



## **INDEXAÇÃO E RECUPERAÇÃO DE CONTEÚDO EM IMAGENS**

**Sérgio Paulo Melo do Souza<sup>1</sup>, João Marques de Carvalho<sup>2</sup>**

### **RESUMO**

Este projeto tem como objetivo final indexar e recuperar informações em documentos multimídia, baseado na análise automática (processamento, extração de características e classificação) das mídias e no compartilhamento das informações específicas de cada uma delas. Foram consideradas, a princípio, as aplicações relacionadas às imagens, sejam estas fotografias ou imagens de documentos de textos previamente digitalizadas. Inicialmente, estudaram-se as técnicas de processamento de imagens. Utilizou-se a linguagem C++ e o ambiente de desenvolvimento Microsoft Visual C++ 2008 Express na criação de um aplicativo, contendo todas as técnicas estudadas. Com este, foram testados os algoritmos implementados.

**Palavras-chave:** processamento de imagens, análise de documentos, documentos multimídia.

## **INDEXING AND RECOVERY OF CONTENT IN IMAGES**

### **ABSTRACT**

This project aims to index and retrieve information in multimedia documents, based on the automatic analysis (processing, extraction of features and classification) and sharing of media specific information. We consider, to start with, applications related to digital images, either photographs or images of text documents, previously scanned. Initially, we study several techniques for image processing. Using the C++ language and the Microsoft Visual C++ 2008 Express development environment to create an application containing all the techniques studied, we implemented and tested the algorithms implemented.

**Keywords:** image processing, document analysis, multimedia documents.

### **INTRODUÇÃO**

A área de Processamento Digital de Imagens vem crescendo devido à necessidade de aprimoramento das técnicas de extração das informações contidas em imagens, tanto para a interpretação humana como para interpretação por máquinas. Nesta área, temos aplicações na medicina, geologia e engenharia, dentre outras. Podemos citar especificamente a aplicação no reconhecimento facial e na recuperação de informações em documentos manuscritos.

No contexto acima descrito, este projeto tem como objetivo indexar e recuperar informações em documentos multimídia, baseado na análise automática (processamento, extração de características e

---

<sup>1</sup>Aluno de Curso de Engenharia Elétrica, Depto. de Engenharia Elétrica, UFPG, Campina Grande, PB, E-mail: [spmsouza@hotmail.com](mailto:spmsouza@hotmail.com)

<sup>2</sup>Engenheiro Eletricista, Prof. Doutor, Depto. de Engenharia Elétrica, UFPG, Campina Grande, PB, E-mail: [carvalho@dee.ufcg.edu.br](mailto:carvalho@dee.ufcg.edu.br)

classificação) das mídias e no compartilhamento das informações específicas de cada uma delas. Serão consideradas, inicialmente, as aplicações relacionadas às imagens.

Na implementação das técnicas de processamento de imagens, utilizamos a linguagem C++ e o ambiente de desenvolvimento Microsoft Visual C++ Express Edition, distribuído gratuitamente pelo fabricante. A biblioteca OpenCV foi utilizada com a finalidade de tornar mais fácil a manipulação das imagens. Ela é livre para uso acadêmico e comercial e para o desenvolvimento de aplicativos na área de visão computacional.

## TÉCNICAS E ALGORITMOS ESTUDADOS

Este trabalho foi desenvolvido no Laboratório de Automação e Processamento de Sinais – LAPS, do Departamento de Engenharia Elétrica do Centro de Engenharia Elétrica e Informática (DEE-CEEI) da Universidade Federal de Campina Grande (UFCG) – PB.

### Imagem

Uma imagem é uma função bidimensional de intensidade de luz  $f(x,y)$ , onde  $x$  e  $y$  representam as coordenadas espaciais. Imagens são obtidas através de sensores, sendo naturalmente analógicas, ou seja, definidas num domínio contínuo (espaço/tempo) e com valores contínuos. Para estudá-las, é preciso converter para um formato digital, ou seja, definido em um domínio discreto e assumindo valores discretos. O processo de conversão A/D (analógico-digital) consiste de duas etapas:

1. Amostragem: discretização espacial da imagem. Cada elemento da matriz indexada de uma imagem é chamado de pixel.
2. Quantização: discretização dos níveis de intensidade de cada ponto de uma imagem.

Após a amostragem e quantização da imagem, esta pode ser considerada uma matriz bidimensional com os índices das linhas e colunas representando um ponto ou pixel da imagem e o valor do elemento da matriz correspondendo ao nível de cinza ou intensidade de brilho naquele ponto. O espaço requerido para armazenar uma imagem simples e monocromática que tenha dimensões  $N \times M$  e  $B$  bits de resolução é  $N \times M \times B$ , em que cada pixel é representado por  $B$  bits.

