



**UNIVERSIDAD NACIONAL DEL LITORAL**  
Facultad de Ingeniería y Ciencias Hídricas

Processamento Digital de Imagens

**Entrega II - A**

Aluna: Carla de Oliveira Barden

Santa Fe, Abril de 2020

## Introdução

Conforme dito anteriormente, em Processamento de Imagens, operações realizadas no domínio espacial consistem na manipulação direta de cada pixel que compõe a imagem. Matematicamente,  $g(x, y) = T[f(x, y)]$ , onde  $g(x, y)$  é a imagem de saída,  $f(x, y)$  é a imagem de entrada e  $T$  é o operador.

No caso onde  $g$  depende apenas do valor de  $f$  em um único ponto  $(x, y)$ ,  $T$  é dito uma **Função de Transformação de Intensidade**, sendo apresentada na forma:  $s = T(r)$ , onde  $s$  e  $r$  são variáveis que indicam a intensidade de  $g$  e  $f$  em qualquer ponto  $(x, y)$ . Há diversos tipos de transformações de intensidade, dentre elas:

### *Operações com somente uma imagem*

- Transformações Lineares (negativo, amplificação, offset)
- Limiarização
- Transformação Logarítmica
- Transformação de Potência (Gama)
- Operação Lógica NOT (operação unária bit-a-bit)

### *Operações entre várias imagens*

- Operações Aritméticas (soma, subtração, média, multiplicação, divisão, etc)
- Operações Lógicas (AND, OR, XOR - todas bit-a-bit)
- Operações Relacionais ( $>$ ;  $<$ ;  $>=$ ;  $<=$ )

Uma das possíveis aplicações desse tipo de operação é o **realce** de características relevantes da imagem, de forma que a imagem transformada seja mais adequada à aplicação que se destina do que a imagem original.

## Problema Proposto

Ao final do processo de manufatura de placas-mãe, da marca ASUS®, modelo A7V600, obtém-se duas classes de produto final:

- A7V600-X,
- A7V600-SE.

Implementar um algoritmo que, a partir de uma imagem, determine que tipo de placa é. Usar as técnicas de realce aprendidas e as imagens a7v600-x.gif y a7v600-SE.gif. Adaptar o método proposto de forma que contemple o reconhecimento de imagens que foram afetadas por um ruído aleatório impulsivo (a7v600-x(RImpulsivo).gif y a7v600-SE(RImpulsivo).gif).

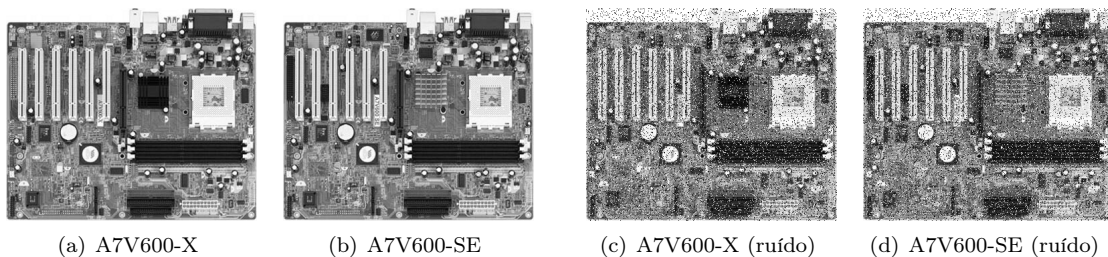


Figura 1: Imagens Fornecidas

## Solução Proposta

A solução proposta encontra-se no seguinte local: <https://tinyurl.com/wkyz599>. Para executá-la, é necessário passar como parâmetro, nesta ordem, o caminho da imagem original da placa A7V600-X, o caminho da imagem original da placa A7V600-SE e o caminho da imagem a ser analisada.

## Metodologia

Observando-se as imagens 1(a) e 1(b), não é possível identificar muitas diferenças além das indicadas na figura 2.

