

INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
CEARÁ
Campus Maracanaú

Instituto Federal de Educação, Ciência e Tecnologia do Ceará
Campus Maracanaú
Coordenadoria de Computação
Curso de Bacharelado em Ciência da Computação
Disciplina: Processamento Digital de Imagens
Professor: Igor Rafael Silva Valente

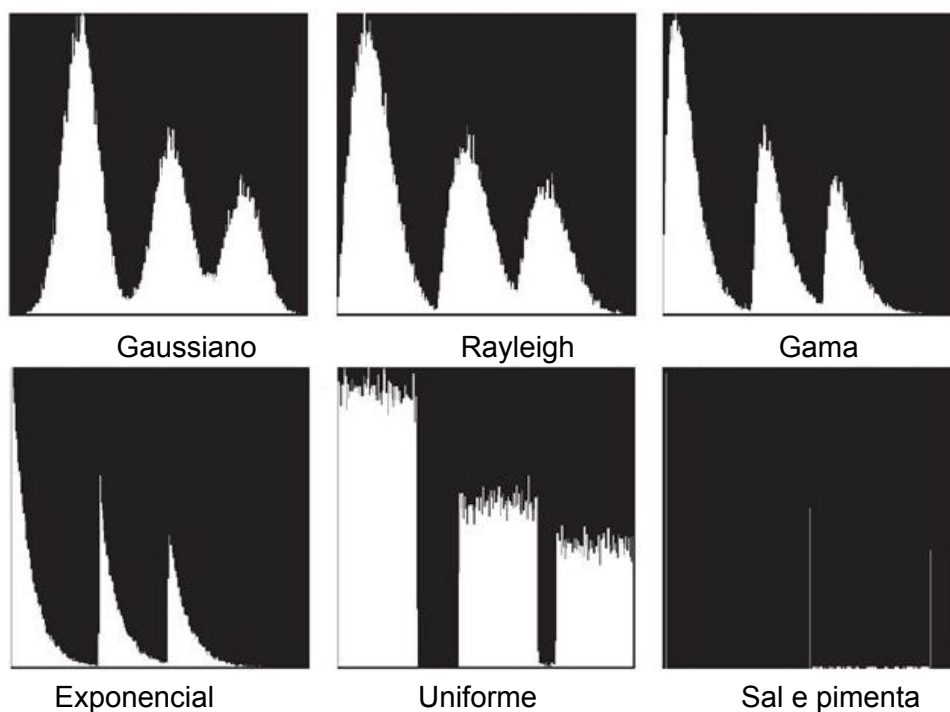
AVALIAÇÃO 02

Semestre: 2020.1 (remoto) Data: 14/09/2020

Nome completo:

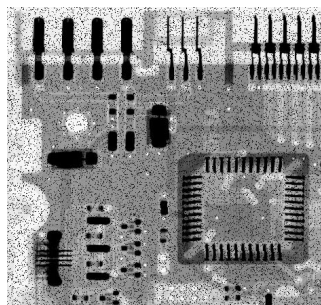
Raul Aquino de Araújo

1. (1 ponto) Uma imagem de teste contendo três regiões homogêneas de tons de cinza foi corrompida com diversos tipos de ruído (um ruído por vez), gerando os histogramas apresentados a seguir:

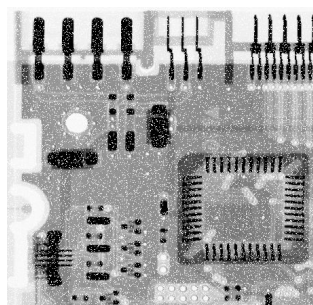


Através da análise de cada histograma, identifique qual tipo de ruído foi utilizado para corromper a imagem: gama, rayleigh, gaussiano, uniforme, sal e pimenta ou exponencial.

