

# Sistemas de Visão e Percepção Industrial

## Aula Prática nº 8

Separação e classificação elementar de objetos

# Sumário

- 1 Operações básicas para separação de objetos
- 2 Separação por fator de forma
- 3 Separação por fator de forma e solidez
- 4 Separação por outros descritores e sua combinação

# Exercício 1 – operações comuns preparatórias

Na imagem 'nuts2a.jpg', e usando as funções disponíveis no Matlab, proceder às seguintes operações:

- Binarização, preenchimento de buracos, remoção de objetos no bordo



- Contar o número de objetos através do número de pontos isolados do esqueleto reduzido obtido com "shrink"; (são 12 objetos)
- Em alternativa, usar a função `bwlabel()` para "separar" os objetos. N.B.: Recomenda-se uma operação de "close" antes da operação de separação.
- Representar cada objeto numa imagem isolada (*subplot* de 2x6 ou 3x4, etc.).



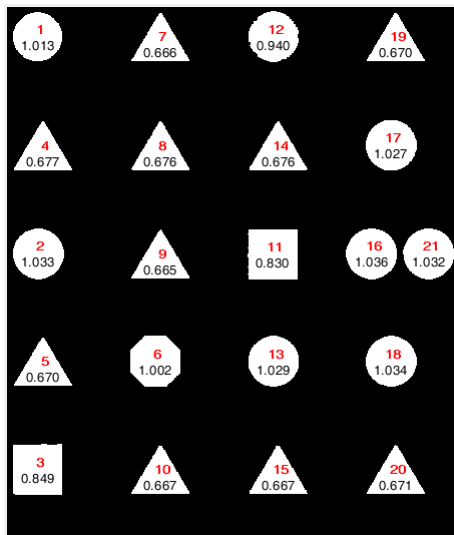
## Exercício 2 – Segmentação por categorias

- Por análise da circularidade (fator de forma) dos diversos sinais ilustrados na imagem "traffic\_signs.jpg", separá-los nas respectivas categorias (circulares, triangulares, outros.).
- Sugestão: usar a respetiva propriedade do `regionprops()`.
- Passos recomendados para a resolução nas páginas seguintes.



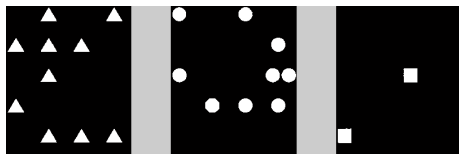
## Exercício 2 – Passos na resolução

- Conversão e pré-processamento da imagem
  - `rgb2gray()`, ...
- Binarização e eventuais operações morfológicas
  - `imbinarize()`, `open/close`, ...
- Obtenção das máscaras dos objetos (regiões)
  - `imfill()`,  
`imclearborder()`, ...
- Etiquetagem das regiões
  - `bwlabel()`, ...
- Obtenção das propriedades das regiões e obter o seu fator de forma (circularidade)
  - `regionprops()`, ...



## Exercício 2 – Etapa final

- Separar e agrupar as regiões em 3 categorias conforme o seu fator de forma (ff);
- Indicam-se os valores teóricos de ff mas devem usar-se intervalos e não valores exatos;
  - Triangulares:  $ff \approx 0.61$
  - Circulares:  $ff \approx 1$
  - Quadrados:  $ff \approx 0.79$
- No código indicado, que valores se podem propor para trLim e cirLim?

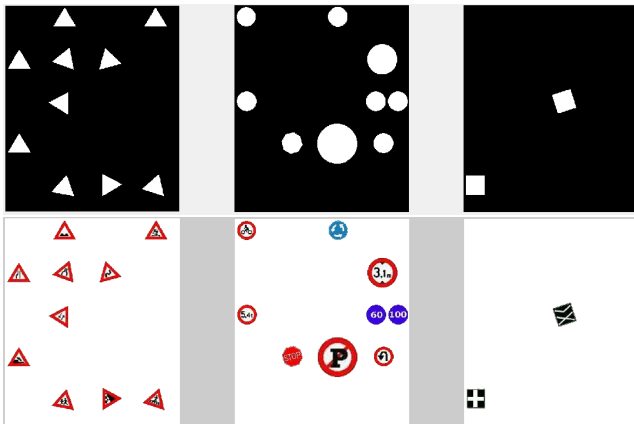


```
...  
ff=[s.Circularity];  
  
triidx=find(ff < trLim);  
ciridx=find(ff > cirLim);  
squidx=find(ff > trLim & ff < cirLim);  
  
TRI =ismember(L, triidx);  
SQUA=ismember(L, squidx);  
CIRC=ismember(L, ciridx);  
  
figure  
subplot(1,3,1), imshow(TRI); title('Triangles');  
subplot(1,3,2), imshow(CIRC);title('Circles');  
subplot(1,3,3), imshow(SQUA);title('Squares');  
...
```

