



FACE DETECTION MODEL

A project by Sristi Dey



01. ABSTRACT

02. NOVELTY

03. RESULT AND DEMO

04. CONCLUSION

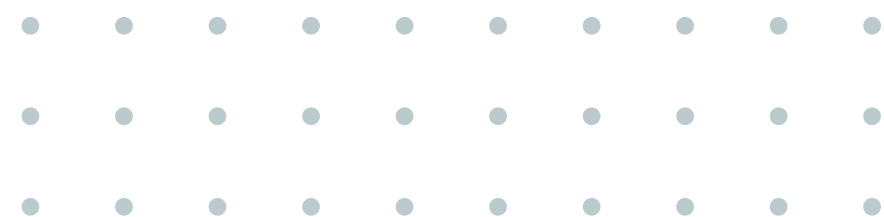
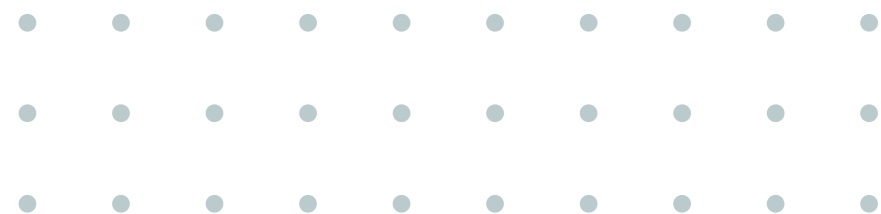


TABLE OF CONTENT



ABSTRACT

The face detection model abstract encompasses the development of a computer vision system adept at identifying and locating human faces within images or video frames. This project involves dataset collection, preprocessing, selection of a deep learning architecture, model training, evaluation, and deployment for applications like security, biometrics, and social media.



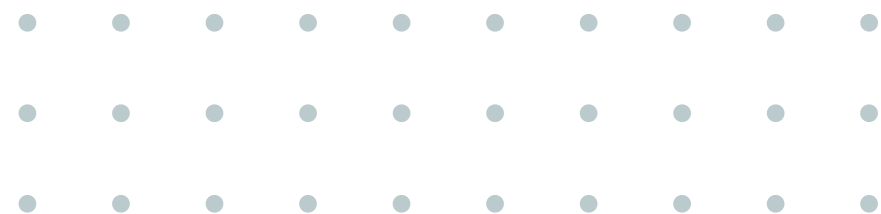
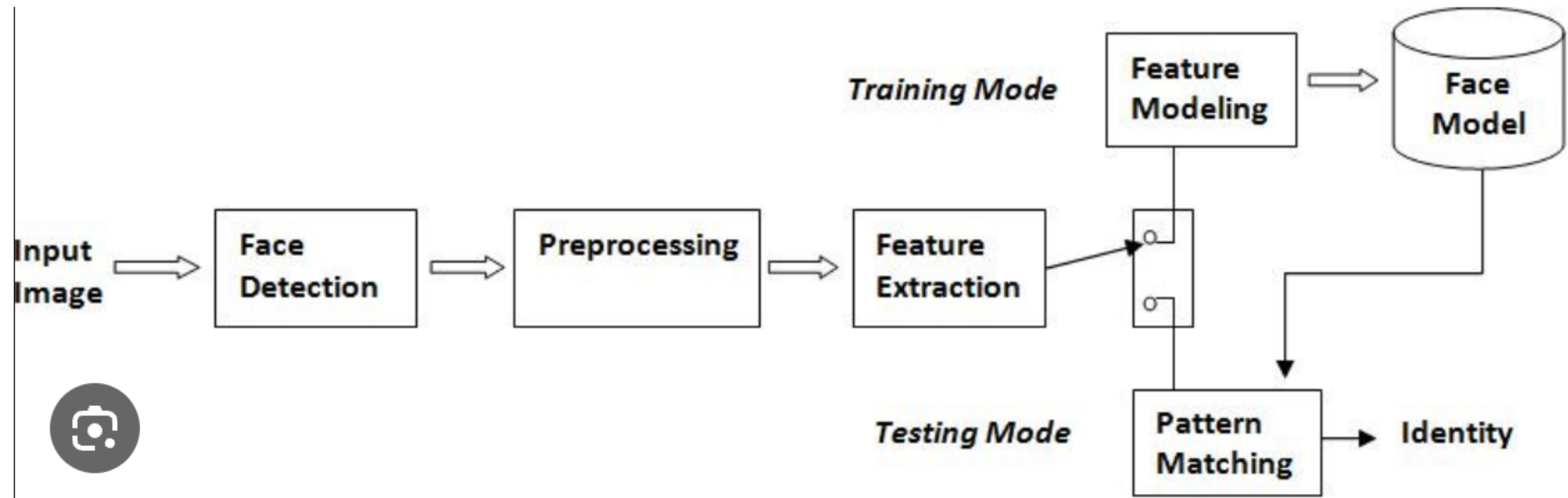


