

UNIVERSIDADE FEDERAL DO CEARÁ  
DEPARTAMENTO DE ENGENHARIA DE TELEINFORMÁTICA  
PROGRAMA GRADUAÇÃO EM ENGENHARIA DE TELEINFORMÁTICA

# Identificação e Localização de Pontos Críticos em Imagens Digitais

Alex Torquato Souza Carneiro

FORTALEZA – CEARÁ  
DEZEMBRO 2007



UNIVERSIDADE FEDERAL DO CEARÁ  
DEPARTAMENTO DE ENGENHARIA DE ELÉTRICA  
PROGRAMA DE PÓS GRADUAÇÃO EM ENGENHARIA DE TELEINFORMÁTICA

# Identificação e Localização de Pontos Críticos em Imagens Digitais

## **Autor**

Alex Torquato Souza Carneiro

## **Orientador**

Prof. Dr. Paulo César Cortez

## **Co-orientador**

Msc. Rodrigo Carvalho Souza Costa

*Proposta de Monografia apresentada à  
Coordenação do Curso de Graduação  
em Engenharia de Teleinformática  
da Universidade Federal do Ceará  
como parte dos requisitos para  
obtenção do título de **Engenheiro de  
Teleinformática.***

FORTALEZA – CEARÁ

DEZEMBRO 2007

ALEX TORQUATO SOUZA CARNEIRO

## **Identificação e Localização de Pontos Críticos em Imagens Digitais**

Esta Monografia foi julgada adequada para a obtenção do título de Engenheiro de Teleinformática e aprovada em sua forma final pela Coordenação do Curso de Graduação em Engenharia de Teleinformática do Centro de Tecnologia da Universidade Federal do Ceará

---

Alex Torquato Souza Carneiro

Comissão Examinadora:

---

Prof. Dr. Paulo César Cortez  
Orientador

---

Eng. Msc. Rodrigo Carvalho Sousa Costa  
Co-orientador

---

Prof. Dr. João César de Moura Mota  
Universidade Federal do Ceará

---

Prof. Msc. Auzir Ripardo de Alexandria  
Centro Federal de Educação Tecnológica do  
Ceará

Fortaleza, 4 de dezembro de 2007

# Resumo

A localização de pontos críticos, característicos a um objeto particular, abre várias oportunidades de aplicações, como por exemplo em: reconhecimento, localização e identificação do objeto presente na cena, reconhecimento de características sutis presentes no objeto, dentre outras. Nesta monografia, é proposta uma abordagem que faz uso da imagem como é obtida, dispensando técnicas de segmentação e extração de contorno presentes em algoritmos de localização de pontos críticos tradicionais. Com relação às técnicas afins usadas: conversão entre espaços de cores, gradientes, transformada *Wavelet* e limiarização, todas são conhecidas da literatura e de fácil utilização. Um aspecto importante desta monografia é sua própria elaboração, em que foram usados ambiente e ferramentas de *software* livre, de modo que não houve nenhum gasto além do *hardware* e ainda podia contar-se com o apoio da comunidade mundial de *software* livre como colaboradores para o desenvolvimento das atividades relacionadas.

Os resultados obtidos na localização dos pontos críticos foram promissores, visto que as imagens de teste, apesar de algumas serem complexas, tiveram os pontos localizados de acordo com o esperado segundo análise visual (qualitativa). Em análises, como neste trabalho, torna-se difícil obter um parâmetro quantitativo para avaliação dos resultados.

**Palavras-chaves:** Processamento Digital de Imagens, Gradiente de Imagens Digitais, Transformada *Wavelet*, Pontos Críticos, *Software* Livre.

# Abstract

The localization of critical points that identifies a particular object opens up several opportunities for applications, for example in recognition, localization and identification of the object in the image, recognition of subtle features in the object, among others. In this monograph, it is proposed an approach that makes use of the image as it is obtained, without use techniques for segmentation and extraction of contour presents in traditional algorithms. With respect to the techniques used in this work: conversion between spaces of colors, gradients, wavelet transform and threshold, all are known from literature and easy to use. An important aspect of this monograph is its own development, in that environment and tools were used to free software, so there has been no spending beyond the hardware and could still account with the support of the world community of free software for the development of related activities.