**Nota:** Devido à definição de 'Perimeter' usada pelas versões de Matlab depois de 2014, e pelo facto de os objetos serem de pequenas dimensões nesta imagem, é possível obter fatores de forma ligeiramente maiores de que 1, o que diverge da definição teórica para formas circulares ideais.

## Exercício 3 – Classificação de sinais

- Testar o programa anterior na imagem `traffic_signs_jam1.jpg` e verificar se se consegue a separação nas categorias abaixo:
  - Sinais de perigo (triangulares)
  - Sinais de prescrição obrigatória (*circulares*)
  - Outros sinais



## Exercício 4a – Obtenção de descritores individuais

- Na imagem `talheres_individuais.jpg`, determinar os seguintes descritores de cada um dos objetos da figura:
  - Fator de forma
  - Solidez



Solidity=0.6935  
Form factor=0.1098



Solidity=0.7561  
Form factor=0.2468



Solidity=0.7220  
Form factor=0.2433



## Exercício 4b – Classificação de objetos

- Usando apenas os descritores de "solidez" e "fator de forma" calculados no exercício anterior, implementar um programa para separar automaticamente a imagem "talheres.jpg" em colheres, garfos e facas:
- Notas:
  - Descartar os objetos que estão no bordo da imagem.
  - Testar ambos os descritores.
  - Permitir tolerância nos valores dos descritores (e.g.  $\pm 5\%$ ) para acomodar eventuais flutuações na imagem.



# Exercício 4b – Ilustração de situações falhadas

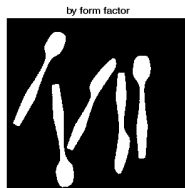
- Casos em que uma separação com as tolerâncias indicadas falha.



Objects type 1  
tol =  $\pm 2.0\%$



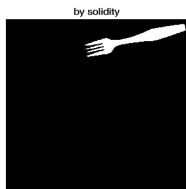
Objects type 2  
tol =  $\pm 2.0\%$



Objects type 3  
tol =  $\pm 2.0\%$



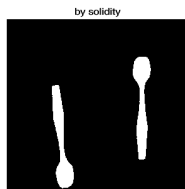
Non classified objects



Objects type 1  
tol =  $\pm 0.5\%$



Objects type 2  
tol =  $\pm 0.5\%$



Objects type 3  
tol =  $\pm 0.5\%$



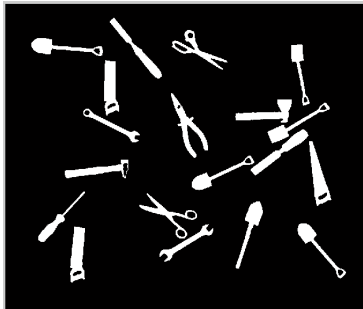
Non classified objects

- Procurar usar outras tolerâncias para fazer uma separação adequada.

## Exercício 5 – Preparação para classificação

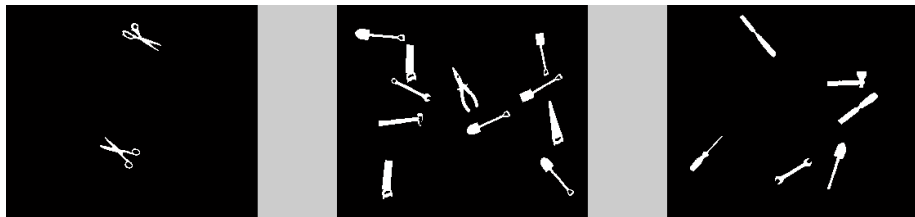
O objetivo é separar os objetos das imagens presentes na sequência Seq1.zip por propriedades geométricas.

