

Programa de Pós-Graduação em Ciências da Computação

Processamento Digital de Imagens - PDI

Trabalho I

Lista 1 – Introdução ao Processamento Digital de Imagens

Discente:

Carlos Estevão Bastos Sousa - **20181018030043**

Docente:

Prof. Dr. Pedro Pedrosa Rebouças Filho

Fortaleza, 03 de Outubro de 2018

Conteúdo

1	Introdução	1
2	Objetivo	1
3	Software e comandos peculiares da linguagem utilizados	1
4	Filtros Passa-baixa	1
5	Filtro da Média	2
5.1	Resultados obtidos	2
6	Filtro da Mediana	3
6.1	Resultados obtidos	4
7	Filtro da Gaussiano	4
7.1	Resultados obtidos	5
8	Filtros Passa-alta	5
9	Filtro Laplaciano	6
9.1	Resultados obtidos	7
10	Filtro Prewitt	8
10.1	Resultados obtidos	9
11	Filtro Sobel	9
11.1	Resultados obtidos	10
12	Outras Operações básicas	10
13	Cálculo de Apresentação do Histograma	11
13.1	Resultados obtidos	11
14	Equalização do Histograma	11
14.1	Resultados obtidos	12
15	Limiarização (<i>Thresholding</i>)	13
15.1	Resultados obtidos	13
16	Multilimiarização	14
16.1	Resultados obtidos	15
17	Conclusão	16
18	Referências	17

1 Introdução

Processamento digital de imagens é a utilização de algoritmos computacionais para aplicar processamento de imagens em imagens digitais. O processamento digital de imagens tem as mesmas vantagens sobre o processamento analógico de imagens que o processamento digital de sinais tem sobre o processamento analógico de sinais: possibilita uma gama muito mais ampla de algoritmos que podem ser aplicados na entrada e pode-se evitar problemas como ampliação de ruído e distorção do sinal durante o processamento. (GONZALEZ; WOODS, 2010)

2 Objetivo

O objetivo deste relatório é documentar as aplicações de operações básicas e de filtros passa-baixa e passa-alta, fazendo uma explanação sobre cada tópico abordado e mostrando os resultados obtidos.

3 Software e comandos peculiares da linguagem utilizados

Para a elaboração dos códigos foram utilizadas a linguagem e IDE Matlab[©]

Além dos comandos básicos de programação foram utilizados também:

Comando	Ação
bar	Imprime um gráfico de barras
histogram	Imprime o histograma de uma imagem
imnoise	Adiciona ruído a uma imagem
imshow	Imprime uma imagem
mean	Retorna a média
median	Retorna a mediana
rgb2gray	Converte uma imagem RGB para níveis de Cinza
sort	Ordena um vetor/matriz
subplot	Organiza várias plotagens em uma só figure
uint8	Transforma os valores de um vetor/matriz em números inteiros de 8 bits (1 byte)

Tabela 1: Alguns comandos utilizados.

4 Filtros Passa-baixa

Os filtros passa-baixa são aplicados à suavização de imagens diminuindo as frequências altas, os mesmos tendem a minimizar ruídos porém apresentam efeito de borramento.

Na figura a seguir apresentaremos 4 imagens, as mesmas serão utilizadas como base para a aplicação de boa parte dos filtros utilizados neste trabalho.



Figura 1: (a) Lena Söderberg, (b) imagem (a) em tons de cinza, (c) imagem (b) com o ruído sal e pimenta (0.1) e (d) imagem (c) com zoom. Fonte: (a) Revista Playboy[©] novembro de 1972 e (b), (c) e (d) autor

5 Filtro da Média

O filtro da média, assim como os demais filtros lineares de suavização tem por objetivo exibir partes de interesse em uma imagem. É possível afirmar que, basicamente, utilizamos a média aritmética da vizinhança e acrescentamos este valor ao centro da matriz. Podemos representar o filtro da média com a seguinte equação:

$$g(x, y) = \frac{\sum_{ki=0}^N \sum_{kj=0}^M w(ki, kj) f(i + ki, j + kj)}{NxM} \quad (1)$$

De uma forma mais simples, representamos a equação utilizando um kernel (máscara) que basicamente é uma matriz de 1's onde cada um desses valores são divididos pela quantidade de posições da mesma, nestes termos, uma matriz 3 x 3, possui o valor 1 dividido por 9 (total de termos) em cada posição.

$$\frac{1}{9} \times \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (2)$$

$$\begin{bmatrix} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{bmatrix} \quad (3)$$

Imaginando uma figura com os valores a seguir:

$$\begin{bmatrix} 23 & 12 & 30 \\ 15 & 55 & 43 \\ 3 & 12 & 50 \end{bmatrix} \quad (4)$$

Resolvemos a equação da seguinte forma:

$$\frac{23 + 12 + 30 + 15 + 55 + 43 + 3 + 12 + 50}{9} = 27 \quad (5)$$

Assim, o pixel central da matriz recebe o valor 27. Esta operação é executada em todos os pixels da imagem.