The results obtained on the location of critical points were promising, because the images of test, despite some are complex, the points were located in accordance with the expected second visual analysis (qualitative). In analysis as in this work is difficult to obtain a quantitative parameter to evaluate the results.

**Keywords:** Digital Image Processing, Digital Image Gradient, Wavelet Transform, Critical Points, Free Software.

Dedico esta monografia à comunidade de *Software* Livre e à minha família.

# Agradecimentos

Agradeço primeiramente a Deus por me dar condições de realizar todas as atividades durante a minha graduação.

A minha família por sempre me dar condições de estudar e crescer profissional e socialmente, em especial aos meus pais, irmã e namorada por sempre estarem ao meu lado.

Aos amigos e colegas que estiveram comigo e me apoiaram.

Aos companheiros do Laboratório de Teleinformática onde realizei as minhas atividades desde 2005: Prof.s Guilherme e Cortez, Isaque, Gustavo, José Maria, Samy, Auzuir, Daniel Colaço, José Arimatéia Jr., Jonh, Victor Hugo, Robson Siqueira, Rodrigues Costa e Freitas, Fábio, Franco, Fred, Raquel, Cincinato, Kim e Antonio Carlos; amigos fiéis e pacientes que me ajudaram em todas as necessidades.

Aos demais alunos dos cursos de graduação e pós-graduação em Engenharia de Teleinformática que conviveram comigo nos últimos 4 anos.

“Todo homem, por natureza, quer saber.”

Aristóteles

# Sumário

<b>Lista de Figuras</b>	<b>x</b>
<b>Lista de Siglas</b>	<b>xi</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Motivação . . . . .	1
1.2 Objetivos . . . . .	2
1.3 Organização da Monografia . . . . .	2
<b>2 Ferramentas e Métodos</b>	<b>3</b>
2.1 Representação de Imagens . . . . .	3
2.2 Imagens Digitais . . . . .	5
2.2.1 Aplicações Utilizando Processamento de Imagens . . . . .	7
2.3 Gradientes . . . . .	7
2.3.1 Gradiente de Campos Escalares . . . . .	8
2.3.2 Gradiente em Imagens Digitais . . . . .	9
2.3.3 Tipos de Gradientes para Imagens . . . . .	13
2.4 Transformada <i>Wavelet</i> . . . . .	14
2.4.1 Análise de Fourier . . . . .	15
2.4.2 Decomposição <i>Wavelet</i> . . . . .	18
2.4.3 Transformada <i>Wavelet</i> 2D . . . . .	19
2.4.4 Transformada <i>Wavelet</i> Haar . . . . .	20
2.4.5 Transformada <i>Wavelet</i> Daubechies . . . . .	22

2.5	Pontos Críticos . . . . .	22
2.5.1	Técnicas de Localização de Pontos Críticos . . . . .	23
<b>3</b>	<b>Metodologia</b>	<b>29</b>
3.1	Ambiente de Desenvolvimento . . . . .	29
3.1.1	Imagens Utilizadas . . . . .	30
3.2	Método Proposto para Detecção de Pontos Críticos . . . . .	32
<b>4</b>	<b>Resultados</b>	<b>36</b>
4.1	Comparação entre Métodos de Gradiente . . . . .	36
4.2	Comparação entre Transformadas <i>Wavelets</i> . . . . .	39
4.3	Localização de Pontos Críticos . . . . .	45
<b>5</b>	<b>Conclusões e Perspectivas Futuras</b>	<b>51</b>
5.1	Perspectivas Futuras . . . . .	51
	<b>Referências Bibliográficas</b>	<b>54</b>

