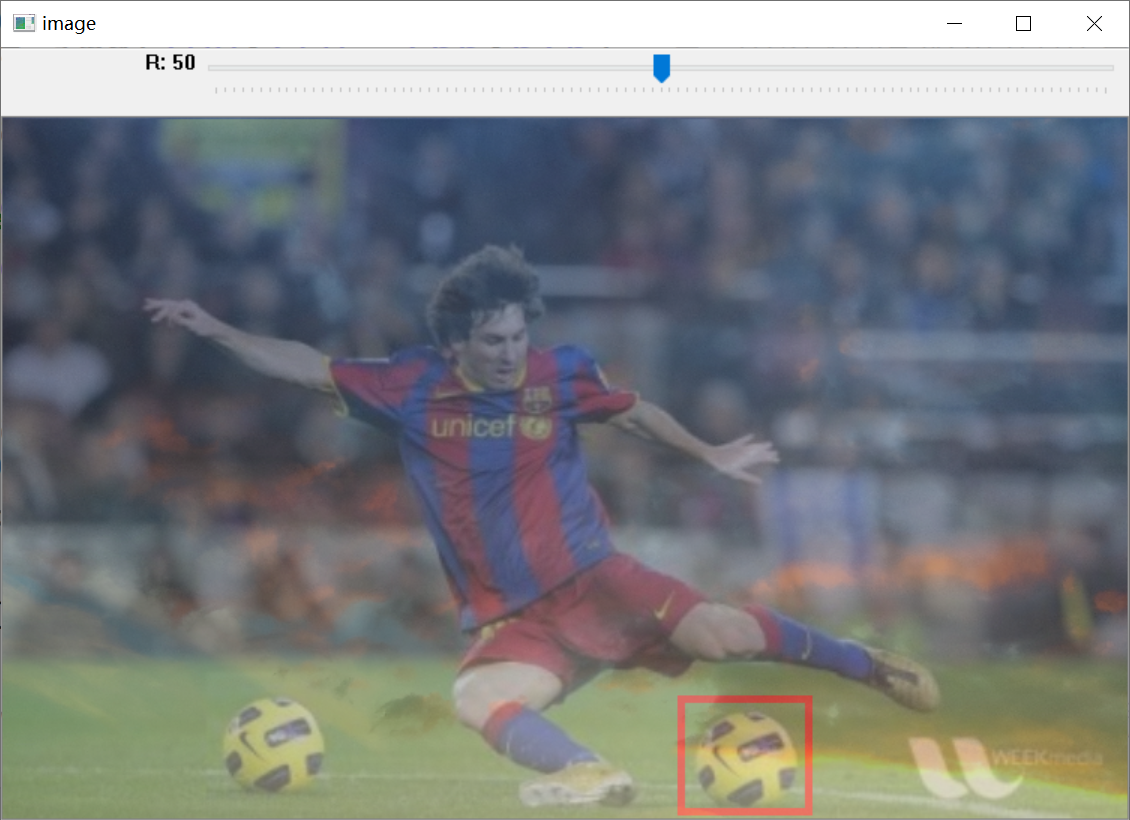
**1**

**import** cv2  
**import** numpy **as** np  
  
  
**def** nothing(x):  
 **pass  
  
  
def** Exchange(img, pd1): *# 创建回调函数* r = cv2.getTrackbarPos(**'R'**, **'image'**) / 100 *# 用滑动条上面显示的数值对两张图片的权重进行改变* **return** cv2.addWeighted(img, 1 - r, pd, r, 0)  
  
  
img = cv2.imread(**'./recourse/football.png'**) *# 读图片*pd = cv2.imread(**"./recourse/photoo.jpg"**)  
pd = cv2.resize(pd, (750, 467)) *# 将图片大小调整为与第一张大小相同*print(img.shape) *# 输出图片的信息*print(img.size)  
print(img.dtype)  
ball = img[390:460, 455:535] *# 用红色矩形框圈出足球*football = ball.copy()  
img[385:465, 450:540] = [0, 0, 225]  
img[390:460, 455:535] = football  
  
cv2.namedWindow(**'image'**) *# 定义窗口名*cv2.createTrackbar(**'R'**, **'image'**, 0, 100, nothing) *# 获得滑动条***while** 1:  
 cv2.imshow(**'image'**, Exchange(img, pd)) *# 显示image窗口* k = cv2.waitKey(1) & 0xff *# 等待输出指令* **if** k == 27: *# Esc退出* **break**cv2.destroyAllWindows()



**2**

**import** cv2  
**import** numpy **as** np  
  
img = cv2.imread(**'./recourse/red block.png'**) *# 读图片*lower\_red = np.array([156, 43, 46]) *# 设置阈值*upper\_red = np.array([180, 255, 255])  
hsv\_red = cv2.cvtColor(img, cv2.COLOR\_BGR2HSV) *# 将BGR图像转换成HSV图像*mask = cv2.inRange(hsv\_red, lower\_red, upper\_red) *# 提取*res = cv2.bitwise\_and(img, img, mask=mask) *# 按位与，通过掩膜显示红色*print(hsv\_red, [[[60, 255, 255]]]) *# 打印红色的HSV值*gray = cv2.cvtColor(res, cv2.COLOR\_BGR2GRAY) *# 变为灰度图*retval, dst = cv2.threshold(gray, 0, 255, cv2.THRESH\_OTSU) *# 大津法二值化*dst = cv2.dilate(dst, **None**, iterations=1) *# 膨胀，白区域变大*dst = cv2.erode(dst, **None**, iterations=1) *# 腐蚀，白区域变小*rows, cols = dst.shape *# 旋转图片并将图片缩小为原来的一半*M = cv2.getRotationMatrix2D((cols / 2, rows / 2), 71, 0.5)  
exc = cv2.warpAffine(dst, M, (2 \* cols, 2 \* rows))  
  
cv2.imshow(**'image'**, exc) *# 显示最后成型的图片*cv2.waitKey(0) *# 按任意键关闭窗口*cv2.destroyAllWindows()

