

# WK 8 Report

Capture face features in MTCNN

Yunting Chiu

A decorative light blue triangle is located in the bottom right corner of the slide, pointing towards the top right.

# Code Bug

```
3388/7000iterations
3389/7000iterations
3390/7000iterations
3391/7000iterations
3392/7000iterations
3393/7000iterations
3394/7000iterations
```

```
-----
IndexError                                Traceback (most recent call last)
```

```
<ipython-input-20-de38fed960e6> in <module>()
```

```
----> 1 extract_face(filename, required_size=(224, 224))
```

```
<ipython-input-18-ec4a89145bf9> in extract_face(filename, required_size)
```

```
    10         results = detector.detect_faces(pixels)
    11         # extract the bounding box from the first face
----> 12         x1, y1, width, height = results[0]['box']
    13         x2, y2 = x1 + width, y1 + height
    14         # extract the face
```

```
IndexError: list index out of range
```

SEARCH STACK OVERFLOW

# Some DeepFake pics cannot be captured

```
Requirement already satisfied: mtcnn in /usr/local/lib/python3.7/dist-packages (0.1.1)  
Requirement already satisfied: keras>=2.0.0 in /usr/local/lib/python3.7/dist-packages (from mtcnn) (2.6.0)  
Requirement already satisfied: opencv-python>=4.1.0 in /usr/local/lib/python3.7/dist-packages (from mtcnn) (4.1.2.30)  
Requirement already satisfied: numpy>=1.14.5 in /usr/local/lib/python3.7/dist-packages (from opencv-python>=4.1.0->mtcnn) (1.19.5)
```

```
{  
  'box': [248, 71, 106, 152],  
  'confidence': 0.7986212968826294,  
  'keypoints': {'left_eye': (273, 133),
```

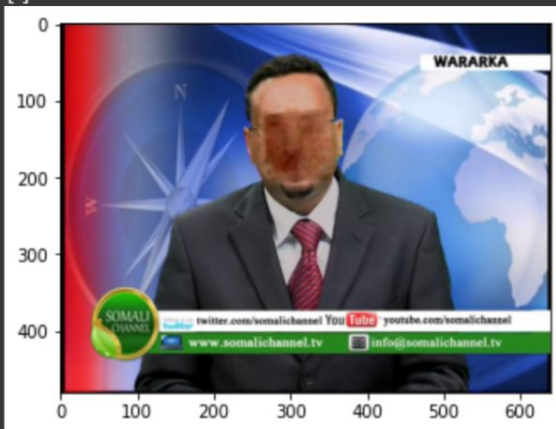
```
    'mouth_right': (321, 188),  
    'nose': (293, 165),  
    'right_eye': (323, 131)}}]
```



**Confidence only 0.79**

# Some DeepFake pics cannot be captured

```
Requirement already satisfied: mtcnn in /usr/local/lib/python3.7/dist-packages (0.1.1)  
Requirement already satisfied: keras>=2.0.0 in /usr/local/lib/python3.7/dist-packages (from mtcnn) (2.6.0)  
Requirement already satisfied: opencv-python>=4.1.0 in /usr/local/lib/python3.7/dist-packages (from mtcnn) (4.1.2.30)  
Requirement already satisfied: numpy>=1.14.5 in /usr/local/lib/python3.7/dist-packages (from opencv-python>=4.1.0->mtcnn) (1.19.5)  
(480, 640, 3)  
[]
```