Os níveis de cinza de cada pixels são inteiros positivos do conjunto  $\{0, \dots, K-1\}$ , onde  $K$  é uma potência de 2 ( $K = 2^x$ ). A quantidade de níveis de cinza possíveis depende da quantidade de bits utilizados para representar cada nível. Assim, para um número de bits  $X=8$ , teremos um total de 256 níveis de cinza. A qualidade da imagem é determinada pela quantidade de bits por pixel.

### O Formato BMP

As imagens de estudo escolhidas estão no formato BMP que é o formato básico de imagem utilizado pelo Windows, em computadores compatíveis com este sistema operacional e com o DOS. É um formato simples e fácil de manipular. Porém, perde para outros formatos em portabilidade, visto que foi ajustado para atender a sistemas operacionais que rodam sobre plataforma Intel. Outra desvantagem desse formato é o grande espaço de memória requerido para o armazenamento de imagens.

Os arquivos BMP possuem quatro partes: cabeçalho de arquivo, cabeçalho de mapa de bits, paleta de cores e área de dados da imagem.

### Histograma

O histograma  $H_f$  de uma imagem digital  $f$  é um gráfico que mostra a frequência de ocorrência de cada nível de cinza nesta imagem, ou seja:

$$H_f(k) = J$$

k: nível de cinza.  $\{0, \dots, K-1\}$   
J: frequência de ocorrência do nível de cinza k.

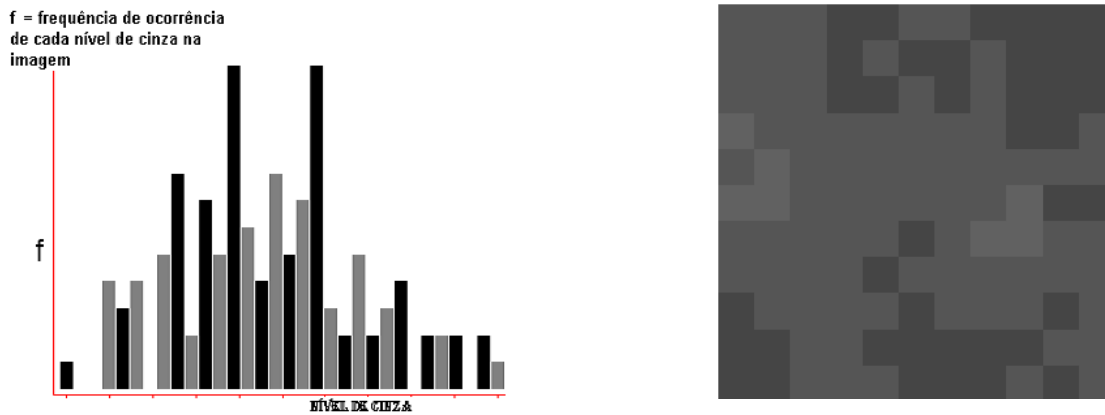


Figura 1 – Imagem à direita corresponde ao histograma da imagem à esquerda.

O histograma é uma ferramenta muito útil para obtermos informações sobre uma imagem. Através do histograma, obtemos informações relativas ao contraste e ao aspecto geral da imagem, dentre outras informações. Algumas técnicas de processamento de imagens trabalham diretamente com o histograma.

Entretanto, não podemos obter uma imagem tendo como informação apenas o seu histograma, pois este é uma representação unidimensional de uma imagem, que é bidimensional. Uma imagem tem somente um histograma, mas um histograma pode representar mais de uma imagem.

Abaixo se encontram alguns exemplos de histogramas de imagem.

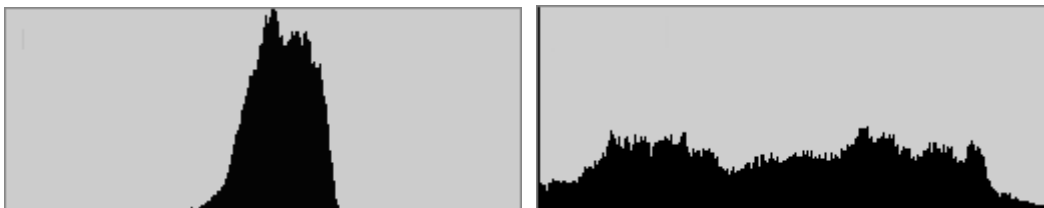


Figura 2 – O histograma da direita corresponde à uma imagem com baixo contraste e a imagem da esquerda com o histograma mais espalhado consiste numa imagem com alto contraste.

