**Face Recognition based Attendance System**

A model trained to detect faces of students in a photo and mark their attendance in database.

**Main Phases in the Development**

**Data Collection**

* We have collected images of different persons about 2-4 solo images of students.
* The images can be of any type(jpeg , png or any other) and size.

**Data Pre-processing:**

* The collected images are passed through MTCNN detector For face detection then the size of photos are changed to (160,160).
* The categorical values i.e name of students are converted into numerical form. This is done using Label Encoder.
* The image array is normalized before training our classification model using Normalizer.
* The collected images are splitted into two sets known as test and train sets using sklearn.90% of examples are taken in train set and 10% are in test set.

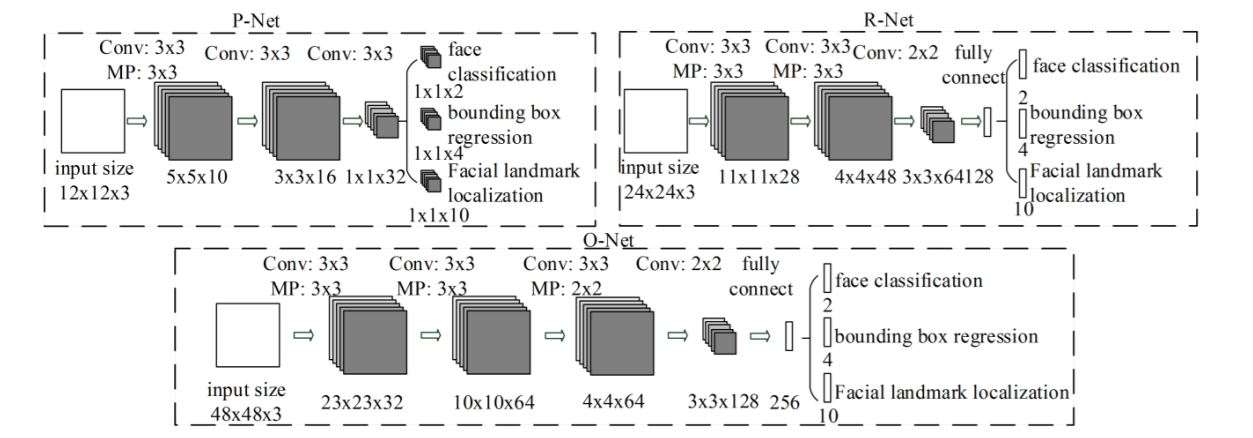
**Model Architecture**

**We have used three different models:**

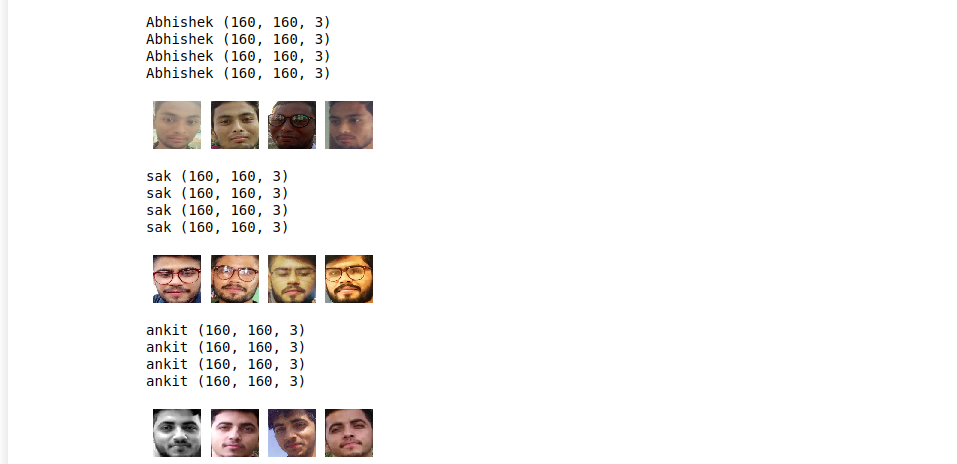
* + **MTCNN**
  + **Facenet**
  + **SVM**

**Multi-task Cascade Convolutional Neural Network**

* Used a pre-trained MTCNN model to detect faces of students.The architecture of model is defined below.
* This model has three convolutional networks (P-Net, R-Net, and O-Net) and is able to outperform many face-detection benchmarks while retaining real-time performance

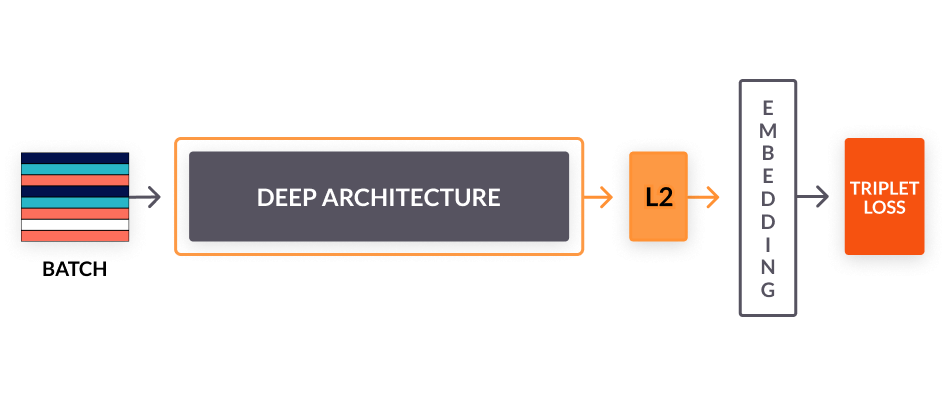


* The model is called a multi-task network because each of the three models in the cascade (P-Net, R-Net and O-Net) are trained on three tasks make**three types of predictions**; they are: **face classification, bounding box regression, and facial landmark localization.**
* Three models are not directly connected and act as a lap of a sprint race. When one round finishes, the next one starts and so on until the 3rd lap finished.
* Non-maximum suppression (**NMS**) is also used to remove the detected faces of lower confidence which increses exraction accuracy **between stages** boxes proposed by the first-stage P-Net prior to providing them to the second stage R-Net model and Finally to the O-net model
* The real time example of face extraction from photos.

