Mass Real Time tracking in CCTV

Project Supervisor:
Prof. Bhogeswar Borah

Submitted by:
Debojyoti Paul (CSB14005)
Bhaskar Sarkar (CSB14029)

Motivation

- A *Grid OF CCTVs* installed in an area can give information about human activities happening in the area.
- Most CCTVs cameras go unmonitored and unmaintained.
- *No Number of Human Being will be able to monitor the terabytes* of information being generated on a second-to-second basis
- Manual video tracking-CCTV analysis mostly relies on teams of specially trained officers watching thousands of hours of footage, waiting for that one crucial second of evidence

Cont..

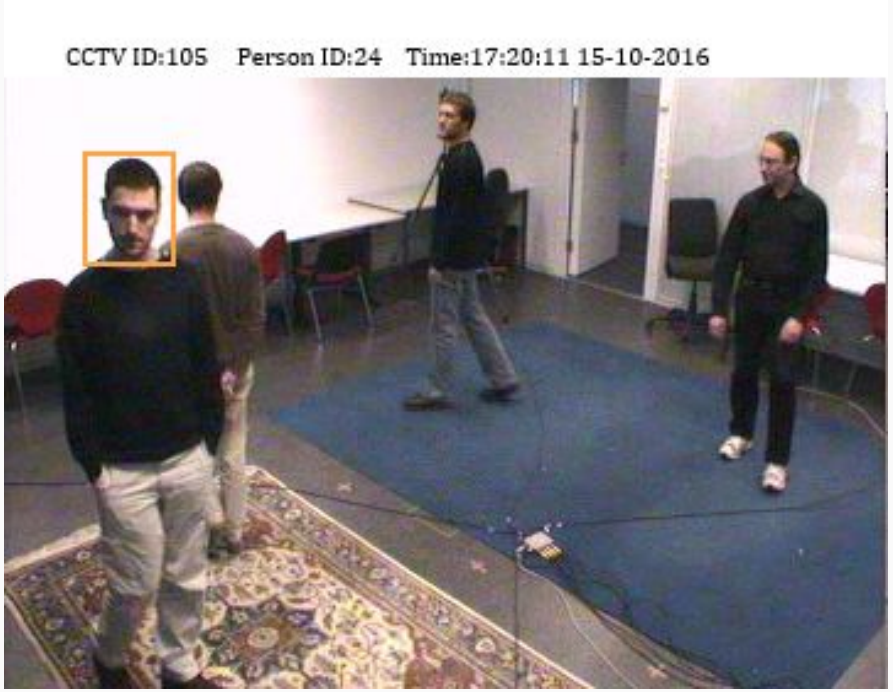
- Advanced Computer Vision technology can monitor the footage (if not disapproved by law) for movement. We just need to bring these cameras online. The data from these live streams can be processed and analyzed in near-real time to produce a wealth of information.

Proposed Solution

Mass Automatic Tracking

- **Mass:** The area covered by the grid of CCTV cameras is very large.
- **Automatic:** Records of tracking information will be generated by our AI system and given to the database, which can be used later.
- **Tracking:** Our smart AI system which should automatically recognize human faces should be able to track it.

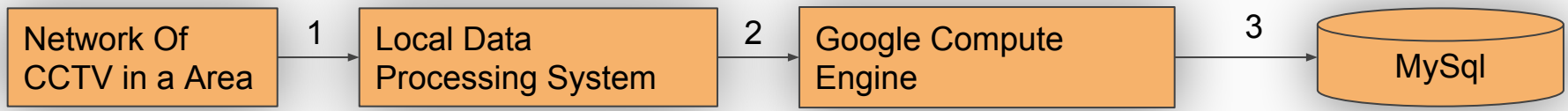

CCTV ID:101 Person ID:24 Time: 09:15:30 13-10-2016


CCTV ID:105 Person ID:24 Time:17:20:11 15-10-2016

Objective

- Develop the Base Architecture that helps in the task of Automated Mass Tracking
- To use an appropriate face recognition model and modify it to use in our Automated tracking system
- Identify problem that may come while developing such an automated system
- Develop a method that handles such problems
- Create a Experimental Evaluation of this method

