**Team Name:**

**Interrupt\_Zero1**

**Title of problem statement:**

##### Facial Recognition to map **Find the photo** of a missing person, from police database

**Names of Team Members:**

**Nishanth D Aluhonnu , Rajat Keshri , Mahadev Mahesh Maitri , Premal singh**

**Email ID of team lead:** [**nishanthdilip@yahoo.co.in**](mailto:nishanthdilip@yahoo.co.in)

**Phone Nos. of team lead: 9036490565**

##### Facial Recognition to map ****Find the photo**** of a missing person, from police database and applying OSINT analysis

1. **Problem Statement/Objective:**

* Sometimes, you need to check if the person who just sent you a friend request on Facebook or LinkedIn is the real person or if that person is using a fake profile photo.
* In sourcing, a reverse image search can be quite handy when you are trying to find more information about someone. For example, when you are trying to find an email address for someone, and it’s not mentioned on one social profile, it could be on another website.
* Because many people use the same profile photo on various social sites, a reverse image search is a handy trick to find other social media profiles.

1. **Proposed Solution:**

Our solution to the problem is to implement **face recognition and extracting unique features from the single image** given as an input. The features will be used to search the **police database, social media websites and Open source databases** like google dorks to find the person.

This person’s other data will then be extracted **using OSINT** methods and data analysis will be done on it. Depending on the received data after facial recognition, we map to criminal info and check if he is a previous offender.

A ML model for facial recognition will be created, which is trained to extract the **right features** from just one of two images of the person.

We would also try possibility of recognizing faces by removing moustache beard and create a model to recognize the person even in disguise.

**Procedure:**

There are different steps involved in the solution.

General algorithm-

Create the **facial recognition model**, which learns to extract unique features for any input image with a face in it. This model extracts the features and stores it in a data structure which then tries to map it onto a new image produced for perfect matching

Apply **facial recognition model** on different websites such as facebook, Instagram, twitter, linkedin to find a match. This done by automating the social media search crawler using the website APIs

Simultaneously, open sources of cctv footage, open source intelligence websites, open source databases and google images are scanned for the person’s face in real time, using OSINT techniques

Once name, number, email or any other relevant information is found, using the reverse image search, public OSINT sources or databases are used, with the help of API keys to get the entire report of the person

**Approach**

Creation of the machine learning model:

First step is to detect whether a face is present in the image or not. We first convert the image to black and white or gray scale.

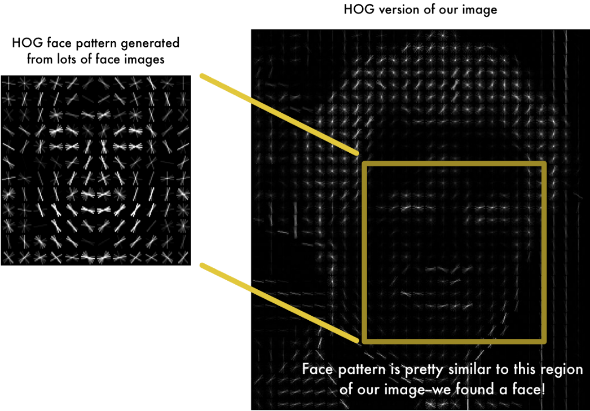
Here now every pixel is of a gradient between 0 to 256.

Next, every pixel is surround by either darker or lighter pixel gradient densities.

Here, for every pixel, we create an arrow mark pointing towards the darker pixel.

We take a window of 64x 64 pixels and compare the density of the 64x64 block with its neighbouring block of pixels, to draw an arrow mark from lighter gradient to darker gradient.

Doing this for the entire image will produce many arrow marks. The end result is we turn the original image into a very simple representation that captures the basic structure of a face in a simple way. This way faces can be easily detected



The next step is to encode the faces and extract features

Feature extraction is done by training deep neural network. Here we extract features by measuring pixel distances between nose and eyes and ears.

But instead of training the network to recognize pictures objects like we did last time, we are going to train it to generate 128 measurements for each face. The training process works by looking at 3 face images at a time:

1. Load a training face image of a known person
2. Load another picture of the same known person
3. Load a picture of a totally different person

Then the algorithm looks at the measurements it is currently generating for each of those three images. It then tweaks the neural network slightly so that it makes sure the measurements it generates for #1 and #2 are slightly closer while making sure the measurements for #2 and #3 are slightly further apart.

A 128 array of embedding will be produced for every image we train and create a generalised model.

Now whenever a new image is put as an input, the 128 length array embedding will be generated and tried to match with the existing image’s embedding hence predict with facial recognition

As the model is trained depending on **the distance of the facial features** with each other, **even if a person is in disguise** with beard or moustache or different hairstyle, the model will **predict accurately**.

Social website crawler:

Using the **Open Source Intelligence**, and **our facial recognition model**, we correlate social media profiles across different sites on a large scale. **Using an automated approach to search popular** social media sites for targets' names and pictures to accurately detect and group a person’s presence, outputting the results into report that a human operator can quickly review.

Using the crawler, an automated gathering of large amounts of social media profiles for use on targeted phishing campaigns will be done. Facial recognition aids this process by removing false positives in the search results, so that reviewing this data is quicker for a human operator.

Every website’s **API keys** will be used to access the website using python, and images will be searched.

Apart from just searching for images, **other details like gender, age** etc will be given as input for faster search

Basically a webs crapper using facial recognition will be implemented.

OSINT search:

An OSINT search engine will be created using web APIs

This search engine will search for the particular culprit’s records, based on his/her name, or social media username etc.

The search engine will use open source data.

Websites like – shodan, Clearbit, Fullcontact, Virus\_Total, EmailHunter will be used to hunt the details.

API keys generated by each website will be used to make the OSINT search engine crawler.

1. **High Level Design (with diagram):**

Saving the model as pickle or h5

Training with Deep learning CNN

Feature extraction on images

Back end Front end

Camera

Input image

Image processing

Web API

Classification using facial recognition model

Using the model on police database and social media websites to search for the username/ name of culprit

Extracting the name and running through the OSINT search engine to generate a report