

Computer Vision

Image Recognition

A64282-Paulo Alves

University of Minho, Department of Informatics,
4710-057 Braga, Portugal

Resumo Este relatório trata da implementação em *MATLAB* de técnicas de segmentação e reconhecimento de imagens. Trata também dos respectivos resultados e comentários sobre os mesmos.

1 Introdução

O presente relatório, no âmbito da unidade curricular de **Visão por Computador** do curso de Mestrado Integrado em Engenharia informática da Universidade do Minho, vai tratar da implementação em *MATLAB* e experimentação de técnicas de segmentação e reconhecimento de imagens com ruído e sem ruído. No primeiro caso é introduzido ruído numa imagem *gray-scaled*, depois aplicado um filtro para redução de ruído e posteriormente aplicam-se as técnicas para detectar e segmentar os objectos da imagem.

2 Algoritmo geral

Dois tipos de imagens foram utilizados, imagens de moedas e imagens de brinquedos. Em ambas o pretendido era detectar cada objecto e fazer a segmentação da imagem pelos mesmos. O algoritmo utilizado tinha que com diferentes imagens de moedas e brinquedos, fundos diferentes, números diferentes de objectos, iluminação não uniforme etc, detectar os objectos o melhor possível e contabilizar. O processo utilizado foi o seguinte :

- Carregar imagem .
- Converter a imagem para *grey-scale* .
- Introduzir ruído do tipo *salt and pepper*.
- Filtrar a imagem utilizando um filtro *Gaussiano* ou filtro de mediana.
- Escolher um threshold que vai ser usado para converter a imagem para preto e branco.

- Depois da conversão da imagem, os espaços são preenchidos com a cor branca.
- Utiliza-se uma função de separação morfológica, com um objecto estruturante do tipo 'disco' na imagem para melhor distinguir os elementos.
- Várias propriedades da imagem são extraídas para posterior uso na segmentação.
- Utilizando as propriedades extraídas são gerados rectângulos a volta de cada objecto de forma a segmentar e são contados quantos elementos diferentes tem a imagem.

3 Thresholding

Uma das partes mais importantes foi escolher o threshold que foi utilizado para converter a imagem de *gray-scale* para preto e branco. Isto é importante pois quanto melhor for a conversão melhor é possível separar os diversos objectos. Enquanto que ruído, pouco contraste entre os objectos e o fundo, objectos sobrepostos, dificultam muito o processo.

3.1 Thresholding - RGB

Uma das técnicas adotadas para melhorar a conversão das imagens de cinza para preto e branco foi, na imagem a cores, separar em 3, uma para vermelho, uma para verde e uma para azul. Depois em cada uma fazer a conversão para cinza e depois para preto. No final juntam-se as 3 em uma única imagem de preto e branco.

Isto foi feito pois ao aplicar um único threshold na conversão da imagem cinza para preto, há muitos objectos que não ficam bem delimitados, o que faz com que o resto do processo gere objetos que vieram do ruído de fundo, por pouco contraste ou por algum outro motivo.

Há também objectos que não são bem delimitados porque o threshold não é bom o suficiente.

Outro fenômeno que acontece muito é o de objectos com cores parecidas ao fundo serem misturados na conversão(pouco contraste).

Por exemplo uma imagem de bolas de várias cores, com uma bola de cor amarela num fundo muito claro, ao converter para preto e branco com um threshold global faz com que muitas vezes essa bola e o fundo sejam convertidos para branco, o que depois ao segmentar faz com que não sejam reconhecidas.

No caso de imagens com um bom contraste entre os objectos e o fundo isso não é necessário, mas na maioria das imagens testadas deu melhor resultado com 3 thresholds(sendo imagens originalmente a cores).

4 III - Preenchimento e estruturação morfológica

Depois de ter uma imagem a preto e branco, com thresholds aceitáveis, o que se faz é preencher as zonas limitadas de branco. O resultado disso por exemplo no caso das imagens de moedas é que as zonas circulares, com algum ruido passam a ser preenchidas a branco formando uma unica zona, e tudo que é fundo deve ficar a preto.

Contudo esse processo não é perfeito e para o melhorar, de seguida usa-se um elemento morfológico estruturante, do tipo 'disco' para separar ainda melhor os objectos entre eles e também do fundo .