5.1 Resultados obtidos

Aplicamos o filtro da média com três tamanhos de matriz:



Figura 2: Aplicação do filtro da média com matriz de tamanho (a) 3×3 , (b) 5×5 e (c) 7×7 . Fonte: autor

Ao fazer uma análise um pouco mais minuciosa percebemos que este filtro não trás um bom resultado para este tipo de problema. As imagens a seguir nos mostram isso:

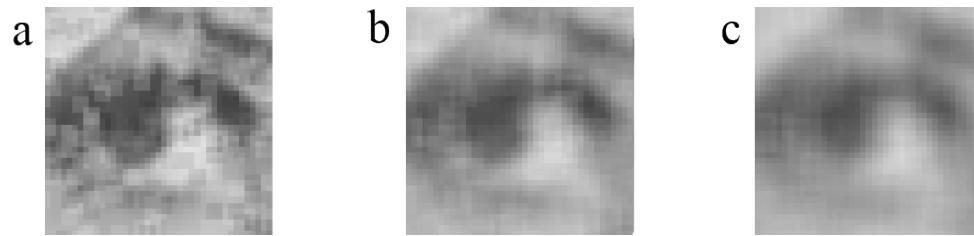


Figura 3: Aplicação do filtro da média com matriz de tamanho (a) 3×3 , (b) 5×5 e (c) 7×7 (zoom). Fonte: autor

Como podemos perceber o filtro da média é incapaz de preservar bordas, detalhes finos e com um kernel grande a imagem tende a ficar borrada, para contornar este problema usamos como uma das alternativas o filtro da mediana que veremos a seguir.

6 Filtro da Mediana

O filtro da mediana é um filtro passa baixa e também apresenta borramento na imagem, é o filtro de estatística de ordem mais conhecido, segundo Gonzalez e Woods (2010) [...] são filtros espaciais não lineares cuja resposta se baseia na ordenação (classificação) dos pixels contidos na área da imagem coberta pelo filtro e substituindo o valor do pixel central pelo valor determinado pelo resultado da classificação.

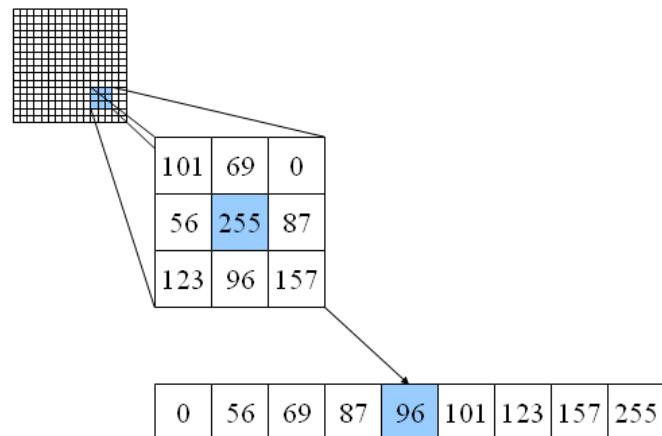


Figura 4: Exemplo do funcionamento do filtro da mediana. Fonte: <https://blitzman.gitbooks.io/cuda/práctica-2-filtro-mediana.html>

6.1 Resultados obtidos



Figura 5: Aplicação do filtro da mediana com matrizes de tamanho (a) 3 x 3, (b) 5 x 5 e (c) 7 x 7. Fonte: autor

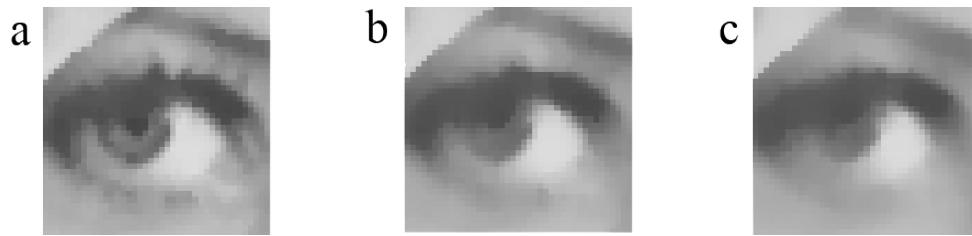


Figura 6: Aplicação do filtro da mediana com matrizes de tamanho 3 x 3, 5 x 5 e 7 x 7 (zoom)

Podemos perceber que apesar de existir um borramento na imagem e que o mesmo se intensifica a medida que o tamanho do kernel aumenta, o filtro da mediana é ótimo na aplicação de problemas com ruído impulsivo (sal e pimenta) e mostrou ótimos resultados.

7 Filtro da Gaussiano

Um filtro gaussiano é utilizado para borrar ou desfocar a imagem na qual ele é aplicado com o objetivo de reduzir os ruídos presentes na imagem (JESUS; JUNIOR, 2014). Sendo também um filtro passa baixa e também possuindo a características de apresentar borramentos na imagem o filtro gaussiano dar-se a partir da aplicação da máscara gaussiana dada pela seguinte fórmula:

$$w(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (6)$$

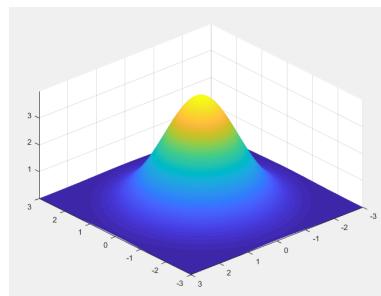


Figura 7: Representação do filtro gaussiano. Fonte: autor

Como podemos perceber o filtro gaussiano possui a capacidade de deixar os elementos ao centro mais próximos da média e os mais distantes, mais longe.

