and eyes appear in the same position on any image based on the landmarks found.



STEP 3: The images of the face are then fed into a deep convolutional neural network and UltraLight Model. This yields 128 measurements, resulting in a 128-dimensional hypersphere. Nobody knows which parts of the face are represented by the 128 measurements [8]. All we know is that for two separate images of the same human, the network generates the same 128 numbers.

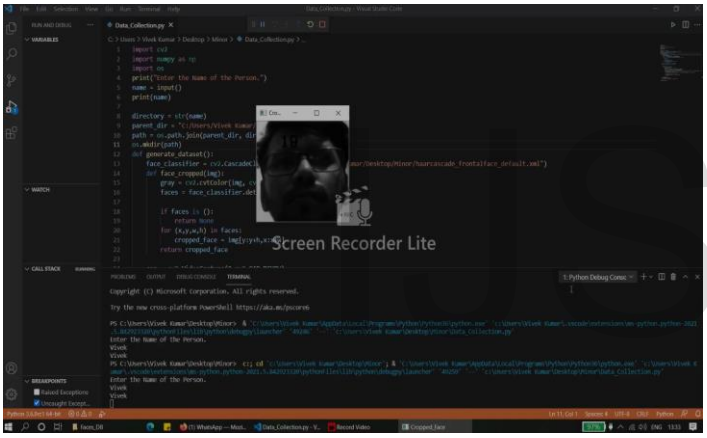
STEP 4: Finally,after preprocessing the image, we will have to prepare the ONNX model and create an ONNX inference session for the face.And then the output is given as matched or not, depending on the ONNX report.

3 IMPLEMENTATION

This Project was designed to search the person and track the mobility of that person. The input provided to the system is either the videos or Multiple photos and Name (ID) of the Person through the provided user interface by web page. The Program processes the inputs and creates a directory to a specific location and stores the given input video of the image file to that directory. Then the processing takes place and the face data is extracted through input files and embedded to the pickle file. The Output is provided in form of the text stating the Camera location and Person name found. Using a webcam view the person can be seen having a face outlined with a red box and showing the Name of the Detected Person.

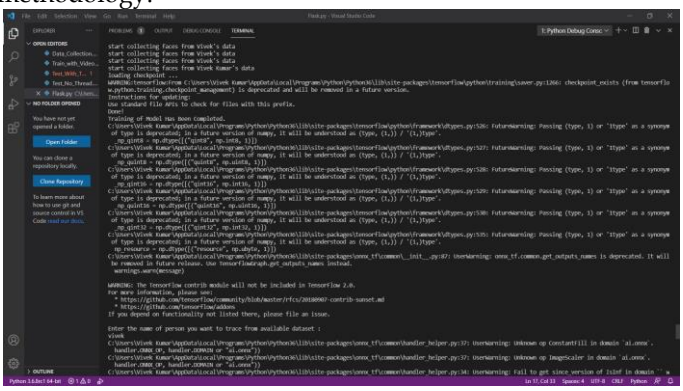
3.1 Data Collection

The initial phase in our research is data gathering, which can be done through a video or by providing photos. The python programme is written in such a way that it takes a video as input.



3.2 Training the Model

Following the creation of the dataset, the next stage is to train the system using it. The software is written in Python, which allows the model to read the dataset and mark the faces with sixty-eight unique points using the face landmark estimation methodology.



3.3 The Main Model

This is the project's main framework. The input is obtained through the use of various cameras. Threading concept in the project application is used to connect multiple cameras. Following the connection, the input is collected and the maximum number of faces is detected by using the ultralight model, the face extraction is now compared to the dataset by comparing all of the unique facial points in both images. When a match is found, the output is either the person who was spotted at this camera or an unknown person whose data isn't in the dataset.

```
import cv2
import numpy as np
print("Enter the Name of the Person.")
name = input()
print(name)
def generate_dataset():
    face_classifier = cv2.CascadeClassifier("C:/Users/Vivek Kumar/Desktop/Minor/haarcascade_frontalface_default.xml")
    def face_cropped(img):
        gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
        faces = face_classifier.detectMultiScale(gray, 1.3, 5)

        if faces is ():
            return None
        for (x,y,w,h) in faces:
            cropped_face = img[y:y+h,x:x+w]
        return cropped_face

    cap = cv2.VideoCapture(0,cv2.CAP_DSHOW)
    img_id = 0 #id number start hoga

    while True:
        ret, frame = cap.read()
        if face_cropped(frame) is not None:
            img_id+=1
            face = cv2.resize(face_cropped(frame), (200,200))
            face = cv2.cvtColor(face, cv2.COLOR_BGR2GRAY)
            file_name_path = "C:/Users/Vivek Kumar/Desktop/Minor/Faces_DB/Face_Image_DB/"+ name +str(img_id)+".jpg"
            cv2.imwrite(file_name_path, face)
            cv2.putText(face, str(img_id), (50,50), cv2.FONT_HERSHEY_COMPLEX, 1, (0,255,0), 2 )

            cv2.imshow("Cropped_Face", face)
            if cv2.waitKey(1)==13 or int(img_id)==250:
                break

    cap.release()
    cv2.destroyAllWindows()
    print("Collecting samples is completed !!!")

generate_dataset()
```