NOVELTY

The novelty of this face detection model lies in its integration of advanced computer vision techniques for accurate and efficient identification of human faces. Leveraging state-of-the-art deep learning architectures, the model excels in diverse environments, offering practical applications in security, biometrics, and social media with enhanced precision and speed.

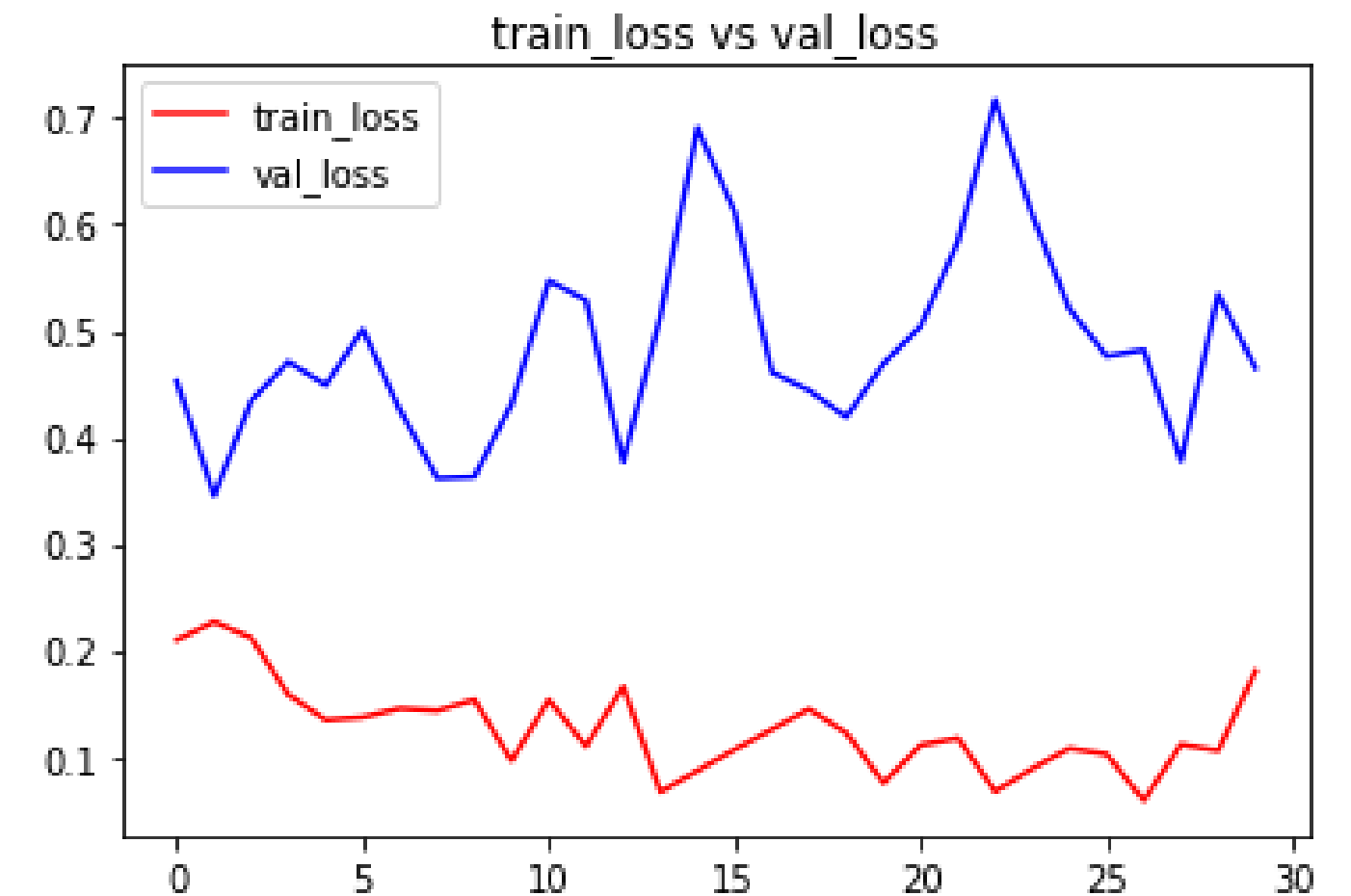


diagram



RESULT AND DEMO

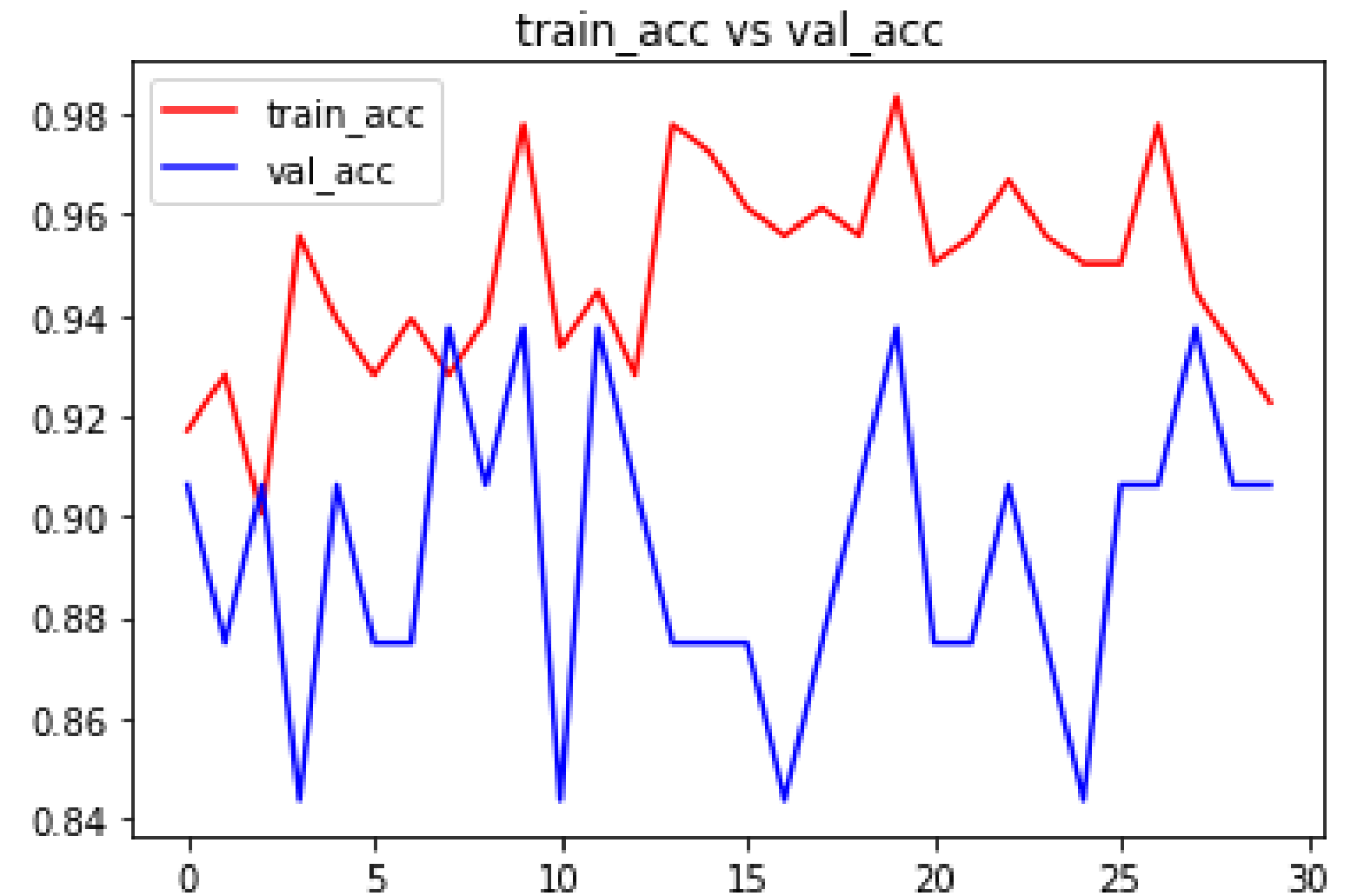
```
1 + import pickle
2 + import cv2
3 + import imutils, face_recognition
4 + import os, glob
5 + # All names
6 + names=[]
7 +
8 + # Getting the file directories
9 + file_list = glob.glob(os.path.join(os.getcwd(), "lfw\*"))
10 + for f in file_list:
11 +     s = f.split(os.path.sep)[-1]
12 +     names.append(s)
13 +
14 + imagePath = []
15 +
16 + for n in names:
17 +     each_imagePath_list = glob.glob(os.path.join(os.getcwd(), "lfw\\"+n+"\*"))
18 +     for ep in each_imagePath_list:
19 +         imagePath.append(ep)
20 +     each_imagePath_list = []
21 +
22 + known_encodings = []