7.1 Resultados obtidos



Figura 8: Aplicação do filtro gaussiano com matrizes de tamanho (a) 3×3 , (b) 5×5 e (c) 7×7 e $\sigma = 1.0$

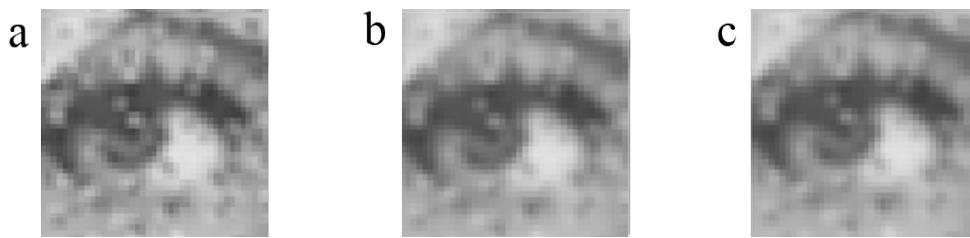


Figura 9: Aplicação do filtro gaussiano com matrizes de tamanho (a) 3×3 , (b) 5×5 e (c) 7×7 e $\sigma = 1.0$ (zoom)



Figura 10: Aplicação do filtro gaussiano com matrizes de tamanho (a) 3×3 , (b) 5×5 e (c) 7×7 e $\sigma = 2.0$



Figura 11: Aplicação do filtro gaussiano com matrizes de tamanho (a) 3×3 , (b) 5×5 e (c) 7×7 e $\sigma = 2.0$ (zoom)

8 Filtros Passa-alta

Diferente dos filtros passa-baixa, os filtros passa-alta realçam os detalhes da imagem, ou seja, as transições se tornam mais perceptíveis. Utilizamos este tipo de filtro para realçar características em imagens.

Para os filtros passa-alta utilizamos a seguinte imagem:



Figura 12: 1- Igreja de São Francisco em Canindé-CE, 2- Torre a direita da imagem.
Fonte: <https://mapio.net/pic/p-41556836/>

9 Filtro Laplaciano

Iniciando este trabalho com os filtros passa-alta, temos os filtros Laplacionais que são diferenciais de segunda ordem e utilizados para detecção de bordas em todas as direções, assim, com o centro do kernel negativo ele remove bordas exteriores e com o centro positivo remove bordas interiores. Em um formato unidimensional, podemos representar o filtro laplaciano pela seguinte equação:

$$\frac{\partial^2 f}{\partial x^2} = f(x+1) + f(x-1) - 2f(x) \quad (7)$$

A equação acima pode ser representada pelos seguintes kernels unidimensionais.

$$\begin{bmatrix} 1 & -2 & 1 \end{bmatrix} \quad (8)$$

$$\begin{bmatrix} -1 & 2 & -1 \end{bmatrix} \quad (9)$$

Para o caso de sinais bidimensionais e, utilizando o conceito de 4 adjacentes temos:

$$\frac{\partial^2 f}{\partial x^2} = f(x+1, y) + f(x-1, y) - 2f(x) \quad (10)$$

$$\frac{\partial^2 f}{\partial x^2} = f(x, y+1) + f(x, y-1) - 2f(x) \quad (11)$$

No qual, podemos representar através dos seguintes kernels.

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad (12)$$

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \quad (13)$$

Porém, utilizando o conceito de 8 adjacentes, temos:

$$\frac{\partial^2 f}{\partial x^2} = f(x+1, y) + f(x-1, y) - 2f(x) \quad (14)$$

$$\frac{\partial^2 f}{\partial x^2} = f(x+1, y+1) + f(x-1, y+1) - 2f(x) \quad (15)$$

$$\frac{\partial^2 f}{\partial x^2} = f(x, y+1) + f(x, y-1) - 2f(x) \quad (16)$$

$$\frac{\partial^2 f}{\partial x^2} = f(x-1, y+1) + f(x-1, y-1) - 2f(x) \quad (17)$$

Seguindo a mesma lógica, representamos o kernel do filtro laplaciano das maneiras a seguir:

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (18)$$

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix} \quad (19)$$

9.1 Resultados obtidos

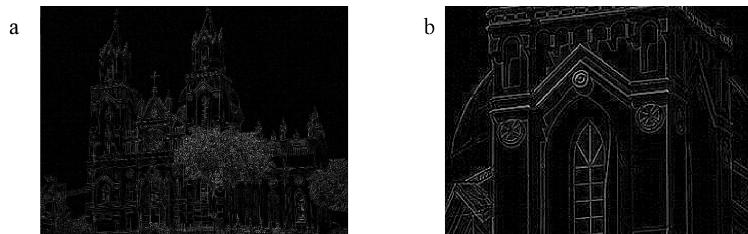


Figura 13: (a) Imagem da Igreja e (b) Torre a direita com aplicação do kernel com centro positivo e 4 vizinhos. Fonte: autor

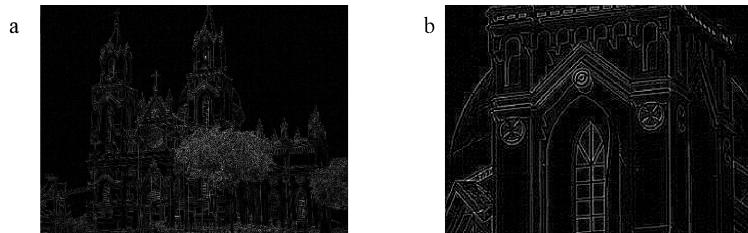


Figura 14: (a) Imagem da Igreja e (b) Torre a direita com aplicação do kernel com centro negativo e 4 vizinhos. Fonte: autor