2. (2 pontos) Os filtros de estatística de ordem máximo e mínimo têm como objetivo encontrar, respectivamente, os pontos mais claros e mais escuros de uma imagem. Desta forma, são úteis para restaurar imagens com ruído impulsivo. Utilizando a ferramenta Octave e as imagens 'circuit-board-salt-prob-pt1.tif' e 'circuit-board-pepper-prob-pt1.tif', faça o que se pede:



circuit-board-pepper-prob-pt1.tif



'circuit-board-salt-prob-pt1.tif'

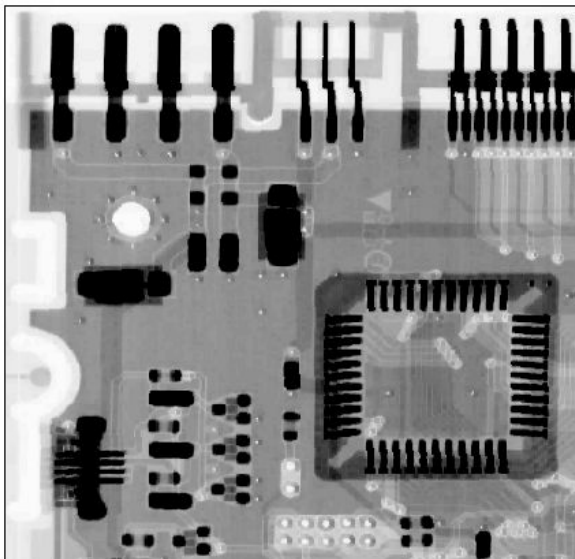
- a. Realize a restauração das imagens originais, conforme os passos descritos a seguir:
 - i. Aplique os filtros máximo e mínimo a cada uma das imagens originais
 - ii. Apresente, lado a lado, a imagem original e o resultado da restauração
 - iii. Descreva o código utilizado em cada uma das operações de restauração
 - iv. Explique os resultados

```

1 clear; clc; clear all; %limpeza
2 pkg load image; %carregando pacote de imagem
3 wSize = 3; %para o tamanho da máscara
4 %carregando imagem com ruído de sal
5 S = imread('circuit-board-salt-prob-pt1.tif');
6 %carregando imagem com ruído de pimenta
7 P = imread('circuit-board-pepper-prob-pt1.tif');
8 %função para o mínimo
9 min_func = @(x) min(x(:));
10 %função para máximo
11 max_func = @(x) max(x(:));
12 %aplicando as máscaras nas imagens
13 S2 = nlfilter(S, [wSize wSize], min_func);
14 S3 = nlfilter(S, [wSize wSize], max_func);
15 P2 = nlfilter(P, [wSize wSize], min_func);
16 P3 = nlfilter(P, [wSize wSize], max_func);
17 %mostrando os resultados
18 figure,
19 subplot(1,2,1), imshow(S2), title('sal min_func'),
20 subplot(1,2,2), imshow(S3), title('sal max_func');
21 figure,
22 subplot(1,2,1), imshow(P2), title('pimenta min_func'),
23 subplot(1,2,2), imshow(P3), title('pimenta max_func');

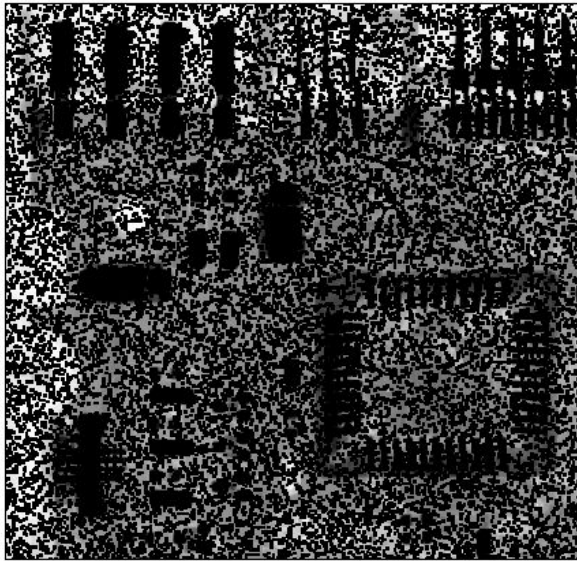
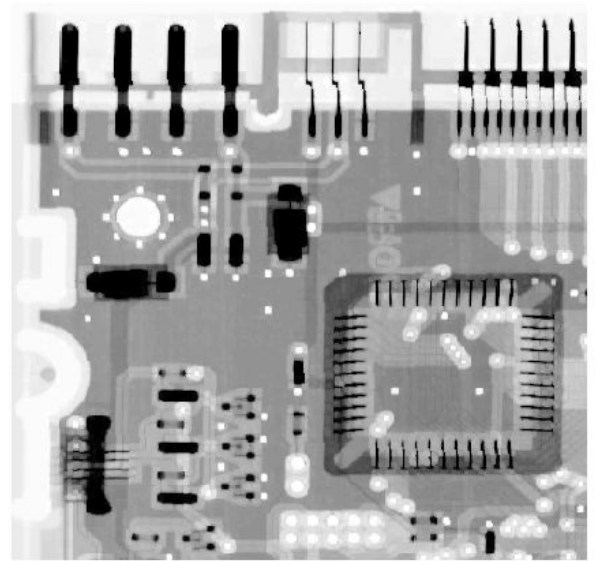
```