Overall Architecture



Components

1. Grid of CCTV cameras

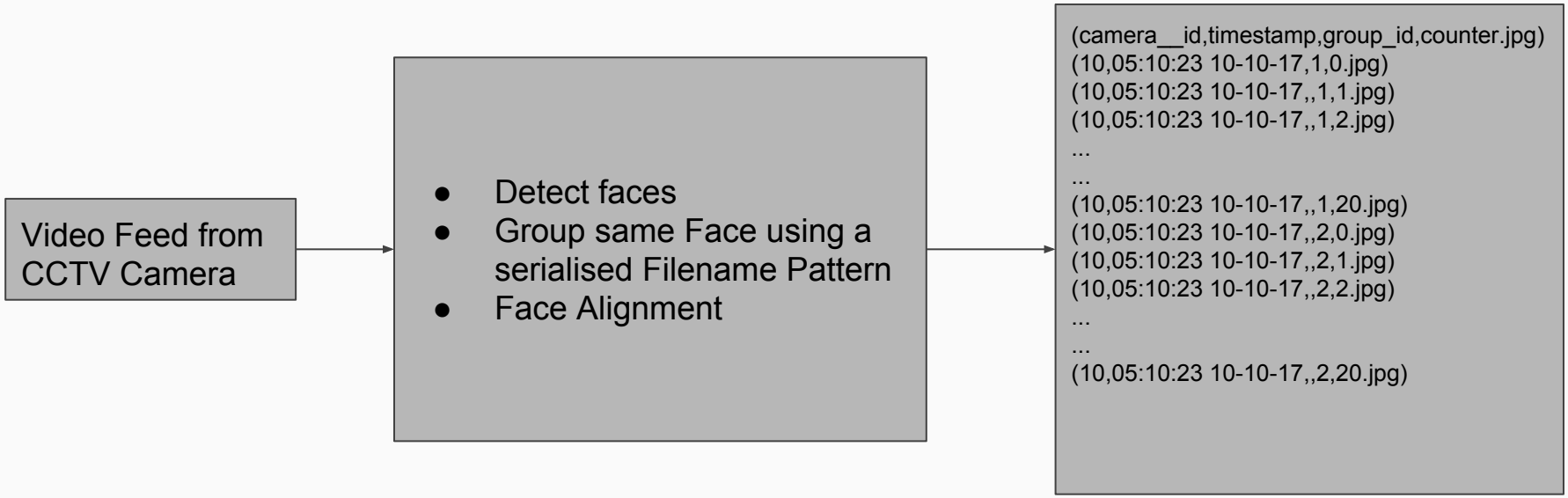
A collection of CCTV cameras connected to a local processing system that sends video Feed to the processing system.

Cont..

2. Local data processing system

An optimal local processing system is a High performance computer that receives video feed from “Grid of CCTV cameras”, process them and groups similar faces together

