

MC920 - Trabalho 3

Fábio Camargo Ricci - 170781

Introdução

O objetivo deste trabalho é obter algumas medidas de objetos presentes em imagens digitais, como área, perímetro, excentricidade e solidez.

Bibliotecas utilizadas

OpenCV - Leitura e escrita de imagens

NumPy - Manipulação de Arrays

Matplotlib - Plotagem de histogramas

Execução

Para executar o programa na raiz do projeto, basta chamar: `python3 ./main.py {caminho_arquivo_entrada} {prefixo_arquivo_saida}`

Ex: `python3 ./main.py ./in/objetos1.png objetos1`

O programa

A partir da imagem de entrada, o programa gera:

- Uma imagem binária.
- Uma imagem com os contornos dos objetos.
- Propriedades dos objetos.
- Histograma com a área dos objetos.

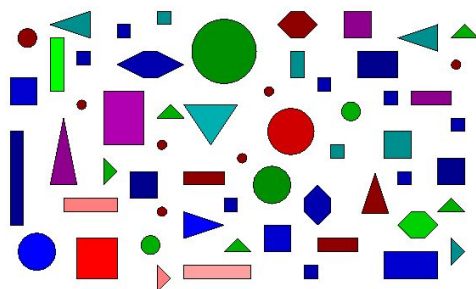
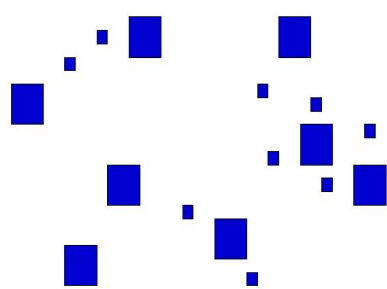
A transformação de cores é feita com os comandos `cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)` (que converte a imagem colorida para níveis de cinza) e `cv2.threshold(image, 200, 255, 0)` para binarizar a imagem, da biblioteca OpenCV.

Para a identificação dos contornos dos objetos, utilizou-se o método de Canny através do método `cv2.Canny(image, 0, 200)`.

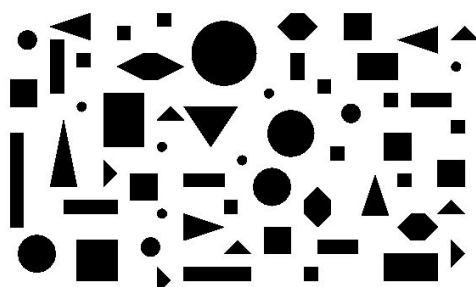
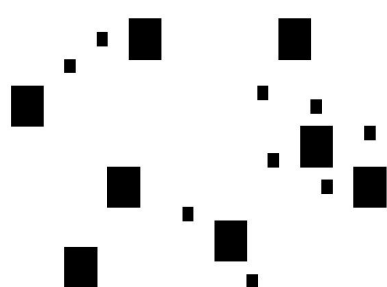
Finalmente, para a obter as propriedades desejadas dos objetos, primeiramente utiliza-se o método `cv2.findContours(image, cv2.RET_TREE, cv2.CHAIN_APPROX_NONE)` para identificar todos os objetos presentes na imagem. Com isso, calcula-se o centróide de cada objeto e todas as propriedades desejadas com os momentos calculados (`cv2.moments(contour)`).

A leitura da imagem é feita a partir do comando `cv2.imread(filepath, cv2.IMREAD_COLOR)`.

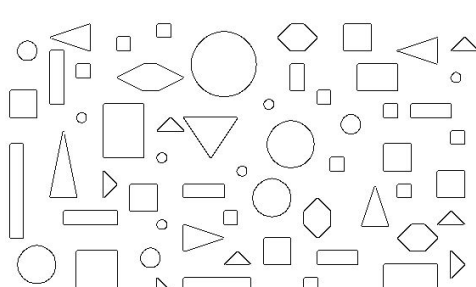
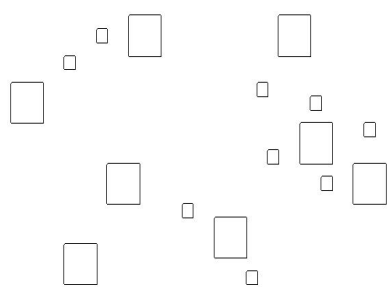
Resultados