## Processamento de imagens em tons de cinza

Existem três tipos básicos de operações que podem ser realizadas em imagens. São elas:

**Operações pontuais:** São operações realizadas em cada pixel individualmente. As interações e relações entre pixels vizinhos não são consideradas. Genericamente, uma operação pontual é da forma  $g(n) = h[f(n)]$ , onde  $h$  é uma função, linear ou não. Segue abaixo a descrição das operações pontuais estudadas:

- Adição (Additive Image Offset):** É uma operação da forma  $g(n) = f(n) + L$  onde  $L$  é um número inteiro e  $|L| < K-1$ . No histograma de uma imagem essa operação desloca o gráfico de  $L$ , ou seja,  $Hg = Hf(k-L)$
- Multiplicação (Multiplicative Image Scaling):** É uma operação da forma  $g(n) = P.f(n)$ , em que  $P$  é um escalar positivo. Essa operação pode alargar (ou espalhar) o histograma se  $P > 1$ , ou comprimir o histograma, se  $P < 1$ .
- Negativo da imagem (Image Negative):** É o resultado de uma operação de adição e de uma operação de multiplicação, expresso como  $g(n) = -1 f(n) + (K-1)$ , em que  $K$  é a quantidade de níveis de cinza possíveis, geralmente 256.
- Espalhamento de Histograma (Full Scale Histogram Stretch - FSHS):** Produz um espalhamento do histograma, fazendo com que uma imagem tenha todos os valores possíveis de nível de cinza em uma faixa predeterminada. Isto deixa a imagem com um melhor contraste. Esta operação é expressa por:

$$g(n) = \left( \frac{K-1}{M-m} \right) \cdot (f(n) - m),$$

em que  $M$  e  $m$  são o maior e o menor valor de nível de cinza do histograma original, respectivamente.

- e. Operação Logarítmica: É a composição de uma operação logarítmica com um espalhamento de histograma, expresso por:  $g(n) = FSHS[\log(1 + f(n))]$
- e. Equalização de Histograma: Além de espalhar o histograma, ela também faz com que os níveis de cinza tenham uma distribuição de probabilidade que aproxima a distribuição uniforme, o que produz uma melhoria de contraste na imagem.

Operações aritméticas: Assim como nas operações pontuais, as relações espaciais entre os componentes da imagem não são consideradas nesta classe de operações. São usadas para redução de ruído, mudança e detecção de movimentos, dentre outras aplicações. São operações ponto-a-ponto entre imagens de mesmo tamanho. Seguem abaixo as operações aritméticas estudadas:

- a. Adição de imagens: É utilizada principalmente para a redução de ruído em imagens, através do cálculo da média de várias imagens do mesmo objeto, o que produz o cancelamento de ruído aleatório aditivo. A soma de  $n$  imagens é dada por:

$$p_1 + p_2 + \dots + p_n = \sum_{m=1}^n p_m$$

em que  $p_1, p_2, \dots, p_n$ , representam os valores de pixel de mesmas coordenadas em cada imagem.

- b. Subtração de imagens: Usada para a detecção de mudanças em imagens, podendo ser utilizada na detecção de movimentos em câmeras de segurança. A subtração de duas imagens é dada por  $p_s = p_1 - p_2$ .

- c. Multiplicação de imagens: Muito utilizada quando manipulamos imagens no domínio da frequência, através da Transformada de Fourier. É dada por:

$$p_1 \times p_2 \times \dots \times p_n = \prod_{m=1}^n p_m$$

em que  $p_1, p_2, \dots, p_n$ , representam os valores de pixel de mesmas coordenadas em cada imagem.

- d. Divisão de imagens: Juntamente com a multiplicação de imagens, utilizamos a divisão quando manipulamos imagens resultantes da Transformação de Fourier. Esta operação é dada por  $g = p_r / p_s$ , em que  $/$  representa o operador de divisão e  $p_s \neq 0$ .

Operações geométricas: São operações opostas às operações pontuais, no sentido de que modificam a posição e as relações espaciais entre os pixels, não modificando seus valores de intensidade. Alteram, portanto, as relações geométricas entre os componentes da imagem. Operações geométricas geralmente são complexas, pois exigem a definição de modelos matemáticos e um grande esforço computacional. Estas operações são realizadas em dois passos:

- I. Calcular as coordenadas de determinado ponto da imagem, armazenando o respectivo valor de intensidade. Se este cálculo resultar em valores não inteiros de coordenadas, faz-se uma interpolação para definir qual o valor de nível de cinza será colocado na imagem de saída. Utilizamos dois tipos de interpolação: interpolação por vizinho mais próximo e interpolação bilinear.
- II. Calcular uma operação geométrica propriamente dita, da forma:

$$g(x,y) = f[a(x,y), b(x,y)],$$

em que  $a(x,y)$  e  $b(x,y)$  são funções que retornam as coordenadas  $x$  e  $y$  da imagem original modificada.