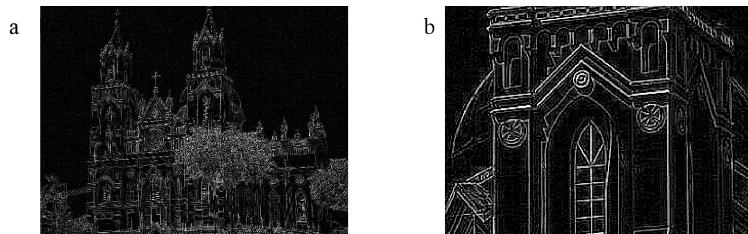


Figura 15: (a) Imagem da Igreja e (b) Torre a direita com aplicação do kernel com centro positivo e 8 vizinhos. Fonte: autor

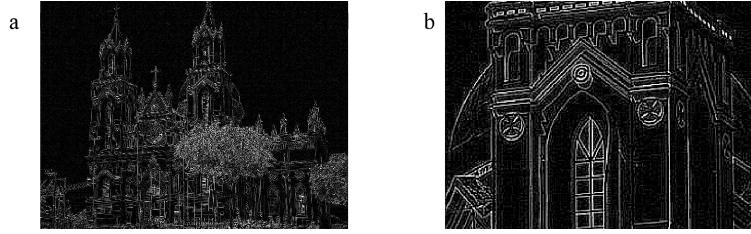


Figura 16: (a) Imagem da Igreja e (b) Torre a direita com aplicação do kernel com centro negativo e 8 vizinhos. Fonte: autor

10 Filtro Prewitt

Tal qual o filtro anterior, o filtro Prewitt também é caracterizado como passa-alta, porém o mesmo é um filtro de primeira ordem, ou seja, pode ser definido como:

$$\frac{\partial f}{\partial x} = f(x+1) - f(x) \quad (20)$$

As derivadas de primeira ordem em processamento de imagens são implementadas utilizando a magnitude do gradiente. Para uma função $f(x, y)$, o gradiente de f nas coordenadas (x, y) é definido como o vetor coluna bidimensional: (GONZALEZ; WOODS, 2010)

$$\nabla(f) = \begin{pmatrix} gx \\ gy \end{pmatrix} = \begin{pmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{pmatrix} \quad (21)$$

Utilizando operadores de vizinhança diagonal o valor dos gradientes dado um ponto central no sinal é:

$$\frac{\partial f}{\partial x} = [f(x-1, y+1) + f(x, y+1) + f(x+1, y+1)] - [f(x-1, y-1) + f(x, y-1) + f(x+1, y-1)] \quad (22)$$

$$\frac{\partial f}{\partial y} = [f(x+1, y-1) + f(x+1, y) + f(x+1, y+1)] - [f(x-1, y-1) + f(x-1, y) + f(x-1, y+1)] \quad (23)$$

As equações (22) e (23) acima podem ser representadas, respectivamente, pelos seguintes kernels.

Horizontal:

$$\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \quad (24)$$

Vertical:

$$\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad (25)$$

10.1 Resultados obtidos

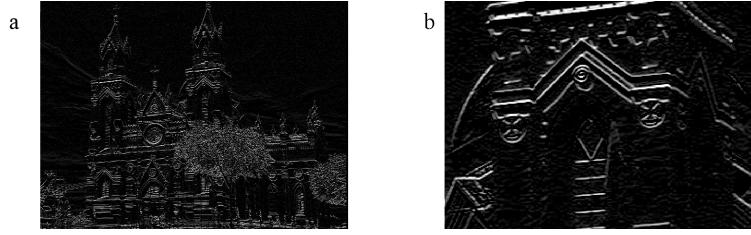


Figura 17: (a) Imagem da Igreja e (b) Torre a direita com aplicação do kernel horizontal.
Fonte: autor

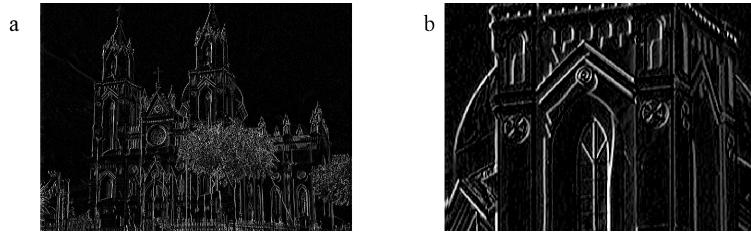


Figura 18: (a) Imagem da Igreja e (b) Torre a direita com aplicação do kernel vertical.
Fonte: autor

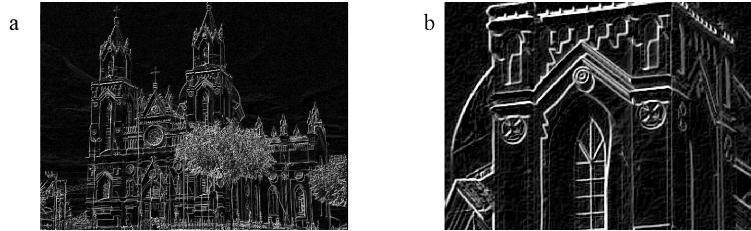


Figura 19: (a) Imagem da Igreja e (b) Torre a direita com aplicação do kernel horizontal e vertical. Fonte: autor

