

8ª Atividade

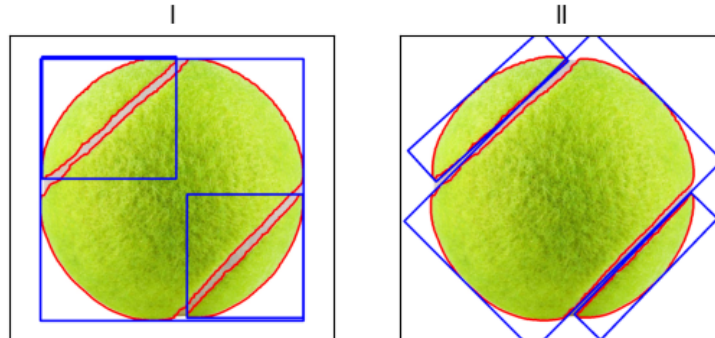
Propriedades dos contornos de uma imagem

Considerando a imagem abaixo:

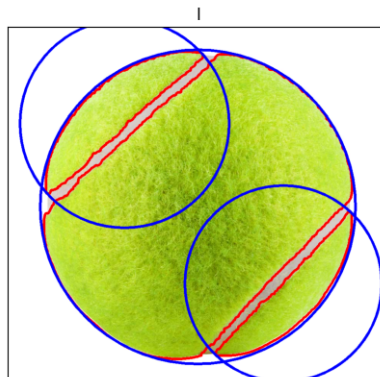


Utilizando métodos de visão computacional, realize as seguintes tarefas:

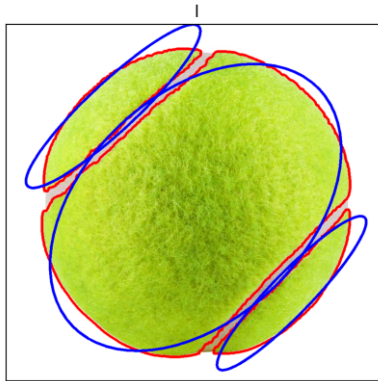
1. Trace os retângulos envolventes normais e rotacionados para cada um dos contornos.



2. Trace os círculos envolventes mínimos para cada um dos contornos.



3. Trace as ellipses circunscritas para cada um dos contornos.

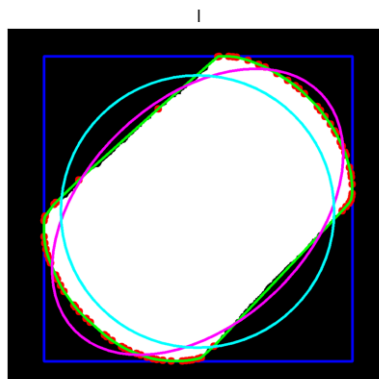


4. Para o segundo contorno obtido a partir da imagem da bola, calcule as seguintes propriedades:

- (a) Proporção.
- (b) Área do retângulo limite.
- (c) Extensão.
- (d) Solidez.
- (e) Diâmetro equivalente.
- (f) Orientação.

5. Para o mesmo contorno da tarefa anterior produza uma imagem contendo:

- (a) A máscara do contorno.
- (b) A sua elipse circunscrita.
- (c) O seu círculo de área equivalente.
- (d) O seu retângulo limite.
- (e) Os pontos da sua casca convexa.
- (f) O polígono formado pela junção dos pontos da sua casca convexa.



Funções de referência

Algumas funções de referência necessárias para implementar as tarefas acima são descritas abaixo:

```
contours, hierarchy = cv.findContours(color_mask,cv.RETR_TREE,
    cv.CHAIN_APPROX_SIMPLE)
```

- Encontra os contornos e a hierarquia dos contornos dos objetos em uma imagem binária.

```
cv.drawContours(img,contours,x,(r,g,b),thickness)
```

- Traça um ou mais contornos em *img* com a cor (r,g,b) e grossura *thickness*. O parâmetro *x* define qual dos contornos contidos em *contours* será desenhado. Para $x = -1$ todos os contornos são desenhados.

```
x = np.vstack(y)
```

- Retorna o vetor *x* obtido a partir do empilhamento vertical dos vetores contidos em *y*.

```
hull = cv.convexHull(contour)
```

- Retorna a casca convexa que engloba um contorno.

```
cv.polylines(img,[points],isClosed,(r,g,b),thickness)
```

- Traça um polígono em *img* a partir de um conjunto de pontos [*points*], que pode ser fechado para *isClosed* = *True* e aberto caso contrário. A linha traçada tem cor (r,g,b) e grossura *thickness*.

```
x,y,w,h = cv.boundingRect(contour)
```

- Retorna as coordenadas do canto superior esquerdo e a largura e a altura do retângulo que engloba um contorno.

```
rect = cv.minAreaRect(contour)
box = cv.boxPoints(rect)
box = np.intp(box)
```

- `cv.minAreaRect()` encontra o retângulo rotacionado de menor área que engloba o contorno. As funções seguintes retornam as coordenadas inteiras dos seus vértices.

```
(x,y),radius = cv.minEnclosingCircle(contour)
```

- Retorna as coordenadas do centro e o raio do menor círculo que engloba um contorno.

```
ellipse = cv.fitEllipse(contour)  
# or  
(x_el,y_el),(MA,ma),angle = cv.fitEllipse(contour)
```

- Retorna a elipse circunscrita a um contorno, definida pelo seu ponto central, eixos maior e menor e ângulo.

```
cv.ellipse(img,ellipse,(r,g,b),thickness)
```

- Desenha uma elipse em *img* de acordo com os parâmetros especificados.