As seguintes operações geométricas foram estudadas:

- a. Translação: translada a imagem de uma quantidade de pixels  $l_1$  e  $l_2$ , as quais são constantes inteiras. Dada por:

$$g(x,y) = f[x - l_1, y - l_2]$$

- b. Rotação: rotaciona a imagem de um ângulo  $\theta$  relativo à horizontal. Dada por:

$$g(x,y) = f[x \cos \theta - y \sin \theta, x \sin \theta + y \cos \theta]$$

- c. Zoom: com esta operação é possível ampliar ou diminuir imagem. Dada por:

$$g(x,y) = f[x/c, y/d]$$

Para  $c \geq 1$  e  $d \geq 1$  temos um aumento da imagem e para  $c < 1$  e  $d < 1$  temos uma diminuição da imagem.

### Processamento em imagens binárias

Uma imagem binária assume apenas 1 bit por pixel, ou seja,  $B=1$ , logo,  $K = 2$ . Esses valores que cada pixel pode assumir são tomados como Booleanos (0 ou 1). Isso pode corresponder à presença ou ausência de determinada característica ou propriedade na imagem. A imagem binária é mostrada em apenas dois tons, preto e branco, os quais correspondem a 0 e 255 em tons de cinza, respectivamente.

Limiarização de imagem: É uma forma simples de binarizar uma imagem. Comparamos o valor de cada pixel da imagem  $f$  com um valor limiar  $T$ . Então:

$$g(n) = \begin{cases} 1, & f(n) \geq T \\ 0, & f(n) < T \end{cases}$$

Porém, a ferramenta básica para entender a limiarização da imagem é o histograma. Esse processo é feito principalmente em imagens que tenham um histograma bimodal, ou seja, o mesmo histograma contenha dois picos, os quais podem ser separados escolhendo um valor de limiar  $T$  conveniente. Histogramas bimodais estão frequentemente associados a imagens que contêm objetos e plano de fundo com diferença de brilho significativa. Um exemplo de histograma bimodal é mostrado abaixo. A seta mostra um possível valor limiar  $T$  para separarmos as duas regiões do histograma.

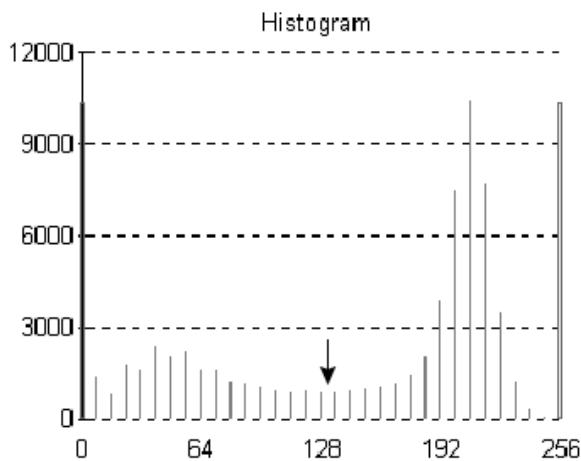


Figura 3 – Histograma bimodal de uma imagem.

### Transformada Discreta de Fourier

Quando aplicamos a transformada em um sinal bidimensional (ex: imagem) o resultado é uma matriz com mesmo tamanho que a imagem de entrada, porém, com valores complexos que representam a Transformada de Fourier da mesma. Como sabemos, um número complexo é caracterizado por sua magnitude e por sua fase. Estas são as formas mais comuns de visualização da DFT de uma imagem.

A magnitude nos diz quanto de cada frequência existe na imagem original e a fase nos diz onde estas frequências estão localizadas. Para reconstrução da imagem original precisamos das duas informações.

A magnitude de uma imagem apresenta, em geral um valor muito alto na frequência zero (fator DC), correspondente ao somatório dos níveis de cinza de toda a imagem (ou ao nível médio, a depender de como a transformada é implementada). Os demais valores são, em geral, muito pequenos comparados a este valor DC. Quando os valores são quantizados em 256 níveis, para visualização, o resultado que podemos obter é um único ponto branco, correspondente à frequência zero, na origem e os demais pontos são pretos, pois foram todos quantizados para o menor nível de cinza, já que são muito menores que o valor máximo.

Para contornar esse problema, fazemos uma transformação que privilegie os valores baixos. A operação escolhida é a operação pontual logarítmica,  $g(n) = \text{FSHS}[\log(1+f(n))]$ , em que FSHS representa a operação de espalhamento de histograma.