# Lista de Figuras

2.1	olho de animal (eqüino). . . . .	4
2.2	sistema de visão artificial. . . . .	4
2.3	<i>pixels</i> dispostos em malha quadrada. . . . .	5
2.4	imagem representada em níveis de cinza. . . . .	6
2.5	imagem representada em cores. . . . .	6
2.6	campo escalar descrito pela Equação 2.3. . . . .	8
2.7	campo vetorial descrito pela Equação 2.4. . . . .	9
2.8	imagem original (demonstrativo do ambiente gráfico KDE). . . . .	13
2.9	componente do gradiente horizontal por a) <i>Prewitt</i> e b) <i>Sobel</i> . . . . .	14
2.10	componente do gradiente vertical por a) <i>Prewitt</i> e b) <i>Sobel</i> . . . . .	14
2.11	gráfico descrito pelas Equações a) 2.15 e b) 2.16. . . . .	16
2.12	Transformada de Fourier das Equações a) 2.15 e b) 2.16. . . . .	17
2.13	exemplo de re-escala do sinal $x(t) = e^{-t}\cos(\omega t)$ em 3 níveis . . . . .	19
2.14	imagem original. . . . .	20
2.15	características extraídas da imagem na Figura 2.14 pelas <i>Wavelets</i> . . . . .	21
2.16	a) função de reescala e b) função mãe da Transformada <i>Wavelet</i> Haar. . . . .	21
2.17	a) função de reescala e b) função mãe da Transformada <i>Wavelet</i> Db4. . . . .	22
2.18	a) função de reescala e b) função mãe da Transformada <i>Wavelet</i> Db6. . . . .	24
2.19	a) função de reescala e b) função mãe da Transformada <i>Wavelet</i> Db8. . . . .	24
2.20	pontos críticos em a) um quadrado, b) losango e c) retas concorrentes. . . . .	24
2.21	fluxograma para localização de pontos críticos . . . . .	25

2.22	possíveis formas de conectividades. . . . .	26
2.23	possíveis formas de conectividades. . . . .	26
2.24	possíveis formas de conectividades. . . . .	27
2.25	imagens usadas no algoritmo de conectividade. . . . .	27
2.26	pontos críticos localizados pelo algoritmo de conectividade. . . . .	28
3.1	computador usado na elaboração da monografia. . . . .	30
3.2	imagem de baixa complexidade a) quadrado e b) retas coordenadas. . . . .	31
3.3	imagem de complexidade moderada a) quadrado e b) retas coordenadas. . . . .	31
3.4	imagem de alta complexidade a) peixe e b) Tux, mascote do Linux. . . . .	32
3.5	imagem de alta complexidade a) mesa e b) tomografia computadorizada de secção do tórax. . . . .	32
3.6	esquema geral do processamento envolvido na localização de pontos críticos. . . . .	33
3.7	a) imagem original e b) imagens geradas pela Transformada <i>Wavelet</i> Haar. . . . .	34
3.8	imagem de detalhes da imagem da Figura 3.7. . . . .	34
3.9	histograma da imagem de detalhes (Figura 3.8). . . . .	35
4.1	imagens de detalhes da Figura 3.2(a). . . . .	36
4.2	imagens de detalhes da Figura 3.2(b). . . . .	37
4.3	imagens de detalhes da Figura 3.3(a). . . . .	37
4.4	imagens de detalhes da Figura 3.3(b). . . . .	37
4.5	imagens de detalhes da Figura 3.4(a). . . . .	37
4.6	imagens de detalhes da Figura 3.4(b). . . . .	38
4.7	imagens de detalhes da Figura 3.5(a). . . . .	38
4.8	imagens de detalhes da Figura 3.5(b). . . . .	38
4.9	imagens de detalhes da Figura 3.2(a). . . . .	39
4.10	imagens de detalhes da Figura 3.2(b). . . . .	40
4.11	imagens de detalhes da Figura 3.3(a). . . . .	40
4.12	imagens de detalhes da Figura 3.3(b). . . . .	41
4.13	imagens de detalhes da Figura 3.4(a). . . . .	41
4.14	imagens de detalhes da Figura 3.4(b). . . . .	42
4.15	imagens de detalhes da Figura 3.5(a). . . . .	43
4.16	imagens de detalhes da Figura 3.5(b). . . . .	44