Em resumo, usando um elemento do tipo 'disco', selecionam-se as zonas pertinentes, de seguida faz-se uma erosão e uma dilatação para separar melhor, no caso do exemplo das moedas, como haviam muitas sobrepostas esse processo ajuda a separar .

5 Propriedades e segmentação

Com a imagem a preto e branco com os objetos já delimitados (na cor branca e o fundo na cor preta) passamos então a extrair várias propriedades que vão ser usadas para fazer a segmentação da imagem, e a contagem do número de objectos. Esse processo foi feito em *MATLAB* usando a função *regionprops* que extrai informação estatística da imagem.

Neste caso o que queríamos fazer era, desenhar um rectangulo que englobasse o volume de cada objecto e fazer uma contagem dos objetos. Para detectar cada um usamos a opção de detectar os centros para o *regionprops*, chamada 'centroid', além disso marcamos com uma *label* cada um deles. No final a contagem é simplesmente o número de *textit{tlabels}* que foram atribuídos, e para delimitar os objectos na imagem original só temos que desenhar um rectangulo seguindo cada uma das áreas que foi extraida.

6 Resultados

De seguida são mostrados alguns dos resultados e comentários para cada um deles.

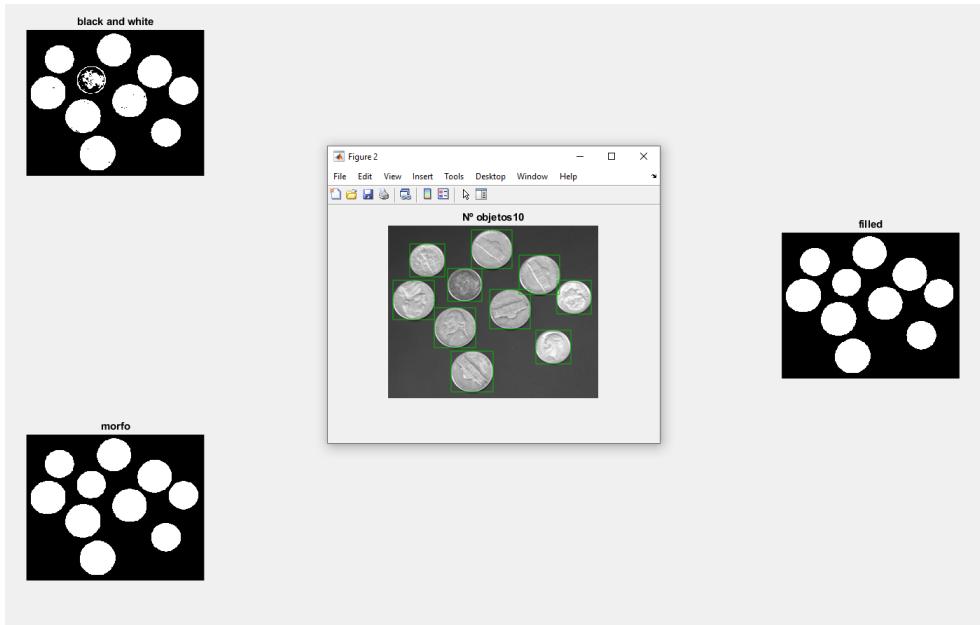


Figura 1: Resultado - coinspng.png

Resultado numa imagem de moedas, sem ruído .

A imagem em questão não tem moedas sobrepostas, além do fundo ser bastante uniforme e haver um bom contraste entre o mesmo e as moedas.

Nestas condições a aplicação do algoritmo é bastante satisfatória .

Vale notar que a imagem já vinha em tons de cinzento, e neste caso o utilizador já indicou isso no início do programa. Sendo assim a imagem é logo convertida para preto e branco como mostrado na figura com título 'black and white' .

De seguida é feito o preenchimento com a cor branca para eliminar ruidos e delimitar as formas das moedas na imagem original e depois utilizando a estrutura morfológica de 'disco' esse resultado é ainda melhorado('morf'), neste caso não muito pois a imagem foi fácil de tratar e já tinha bons resultados logo quando se faz o preenchimento('filled').