11 Filtro Sobel

Outro filtro de primeira ordem que também podemos destacar é o filtro de Sobel, o mesmo também pode ser representado pela seguintes equações:

$$\frac{\partial f}{\partial x} = f(x+1) - f(x) \quad (26)$$

$$\nabla(f) = \begin{pmatrix} gx \\ gy \end{pmatrix} = \begin{pmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{pmatrix} \quad (27)$$

$$\frac{\partial f}{\partial y} = [f(x+1, y-1) + 2f(x+1, y) + f(x+1, y+1)] - [f(x-1, y-1) + 2f(x-1, y) + f(x-1, y+1)] \quad (28)$$

$$\frac{\partial f}{\partial x} = [f(x-1, y+1) + 2f(x, y+1) + f(x+1, y+1)] - [f(x-1, y-1) + 2f(x, y-1) + f(x+1, y-1)] \quad (29)$$

As equações acima podem ser representadas, respectivamente, pelos seguintes kernels.

Horizontal:

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad (30)$$

Vertical:

$$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (31)$$

Vale salientar que, pelo fato do filtro de Sobel ser um operador derivativo, o somatório dos valores de sua máscara deverá ser sempre 0 (zero) e, diferente do filtro anterior, seu kernel considera maior os pesos centrais.

11.1 Resultados obtidos

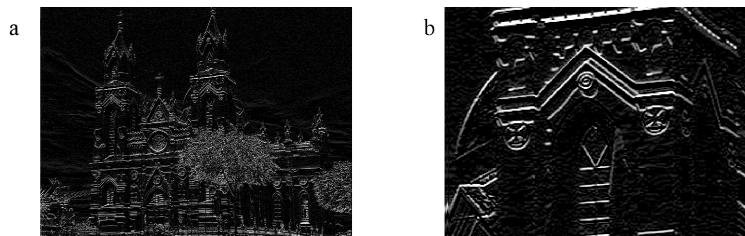


Figura 20: (a) Imagem da Igreja e (b) Torre a direita com aplicação do kernel horizontal.
Fonte: autor

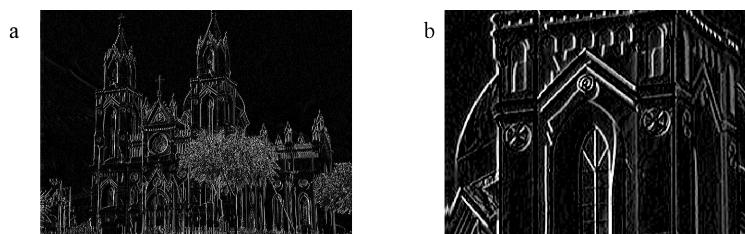


Figura 21: (a) Imagem da Igreja e (b) Torre a direita com aplicação do kernel vertical.
Fonte: autor

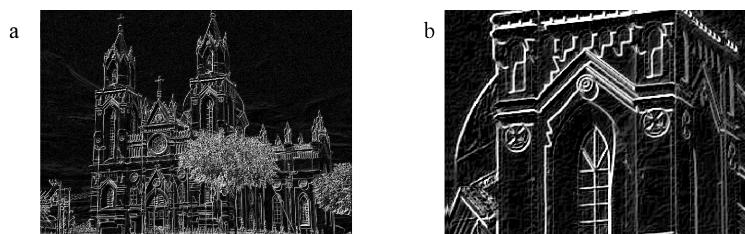


Figura 22: (a) Imagem da Igreja e (b) Torre a direita com aplicação do kernel horizontal e vertical. Fonte: autor

12 Outras Operações básicas

Nesta seção iremos trabalhar com o cálculo, apresentação e equalização do Histograma, limiarização e multi-limiarização.

13 Cálculo de Apresentação do Histograma

O histograma é uma excelente ferramenta para análise de níveis de cores de uma imagem, através do mesmo podemos verificar se uma imagem possui muitos tons brancos, pretos ou se a imagem está equalizada. De uma forma mais simples, podemos dizer que o histograma de uma imagem funciona como uma tabela que indica o número ou o percentual de pixels que a imagem possui em cada tom.

13.1 Resultados obtidos

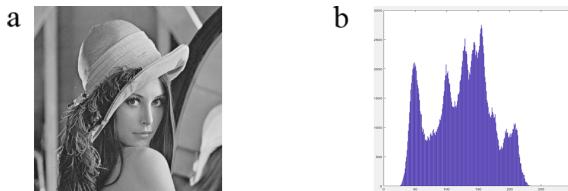


Figura 23: (a) Lena Söderberg em escala cinza e (b) histograma da imagem. Fonte: autor

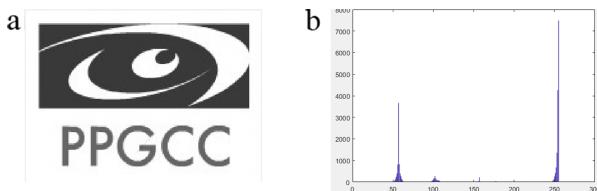


Figura 24: (a) Símbolo PPGCC- IFCE em escala cinza e (b) histograma do mesmo. Fonte: (a) ppgcc.ifce.edu.br e (b) autor

