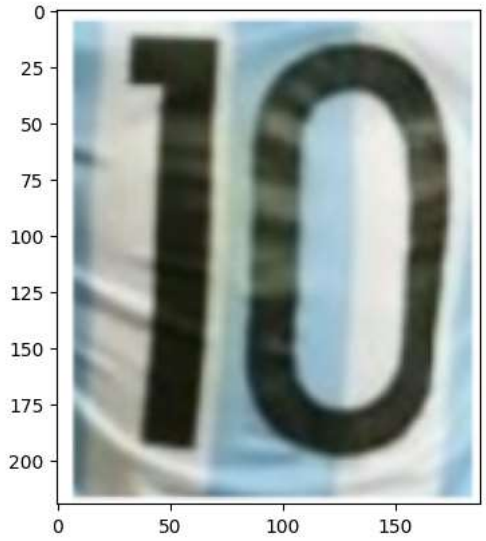
```
import numpy as np
import matplotlib.pyplot as plt
import cv2
%matplotlib inline
```

```
img=cv2.imread('/content/image.png')
img=cv2.cvtColor(img,cv2.COLOR_BGR2RGB)
plt.imshow(img)
```

```
<matplotlib.image.AxesImage at 0x78adfa636cb0>
```


```
img1=cv2.imread('/content/template.png')
img1=cv2.cvtColor(img1,cv2.COLOR_BGR2RGB)
plt.imshow(img1)
```

```
<matplotlib.image.AxesImage at 0x78adf9c84340>
```




```
height,width,channels=img1.shape # template image
height
```

```
220
```

```
width
```

```
188
```

```
channels
```

```
3
```

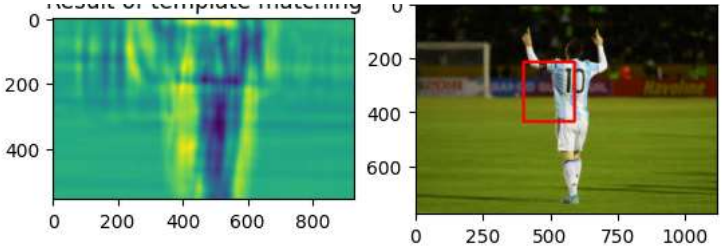
```
#Eval function ()  --->>> these brackets are called Eval function
#It will convert keyword to function
#Eval takes the already available functions like max ,min and other sort of in build python functions
```

```
# all the 6 methods for comparing in the list
methods = ['cv2.TM_CCOEFF','cv2.TM_CCOEFF_NORMED','cv2.TM_CCORR','cv2.TM_CCORR_NORMED','cv2.TM_SQDIFF','cv2.TM_SQDIFF_NORMED']


for m in methods:
  # Create a copy of the image
  img_copy=img.copy()
  # get actual function instead of string
  method=eval(m)
  res=cv2.matchTemplate(img_copy,img1,method)
  # grab their min and max values, plus their locations
  min_val,max_val,min_loc,max_loc=cv2.minMaxLoc(res)
   # Set up drawing of the rectangle
   # If the method is TM_SQDIFF or TM_SQFIFF_NORMED, take minimum
   # Notice the coloring on the last 2 left hand side images
  if method in [cv2.TM_SQDIFF,cv2.TM_SQDIFF_NORMED]:
    top_left=min_loc
  else:
    top_left=max_loc
    # assign the bottom right of the rectangle
  bottom_right=(top_left[0]+width,top_left[1]+height)
    # draw the red rectangle
  cv2.rectangle(img_copy,top_left,bottom_right,255,10) # 255 is the color of the rectangle and 10 is the thickness of the rectangle

  plt.subplot(121)
  plt.imshow(res)
  plt.title('Result of template matching')
  plt.subplot(122)
  plt.imshow(img_copy)
  plt.title('detected Point')
  plt.suptitle(m)
  plt.show()
  print('\n')
  print('\n')
```

cv2.TM_CCOEFF



cv2.TM_CCOEFF_NORMED

detected Point