4.17 pontos críticos localizados na Figura 3.2(a).	45
4.18 pontos críticos localizados na Figura 3.2(b).	46
4.19 pontos críticos localizados na Figura 3.3(a).	46
4.20 pontos críticos localizados na Figura 3.3(b).	47
4.21 pontos críticos localizados na Figura 3.4(a).	47
4.22 pontos críticos localizados na Figura 3.4(b).	48
4.23 pontos críticos localizados na Figura 3.5(a).	49
4.24 pontos críticos localizados na Figura 3.5(b).	50

# Lista de Siglas

KDE	K Desktop Environment
GNU	GNU is Not Unix
RGB	Red Green Blue
CMY	Cian Magenta Yellow
YCbCr	Luma Chroma Blue and Red
HSV	Hue Saturation Value
Db	Daubechies
DDR2	Double Data Rating 2
RAM	Random Access Memory

## Introdução

Os pontos críticos podem representar os objetos presentes em uma cena (na imagem) e as suas formas. Existem diferentes tipos de pontos críticos, entre eles os chamados cantos, os quais representam um sub-conjunto dos pontos críticos. Os cantos podem representar características do objeto em análise.

Algumas das aplicações mais tradicionais em que usa-se pontos críticos são reconhecimento de padrões e rastreamento, em que os pontos são usados para identificar e localizar o objeto de interesse.

### 1.1 Motivação

---

O uso de câmeras representa atualmente uma das formas de sensoriamento e aquisição de informações com maior potencial de aplicações e inovações, cujas imagens contêm informações importantes sobre os objetos presentes; destacando-se as bordas e os cantos, os quais formam um sub-conjunto dos pontos críticos.

À partir dos pontos críticos é possível fazer o reconhecimento de objetos existentes na imagem original fazendo uso de técnicas de reconhecimento de padrões, como, por exemplo, classificadores baseados em redes neurais. No trabalho desenvolvido por Gomes (2006) fez-se o uso de Transformadas *Wavelet* para localização de objetos, sendo esta uma das principais motivações desta monografia.

Outra aplicação interessante que depende da localização de pontos críticos consiste na utilização de algoritmos de rastreamento, de modo que a abordagem proposta permitiria o desenvolvimento de uma aplicação com potencial de ser adotada em várias situações.

---

## 1.2 Objetivos

---

Esta monografia tem como principal objetivo o desenvolvimento de um sistema, em plataforma *GNU/Linux* (livre), para a identificação e localização de pontos críticos em imagens digitais.

Como objetivos específicos, almeja-se:

- ▶ elaborar uma técnica completa para localização dos pontos críticos desde a captura da imagem até a definição das coordenadas dos pontos desejados;
- ▶ permitir que o algoritmo apresentado possa ser usado aplicações que envolvam processamento digital de imagens, de acordo com a necessidade relativa à sensibilidade na identificação de um ponto como crítico;
- ▶ implementar o sistema (*software*) em plataforma *GNU/Linux* fazendo uso apenas de ferramentas de *software* livre na elaboração dos algoritmos para simulações e análise dos resultados obtidos.

---

## 1.3 Organização da Monografia

---

Esta monografia está organizada em 5 Capítulos. O Capítulo 2 apresenta os fundamentos teóricos e ferramentas matemáticas utilizadas. O Capítulo 3 apresenta e descreve a plataforma de desenvolvimento (*hardware* e *software*), o algoritmo proposto e sua metodologia. O Capítulo 4 apresenta e avalia os resultados obtidos e o Capítulo 5 faz as considerações finais e as perspectivas de trabalhos futuros.

# Capítulo 2

## Ferramentas e Métodos

O processamento digital de imagens é a área do conhecimento que trata da modelagem matemática, análise, projeto e implementação em *software* e em *hardware* de sistemas digitais voltados ao tratamento de informação pictorial com o propósito de torná-la mais adequada à interpretação por seres humanos ou máquinas, ou ainda para obter maior eficiência de armazenamento e transmissão (GONZALEZ; WOODS, 2002).