sal min_func



sal max_func



pimenta min₁uncpimenta max₁unc

Como podemos observar na imagem que tem o ruído de sal, ao aplicar o filtro de de máx temos que os valores que predominam são os valores brancos, piorando assim a imagem e quando aplicado a máscara com a função de mínimo já temos um resultado mais satisfatório. Para o de pimenta temos basicamente a mesma coisa que para o ruído de sal, porém ao contrário, quando aplicamos a máscara de mínimo os valores predominantes na cor preta acabam por piorar a imagem, o que não acontecem quando usamos o função max.

- b. Se a imagem original for contaminada com os dois ruídos (sal e pimenta) ao mesmo tempo, qual filtro poderia ser utilizado para realizar a restauração de forma adequada?

Um bom filtro que poderia resolver o problema seria o da média harmônica, teríamos uma imagem com boa qualidade mesmo bastante ruidosa.

3. (2 pontos) Sobre segmentação de imagens, faça o que se pede:

- a. Explique seu conceito.

A segmentação é o processo de separar os objetos constituintes de uma cena, sendo um dos passos principais que precedem a extração de atributos

Definição: Processo de subdividir uma imagem em seus objetos ou regiões constituintes. O nível de subdivisão depende do nível de precisão necessário por cada aplicação.

Determina a qualidade das tarefas posteriores de extração de características e reconhecimento de padrões

- b. Quais seus tipos?

Os algoritmos de segmentação de imagens monocromáticas são divididos em duas categorias:

Descontinuidade e Similaridade. Na descontinuidade, pressupõe-se que as fronteiras entre as regiões são suficientemente diferentes entre si e em relação ao fundo da imagem para permitir a detecção de limite com base nas descontinuidades locais em intensidade.

A segmentação baseada nas bordas é a principal abordagem desta categoria. Na similaridade, também chamada de abordagem de segmentação baseada em região, o processo é realizado através da divisão de uma imagem em regiões que sejam similares de acordo com um conjunto de critérios predefinidos (como, por exemplo, o valor de intensidade)