3.4 Creating the Web Page

Now we've created an end-user application in the form of a webpage in HTML/CSS. This website's main function would be to enter the name of the individual we need to trace around our cameras. The video or photographs of the person in question is then uploaded. The purpose of this website is to make it as simple as possible for the user to use this project

3.5 Flask

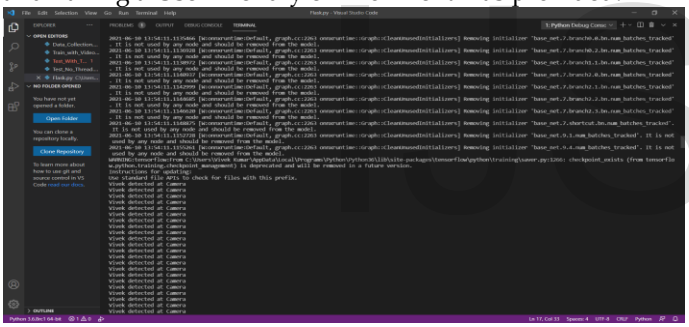
Flask is a popular Python web framework, which is a Python library that is used to create web applications.We're using this to link our Python module to a website. So the name and video that are entered into the website may be processed by the python module, and all of the work is completed and the required output is obtained.

4 TESTING THE MODEL

The system's requirements and specifications are assessed first. To see if SUT processes valid inputs successfully, the tester selects legitimate inputs (positive test scenario). In addition, certain faulty inputs are chosen (negative test scenario) to ensure that the SUT can detect them. For each of those inputs, the tester determines predicted outcomes. With the chosen inputs, the software tester creates test cases. The test cases are carried out. The actual outputs are compared to the expected outputs by the software tester. If there are any flaws, they are corrected and retested. Black Box testing was conducted to test the real time implementation of this project.

4.1 Testing Strategy

For testing, black box testing was performed, in that the project was given to test to users without having knowledge on internal functionality. User runs the program and with the help of the user interface provided, it was very easy to go with the flow. after submitting the name. The web page was shown immediately through which the user was able to submit multiple or single video files. In correspondence to that the webcam opens and starts detecting the faces and recognising the faces detected. The output was observed and checked by the user, that the program was working smoothly and efficiently and having a User friendly environment was provided.



5 RESULTS & DECISIONS

5.1 Efficiency of the Proposed System

Day by day we are using modern technology to improve our various systems and their functionality. Implementing this system in the police department is going to help them a lot to trace someone by using their face identity. With the use of this system searching people is going to be so easy and efficient. This is a very efficient way to search for any person using his face identity with the help of this supervised machine learning neural network model[8]. This system uses footage of cctv cameras pre installed to the various locations and searches for the given face identity. In comparison to the traditional method to find any person that takes a lot of time and effort, this system takes minimal efforts and sources to do the task.

5.2 Comparison of Existing and Proposed System

Existing system for detecting the face using the MTCNN/CNN method is less efficient and time consuming.In a comparison test, Face Recognition CNN was able to detect only 70 faces in 108 second and Face Recognition, MTCNN was able to detect 123 faces in 3 seconds. but the ultralight onnx model can detect 80 faces in a minute time of 0.1 second. The Existing System of finding any person using traditional methods by pasting posters and offering rewards, is too complex, time-consuming and a less efficient technique. Using modern tech to overcome this problem by implementing the proposed system makes the process very easy and effective as you will have to provide only the photos and videos of the person with name and trained model can search for that face over all cities using a network of CCTV cameras.

5.3 Advantages of proposed system

This system helps to find the lost person/suspect using his face identity and can cover a large area in such a less amount of time. This project involves a network of CCTV spreaded all over the city and hence no manforce is required to find anyone in between areas covered by the cameras network.

6 REQUIREMENTS AND SAMPLE CODE

6.1 Requirements

- Heroku
- Python 3.6 or above
- Flask
- Python libraries: CV2, dlib, numpy, tensorflow 1.14, tqdm, os, tflearn, pillow, random.
- 64 bit-Operating System
- Window 10
- HTML5
- CSS
- SQL
- Any Latest Browser