Este capítulo descreverá o estado da arte das técnicas de detecção de pontos críticos em uma imagem digital e as ferramentas necessárias para desenvolver este tipo de aplicação.

### 2.1 Representação de Imagens

---

Imagens são formadas a partir da percepção da luz como estímulos dos órgãos foto-sensitivos (olhos) dos animais, Figura 2.1. As estruturas visuais dos animais encontram-se dentre as mais complexas e de maior relevância para a adaptação do animal em seu ambiente. Estudos ainda da década de 90 indicam a presença de recursos existentes no sistema visual especializados em realizar operações não-lineares. Alguns desses recursos incluem a localização de cantos ou fins de linha (MOTA, 1999).

Os animais têm os olhos como a principal interface de comunicação com o meio em que encontram-se, assim podendo procurar por indivíduos da mesma espécie, alimentos (vegetais ou caça, para animais herbívoros ou carnívoros, respectivamente) ou abrigo contra os perigos existentes.

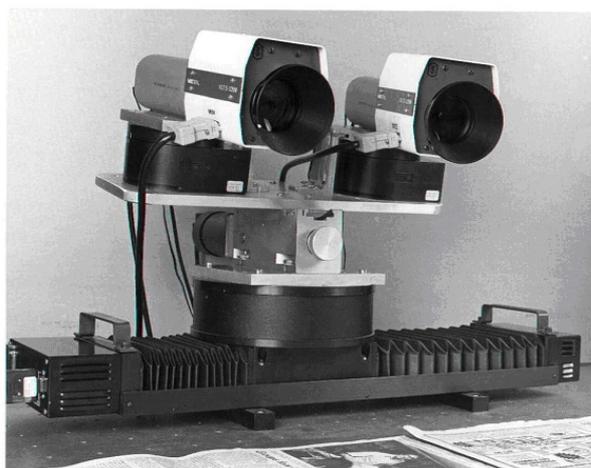


**Figura 2.1:** olho de animal (equino).

Inicialmente, as imagens foram concebidas para a representação de ambientes, objetos, seres vivos, dentre outros.

Baseando-se no que existe na natureza há milhões de anos, hoje existem sistemas de visão artificial, em que dispositivos foto-sensitivos captam a luz emitida pelo ambiente, gerando imagens a partir desta e possibilitam que sejam realizados processamentos para aplicações específicas.

Com o advento dos computadores digitais, tornou-se muito conveniente ter as imagens digitalizadas para que pudessem ser processadas e analisadas por estes. Assim são criados os sistemas de visão computacional. Na Figura 2.2 é mostrado um sistema de visão computacional típico baseado no sistema visual humano.



**Figura 2.2:** sistema de visão artificial.

Os sistemas artificiais de visão, conforme apresentado, fazem uso de imagens digitalizadas que consistem em matrizes de elementos escalares ou vetores a depender

do tipo de imagem.

## 2.2 Imagens Digitais

Uma imagem  $I$  pode ser definida como uma função vetorial  $\mathbf{I} : A \rightarrow B$ , em que  $A$  e  $B$  são dois conjuntos finitos, em que  $A \subseteq \mathbb{Z}^2$  e  $B \subseteq \mathbb{Z}^n, n \in \mathbb{N}_0$ .

Em imagens planas, um elemento  $\mathbf{a} \in A$  é definido como a posição de um determinado *pixel* (*picture element*) na imagem, sendo dado por:

$$\mathbf{a} = (x, y) \quad (2.1)$$

Um elemento  $\mathbf{b} \in B^n$ , tal que  $n = 1$ , é um escalar cujo valor  $p$  (*pixel*) na posição  $\vec{x}$ .

Define-se então *pixel* como o elemento básico da imagem, cuja posição ( $\mathbf{a}$ ) possui um valor de intensidade ( $p$ ) a ela associado (GONZALEZ; WOODS, 2002), conforme mostrado na Figura 2.3.