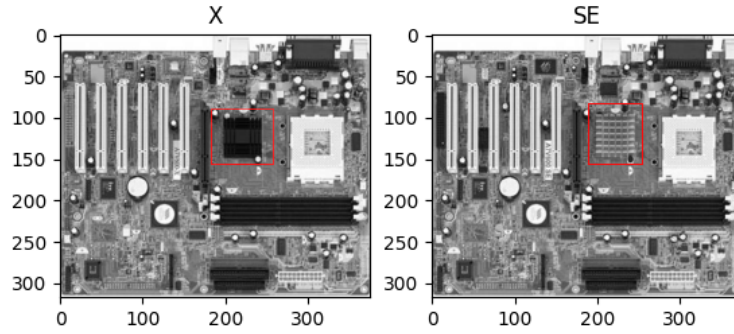


Figura 2: Diferença entre as placas facilmente identificada por um observador.

Porém, ao aplicar-se uma técnica de realce conhecida como *Limiarização*, geram-se duas imagens binárias e tornam-se mais nítidas as diferenças entre as duas placas (figura 3).

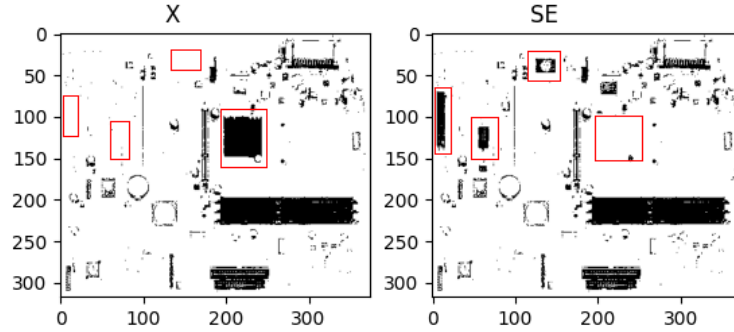


Figura 3: Diferença entre as placas após Limiarização.

Matematicamente, a Limiarização é definida como  $s = T(r)$ , onde:

$$s = \begin{cases} 0, & \text{se } r < k; \\ 255, & \text{se } r \geq k. \end{cases}$$

Para esta aplicação, o valor de  $k$  foi empiricamente definido como 50.

Sendo assim, a estratégia escolhida para a resolução do problema foi criar uma máscara para cada placa e compará-la com a imagem a ser analisada. A fim de criar as máscaras, efetuou-se a operação de *subtração* entre as imagens binárias, como segue:

$$\text{mascara } X = \text{imagem } SE \text{ binaria} - \text{imagem } X \text{ binaria}$$

$$\text{mascara } SE = \text{imagem } X \text{ binaria} - \text{imagem } SE \text{ binaria}$$

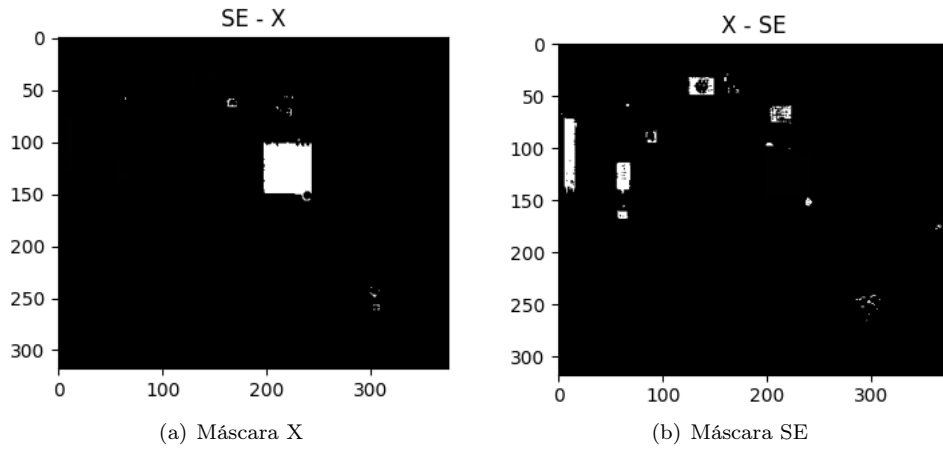


Figura 4: Máscaras geradas

Comparando-se as figuras 3 e 4, fica nítido que, se aplicada a *operação lógica AND* entre a imagem binária da placa a ser analisada e as máscaras, a imagem resultante será composta somente de zeros (preta) se e somente se a máscara e a imagem analisada corresponderem à mesma placa (figura 5). A operação AND é realizada bit-a-bit entre as imagens e equivale a uma multiplicação.