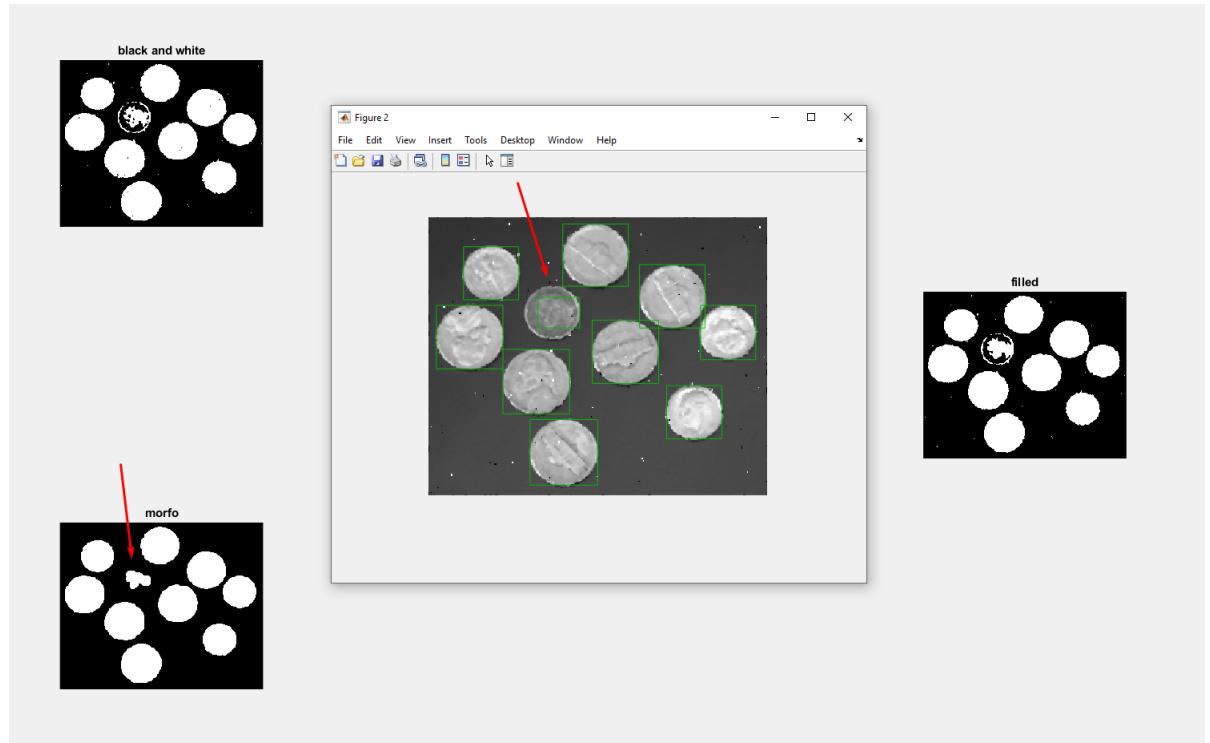


Figura 2: Resultado - coins.png com ruído salt and pepper

Resultado na mesma imagem mas onde foi introduzido ruído do tipo salt pepper e posteriormente feita uma filtragem com um filtro de mediana, que foi escolhido pela sua simplicidade e bons resultados. Nota-se que uma das moedas foi mal delimitada por causa do ruído introduzido.

De seguida são mostrados alguns resultados de imagens a cores, onde o processo inicial é ligeiramente diferente pois as imagens são separadas em R,G,B antes de fazer o threshold, cada uma com o seu próprio threshold e depois são sobrepostas sobre a mesma imagem. O processo então segue conforme seria nas imagens a cinzento .



Figura 3: Resultado intermédio - coins3.png

Imagens intermédias, as 3 primeiras são a separação em R,G,B onde são convertidas para preto e branco cada uma com um threshold adequado.

A 4º imagem é a soma destas 3, que é a que se vai utilizar no resto do processo.

De seguida só é feita uma inversão de cores para se trabalhar com fundo preto. O resto do processo é o mesmo .

De notar que uma das moedas devido ao pouco contraste com o fundo não foi bem delimitada. Além disso outras duas estão muito próximas, isso resolve-se modificando os thresholds e o tamanho do disco que se usa como objecto morfológico estruturante. Porém depois de vários testes nunca se obteve um resultado perfeito.