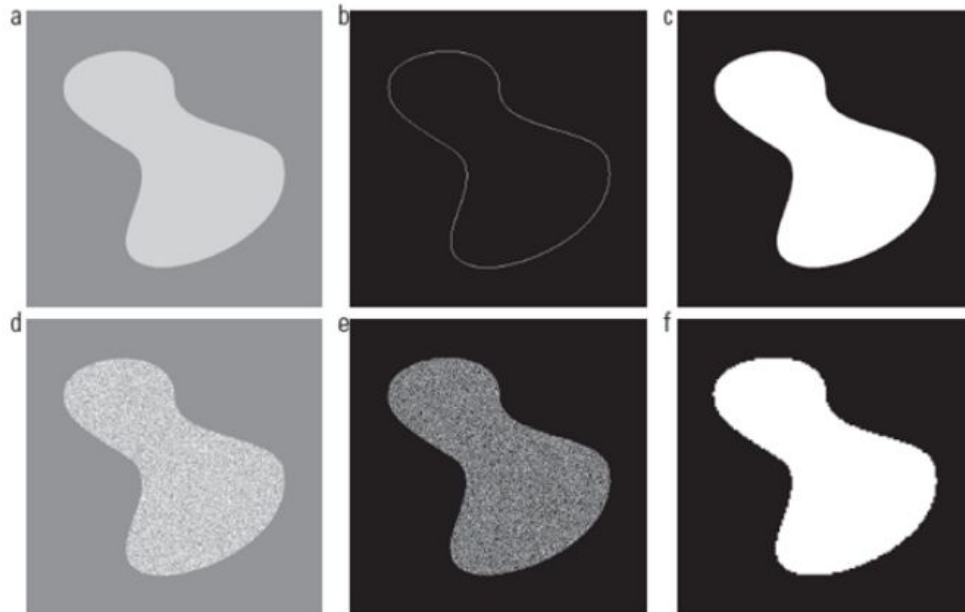


Figura 10.1 (a) Imagem que contém uma região de intensidade constante. (b) Imagem mostrando a fronteira da região interior, obtida a partir das descontinuidades de intensidade. (c) Resultado da segmentação da imagem em duas regiões. (d) Imagem que contém uma região com textura. (e) Resultado do cálculo da borda. Repare no grande número de pequenas bordas que estão conectadas com a fronteira original, o que torna difícil encontrar um limite único utilizando apenas a informação das bordas. (f) Resultado da segmentação baseada nas propriedades da região.

c. Para cada tipo, cite um exemplo real em que pode ser aplicado.

Imagens Médicas

- Localização de tumores e outras patologias;
- Medida de volume de tecidos;
- Cirurgia guiada por computador;
- Diagnóstico de doenças;

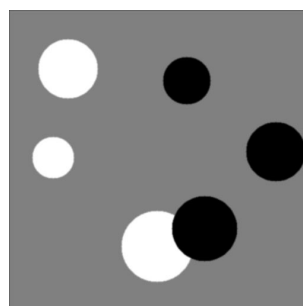
Planos de tratamento;

- Estudo da estrutura anatômica.
- Localização de objetos em imagens de satélite (estradas, florestas, etc.)
- Sistemas de reconhecimento facial;
- Sistemas de controle automático de tráfego;
- Sistemas de visão computacional.

d. Por que a segmentação é uma etapa importante na maior parte dos sistemas de PDI?

A segmentação é um passo importante para diversas aplicações que visam trabalhar com o objeto segmentado ou isolado da imagem. Portanto, a segmentação é o meio para se realizar medições de forma, de tamanho (área e comprimento) e ou identificar características do objeto como centro de massa. Uma outra aplicabilidade da segmentação é facilitar e tornar possível a mensuração da quantidade de partículas específicas presentes em determinada imagem.

4. (2,5 pontos) Utilizando a ferramenta Octave e a imagem 'circlesBrightDark.png', faça o que se pede:



'circlesBrightDark.png'

- a. Com o intuito de segmentar os círculos brancos e pretos apresentados na imagem, aplique o método de Otsu para calcular automaticamente 2 limiares