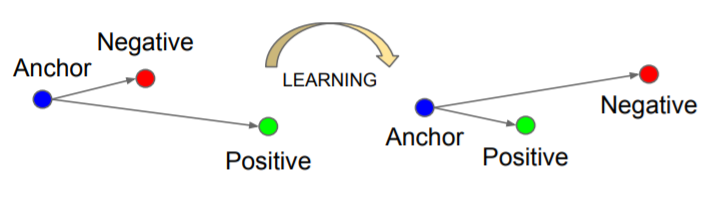


**Google’s Face Net Model:**

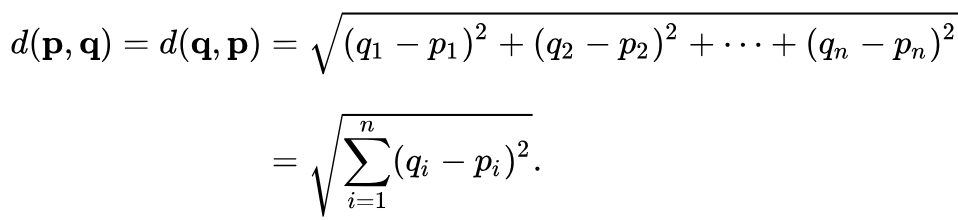
* FaceNet is a start-of-art face recognition, verification and clustering neural network. FaceNet is introduced in 2015 by Google researchers.
* The main purpose of using this model is to convert image into embeddings which can be further used for classification.
* The simplified architecture of Facenet Model



* It transforms the face into 128D Euclidian space similar to word embedding.
* Once the FaceNet model having been trained with triplet loss for different classes of faces to capture the similarities and differences between them, the 128 dimensional embedding returned by the FaceNet model can be used to clusters faces effectively.
* Once such a vector space(embedding) is created, tasks such as face recognition, verification and clustering can be easily implemented using standard techniques with FaceNet embeddings as feature vectors.
* In a way, distance would be closer for similar faces and further away for non-similar faces.
* The paper describes that network consists of a batch input layer and a deep CNN followed by L2 normalization, which results in the face embedding. This is followed by the triplet loss during training.

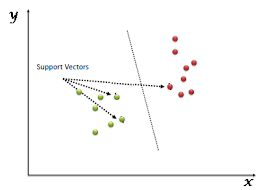


* One the FaceNet model is trained, we can create the embedding for the face by feeding into the model. In order to compare two images, create the embedding for both images by feeding through the model separately.
* Then we can use below formula to find the distance which will be lower value for similar faces and higher value for different face.

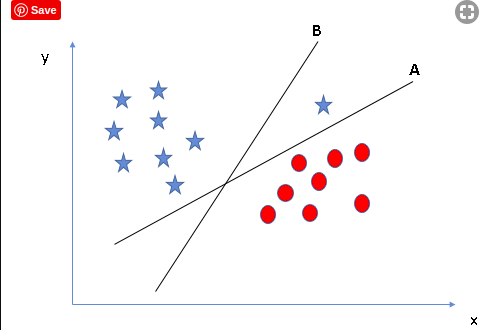


**Support vector Machine**

* We are using this model for classification purpose i.e to classify face embedding into different categories.
* Support Vector Machine” (SVM) is a supervised [machine learning algorithm](https://courses.analyticsvidhya.com/courses/introduction-to-data-science-2?utm_source=blog&utm_medium=understandingsupportvectormachinearticle) which can be used for both classification or regression challenges.



* However,  it is mostly used in classification problems. In this algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate.
* Then, we perform classification by finding the hyper-plane that differentiates the different classes very well.



* After converting an image to face embeddings the model is trained to predict the name of students in the photo.
* The model is trained on training dataset and achieved an accuracy of more than 98%

**Deployment Phase:**

* The model is deployed on Local Server using Python framework known as flask.

It consist of mainly three pages:

* Login page : For faculty login
* Upload page: To upload Photo of students
* Prediction Page: To display the name of students whose attendance has been marked.

**Database:**

* Sql3 has been used with the help of python.

Mainly two tables are created:

* Faculty Table: This table consist of name and password of different faculties along with their password.
* Student Table: This table consist of name,roll no,password of different studets with initially their attendance marked as ‘A’ Which indicates Absent.
* The student table attendance field gets updated according to prediction made by our model.

**Technologies Used**

* Python
* Sql
* HTML
* CSS
* Javascript
* Flask

**Software Requirements To Run Project**

* Python 3.7 with following Packages
  + - tensorflow==2.0.0
    - numpy==1.17.3
    - Flask==1.1.1
    - Keras==2.2.5
    - mtcnn==0.0.9
    - Pillow==6.2.1
    - scikit-learn==0.21.3
    - opencv-contrib-python==4.1.1.26
    - Pickle
    - Sql3
* A web browser(Google Chrome Preferred)

**Hardware requirements**

* Processor:i3 or higher
* Graphic: Intel Integrated(Dedicated Graphics recommended)
* Ram:4 Gb or higher
* Clock Speed: 1.6Ghz or Higher