Local Processing System



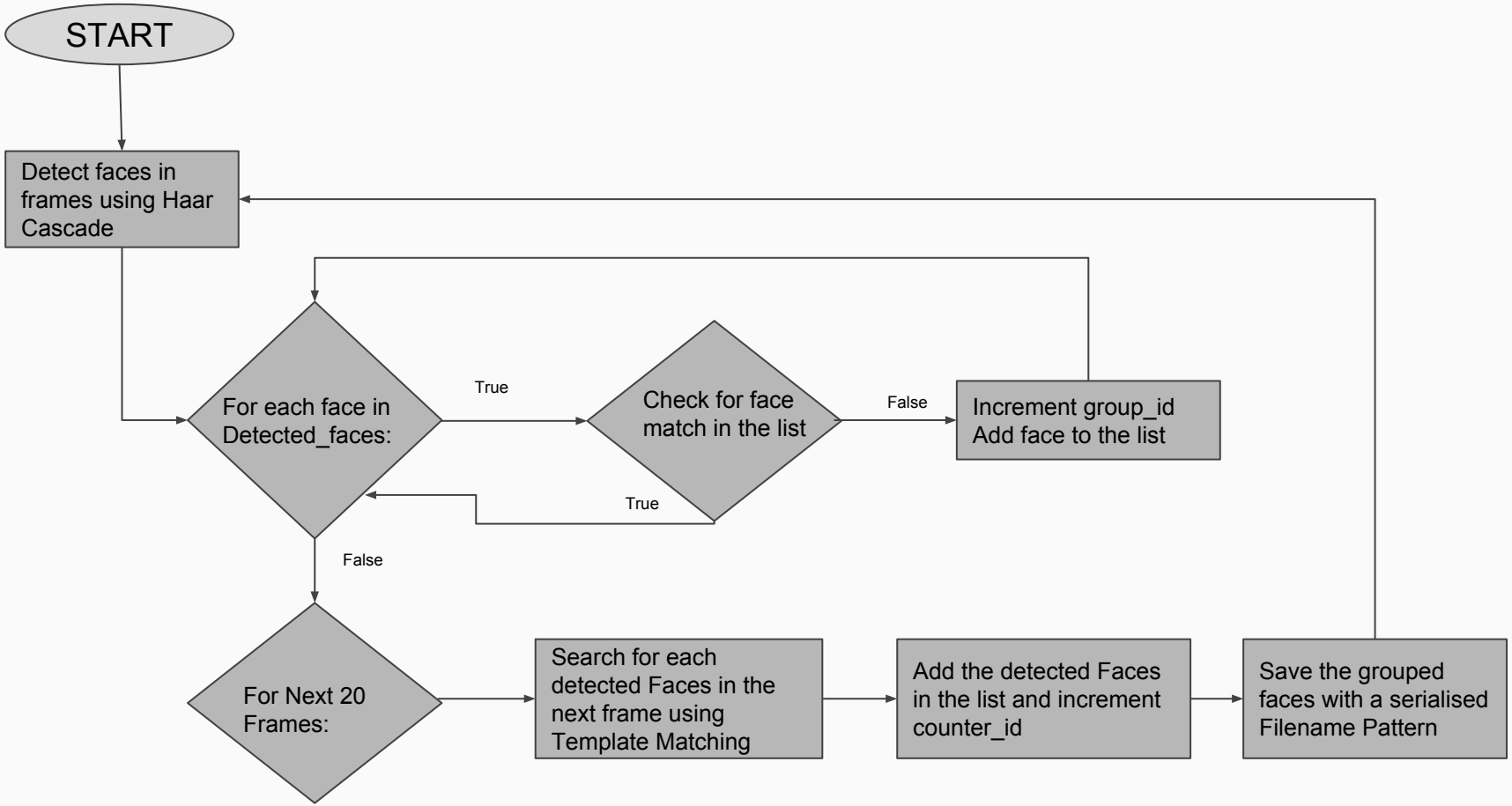


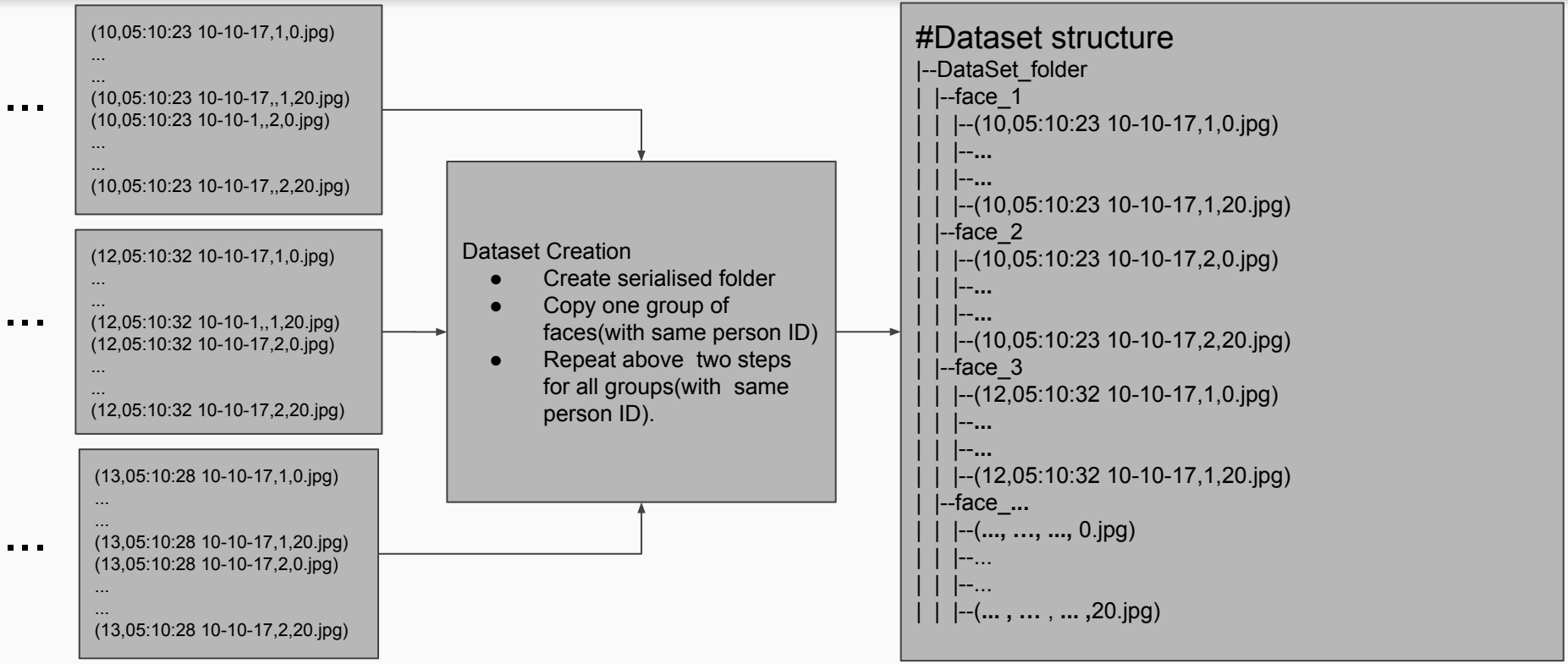
Figure: Flow chart for grouped faces generation

Cont..