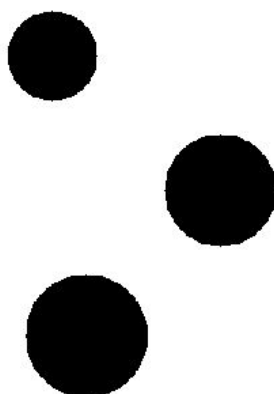

```

129 U = imread('circlesBrightDark.png');
130 limiar = graythresh(U);
131 U2 = im2bw(U,limiar);
132 U3 = im2bw(U, 158/255);
133 figure,
134 subplot(1,2,1), imshow(U2), title('a'),
135 subplot(1,2,2), imshow(U3), title('b');

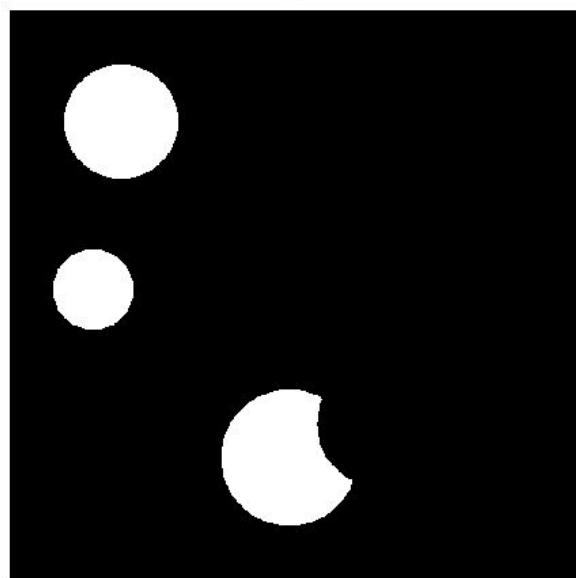
```

- b. Utilizando os 2 limiares calculados, faça a limiarização gerando duas imagens binárias/lógicas de resultado, A e B, nas quais somente os círculos brancos e pretos, respectivamente, são exibidos.

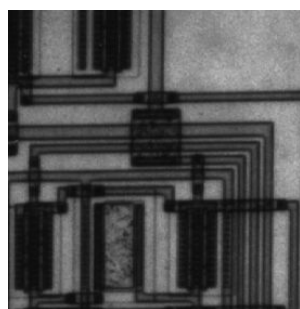
a



b



5. (2,5 pontos) Utilizando a ferramenta Octave, a imagem 'circuit.png' e os operadores de detecção de bordas horizontais e verticais de Prewitt, faça o que se pede:



'circuit.png'

-1	-1	-1	-1	0	1
0	0	0	-1	0	1
1	1	1	-1	0	1

Prewitt

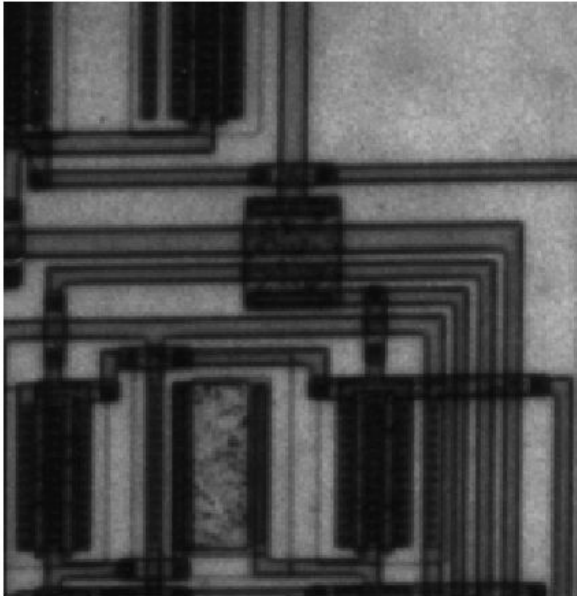
- Aplique os operadores de detecção de bordas horizontais e verticais de Prewitt na imagem 'circuit.png', gerando as imagens da magnitude do gradiente Gx e Gy
- Gere a imagem G da magnitude do gradiente a partir de aproximação usando Gx e Gy

```

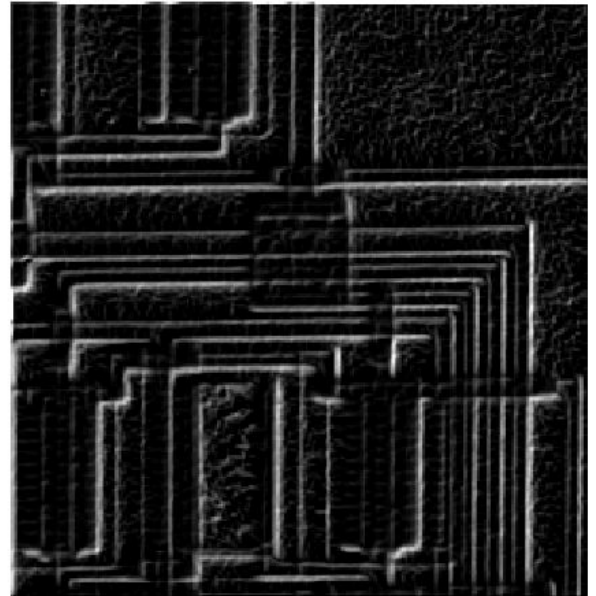
25 Z = imread('circuit.png');
26 prewittX = [-1 -1 -1; 0 0 0; 1 1 1]
27 prewittY = [-1 0 1; -1 0 1; -1 0 1]
28 Gx = im2double(imfilter(Z, prewittX));
29 Gy = im2double(imfilter(Z, prewittY));
30 G = abs(Gx) + abs(Gy);
31 figure,
32 subplot(1,2,1), imshow(Z), title('circuito original'),
33 subplot(1,2,2), imshow(G), title('prewitt');