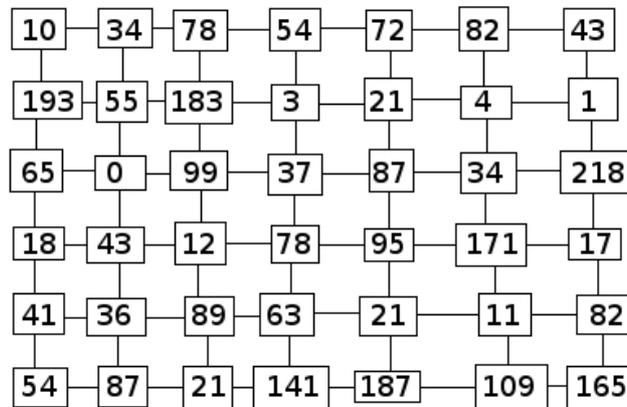


Figura 2.3: *pixels* dispostos em malha quadrada.

Desta forma, uma imagem digital pode ser considerada como sendo uma matriz (ou *grid*) cujos índices de linhas ( $x$ ) e de colunas ( $y$ ) identificam um ponto na imagem, e  $p$  é o correspondente valor do elemento da matriz o qual identifica o nível de intensidade naquele ponto ( $x, y$ ).

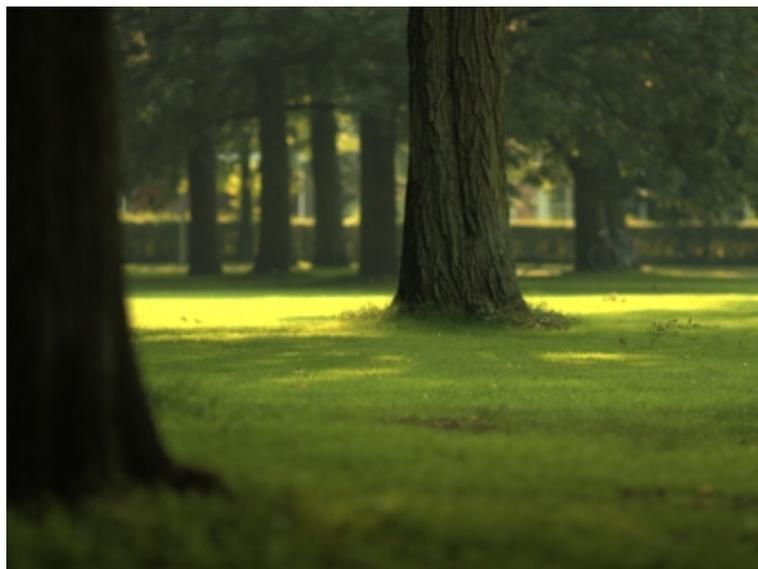
As imagens podem ser classificadas de acordo com a dimensão do valor associado a cada *pixel* da imagem. Conforme citado, quando  $p$  é um vetor uni-dimensional,

ou seja é um escalar ( $p \in B$  com  $B \subset \mathbb{Z}$ ), a imagem é considerada como **monocromática**, exemplificada pela Figura 2.4. Os valores de  $p$  indicam níveis de cinza.



**Figura 2.4:** imagem representada em níveis de cinza.

Contudo, quando  $n \geq 2$ ,  $\underline{\mathbf{b}}$  é um vetor multi-dimensional ( $\underline{\mathbf{b}} \in B$  com  $B \subset \mathbb{Z}^n$ ), em que  $n$  é o número de dimensões e a imagem é representada como **imagem colorida**, representada pela Figura 2.5.



