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

# All the 6 methods for comparison in a list

methods = ['cv2.TM\_CCOEFF', 'cv2.TM\_CCOEFF\_NORMED', 'cv2.TM\_CCORR',

'cv2.TM\_CCORR\_NORMED', 'cv2.TM\_SQDIFF', 'cv2.TM\_SQDIFF\_NORMED']

for meth in methods:

img = img2.copy()

method = eval(meth)

# Apply template Matching

res = cv2.matchTemplate(img,template,method)

min\_val, max\_val, min\_loc, max\_loc = cv2.minMaxLoc(res)

# If the method is TM\_SQDIFF or TM\_SQDIFF\_NORMED, take minimum

if method in [cv2.TM\_SQDIFF, cv2.TM\_SQDIFF\_NORMED]:

top\_left = min\_loc

else:

top\_left = max\_loc

bottom\_right = (top\_left[0] + w, top\_left[1] + h)

cv2.rectangle(img,top\_left, bottom\_right, 255, 2)

plt.subplot(