3. A Centralized server

- Our smart AI that tracks the people has to recognize faces and assigns them a serial id. For example, in figure 1: serial id of 25 was assigned to the person.
- we are not interested in the actual label (i.e, the name and details of the persons). We are just interested in assigning an serial id to that face which will be used later as the label for our training.

Centralized server

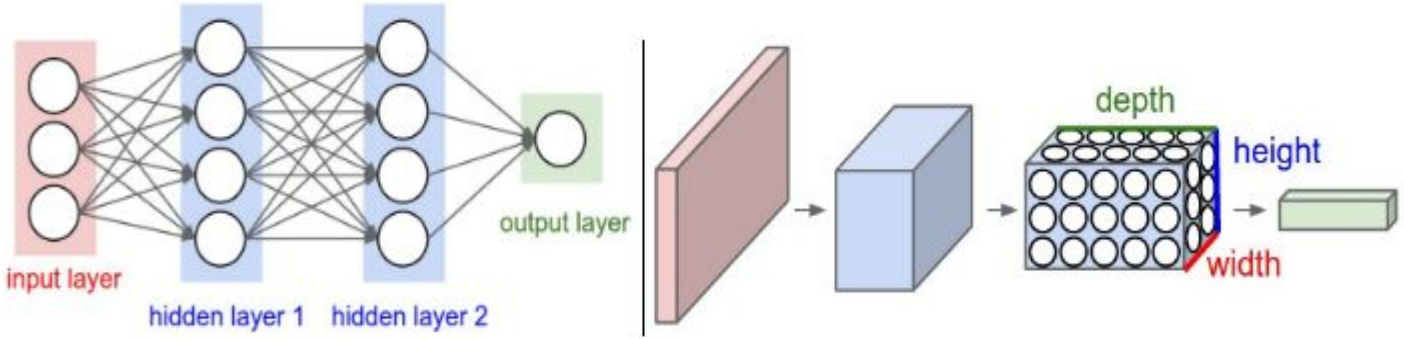


Face Recognition

- We need a mechanism to map an incoming face to to a correct label that our system can identify.
- One of the ways to do this is by comparing selected facial features from the image and a compare with the faces our system already knows.

- The state of the art system is Convolutional Neural Network .

Convolution Neural Network



Left: A regular 3-layer Neural Network. Right: A ConvNet arranges its neurons in three dimensions (width, height, depth), as visualized in one of the layers. Every layer of a ConvNet transforms the 3D input volume to a 3D output volume of neuron activations. In this example, the red input layer holds the image, so its width and height would be the dimensions of the image, and the depth would be 3 (Red, Green, Blue channels).