**Figura 2.5:** imagem representada em cores.

Uma imagem colorida representa a cor em cada *pixel* da imagem baseado em

um padrão de cor que facilita a interpretação do formato de cores. Os formatos de cores mais conhecidos são (GONZALEZ; WOODS, 2002):

- ▶ RGB: descrito através das cores primárias (vermelho, verde e azul);
- ▶ CMY: muitos dispositivos de impressão utilizam este padrão através da deposição dos pigmentos ou cores secundárias no papel;
- ▶ YCrCb: descreve a cor baseada na luminância (Y) e em termos de pigmentação vermelha (Cr) e azul (Cb), muito utilizado em transmissão e codificação de vídeo (TV, video componente e *Color Stream*);
- ▶ HSV: descreve a cor baseado na intensidade de cor (H), saturação (S) e brilho (V);

As componentes do vetor  $\mathbf{b}$  em imagens coloridas expressam a intensidade  $p$  do pixel, também os seus formentos de cores.

### 2.2.1 Aplicações Utilizando Processamento de Imagens

As definições das imagens descritas acima são úteis para a utilização de técnicas de processamento digital de imagens e visão computacional (GONZALEZ; WOODS, 2002); as quais são importantes em aplicações tais como:

- ▶ supervisão de processos industriais, principalmente em ambientes insalubres;
- ▶ análise de imagens médicas visando o auxílio ao diagnóstico;
- ▶ sensoriamento e localização de objetos (GOMES, 2006);
- ▶ reconhecimento de padrões em objetos e ambientes;
- ▶ monitoramento residencial ou corporativo, para identificação de indivíduos e segurança.

## 2.3 Gradientes

---

No contexto do processamento de imagens, o estudo dos vetores gradientes têm como efeito mais significativo a avaliação e o realce dos detalhes, tais como bordas, porque suas componentes representam taxas de variações dos elementos de imagem e relação e variação de posição do *pixel*.

### 2.3.1 Gradiente de Campos Escalares

Um campo escalar é definido como uma função  $f$  que tem como domínio um espaço vetorial  $\mathbb{R}^n$  e como contra-domínio um espaço escalar  $\mathbb{R}$  (APOSTOL, 1998).

O gradiente de um campo escalar do tipo  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  é definido por:

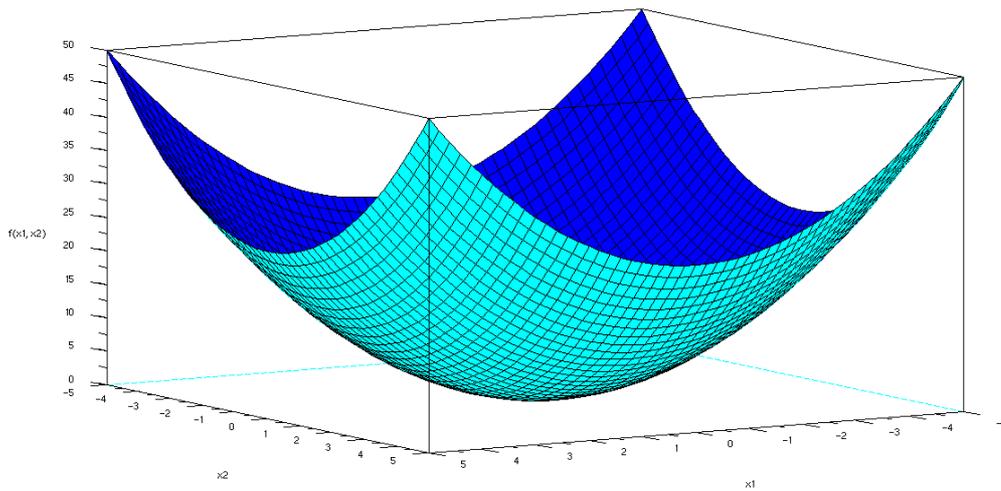
$$\nabla f(x_1, x_2, \dots, x_n) = \frac{\partial f}{\partial x_1} \vec{a}_1 + \frac{\partial f}{\partial x_2} \vec{a}_2 + \dots + \frac{\partial f}{\partial x_n} \vec{a}_n \quad (2.2)$$

em que  $\vec{a}_i$  é o vetor unitário na direção  $x_i$ .