6.2 Sample Code

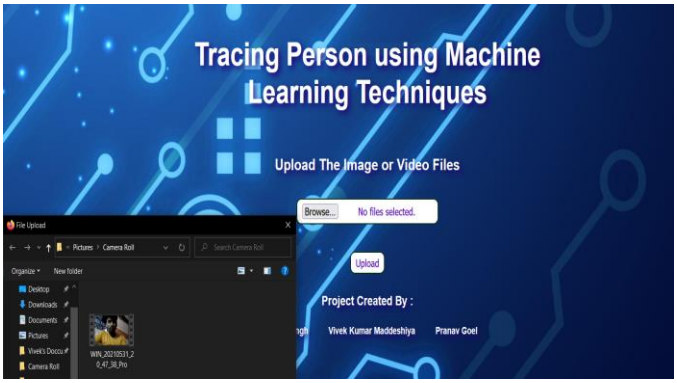
Full implementation of code can be found at https://github.com/vivekkumar3108/Minor_Project

```
import os.path
from werkzeug.utils import secure_filename
import os
from flask import Flask, flash, request, redirect, render_template
arr=[]
app = Flask(__name__, template_folder='template')
app.secret_key = "secret key"
APP_ROOT = "C:/Users/Vivek Kumar/Desktop/Minor/Faces_DB/"


@app.route("/")
def index():
    return render_template("Create.ht
ml")

@app.route('/', methods =["GET", "POST"])
def gfg():
    if request.method == "POST":
        p = request.form.get("name")
        p= str(p)
        arr.append(p)
        parent_dir = "C:/Users/Vivek Kumar/Desktop/Minor/Faces_DB"
        path = parent_dir+'/'+ p
        path=path.replace(" ","")
        path=path.replace("VivekKumar","Vivek Kumar")
        os.mkdir(str(path))
    return render_template("Upload.html")
```

6.1 OUTPUT



7 CONCLUSION AND FUTURE ENHANCEMENTS

7.1 Conclusion

So from this work, we can conclude that the ultralight onnx model is highly efficient and fast to detect the maximum number of faces in a given frame in a fraction of second and hence can be used with even high fps cameras to increase the working efficiency. Ultralight neural network models have already been trained for detecting faces in less time using the WIDER FACE dataset. So the ultralight model is the best model to be used for this task as within fraction of seconds it can detect max faces as possible despite being of less weight in terms of memory. and this project is going to help in solving cases of missing people and tracing the criminal or wanted persons and will help to reduce crime scene.

7.2 Future Enhancements

This project is of high future scope and with modification we will be able to increase its working efficiency and can be used in real time scenarios. In future we can upload this model on the cloud services such as heroku, AWS etc. and can be accessed from anywhere. A network of cameras from all over the area can be built and can be connected to this system using cloud and all the processing can be done on cloud to save physical resources. By making various instances of the model class we will be able to process multiple video streams at a single time and will be able to get results at the user's fingertips. This model deployed on the cloud will be able to work faster than in comparison to the computer hardwares of limited resources

ACKNOWLEDGMENT

We Shourya Singh, Vivek Kumar Maddeshiya, Pranav Goel, Sai Venkat Raparthi as the authors of this paper wish to thank our project guide Mr. A.S. Syed Fiaz and School of Computing of VelTech RR. Dr. SR. R&D Institute of Science and Technology. This Project work was supported in part as a minor project by a grant from the Department of Computer Science and Engineering, VelTech.

REFERENCES

[1] Gagandeep Singh, Find Missing People using Facial Recognition, International Journal of Engineering and Advanced Technology (IJEAT)

[2] Thomas M. Omweri, Andrew M. Kahonge. Using a Mobile Based Web Service to Search for Missing People – A Case Study Of Kenya . International Journal of Computer Applications Technology and Research Volume 4– Issue 7,2015

[3] Shengcai Liao,Anil K. Jain,Stan Z. L.A Fast and Accurate Unconstrained Face Detector.IEEE Transactions on Pattern Analysis and Machine Intelligence(Volume: 38, Issue: 2, Feb. 1 2016)

[4] https://github.com/Linzaer/Ultra-Light-Fast-Generic-Face-Detector-1Mb -Linzaer

[5] Swapnil Vitthal Tathe*Abhilasha Sandipan Narote , Sandipan Pralhad Narote. Face Recognition and Tracking in Videos.Advances in Science, Technology and Engineering Systems Journal Vol. 2, No. 3, 1238-1244 (2017)

[6] Nikhil Chug Samarth Mehrotra Sameya Alam Anand Gaurav Girish.N. Real Time Face Parsing Using Enhanced KNN and DLIB. International Journal of Innovative Science and Research Technology,Volume 5,Issue 2, February – 2020 77

[7] Fahad Parvez Mahdia,, Md. Mahmudul Habibb, Md. Atiqur Rahman Ahadb, Susan Mckeeverc, A.S.M. Moslehuddinband Pandian Vasant.Face recognition-based real-time system for surveillance.Intelligent Decision Technologies 11 (2017) 79–92 79 DOI 10.3233/IDT-160279

[8] Bharath Darshan Balar, D S Kavya, Chandana M, Anush E, Vishwanath R Hulipalled , May 2019, Efficient Face Recognition System for Identifying Lost People International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249-8958,Volume-8, Issue-5S,

[9] Efficient Face Recognition System for Identifying Lost People ,Bharath Darshan Balar, D S Kavya, Chandana M, Anush E, Vishwanath R Hulipalled, IJEAT ISSN: 2249 – 8958,Volume-8, Issue-5S, May 2019