# OPENCV REFERENCES Line-detection

**LINESDETECTION.PY**  
  
**Lines detection with Hough Transform**https://pysource.com/2018/03/07/lines-detection-with-hough-transform-opencv-3-4-with-python-3-tutorial-21/  
  
**Line Finding with Hough Lines**  
<https://www.youtube.com/watch?v=lhMXDqQHf9g>  
  
**Hough Transform and Line Detection with Python**

<https://www.youtube.com/watch?v=G019Av7XhGo>  
  
**(ADVANCED) Finding Lane Lines on the Road**  
<https://towardsdatascience.com/finding-lane-lines-on-the-road-30cf016a1165>  
  
  
Template matching  
  
**Template Matching - OpenCV with Python for Image and Video Analysis 11**  
https://pythonprogramming.net/template-matching-python-opencv-tutorial/  
  
**Template matching-multiple objects**  
http://answers.opencv.org/question/165740/template-matching-multiple-objects/  
  
**Tutorial Template matching OPENCV**  
<https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_template_matching/py_template_matching.html>

**Template matching – OpenCV with python**https://pysource.com/2018/03/05/template-matching-opencv-3-4-with-python-3-tutorial-20/  
  
**MULTIPLE TEMPLATES FOR MULTIPLE OBJECTS**#FiRST template

template = cv2.imread("barts\_face.jpg", cv2.IMREAD\_GRAYSCALE)

w, h = template.shape[::-1]

result = cv2.matchTemplate(gray\_img, template, cv2.TM\_CCOEFF\_NORMED)

loc = np.where(result >= 0.4)

for pt in zip(\*loc[::-1]):

cv2.rectangle(img, pt, (pt[0] + w, pt[1] + h), (0, 255, 0), 3)

# SECOND TEMPLATE

template2 = cv2.imread("barts\_face.jpg", cv2.IMREAD\_GRAYSCALE)

w2, h2 = template.shape[::-1]

result2 = cv2.matchTemplate(gray\_img, template, cv2.TM\_CCOEFF\_NORMED)

loc2 = np.where(result >= 0.4)

for pt in zip(\*loc2[::-1]):

cv2.rectangle(img, pt, (pt[0] + w, pt[1] + h), (0, 255, 0), 3)﻿

**Python OpenCV find image in folder of images**

<https://stackoverflow.com/questions/39601465/python-opencv-find-image-in-folder-of-imagesroad-sign-recognition>  
  
**C++ Template matching with mutiple templates**http://answers.opencv.org/question/35469/template-matching-with-mutiple-templates/  
  
**Github project**  
<https://github.com/nikgens/TankRobotProject/tree/master/signRecognition>  
  
  
**KIJK DEZE \*GEBRUIKT GLOB\***<https://stackoverflow.com/questions/43339287/template-matching-with-multiple-source-images-in-opencv-and-python>  
  
VERKEERSLICHT  
https://github.com/ptd006/redlightdetection/blob/master/detect\_red.py