Layers in a Convolutional Neural Network

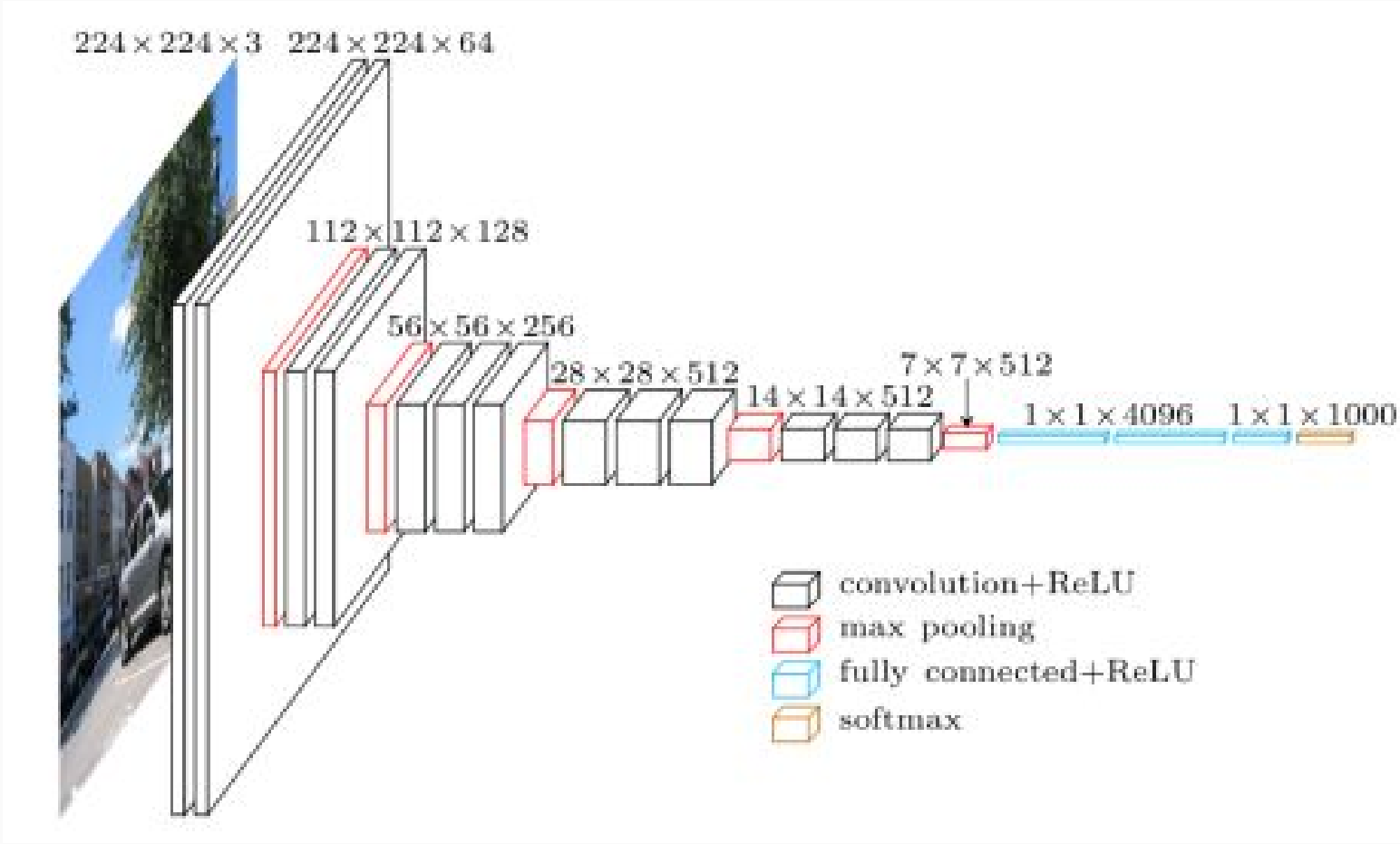
Convolution2D: In, this layer the output of neurons that are connected to local regions in the input, each computing a dot product between their weights and a small region they are connected to in the input volume.

Pooling: This layer will perform a downsampling operation along the spatial dimensions (width, height), resulting in volume

Fully Connected Layer:This layer will compute the class scores.

VGG-face : Model for face recognition

- Part of Deep face recognition model at Visual Geometry Group Department of Engineering Science University of Oxford.
- Uses 2.6M images, over 2.6K people for model training
- Dataset collected were celebrities and public figures, such as actors or politicians
- Takes input image of dimension 224 X 224 X 3.
- Evaluated on the LFW and YTF dataset.
- Accuracy of 98.95 on the LFW dataset.


224 × 224 × 3
224 × 224 × 64
112 × 112 × 128
56 × 56 × 256
28 × 28 × 512
14 × 14 × 512
7 × 7 × 512
1 × 1 × 4096
1 × 1 × 1000
convolution+ReLU
max pooling
fully connected+ReLU
softmax

Transfer Learning

- In a trained ConvNet, the earlier features of the layers contain more generic features and the later layers contains more specific features of the classes ,contained in the original dataset.
- Since training an entire Convolutional Network from scratch requires a lot of time, it is common to pre-train a ConvNet on a very large dataset and then use the ConvNet either for fixed feature extractor for the task of interest or fine tuning it with our dataset.

