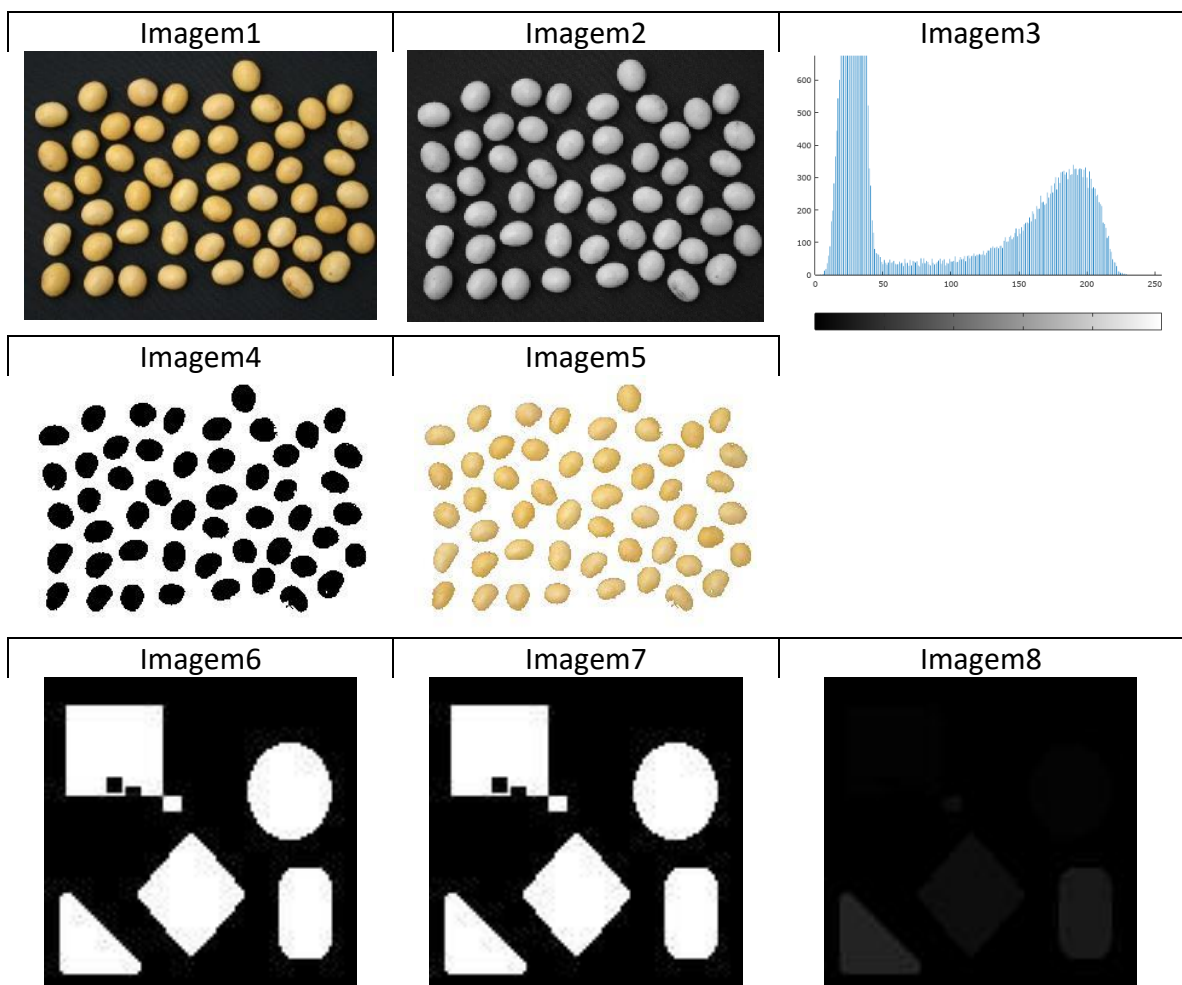




Prática 3 – Laboratório de PDI

Tópicos: limiarização por corte de histograma, rotulação componentes conectados, identificação regiões e contagem de elementos.

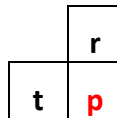
Imagens utilizadas



1. Ler a imagem soja.jpg na variável im (Imagem1) e mostrar em uma janela com o título “Imagem original: soja”;
2. Converter a imagem im para tons de cinza (Imagem2) e mostrar seu histograma (imagem3);
3. Realizar a limiarização da imagem soja a partir do corte do histograma (Imagem4). Verifique, observando o histograma, qual seria o melhor valor para utilizar como corte (limiar) e faça a limiarização utilizando este valor (fixo).

4. Segmente a imagem original (Imagem1) utilizando a imagem limiarizada (Imagem4) como máscara, gerando a imagem final (Imagem5). Mostre a imagem final.
5. Ler a imagem objetos1.jpg na variável im2 (Imagem6) e mostrar em uma janela com o título "Imagem original: objetos PB";
6. Transformar a imagem original (im2) em uma imagem binária (imBW), considerando 1-branco e 0-preto (não utilizar a função im2bw do Octave);
7. Mostrar, em intensidades de cinza diferentes, os objetos individuais da imagem conforme Imagem3. Para este passo, observe as informações a seguir:

- Considerando vizinhança-4, temos o seguinte esquema explicativo:



- Seja p um pixel a ser analisado e sejam r e t os pixels de cima e da esquerda, respectivamente. Considerando que a varredura se dará da esquerda para a direita e de cima para baixo, quando o pixel p for analisado, r e t já terão sido analisados e rotulados (se tiverem satisfeito o critério de similaridade $C_s=1$).
- Procedimento a ser seguido pelo algoritmo:
 - Se $p = 0$ então verifica o próximo pixel;
 - Se $p = 1$, examina r e t
 - Se $(r = 0 \text{ e } t = 0)$ então rotula p com **novo rótulo**;
 - Se $(r = 1 \text{ e } t = 0)$ ou $(r = 0 \text{ e } t = 1)$ rotula p com **o rótulo de r ou de t** (o que é igual a 1);
 - Se $(r = 1 \text{ e } t = 1)$ e possuem o mesmo rótulo então rotula p com **este rótulo**;
 - Se $(r = 1 \text{ e } t = 1)$ e possuem rótulos diferentes então rotula p com **um dos rótulos** e armazena **ambos os rótulos**, que são diferentes mas que marcam uma mesma região (equivalências de rótulos) e, portanto, deveriam ser iguais, para posterior correção de erro.

No final do procedimento, todos os pixels que satisfizeram o critério de similaridade C_s estarão rotulados, mas alguns com rótulos iguais (indicando uma mesma região, considerando vizinhança-4).

- Percorrer a imagem removendo equivalências de rótulos (erros);
 - Ao final, percorrer novamente a imagem substituindo o valor de cada pixel de um mesmo rótulo por uma intensidade de cinza diferente atribuída a sua região.
8. Finalizando, mostrar a quantidade de objetos (regiões distintas do *foreground*) encontrados na imagem.