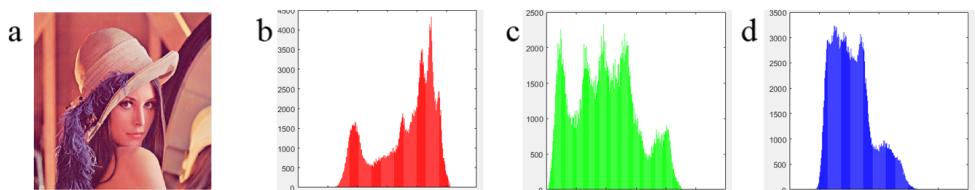


Figura 25: (a) Lena Söderberg , (b) Histograma canal vermelho, (c) verde e (d) azul. Fonte: autor

Através das imagens anteriores podemos perceber que a imagem (23) pode ser equalizada e obter melhores resultados nos detalhes, que a figura (24) possui muitos tons nos valores nos tons 50 e 250 (escala cinza) e percebe-se na imagem (25) que a mesma possui uma grande quantidade de tons vermelhos (b) de 200 a 250 (canal Red).

14 Equalização do Histograma

A equalização de histograma é uma técnica a partir da qual se procura redistribuir os valores de tons de cinza dos pixels em uma imagem, de modo a obter um histograma uniforme, no qual o número ou percentual de pixels de qualquer nível de cinza é praticamente o mesmo. Para tanto, utiliza-se uma função auxiliar, denominada função de transformação.

A maneira mais comum de equalizar um histograma é através da função de distribuição acumulada (CDF - *Cumulative Distribution Function*) que pode ser descrita como:

$$S_k = T(r_k) = \sum_{j=0}^k \frac{n_j}{n} = \sum_{j=0}^k p_r(r_j) \quad (32)$$

De uma maneira mais simples, podemos equalizar um histograma de uma imagem da seguinte maneira:

Tons	Ocorrência	Probabilidade	Somatório	Probabilidade X Total de Tons	Novo tom
0	200	0,07	0,07	0,52	1
1	213	0,08	0,15	1,07	1
2	850	0,32	0,47	3,28	3
3	656	0,24	0,71	4,98	5
4	329	0,12	0,83	5,84	6
5	245	0,09	0,92	6,47	6
6	122	0,05	0,97	6,79	7
7	81	0,03	1,00	7,00	7
-	2696	-	-	-	-

Tabela 2: Exemplo da equalização do Histograma de uma imagem.

14.1 Resultados obtidos

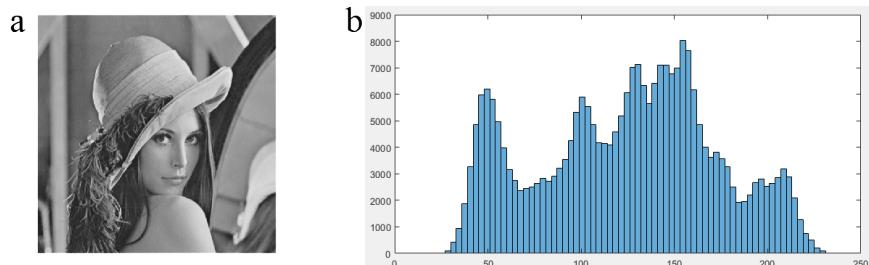


Figura 26: (a) Lena Söderberg, (b) Histograma da imagem. Fonte: autor

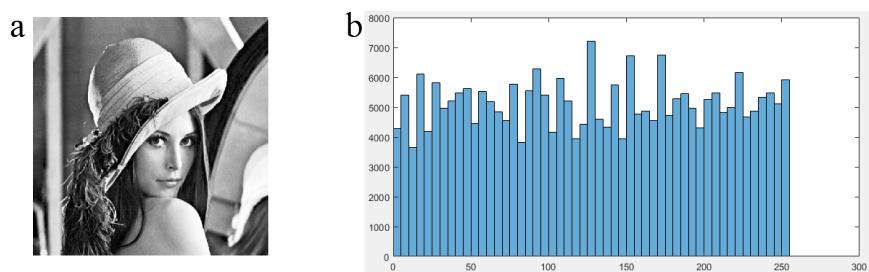


Figura 27: (a) Lena Söderberg, (b) Histograma da imagem equalizada. Fonte: autor

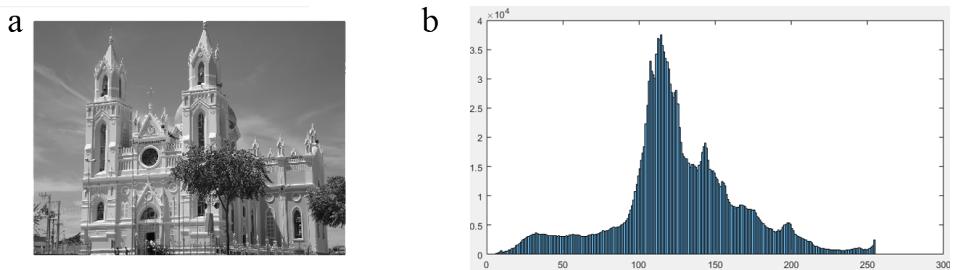


Figura 28: (a) Igreja de São Francisco em Canindé-CE, (b) Histograma da mesma. Fonte: autor