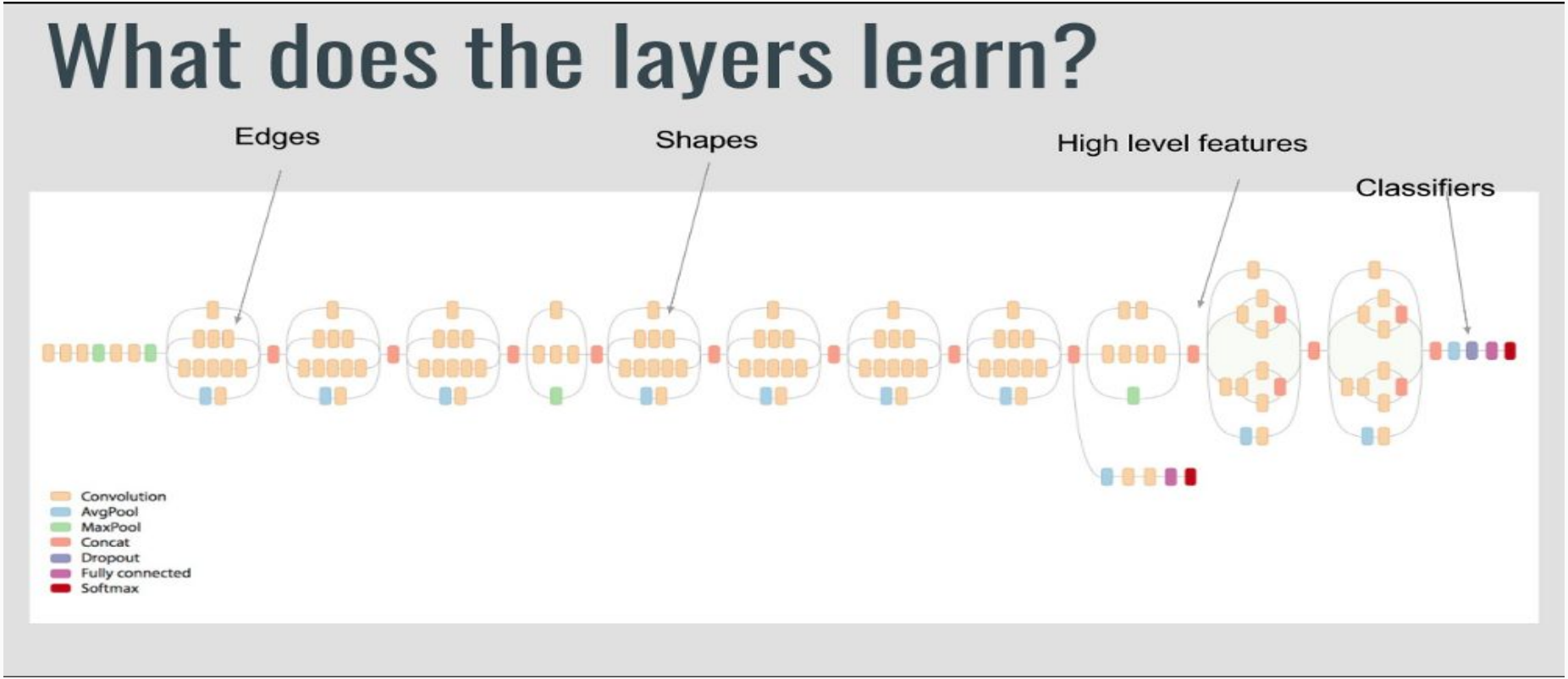
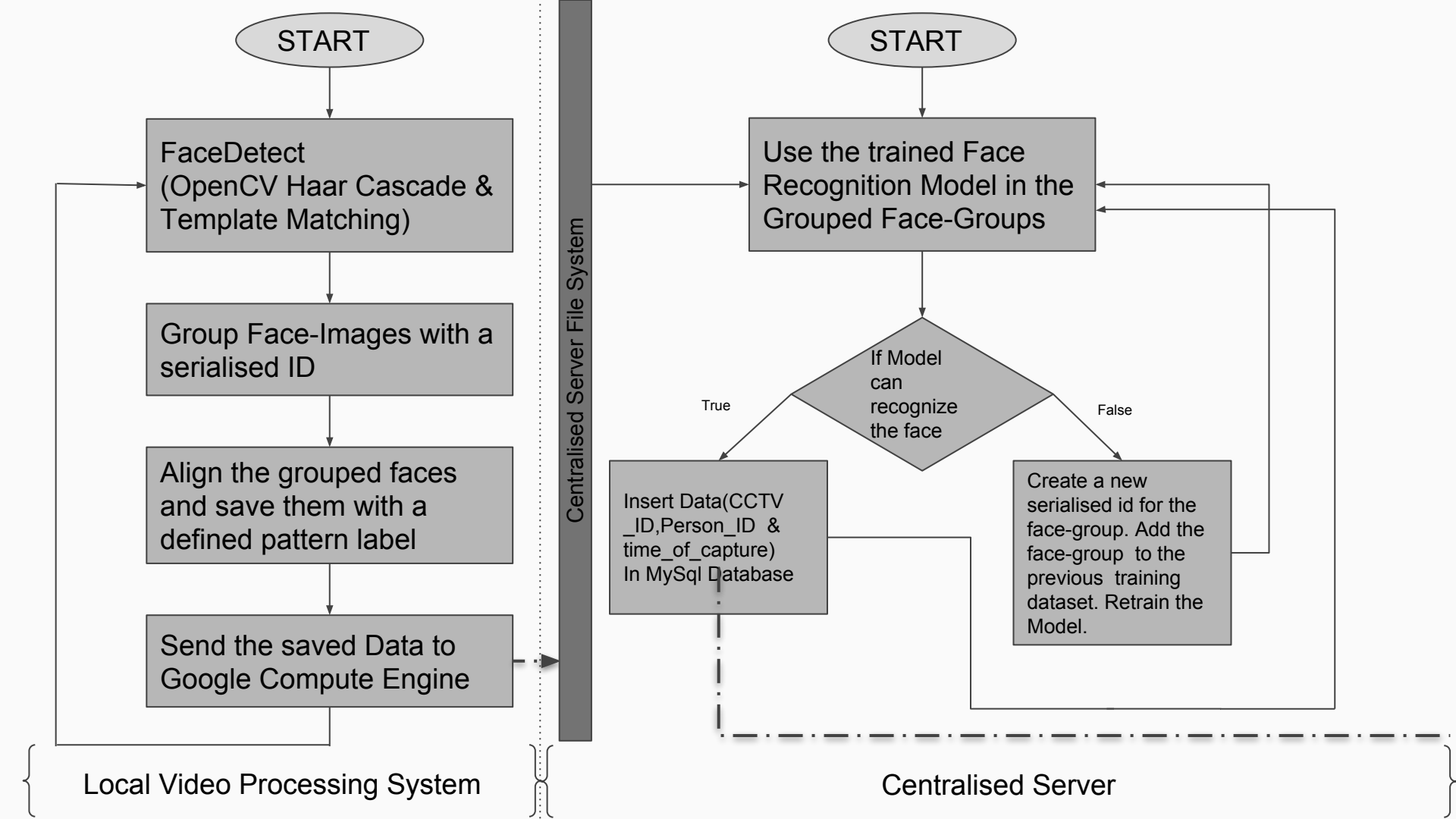