A moeda mais superior sempre foi mal delimitada devido ao pouco contraste, e caso se ajuste os thresholds para a delimitar melhor , as outras acabam por ficar mal delimitadas. No entanto o resultado foi bom no geral.



Figura 4: Resultado final - coins3.png

Resultado final, as moedas não foram bem delimitadas dada a sobreposição que se verificou anteriormente. Isso foi devido à má escolha de thresholds. Além disso a moeda de 10 cêntimos não foi bem segmentada, como era esperado ao ver a imagem que se gerou a preto e branco

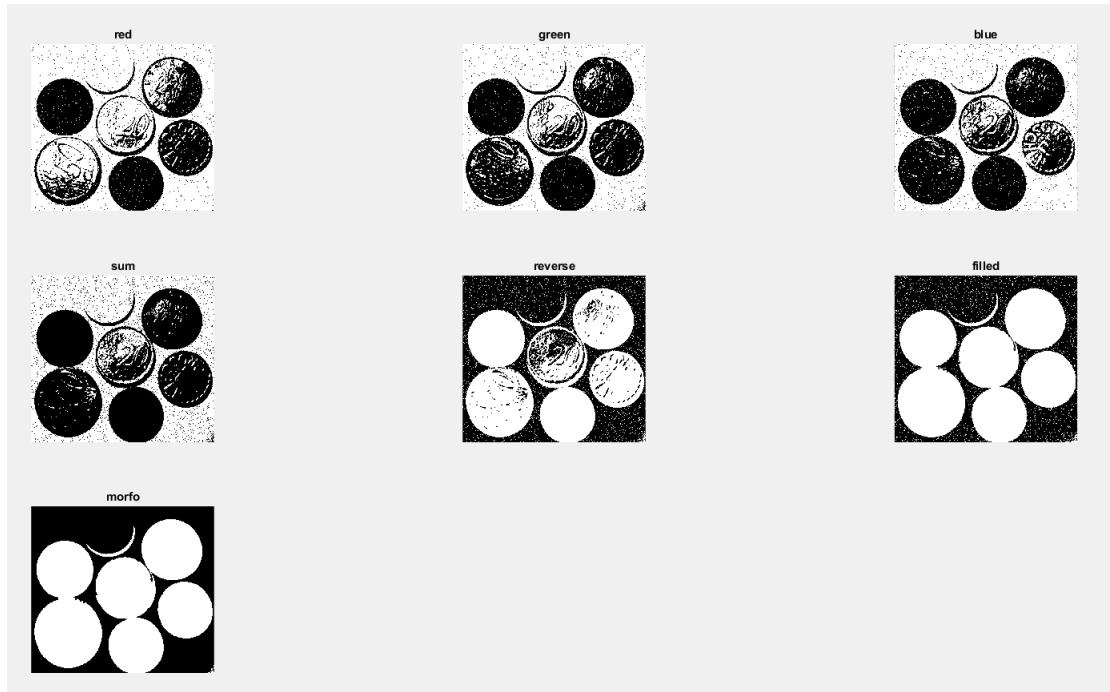


Figura 5: Resultado final - coins3.png

Mesma imagem mas agora com ruído salt and pepper



Figura 6: Resultado Final - coins3.png com ruído

Resultado final . Pelo ruído foram introduzidos alguns aterfactos , mas no geral não alterou o resultado. No entanto alterou o número de moedas contabilizadas.



Figura 7: Resultado intermédio - toys.jpg

Mesmo teste mas com uma imagem de brinquedos.



Figura 8: Resultado Final - toys.jpg

Resultado satisfatório, havia bom contraste e pouca sobreposição. No entanto alguns brinquedos foram separados (gorro do macaco) quando deveriam ter ficado como o mesmo objecto e outros não foram separados, como as peças de legos e deveriam ter sido, isso foi devido à sombra perto deles que faz com que seja difícil a separação. É possível corrigir boa parte destes erros com outros thresholds e modificando a forma do objecto estruturante para rectangular(possivelmente).

Referências

1. MATLAB help page.
<https://www.mathworks.com/>.