23 + known_faces = []
24 + for i,im in enumerate(imagePath):
25 +     print("Processes Image {}/{}".format(i+1, len(imagePath)))
26 +     name = im.split(os.path.sep)[-2]
27 +     image = cv2.imread(im)
28 +     rgb = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
29 +     boxes = face_recognition.face_locations(rgb)
30 +     encodings = face_recognition.face_encodings(rgb, boxes)
31 +     for encoding in encodings:
32 +         known_encodings.append(encoding)
```




```

33 +         known_faces.append(name)
34 +
35 + # Dumping all using pickle so to use later
36 + data = {"encodings":known_encodings,"names":known_faces}
37 +
38 + '''----- RECOGNIZING PART -----'''
39 + ''' This code is used to get the test image and Perform recognition '''
40 + import cv2
41 + import imutils, face_recognition
42 + import os, glob
43 +
44 +
45 + # Loading the input image and convert it from BGR to RGB
46 + test_image = cv2.imread("test_image.png")
47 + test_rgb = cv2.cvtColor(test_image, cv2.COLOR_BGR2RGB)
48 +
49 + test_boxes = face_recognition.face_locations(test_rgb)
50 + print("Performed Encoding.....")
51 + test_encodings = face_recognition.face_encodings(test_rgb, test_boxes)
52 +
53 + # initialize the list of names for each face detected
54 + test_names = []
55 +
56 + for encoding in test_encodings:
57 +     matches = face_recognition.compare_faces(data["encodings"], encoding)
58 +     name1 = "Unknown"
59 +     if True in matches:
60 +         matchedIdxs = [i for (i, b) in enumerate(matches) if b]
61 +         counts = {}
62 +         for i in matchedIdxs:
63 +             name1 = data["names"][i]
64 +             counts[name1] = counts.get(name1, 0) + 1

```



CONCLUSION

In conclusion, the face detection model abstract outlines the development of an innovative computer vision system. By leveraging advanced deep learning techniques, this model achieves precise and efficient identification of human faces across various environments. Its practical applications in security, biometrics, and social media underscore its significance in advancing facial recognition technology.

git hub link- <https://github.com/SRISTI04/face-detection-model->