Figure : Inception V3 Google Research

Training the model

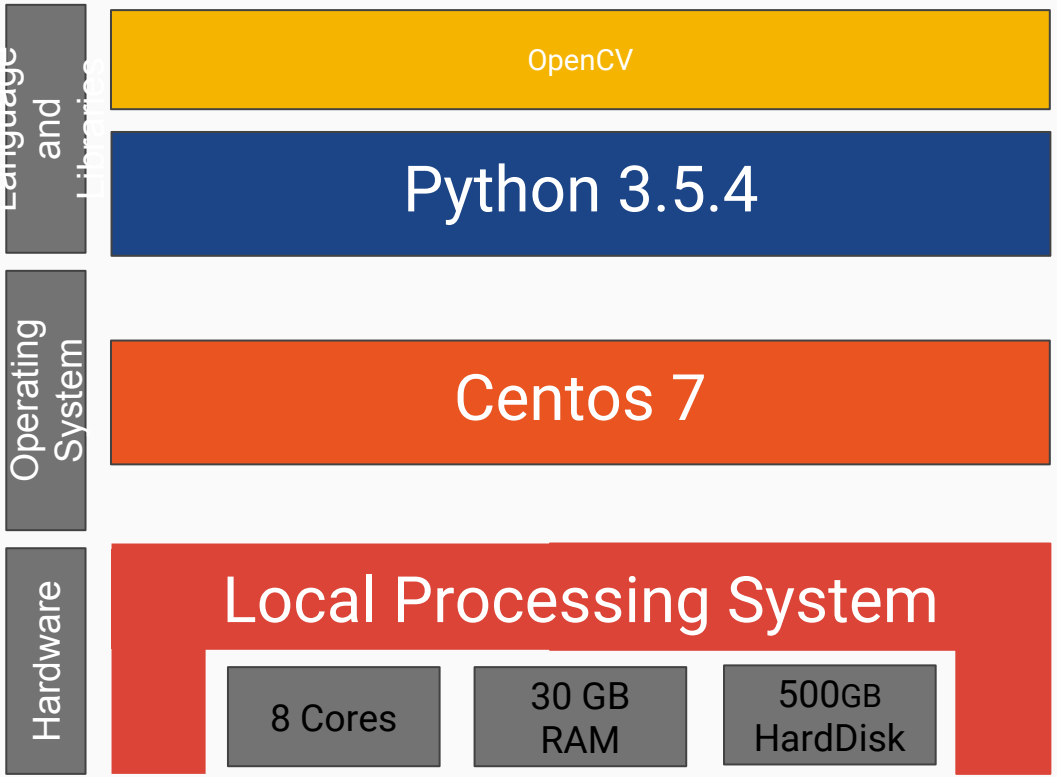
The VGG model has 16 layers.

Freezing all but the last 3 layers.