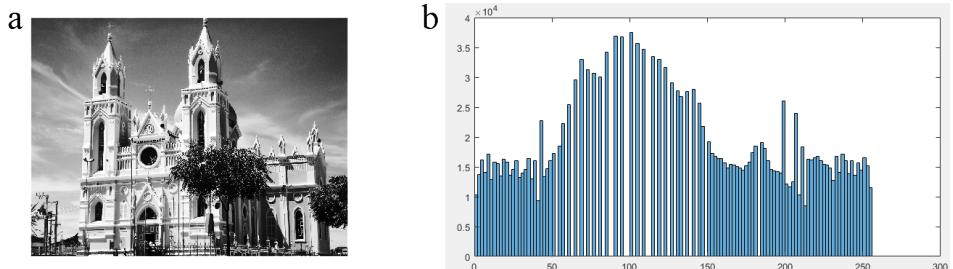


Figura 29: (a) Imagem equalizada da Igreja de São Francisco em Canindé-CE, (b) Histograma da mesma. Fonte: autor

15 Limiarização (*Thresholding*)

A ideia de limiarização representa a separação de regiões. De acordo com Marques e Neto (1999) a forma mais simples de limiarização consiste na bipartição do histograma, convertendo os pixels cujo tom de cinza é maior ou igual a um certo valor de limiar (T) em brancos e pretos.

15.1 Resultados obtidos



Figura 30: Lena Söderberg, (a) limiar = 50, (b) limiar = 100, (c) limiar = 150 e (d) limiar = 200. Fonte: autor

Como podemos perceber na figura a seguir, o histograma apresenta dois picos situados exatamente nas extremidades do histograma.

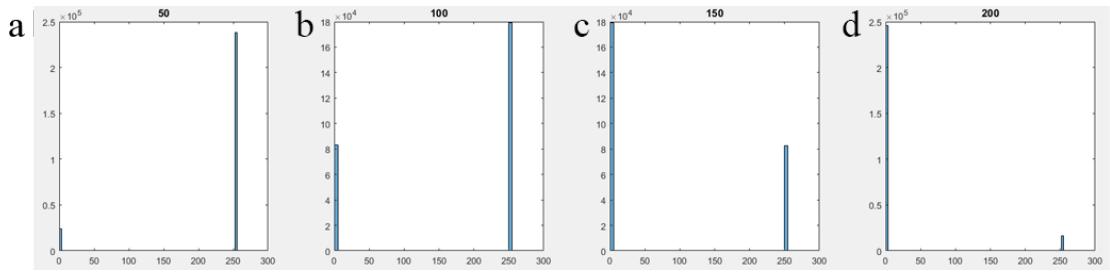


Figura 31: Histogramas: (a) Limiar = 50, (b) limiar = 100, (c) limiar = 150 e (d) limiar = 200. Fonte: autor

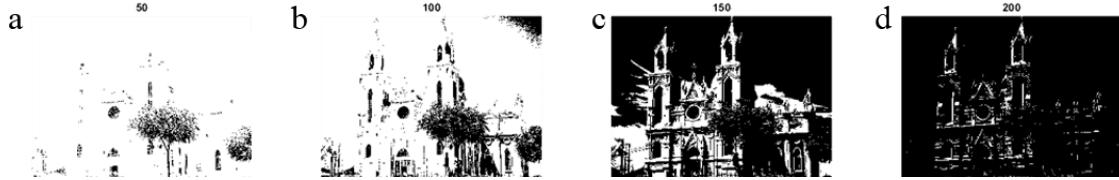


Figura 32: Igreja de São Francisco em Canindé-CE, (a) Limiar = 50, (b) limiar = 100, (c) limiar = 150 e (d) limiar = 200. Fonte: autor

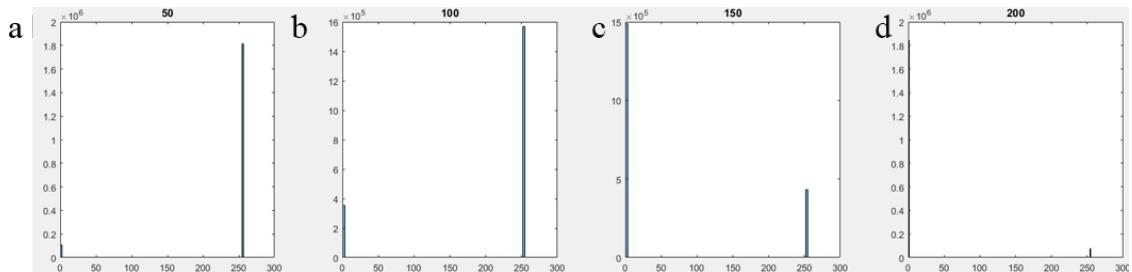


Figura 33: Histogramas: (a) Limiar = 50, (b) limiar = 100, (c) limiar = 150 e (d) limiar = 200. Fonte: autor

16 Multilimiarização

Semelhante a seção anterior (limiarização), a multilimiarização, como o próprio nome já torna claro, funciona com mais de um limiar, assim, podemos representar a mesma da seguinte maneira:

$$\begin{aligned}
 g(x, y) &= 0 & \text{se } f(x, y) < L_{inf} \\
 g(x, y) &= h & \text{se } L_{inf} < f(x, y) < L_{sup} \\
 g(x, y) &= \max & \text{se } f(x, y) \geq L_{sup}
 \end{aligned} \tag{33}$$

16.1 Resultados obtidos



Figura 34: Lena Söderberg, (a) $\text{Linf} = 50$ e $\text{Lsup} = 100$, (b) $\text{Linf} = 75$ e $\text{Lsup} = 150$, (c) $\text{Linf} = 100$ e $\text{Lsup} = 150$, (d) $\text{Linf} = 200$ e $\text{Lsup} = 250$. Fonte: autor

