



PROJETO II

Espera-se do projeto:

A clara identificação do problema a ser resolvido. Descrição completa da solução proposta e resultados. Identificação objetiva e avaliação dos resultados.

O que deve ser entregue?

- Relatório feito pelos alunos, em 3 páginas como mínimo e 5 como máximo. Detalhes sobre o relatório, ler o arquivo "LER ANTES DE FAZER OS RELATÓRIOS" no Moodle.
- Código da solução, comentado e com descrição de uso e extensão.

Questão 1

Faça um programa que utilize como entrada a imagem `morf_test.png`. O programa deve entregar uma imagem binária como saída, com o fundo branco e as imagens pretas. Teste, e mostre os resultados das seguintes operações no relatório:

2.1 A aplicação da transformada top-hat e/ou bottom-hat antes da binarização

2.2 Tentar criar uma imagem que seja somente o fundo (mediante operações morfológicas), e subtrair essa imagem da original.

2.3 Aplique filtros prévios para tentar melhorar o resultado de 2.1 e 2.2

2.4 No melhor resultado obtido aplique operações morfológicas binárias (tipo abertura, fechamento) para tentar evitar símbolos desconectados ou ruídos.

DICA: Para achar o limiar de binarização na parte 2.1 utilize a função `graythresh` no matlab ou `cvThreshold` no openCV.

Questão 2

Faça um programa para realizar o processo de filtro rejeita-**notch**, mediante o uso de filtros passa-altas Butterwoth (com $n=4$). A imagem que deve ser usada é o arquivo "MOIRE.TIF". No relatório desta parte deve estar incluído:

3.1 Uma Figura com a imagem original

3.2 Uma Figura com a imagem filtrada utilizando 4 pares notch com a seguinte características (usando padding)

3.3 Comentários sobre os resultados obtidos.

Questão 3

Em sistemas automáticos de inspeção de circuitos impressos é necessário verificar a quantidade e diâmetro dos buracos para verificar se os componentes eletrônicos podem ser inseridos sem dificuldades. Faço um programa que use como entrada a imagem 'pcb.jpg', e mediante operações morfológicas conte o número de buracos e indica seu diâmetro em pixels.

DICA: uma solução possível seria utilizando as funções *imfill* e *regionprops* do Matlab (ou equivalente em OpenCV).

No relatório desta parte deve estar:

1.1 Mostrar a imagem original, e a imagem depois de cada um dos operadores morfológicos utilizados.

1.2 O resultado do programa indicando a quantidade de buracos e diâmetro.