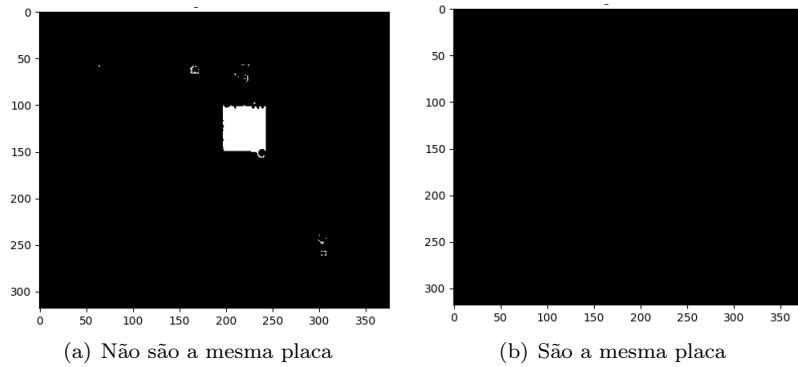


Figura 5: Resultado da operação AND entre máscara e imagem analisada

Computacionalmente, isso consiste em contar o número de pixels brancos na imagem. Se esse número for maior que zero, a máscara e a imagem analisada não correspondem à mesma placa. Para tratar o caso de imagens de entrada ruidosas, a aplicação considera que se até 1% dos pixels forem brancos (valor obtido empiricamente), ainda se trata da mesma placa (figura 6).

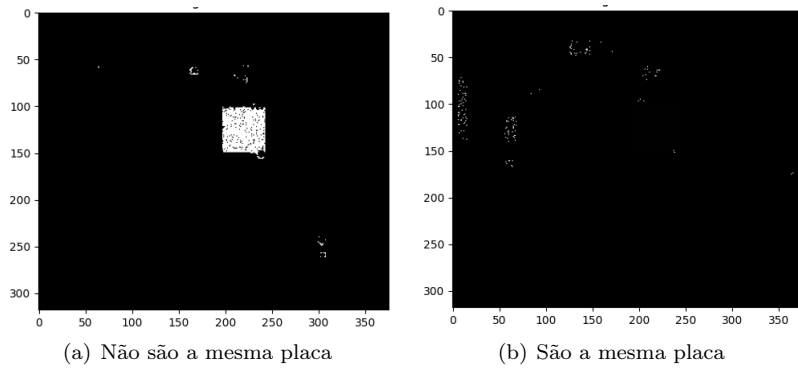


Figura 6: Resultado da operação AND entre máscara e imagem ruidosa

## Resultados

A aplicação foi capaz de reconhecer corretamente as placas (mediante passagem correta de parâmetros). Ela imprime no *shell* onde foi invocada o resultado da análise e mostra a imagem analisada e o resultado em uma janela (figura 7).

Exemplo de execução:

```
python3 000-placas.py -ipx imgs/a7v600-X.gif -ips imgs/a7v600-SE.gif -img imgs/a7v600-XRImpulsivo.gif
```

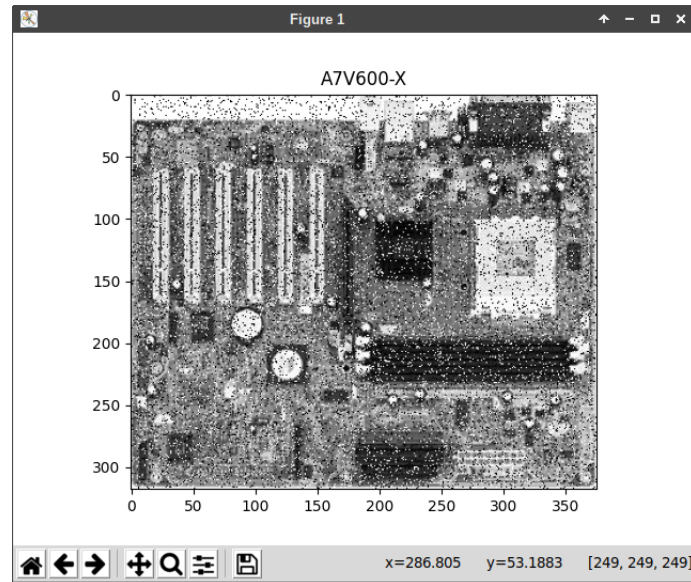


Figura 7: Exemplo de Execução da Aplicação.

## Referências

Videoaulas da Disciplina;

Processamento Digital de Imagens - Rafael C. Gonzalez e Richard E. Woods - 3a edição.