```

circuito original



prewit



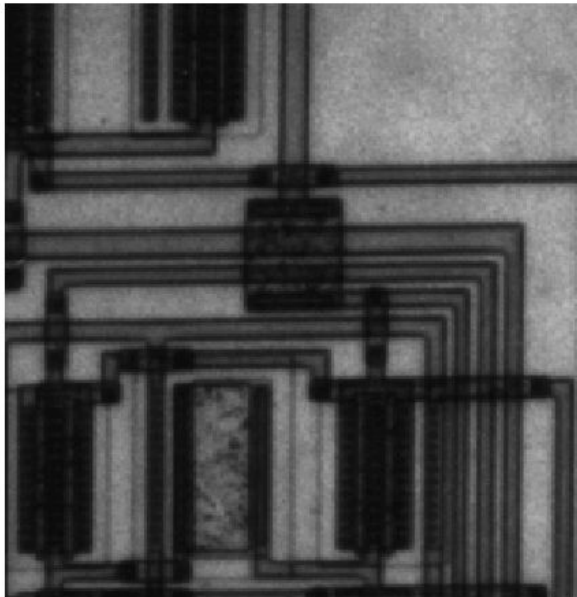
- c. Utilizando o método de Otsu, calcule automaticamente o limiar para segmentar as bordas da imagem G, gerando a imagem R, resultado da limiarização da imagem G pelo limiar calculado

```

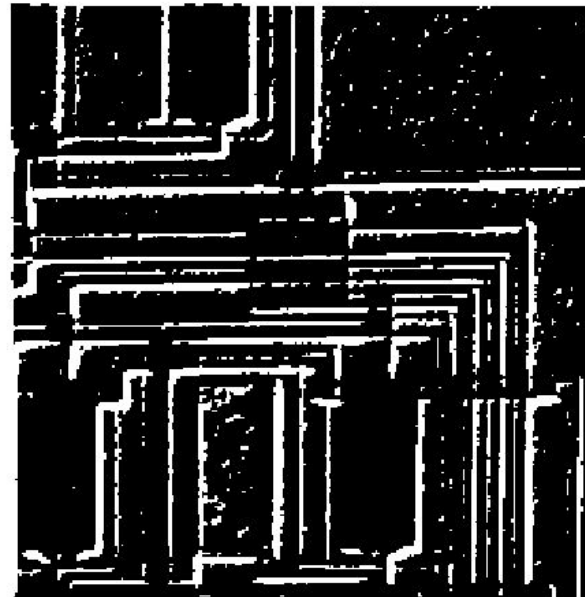
34 T1 = graythresh(G)
35 R = im2bw(G, T1);
36 figure,
37 subplot(1,2,1), imshow(Z), title('circuito original'),
38 subplot(1,2,2), imshow(R), title('limiarizado');

```

circuito original



limiarizado



Boa sorte! Prof. Igor.