A DFT 2-D (bidimensional) é dada pela expressão:

$$F(u, v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) e^{-j2\pi(ux/M + vy/N)} \quad (1)$$

onde  $F(u, v)$  representa o valor da transformada de  $f(x, y)$  e a expressão é calculada para  $u = 0, 1, 2, \dots, (M - 1)$  e  $v = 0, 1, 2, \dots, (N - 1)$ .

### Transformada Rápida de Fourier (FFT)

O número de multiplicações e adições complexas necessárias para calcular a DFT de uma imagem dada pela equação (1) é proporcional a  $N^2$ . Uma decomposição adequada da equação (1) pode tornar o número de operações de multiplicações e adições proporcional a  $N \cdot \log_2(N)$ . Esse procedimento de decomposição é chamado de Transformada Rápida de Fourier (FFT). A implementação computacional da DFT a partir da equação (1) não é viável, para computar a transformada de uma imagem utilizamos o algoritmo de FFT.

Este algoritmo se baseia no chamado método de dobramentos sucessivos, que consiste na divisão do somatório em dois somatórios, um para os números  $n$  pares e outra para os número  $n$  ímpares.

$$X_k = E_k e^{j\frac{2\pi}{N}k} + D_k \quad 0 \leq k < \frac{N}{2}$$

$$E_k = \sum_{n=0}^{\frac{N}{2}-1} E_{2n} e^{j\frac{2\pi}{N}kn}$$

$$D_k = \sum_{n=0}^{\frac{N}{2}-1} D_{2n+1} e^{j\frac{2\pi}{N}k(2n+1)}$$

$X_k$  representa o valor da transformada e  $E_k$  e  $D_k$  representam as componentes par e ímpar da transformada, respectivamente.

### **Filtragem**

A filtragem digital consiste num conjunto de técnicas destinadas a corrigir e realçar características de uma imagem. Correção é a remoção de características indesejáveis, enquanto melhoria/realce é a acentuação de características desejáveis. O uso de máscaras espaciais para processamento de imagens é usualmente chamado de filtragem espacial. Aqui utilizaremos filtros espaciais lineares e não-lineares para realce de imagens. Os denominados filtros passa-baixas atenuam ou eliminam as componentes de alta frequência no domínio da frequência enquanto as baixas frequências permanecem inalteradas.

Filtros passa-altas são complementares aos filtros passa-baixas, no sentido de que eliminam ou atenuam as componentes de baixa frequência e deixam passar as componentes de alta frequência.

Componentes de alta frequência caracterizam bordas e outros detalhes finos de uma imagem, assim o resultado de uma filtragem passa-baixas em uma imagem é o borramento desta. Características que variam lentamente em uma imagem, tais como contraste total e intensidade média são definidos pelas componentes de baixa frequência, assim o resultado de uma filtragem passa-altas é a redução destas características, correspondendo a uma aparente agudização da imagem.

Existe ainda outro tipo de filtragem denominado filtragem passa-faixa que remove regiões selecionadas de frequências entre altas e baixas frequências. Esta filtragem é usada principalmente para a restauração de imagens e raramente para o realce.

A técnica de filtragem espacial consiste em somar os produtos entre os coeficientes da máscara e as intensidades dos pixels sob a máscara numa posição específica da imagem.

$$\begin{bmatrix} w_1 & w_2 & w_3 \\ w_4 & w_5 & w_6 \\ w_7 & w_8 & w_9 \end{bmatrix}$$

A figura acima mostra uma máscara genérica 3x3. Denotando os níveis de cinza, isto é, intensidade de brilho em cada pixel, sob a máscara em qualquer posição por  $z_1, z_2, \dots, z_9$ , a resposta de uma máscara linear é:

$$R = w_1 z_1 + w_2 z_2 + \dots + w_9 z_9$$

Assim se o centro da máscara estiver numa posição  $(x, y)$  na imagem, o nível de cinza do pixel posicionado nesta mesma posição será substituído por  $R$ . A máscara é então movida para a próxima posição de pixel na imagem e o processo se repete. Isso continua até que todos os pixels tenham sido cobertos.

Abaixo especificamos alguns dos tipos de filtros mais usados.

Filtro de Suavização ou Filtragem espacial passa-baixas (filtro da média):

São usados para obter borramento e redução de ruído em imagens. O borramento é utilizado em pré-processamento para remoção de pequenos detalhes (menores que a máscara) em uma imagem, ou para preencher pequenos espaços entre objetos.

- Remoção de ruído: ruído é geralmente caracterizado por transições nítidas (sharp) nos tons de cinza. O filtro suaviza essas transições "sharp".
- Efeito colateral: imagem borrada (pois os contornos são também suavizados ...)

