

Trabalho de Processamento de Imagens

Atividade 1 - Imagem distância e rotulação de componentes conexos

Objetivo

O objetivo dessa atividade é explorar os conceitos iniciais da disciplina de processamento de imagens como: varredura da imagem, componentes conexos, distância de pixels, 8 e 4-conectividade e 8 e 4-adjacência.

Problema

Considere a Figura 1 de uma bandeja de feijões:

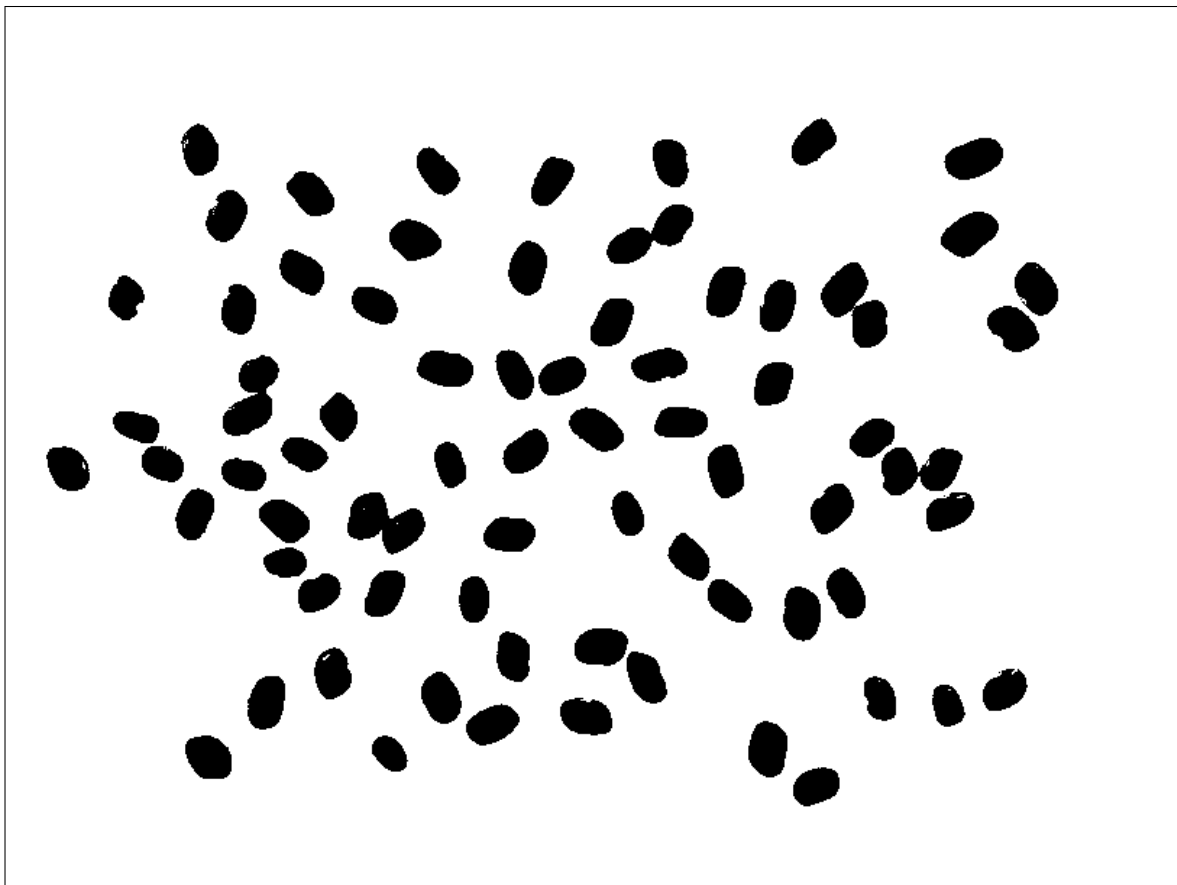


Figura 1: Imagem de feijões

Uma sequência de transformações de processamento de imagens podem ser úteis para resolver alguns problemas práticos, como o desse trabalho. Considere que o seu problema é desenvolver um programa de processamento de imagens para contar feijões, conforme imagem.

Alguns algoritmos já estudados devem ser utilizados para esse problema, como: limiarização, rotulação de componentes conexos e imagem distância. Por exemplo, a partir de uma imagem

em tons de cinza é possível produzir uma imagem binária separando a forma (feijão) do fundo, usando o algoritmo de limiarização. Para atribuir um rótulo diferente para cada região conexa (feijão) pode-se utilizar o algoritmo de rotulação de componentes conexos que atribui um rótulo diferente para cada região. Ainda existe o problema de alguns feijões muito próximos aparecerem como uma única região conexa. Uma alternativa para resolver isso é utilizar o algoritmo que calcula a distância dos pixels de uma forma para o fundo. Se tomarmos somente um pequeno raio da imagem distância pode-se separar os centros de cada feijão para ter um resultado de contagem mais preciso.

Descrição

1. Desenvolva um programa, baseado no fonte que calcula o negativo de uma imagem disponível no MOODLE, para: (a) carregar a imagem em formato PGM; (b) contar as formas (feijões) (usando os algoritmos de rotulação de componentes conexos, limiarização e imagem distância) e (c) apresentar o resultado.
2. O programa deve obter, na linha de comando, o nome do arquivo e apresentar como resultado uma linha com o seguinte texto:

```
#componentes= XX
```

Onde XX é o número de componentes conexos encontrado na imagem. Esse formato é fundamental para permitir a correção automática do programa.

3. Opcionalmente, pode ser gerada uma imagem PGM, resultante da rotulação de componentes conexos.
4. Experimente o programa para vários arquivos diferentes.

Entrega

1. Incluir um comentário no cabeçalho de cada programa fonte com o seguinte formato:

```
1  /*-----
2  *          UNIFAL – Universidade Federal de Alfenas.
3  *          BACHARELADO EM CIENCIA DA COMPUTACAO.
4  * Trabalho...: Contagem de feijoes
5  * Professor.: Luiz Eduardo da Silva
6  * Aluno.....: Fulano da Silva
7  * Data.....: 99/99/9999
8  *-----*/
```

2. O projeto deverá incluir um arquivo MAKEFILE para construção do executável que deverá ser nomeado de **contafeijao**.
3. O programa deverá ser chamado em linha de comando da seguinte forma (com a imagem de entrada como parâmetro), como por exemplo:

```
1  ./contafeijao <nomedaimagem [.pgm]>
```

4. Enviar num arquivo único (.ZIP), com todos os arquivos fonte do projeto através do Envio de Arquivo do MOODLE.

5. Não utilizar qualquer biblioteca com os algoritmos. Utilizar as implementações dos algoritmos discutidos em aula.