START
FaceDetect
(OpenCV Haar Cascade &
Template Matching)
Group Face-Images with a
serialised ID
Align the grouped faces
and save them with a
defined pattern label
Send the saved Data to
Google Compute Engine
Centralised Server File System
START
Use the trained Face
Recognition Model in the
Grouped Face-Groups
If Model
can
recognize
the face
True
False
Insert Data(CCTV
_ID,Person_ID &
time_of_capture)
In MySql Database
Create a new
serialised id for the
face-group. Add the
face-group to the
previous training
dataset. Retrain the
Model.
Local Video Processing System
Centralised Server

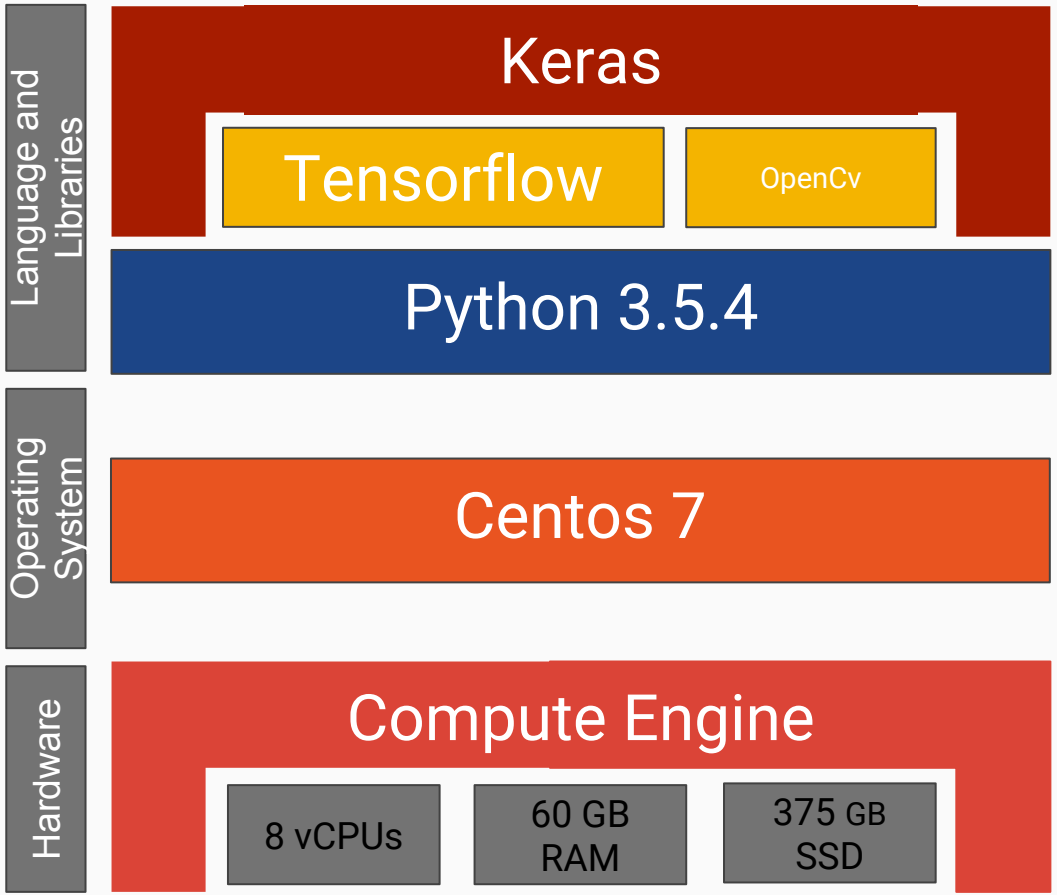
Local Processing System

1. Hardware Stack
   a. 8-cores
   b. 30 GB RAM
   c. 500GB Hard Disk
2. Software Stack
   a. Linux Based OS
   b. Python : 3.5.4
   c. OpenCV 3.2.0



Server Stack

1. Hardware Stack
   a. 8vCPU
   b. 60 GB RAM
2. Software Stack
   a. Centos 7
   b. Python : 3.5.3
   c. TensorFlow 1.1
   d. Keras 2.0.5



Conclusion

- A Base architecture that does the tracking has been constructed.
- The CCTV video feed is effectively processed to generate dataset of grouped faces.
- Used an appropriate face recognition model and modify it to use in our Automated tracking system
- An highly accurate face recognition model, VGG-FACE is used for Fine Tuning.

Future Work

- Our initial goal of the system which can can handle a large amount of training classes with huge amount of dataset while retaining the the model accuracy and training time by splitting the grouped dataset so that each model has a finite number of output classes.
- We plan to deploy this technique and evaluate weather it will work in a real world system involving lot of training data.

Cont..

- The problem of Mass automated tracking can be solved and this technique of splitting the grouped dataset can be applied here.
- As we have developed an model that uses transfer learning and also developed an local processing system that continuously generates dataset,applying this new methodology will not be difficult and tracking logic can be written to build an end user application.

Thank you!