231501008

EXP NO: 09
DATE: 26-09-2025

Face Detection

Aim: Face Detection on available online human face image datasets

Algorithm:

1. Load pretrained Haar Cascade or DNN face detector.
2. Read image or video frame.
3. Convert to grayscale for faster detection.
4. Detect faces using detectMultiScale().
5. Draw bounding boxes around faces.
6. Display and save output.

Code:

```
import cv2

from google.colab.patches import cv2_imshow

face_cascade =
cv2.CascadeClassifier(cv2.data.haarcascades+'haarcascade_frontalface_default.xml')

img = cv2.imread('face.jpg'); gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

faces = face_cascade.detectMultiScale(gray, 1.3, 5)

for (x,y,w,h) in faces: cv2.rectangle(img,(x,y),(x+w,y+h),(0,255,0),2)

cv2_imshow(img)


import dlib, cv2

from google.colab.patches import cv2_imshow

detector = dlib.get_frontal_face_detector()

img = cv2.imread('face.jpg'); rgb = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)

for d in detector(rgb): cv2.rectangle(img,(d.left(),d.top()),(d.right(),d.bottom()),(0,255,0),2)

cv2_imshow(img); cv2.waitKey(0); cv2.destroyAllWindows()
```

231501008

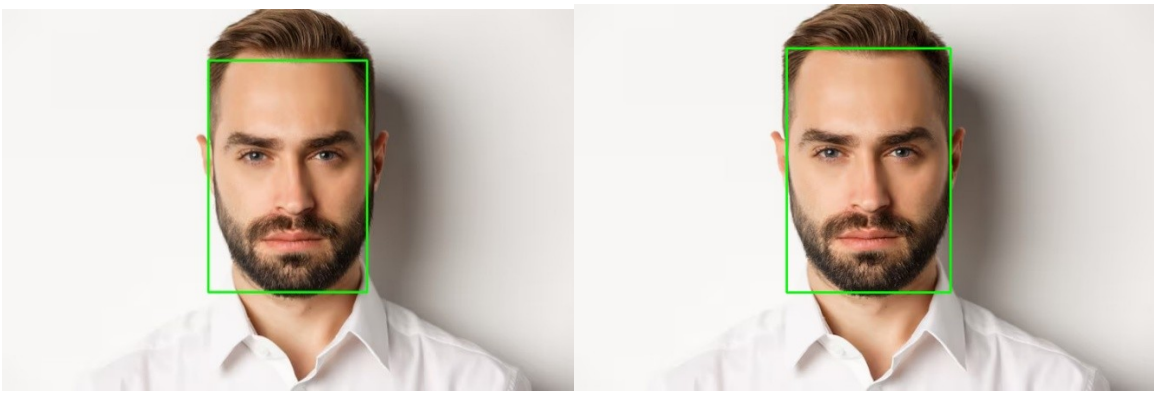
```
from mtcnn import MTCNN
import cv2
from google.colab.patches import cv2_imshow
detector = MTCNN(); img = cv2.imread('face.jpg')
res = detector.detect_faces(cv2.cvtColor(img, cv2.COLOR_BGR2RGB))
for r in res: x,y,w,h = r['box']; cv2.rectangle(img,(x,y),(x+w,y+h),(0,255,0),2)
cv2_imshow(img); cv2.waitKey(0); cv2.destroyAllWindows()
```

```
import cv2
from google.colab.patches import cv2_imshow

net =
cv2.dnn.readNetFromCaffe('deploy.prototxt','res10_300x300_ssd_iter_140000.caffemodel')
img = cv2.imread('face.jpg'); h,w = img.shape[:2]
blob = cv2.dnn.blobFromImage(cv2.resize(img,(300,300)),1.0,(300,300),(104,177,123))
net.setInput(blob); dets = net.forward()
for i in range(dets.shape[2]):
    if dets[0,0,i,2]>0.5:
        box = dets[0,0,i,3:7]*[w,h,w,h]; x1,y1,x2,y2 = box.astype(int)
        cv2.rectangle(img,(x1,y1),(x2,y2),(0,255,0),2)
cv2_imshow(img); cv2.waitKey(0); cv2.destroyAllWindows()
```

231501008

Output:



Result: Thus, Face Detection on available online human face image datasets was implemented successfully.