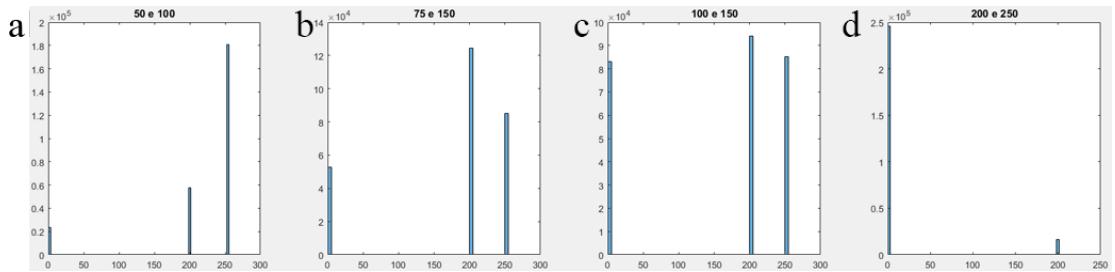


Figura 35: (a) $\text{Linf} = 50$ e $\text{Lsup} = 100$, (b) $\text{Linf} = 75$ e $\text{Lsup} = 150$, (c) $\text{Linf} = 100$ e $\text{Lsup} = 150$, (d) $\text{Linf} = 200$ e $\text{Lsup} = 250$. Fonte: autor

Tal qual a limiarização, a multilimiarização se torna uma ferramenta poderosa se tratando de imagens de cores distintas (preto e branco) com a eficácia de poder ser aplicada a mais cores e não só duas como visto na seção anterior (limiarização).



Figura 36: Igreja de São Francisco em Canindé-CE, (a) $\text{Linf} = 50$ e $\text{Lsup} = 100$, (b) $\text{Linf} = 75$ e $\text{Lsup} = 150$, (c) $\text{Linf} = 100$ e $\text{Lsup} = 150$, (d) $\text{Linf} = 200$ e $\text{Lsup} = 250$. Fonte: autor

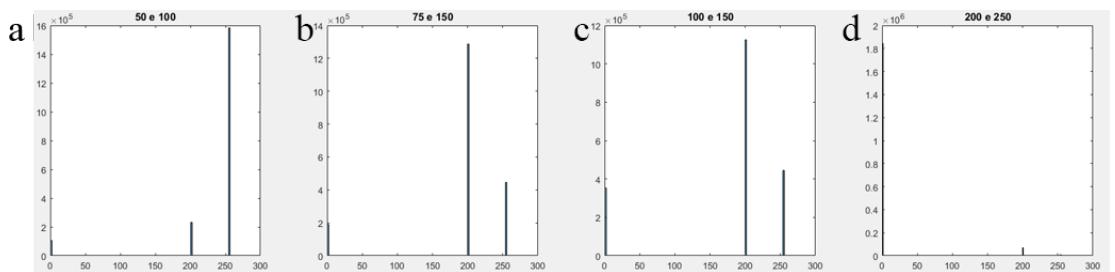


Figura 37: (a) $\text{Linf} = 50$ e $\text{Lsup} = 100$, (b) $\text{Linf} = 75$ e $\text{Lsup} = 150$, (c) $\text{Linf} = 100$ e $\text{Lsup} = 150$, (d) $\text{Linf} = 200$ e $\text{Lsup} = 250$. Fonte: autor

17 Conclusão

Através do presente trabalho concluímos que os filtros passa-baixa servem para suavizar uma imagem e podem reduzir ruídos na mesma. Por outro lado, os filtros passa-alta são ótimos para melhorar a nitidez, deixando seus detalhes mais perceptíveis, percebemos também, que através de operações básicas podemos verificar o nível de luminosidade de uma imagem, equalizar a mesma afim de se obter um histograma uniforme e definir um ou vários limiares com o intuito, por exemplo, de separar objetos do fundo da imagem.

18 Referências

- [1] AQUAPHOENIX. Signal Processing - Digital Image processing. Disponível em: <<http://www.aquaphoenix.com/lecture/matlab10/page3.html>>. Acesso em: 29 set. 2018.
- [2] Gaussian filter, or Gaussian blur. Disponível em: <<http://www.librow.com/articles/article-9>>. Acesso em: 25 set. 2018.
- [3] GONZALEZ, Rafael C.; WOODS, Richard E.. Processamento Digital de Imagens. São Paulo: Pearson, 2010.
- [4] IMAGE PROCESSING. Disponível em: <<https://www.imageprocessing.com/2013/07/image-sharpening-using-second-order.html>>. Acesso em: 27 set. 2018.
- [5] JESUS, E.O; JUNIOR, R. C.; A Utilização de Filtros Gaussianos na Análise de Imagens Digitais Universidade Federal de Itajubá, UNIFEI, Instituto de Matemática e Computação, IMC, Natal-RN, 2014.
- [6] INPE. Teoria : Processamento de Imagens. Disponível em: <<http://www.dpi.inpe.br/spring/teor>>. Acesso em: 30 set. 2018.
- [7] MARQUES FILHO, Ogê; VIEIRA NETO, Hugo. Processamento Digital de Imagens, Rio de Janeiro: Brasport, 1999. ISBN 8574520098.