```
# face detection with mtcnn on a photograph
from matplotlib import pyplot
from matplotlib.patches import Rectangle
from matplotlib.patches import Circle
from mtcnn.mtcnn import MTCNN
```

```
!pip install mtcnn
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Collecting mtcnn
  Downloading mtcnn-0.1.1-py3-none-any.whl (2.3 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 2.3/2.3 MB 34.7 MB/s eta 0:00:00
Requirement already satisfied: keras>=2.0.0 in /usr/local/lib/python3.8/dist-packages (from mtcnn) (2.11.0)
Requirement already satisfied: opencv-python>=4.1.0 in /usr/local/lib/python3.8/dist-packages (from mtcnn) (4.6.0.
Requirement already satisfied: numpy>=1.17.3 in /usr/local/lib/python3.8/dist-packages (from opencv-python>=4.1.0-
Installing collected packages: mtcnn
Successfully installed mtcnn-0.1.1
```

```
# draw an image with detected objects
def draw_image_with_boxes(filename, result_list):
   # load the image
   data = pyplot.imread(filename)
   # plot the image
   pyplot.imshow(data)
   # get the context for drawing boxes
   ax = pyplot.gca()
   # plot each box
   for result in result_list:
     # get coordinates
     x, y, width, height = result['box']
     # create the shape
     rect = Rectangle((x, y), width, height, fill=False, color='red')
     # draw the box
     ax.add_patch(rect)
     # draw the dots
```

✓ 9s completed at 10:35 AM

```
        # create and draw dot
        dot = Circle(value, radius=2, color='red')
        ax.add_patch(dot)
    # show the plot
    pyplot.show()


filename = '/content/drive/MyDrive/Ndata/test1.jpg'
# load image from file
pixels = pyplot.imread(filename)
# create the detector, using default weights
detector = MTCNN()
# detect faces in the image
faces = detector.detect_faces(pixels)
# display faces on the original image
draw_image_with_boxes(filename, faces)
```

Colab paid products - Cancel contracts here