Um exemplo de máscara para filtragem passa-baixas é mostrado abaixo:

$$\frac{1}{9} \cdot \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Esta máscara borra a imagem e remove ruídos do tipo transições abruptas nos tons de cinza, suavizando estas transições. Também suaviza os contornos. Para normalizar dividimos o valor de  $R$  por um fator de acordo com o tamanho da máscara. O valor calculado corresponde à média das intensidades dos pixels sobrepostos pela máscara.

Filtro de aguçamento ou Filtragem passa-altas:

Tem como objetivo enfatizar detalhes finos em uma imagem ou realçar detalhes que tenham sido borrados em consequência de erros. Abaixo se encontra uma máscara para filtragem passa-altas:

$$\frac{1}{9} \cdot \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

Filtro de auto reforço:

Uma imagem filtrada por um filtro passa-altas pode ser computada pela diferença entre a imagem original e a versão filtrada passa-baixas daquela imagem, ou seja:

$$PA = O - PB$$

PA, PB e O representam a imagem após filtragem passa-altas, passa-baixas e a imagem original respectivamente.

Multiplicando-se a imagem original por uma fator de amplificação, denotado por  $A$ , resulta a definição de um filtro auto reforço.

$$\begin{aligned} \text{Alto Reforço} &= A \cdot O - PB \\ \text{Alto Reforço} &= A \cdot O - O + PA \\ \text{Alto Reforço} &= (A-1) \cdot O + PA \end{aligned}$$

Quando  $A$  for igual a 1 temos o passa-altas padrão. Quando  $A$  for maior ou que 1 parte da imagem original é adicionada de volta ao resultado passa-altas. Uma imagem após filtragem de auto reforço assemelha-se à imagem original com relativo grau de realce de bordas, que depende do valor de  $A$ .

Em termos de implementação, a máscara de filtragem auto reforço básica é:

$$\frac{1}{9} \bullet \begin{bmatrix} -1 & -1 & -1 \\ -1 & w & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

em que  $w$  é dado por:

$$w = 9.A - 1, A \geq 1.$$

O valor de  $A$  determina a natureza do filtro.

#### Filtragem por derivadas:

Utilizamos a filtragem por derivadas para detecção de bordas e regiões de transição rápida de nível de cinza. O método mais comum de diferenciação em aplicações de processamento de imagens é o gradiente. Para uma função  $f(x,y)$  o gradiente de  $f$  nas coordenadas  $(x, y)$  é definido como o vetor:

$$\mathbf{Y} = \frac{\partial f}{\partial x} \vec{i}_x + \frac{\partial f}{\partial y} \vec{i}_y$$

, onde  $\vec{i}_x$  e  $\vec{i}_y$  são os vetores unitários nas direções  $x$  e  $y$ , respectivamente.

Utilizamos a magnitude do vetor gradiente para as várias abordagens de diferenciação de imagens. A magnitude do vetor gradiente é dada por:

$$|\mathbf{Y}| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$$

Podemos aproximar a magnitude do gradiente de várias formas. Aqui usaremos a soma das seguintes diferenças em módulo para uma máscara 3x3:



O primeiro módulo da expressão acima aproxima a derivada na direção  $x$  e o segundo módulo aproxima a derivada na direção  $y$ .

Existem várias máscaras para implementar o gradiente de uma imagem. Duas delas muito utilizadas foram implementadas.

Os operadores de Prewitt consistem em duas máscaras 3x3. O resultado é a soma dos resultados de cada máscara. Abaixo se encontram estes operadores.

$$\frac{1}{9} \bullet \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad \frac{1}{9} \bullet \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

Do mesmo modo, os operadores de Sobel também consistem em duas máscaras e o resultado final á a soma da aplicação de cada máscara. Abaixo temos estes operadores.

$$\frac{1}{9} \bullet \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad \frac{1}{9} \bullet \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

Existem ainda outras máscaras para a implementação da filtragem por derivadas em imagens como os operadores cruzados de gradiente de Roberts. Porém, este operador é uma máscara 2x2, o que na prática torna-se mais difícil sua implementação.



Além o gradiente, podemos utilizar o laplaciano(derivada segunda) para realizarmos a filtragem por derivadas. O laplaciano é definido por:



O laplaciano de um imagem pode ser obtido aplicando a máscara abaixo:

$$\frac{1}{9} \cdot \begin{bmatrix} 0 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Porém, o laplaciano tem desvantagens em relação aos operadores acima. Ele é muito sensível ao ruído e ele gera bordas duplas.

### Rastreamento de bordas

Rastreamento de bordas é a técnica aplicada em imagens digitais para extrairmos bordas ou contornos. Ao extrairmos as bordas de um objeto em uma imagem, obtemos informações sobre a forma geral do mesmo. Com estas informações diferentes características são obtidas e utilizadas como elementos que mais tarde podem ser comparadas com diversos padrões de classificação. Portanto, uma correta extração de bordas da imagem irá nos fornecer características mais precisas, aumentando as chances de correta classificação.

