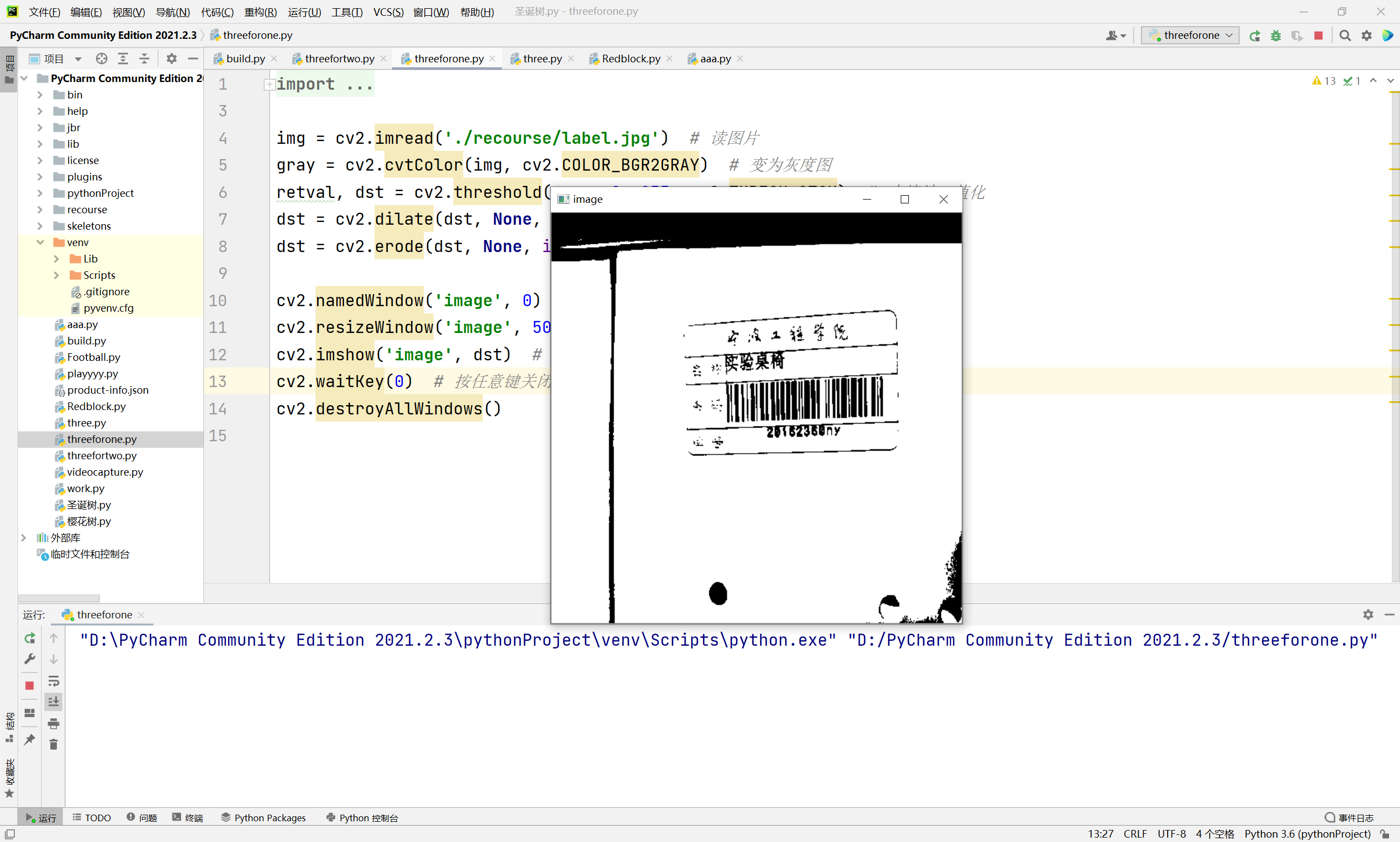
**import** cv2  
**import** numpy **as** np  
  
img = cv2.imread(**'./recourse/label.jpg'**) *# 读图片*gray = cv2.cvtColor(img, cv2.COLOR\_BGR2GRAY) *# 变为灰度图*retval, dst = cv2.threshold(gray, 0, 255, cv2.THRESH\_OTSU) *# 大津法二值化*dst = cv2.dilate(dst, **None**, iterations=1) *# 膨胀，白区域变大*dst = cv2.erode(dst, **None**, iterations=1) *# 腐蚀，白区域变小*cv2.namedWindow(**'image'**, 0) *# 定义窗口名*cv2.resizeWindow(**'image'**, 500, 500) *# 设定窗口大小*cv2.imshow(**'image'**, dst) *# 显示最后成型的图片*cv2.waitKey(0) *# 按任意键关闭窗口*cv2.destroyAllWindows()



**import** cv2  
**import** numpy **as** np  
  
img = cv2.imread(**"./recourse/gun.jpg"**) *# 读图片*lower = np.array([0, 100, 0]) *# 设置阈值*upper = np.array([10, 255, 255])  
hsv = cv2.cvtColor(img, cv2.COLOR\_BGR2HSV) *# 将BGR图像转换成HSV图像*mask = cv2.inRange(hsv, lower, upper) *# 提取*res = cv2.bitwise\_and(img, img, mask=mask)  
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*(drink, \_) = cv2.findContours(dst.copy(), cv2.RETR\_EXTERNAL, cv2.CHAIN\_APPROX\_SIMPLE) *# 求最小面积矩形*c = sorted(drink, key=cv2.contourArea, reverse=**True**)[0] *# 计算最大轮廓的旋转边界框*draw = img.copy()  
x, y, w, h = cv2.boundingRect(c) *# 画矩形*res = cv2.rectangle(draw, (x, y), (x + w, y + h), (255, 255, 0), 2)  
print(**'('**, round(x + w / 2), **','**, round(y + h / 2), **')'**, h, w) *# 打印中心点，矩形的长宽*(x, y), radius = cv2.minEnclosingCircle(c) *# 画圆*center = (int(x), int(y))  
radius = int(radius)  
res = cv2.circle(res, center, radius, (0, 255, 0), 2)  
print(center, radius) *# 打印圆心，半径*cv2.namedWindow(**'dst'**, 0) *# 定义窗口名*cv2.resizeWindow(**'dst'**, 500, 500) *# 设定窗口大小*cv2.imshow(**"dst"**, res) *# 图像显示*cv2.waitKey(0) *# 等待窗口*cv2.destroyAllWindows()