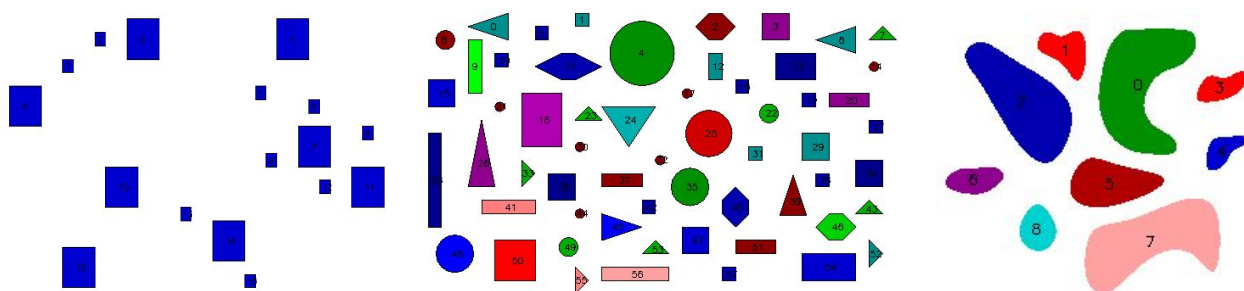
Imagens Originais



Imagens Binárias



Contornos dos Objetos



Regiões Rotuladas

Propriedades dos objetos da imagem “objetos3.png”:

Número de regiões: 9

Região 0: área: 4107.0 perímetro: 319.421354 excentricidade: 0.813036 solidez: 0.754963

Região 1: área: 843.5 perímetro: 125.639609 excentricidade: 0.735146 solidez: 0.904558

Região 2: área: 3690.5 perímetro: 265.119838 excentricidade: 0.896234 solidez: 0.978264

Região 3: área: 584.0 perímetro: 104.911687 excentricidade: 0.882918 solidez: 0.913928

Região 4: área: 478.0 perímetro: 94.426406 excentricidade: 0.85057 solidez: 0.92546

Região 5: área: 1761.5 perímetro: 179.781745 excentricidade: 0.864278 solidez: 0.971862

Região 6: área: 688.5 perímetro: 108.669047 excentricidade: 0.882704 solidez: 0.972458

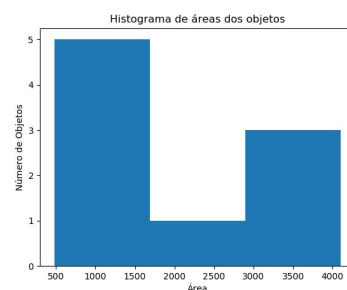
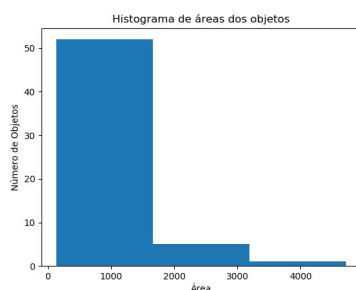
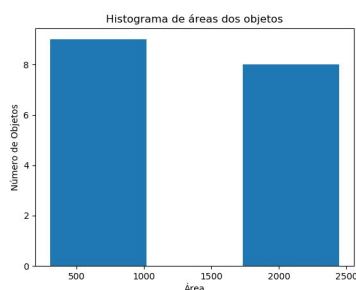
Região 7: área: 4067.0 perímetro: 311.078208 excentricidade: 0.908225 solidez: 0.780689

Região 8: área: 716.5 perímetro: 101.982755 excentricidade: 0.61053 solidez: 0.980164

Número de regiões pequenas: 5

Número de regiões médias: 1

Número de regiões grandes: 3



Histograma das Áreas dos Objetos

Comparando-se os resultados das propriedades dos objetos com outros colegas, notou-se uma diferença em relação ao uso das bibliotecas OpenCV e skimage na determinação da área, perímetro, etc. Isso provavelmente se deve às diferenças no processo de determinação dos contornos das duas bibliotecas, como o algoritmo ou margens utilizadas.

Conclusão

Ao final desse trabalho, pôde-se estudar métodos de detecção de bordas e objetos em imagens, com obtendo-se propriedades geométricas como área, perímetro, entre outras. É interessante notar que para o processo de identificação de objetos funcionar, foi necessário binarizar a imagem original, para assim utilizar métodos da biblioteca OpenCV para obter os resultados desejados.