**No outcomes in the  
list [ ]**

# Solution

- get rid of confidence < 0.9
- results != []

```
1 #filename = '/content/drive/MyDrive/American University/2021_Fall/DATA-793-001_D
now site information

3 def extract_face(filename, required_size=(320, 320)):
4     errCount = 0
5     for i in range(len(filename)):
6         # load image from file
7         pixels = imageio.imread(filename[i])
8         # create the detector, using default weights
9         detector = MTCNN()
10        # detect faces in the image
11        results = detector.detect_faces(pixels)
12        # make sure the captured probability > 0.9 and 4 values in box.keys()
13        if results:
14            if results[0]['confidence'] >= 0.9 and len(results[0]['box']) == 4:
15                # extract the bounding box from the first face
16                x1, y1, width, height = results[0]['box']
17                x2, y2 = x1 + width, y1 + height
18                # extract the face
19                print("The confidence is " + str(results[0]['confidence']))
20                face = pixels[y1:y2, x1:x2]
21                # resize pixels to the model size
22                image = Image.fromarray(face)
23                image = image.resize(required_size)
24                # return face_array
25                face_array = np.asarray(image)
26                name = './DeepFake facial extractions/image' + "_" + str(i) + '.jpg'
27                # writing the extracted images
28                imageio.imwrite(name, face_array)
29                #print(str(i) + "/" + str(len(filename)) + "iterations")
30                print('creating...' + name)
31            else:
32                errCount += 1
33                print("confidence < 0.9")
34                continue
35        else:
36            errCount += 1
37            print("The face cannot be captured")
38            continue
39    return errCount
```

Capturing...

Saving...

```
creating..../DeepFake_facial_extractions/image_6986.jpg
The confidence is 0.9977972507476807
creating..../DeepFake_facial_extractions/image_6987.jpg
The confidence is 0.9973077774047852
creating..../DeepFake_facial_extractions/image_6988.jpg
The confidence is 0.9985552430152893
creating..../DeepFake_facial_extractions/image_6989.jpg
The confidence is 0.9993975162506104
creating..../DeepFake_facial_extractions/image_6990.jpg
The confidence is 0.999772846698761
creating..../DeepFake_facial_extractions/image_6991.jpg
The confidence is 0.9997920393943787
creating..../DeepFake_facial_extractions/image_6992.jpg
The confidence is 0.9996356964111328
creating..../DeepFake_facial_extractions/image_6993.jpg
The confidence is 0.9998679161071777
creating..../DeepFake_facial_extractions/image_6994.jpg
The confidence is 0.9998793601989746
creating..../DeepFake_facial_extractions/image_6995.jpg
The confidence is 0.9998241066932678
creating..../DeepFake_facial_extractions/image_6996.jpg
The confidence is 0.9998055100440979
creating..../DeepFake_facial_extractions/image_6997.jpg
The confidence is 0.9999107122421265
creating..../DeepFake_facial_extractions/image_6998.jpg
The confidence is 0.999893069267273
creating..../DeepFake_facial_extractions/image_6999.jpg
16
```



# Deepfake - After MTCNN



image\_176.jpg



image\_177.jpg



image\_178.jpg



image\_179.jpg



image\_180.jpg



image\_181.jpg



image\_182.jpg



image\_183.jpg



image\_184.jpg



image\_185.jpg



image\_186.jpg



image\_187.jpg



image\_188.jpg



image\_189.jpg



image\_190.jpg



image\_191.jpg



image\_192.jpg



image\_193.jpg



image\_194.jpg



image\_195.jpg



image\_196.jpg



image\_197.jpg



image\_198.jpg



image\_199.jpg



image\_200.jpg



image\_201.jpg



image\_202.jpg



image\_203.jpg



image\_204.jpg



image\_205.jpg



image\_206.jpg



image\_207.jpg



image\_208.jpg



image\_209.jpg



image\_210.jpg



image\_211.jpg



image\_212.jpg



image\_213.jpg



image\_214.jpg



image\_215.jpg

# Real - After MTCNN





# Summary

## Deepfake

```
1 err_images = 16
2 print("MTCNN model cannot capture {} images in the Deepfake set".format(err_images))
3 capture_rate = (1 - (err_images/len(filename)))*100
4 print("The Capture Rate of Deepfakes is " + str(round(capture_rate, 2)) + "%")
```

MTCNN model cannot capture 16 images in the Deepfake set  
The Capture Rate of Deepfakes is 99.77%

## Real

```
[ ] 1 err_images = 0
2 print("MTCNN model cannot capture {} images in the Real set".format(err_images))
3 capture_rate = (1 - (err_images/len(filename)))*100
4 print("The Capture Rate of Reals is " + str(round(capture_rate, 2)) + "%")
```

MTCNN model cannot capture 0 images in the Real set  
The Capture Rate of Reals is 100.0%

# Recall: How I generate the data