- Resolver para uma imagem (TP2\_img\_01\_01.png) com a intenção de aplicar nas restantes;
- Converter para cinzento e binarizar a um nível adequado para captar até os objetos mais escuros.
- Eliminar todos os objetos do bordo.
  - NB. Pode-se descartar os objetos com área menor que 100 pixels.



## Exercício 6 – Número de Euler

- Criar máscaras para separar os objetos por categorias conforme o número de furos:
  - 2, 1 ou 0 (usar EulerNumber).



NB. O martelo foi mal classificado devido a uma imagem pobre e de binarização difícil.

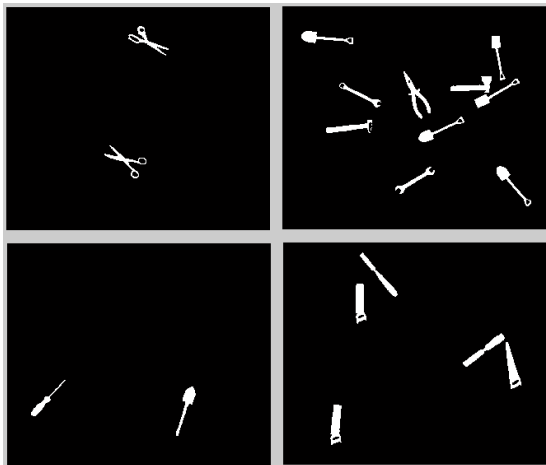
# Questões de programação

Para agilizar o processo de programação e resolução rápida dos exercícios seguintes sugere-se o uso de código reutilizável. Pode-se admitir que se tem sempre um número fixo de categorias para cada descritor (4, por exemplo) e que portanto há 4 intervalos. Se a imagem pronta a ser analisada for B, pode-se usar um código similar ao seguinte para visualizar as separações de objetos por intervalos. Para outros descritores e intervalos será necessária uma pequena adaptação.

```
L=bwlabel(B); %obter matriz de 'labels'
s=regionprops(L,'All'); %obter lista das propriedades todas
soli=[0    0.5  0.6    0.7    1]; %limites dos intervalos
lins=2, cols=2; %medidas para o subplot
for i=2:numel(soli)
    idx=find([s.Solidity]>soli(i-1)&[s.Solidity]<=soli(i));
    m=ismember(L,idx); %imagem binaria dos objetos detetados
    subplot(lins,cols,i-1); imshow(m);
    str=sprintf('Sol>%0.2f&Sol<=%0.2f',soli(i-1),soli(i));
    title(str);
end
```

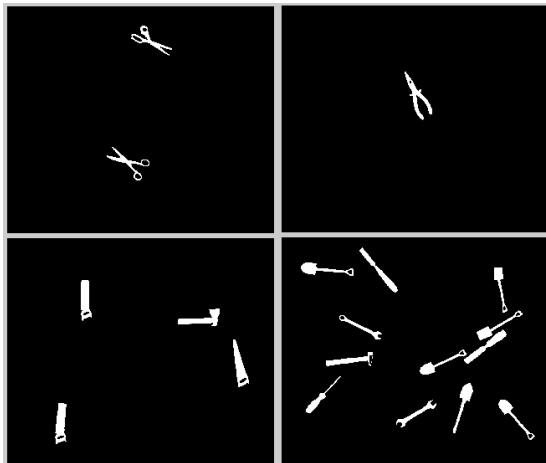
## Exercício 7 – Solidez

- Separar os objetos pela sua solidez (solidity) nos seguintes intervalos:
  - 0.0 - 0.5 - 0.6 - 0.7 - 1.0



## Exercício 8 – Excentricidade

- Separar os objetos pela excentricidade (eccentricity) da elipse envolvente nos seguintes intervalos:
  - 0.0 - 0.94 - 0.96 - 0.98 - 1.00



## Exercício 9 – Fator de Forma

- Separar os objetos pelo fator de forma (circularity) nos seguintes intervalos:
  - 0.0 - 0.15 - 0.2 - 0.3 - 1.0