Para realizar o estudo de rastreamento de bordas é necessário, primeiramente, ter noção sobre vizinhança entre pixels. Existem diversos tipos de vizinhanças, porém, utilizou-se apenas uma, a vizinhança-4. Esta corresponde aos pixels que estão logo acima, abaixo, à direita e à esquerda do pixel avaliado. A imagem abaixo ilustra uma vizinhança-4.

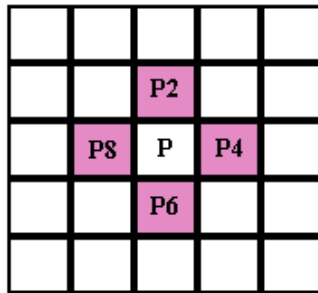


Figura 4 – Os pixels P2, P4, P6 e P8 são vizinhos-4 de P.

Para o rastreamento de bordas realiza-se os seguintes passos:

1. Faz-se uma varredura na imagem procurando o pixel de nível de cinza mais alto. Este será o pixel inicial, pelo qual a varredura irá começar;
2. Verifica-se a vizinhança-4 do pixel inicial procurando-se o de maior nível de cinza. Este será o próximo pixel base.
3. A partir destes dois pixels iniciais, determinaremos a direção e os pixel a serem avaliados. Abaixo está ilustrado como determinamos a direção a partir do pixel base atual (Patual) e do pixel base anterior (Panterior). Após determinada a direção da borda, verifica-se os pixels P1, P2 e P3 fazendo do pixel de maior nível de cinza dentre eles o pixel base atual. Caso mais de um pixel tenha o maior nível de cinza, pode-se escolher qualquer dos dois como pixel base atual.

<u>Direção vertical</u>	<u>Direção horizontal</u>	<u>Direção diagonal</u>
<i>A P B</i>	<i>x x A</i>	<i>Panteriorx</i>
<i>x Patual</i>	<i>PanteriorPatual</i>	<i>x Patual</i>
<i>x Panterior</i>	<i>x x P</i>	<i>x B P</i>

- Realiza-se os passos anteriores sucessivamente até que não haja mais nenhum pixel de borda para ser rastreado, ou ainda, quando o próximo pixel base for o pixel inicial, ou seja, o contorno tenha sido fechado.



Realizando os passos anteriores corretamente consegue-se rastrear as bordas de uma imagem previamente pré-processada. É importante que haja o pré-processamento da imagem antes de aplicarmos a técnica de rastreamento de bordas, para que a borda seja realçada não contenha falhas, as quais podem gerar um caminho errado que resultaria em uma borda parcialmente rastreada.












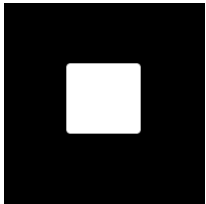
## RESULTADOS




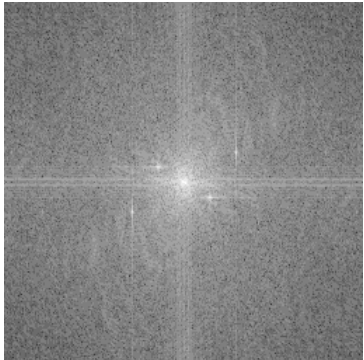



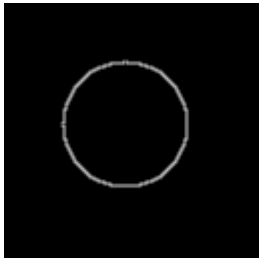
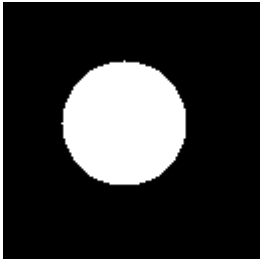
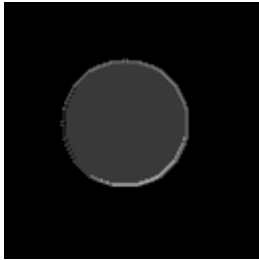
Utilizou-se a linguagem C++ aliada ao ambiente de desenvolvimento Microsoft Visual C++ 2008 Express Edition na implementação das técnicas estudadas e para a criação de um aplicativo reunindo todas elas. A seguir, temos algumas imagens da interface gráfica do aplicativo desenvolvido.

