



Finding BLOBS in LEGOLAND®

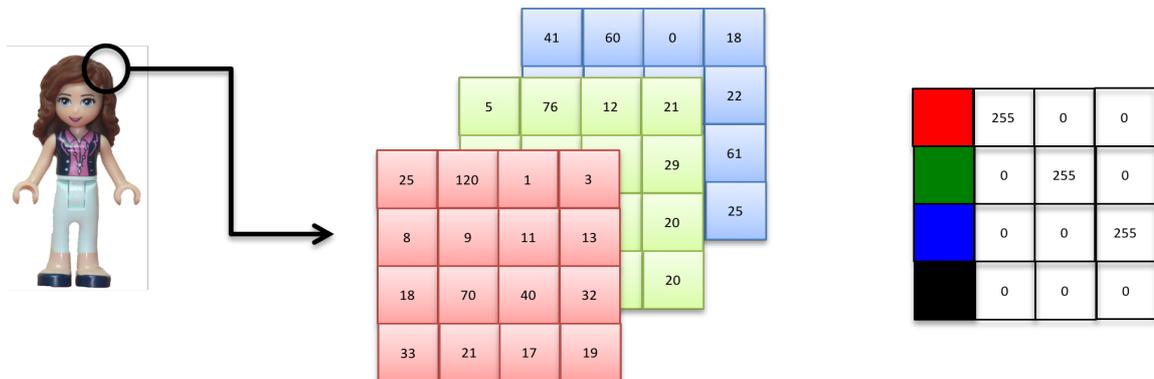


"A cidade de LEGOLAND está muito animada! Várias pessoas passeiam a pé e de bicicleta pela praça central, bem próximo do hotel e do hospital! No centro da praça, a fonte dá água fresca a todos as personagens desta fantástica cidade! Vendem-se gelados e até o Bóbi e o Tareco brincam alegremente na rua"

No entanto, por segurança, toda a cidade está vigiada por um conjunto de câmaras de vídeo-vigilância, que monitorizam toda a atividade da cidade. O objetivo é que, caso algo errado aconteça, se possam analisar as imagens e descobrir o culpado..."

Neste trabalho pretendem-se definir estruturas de dados que permitam procurar por "BLOBS" em imagens, com base em informação de cor e num determinado valor de "tolerância de pesquisa". No domínio da imagem digital, um BLOB define-se como uma região da imagem que difere das restantes em termos de alguma propriedade (por exemplo, textura ou brilho). Neste trabalho, por questões de simplicidade, iremos analisar simplesmente a cor de cada ponto (pixel) da imagem.

Como saberá, uma imagem é representada digitalmente por um conjunto bidimensional (matriz) de pontos (chamados píxeis), cada um com 3 valores numéricos (0..255) no caso de se tratar de imagem de cor.

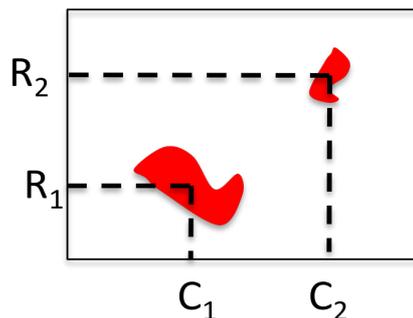




Assim, o objetivo deste trabalho consiste na leitura de um ficheiro de texto que representa um conjunto de imagens, e na pesquisa por regiões (BLOBs) de cada uma das imagens que correspondam a uma determinada cor, especificado pelo utilizador.

Após a leitura dos ficheiros de dados, o sistema deverá começar por pedir ao utilizador informação sobre uma cor (R, G, e B) e uma tolerância (d) deverá devolver as posições centrais na imagem que contém regiões conectadas próximas dessa cor.

Por exemplo, considerando a seguinte imagem ("teste.jpg"):



O sistema deveria começar por receber a cor a procurar e o intervalo de tolerância:

R=255
G=0
B=0

d=10

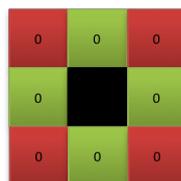
De seguida deveria procurar regiões da imagem cujos pixéis estivessem no intervalo:

RGB=[255 ± 10, 0 ± 10, 0 ± 10]

De seguida, considerando **4-vizinhança**, deveria identificar as regiões com centro em (R₁, C₁) e (R₂, C₂) como as que correspondem ao intervalo pretendido. Finalmente, deveria ser mostrada a seguinte listagem (ordenada decrescentemente pelo tamanho das regiões):

teste.jpg; (R₁, C₁): 240 pixéis
teste.jpg; (R₂, C₂): 85 pixéis

Nota 1: A definição de 4-vizinhança diz que, para uma dada posição, apenas os pixéis localizados imediatamente acima, abaixo, à esquerda e à direita são considerados seus vizinhos, tal como ilustrado pelas posições a verde no exemplo abaixo. Pelo contrario, as posições a vermelho não são consideradas vizinhas da posição a preto.





Nota 2: O ficheiro de texto a ler tem o seguinte formato:

Nome da imagem 1
Nº de linhas + Nº de colunas + Nº de canais
Pixel (linha 1, coluna 1, canal 1)
Pixel (linha 1, coluna 1, canal 2)
Pixel (linha 1, coluna 1, canal 3)
Pixel (linha 1, coluna 2, canal 1)
...
Pixel (linha 1, coluna 200, canal 3)
Pixel (linha 2, coluna 1, canal 1)
...
Pixel (linha 300, coluna 200, canal 3)
Nome da imagem 2
...

Exemplo de ficheiro de texto:

Teste.jpg
10 10 3
150
112
135
121
...