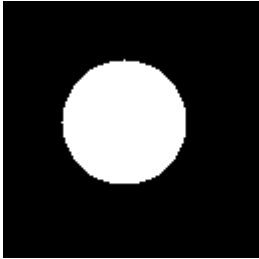
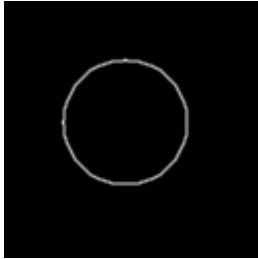
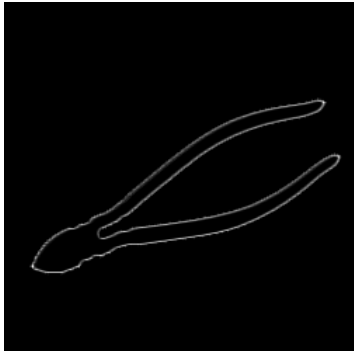
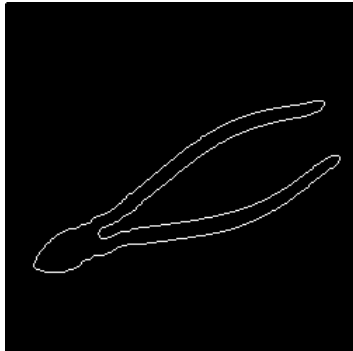


Abaixo encontra-se uma tabela com as imagens originais utilizadas nos teste e as imagens resultantes da aplicação de cada uma das técnicas implementadas.

Operação	Imagem(ns) original	Imagem processada
Adição pontual (L=50)		

<p>Multiplicação pontual (<math>P=2</math>)</p>		
<p>Limiarização (<math>T=125</math>)</p>		
<p>Espalhamento de histograma</p>		
<p>Adição de imagens</p>		
<p>Subtração de imagens</p>		
<p>Multiplicação de imagens</p>		

<p>Filtragem passa-baixas</p>		
<p>Transformada Rápida de Fourier (FFT)</p>		
<p>Filtragem passa-altas</p>		
<p>Filtragem por derivadas (operadores de Sobel)</p>		
<p>Filtragem por derivadas (operadores de Prewitt)</p>		

Filtragem por derivadas segunda (Laplaciano)		
Rastreamento de bordas		

### CONCLUSÕES

Os resultados obtidos estão de acordo com o esperado, significando que os algoritmos foram implementados corretamente e que podem ser utilizados em aplicações que requerem um pré-processamento de imagens. Na continuação deste projeto, estes algoritmos serão utilizados em tarefas de indexação e recuperação de informação em imagens de documentos.

### AGRADECIMENTOS

Ao CNPq pela bolsa de Iniciação Científica.

### REFERÊNCIAS BIBLIOGRÁFICAS

- BOVIK, A.I. Handbook of Image & Video Processing. Academic Press, 1<sup>st</sup> Edition, 2000.
- Carvalho, B. M. **Filtragem de Imagens no Domínio Espacial**. Disponível em: <<http://www.dimap.ufrn.br/~motta/dim411/Filtragem%20Espacial.pdf>> Acesso em: 22 jul. 2009.
- Costa, António. **Processamento de Imagem**. Disponível em: <[www.dei.isep.ipp.pt/~ACC/etc/PDI-tutorial.ppt](http://www.dei.isep.ipp.pt/~ACC/etc/PDI-tutorial.ppt)> Acesso em: 10 jun. 2009.
- Falcão, A. X.; Leite, N. J. **Fundamentos de Processamento de Imagem Digital**. Disponível em: <<http://www.dcc.unicamp.br/~cpg/material-didatico/mo815/9802/curso/curso.html>> Acesso em: 27 jul. 2009.
- Filho, K. S. O. **Filtragem**. Disponível em: < <http://www.if.ufrgs.br/ast/med/imagens/node38.htm>> Acesso em: 25 jul. 2009.
- Fisher, R.; Perkins, S.; Walker, A.; Wolfart, E. **Image Processing Operator Worksheets**. Disponível em: < <http://homepages.inf.ed.ac.uk/rbf/HIPR2/wksheets.htm> > Acesso em: 15 mar. 2009.
- GONZALEZ, R. C.; WOODS, R. E. Processamento de Imagens Digitais. Edgard Blucher, São Paulo, 2000.

HIRATA, Nina S. T. **Visão e Processamento de Imagens**. Disponível em:  
<<http://www.vision.ime.usp.br/~nina/cursos/mac0417-02/index.html#aulas>> Acesso em: 27 jul. 2009.

William, K. Pratt. Digital Image Processing, 3ª edição. John Wiley & Sons, Inc. Los Altos, California, 2001.

Young, I.T.; Gerbrands, J.J.; van Vliet, L.J. **Image Processing Fundamentals**. Disponível em:  
<<http://www.ph.tn.tudelft.nl/Courses/FIP/noframes/fip-Derivati.html>> Acesso em: 27 jul. 2009.