Para exemplificar a informação apresentada, toma-se o seguinte campo escalar  $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ , definido como:

$$f(x_1, x_2) = x_1^2 + x_2^2, \quad (2.3)$$

em que este campo é mostrado ilustrado na Figura 2.6.

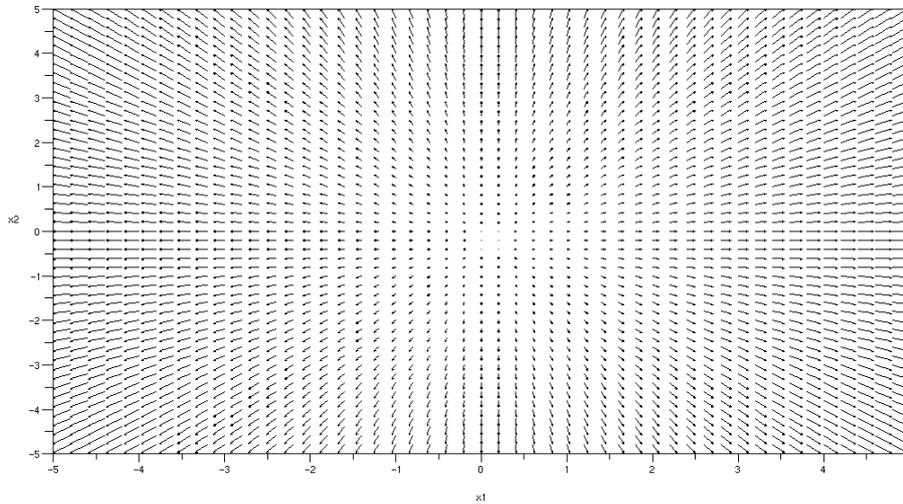


**Figura 2.6:** campo escalar descrito pela Equação 2.3.

O campo escalar usado como exemplo tem sua definição feita por uma expressão analítica em  $\mathbb{R}^2$ , podendo então ser definido o seu gradiente analiticamente, aplicando as técnicas de derivação parcial utilizando a Equação 2.2, resultando na Equação 2.4:

$$\nabla f(x_1, x_2) = \frac{\partial f}{\partial x_1} \vec{a}_1 + \frac{\partial f}{\partial x_2} \vec{a}_2 = 2(x_1 \vec{a}_1 + x_2 \vec{a}_2) \quad (2.4)$$

O campo vetorial obtido através desta equação é representado na Figura 2.7.



**Figura 2.7:** campo vetorial descrito pela Equação 2.4.

Um campo gradiente possui as seguintes propriedades (LARSON; HOSTETLER; EDWARDS, 1998):

- ▶ quando  $\nabla f = 0$ , então  $D_{\vec{u}}f = 0, \forall \vec{u}$ , em que  $D_{\vec{u}}f$  é a derivada de  $f$  na direção de  $\vec{u}$ ;
- ▶  $\nabla f$  determina a direção de maior crescimento de  $f$ ;
- ▶  $-\nabla f$  determina a direção de maior decréscimo de  $f$ ;

Conclui-se então que, para campos escalares  $f : \mathbb{R}^2 \rightarrow \mathbb{R}$  :

- ▶ o conjunto de pontos de vetores de gradiente são nulos representa pontos de área plana;
- ▶ o conjunto de pontos de valores de gradiente são mais elevados indica zona de maior aclividade ou declividade (transição entre níveis).

### 2.3.2 Gradiente em Imagens Digitais

Em um sinal discreto unidimensional  $f[n]$ , em que  $n \in \mathbb{Z}$ , a derivada de 1ª ordem (taxa de variação da função) é implementada numericamente conforme a equação (MANOLAKIS, 2007):

$$f'[n] = \frac{f[n] - f[n - 1]}{\Delta n}, \quad (2.5)$$

em que  $\Delta n = n - (n - 1) = 1$ , desta forma a expressão final obtida para a derivada é dada por:

$$f'[n] = f[n] - f[n - 1]. \quad (2.6)$$

Esta forma é conveniente quando o sinal usa interpolação de primeira ordem, contudo, pela Série de Taylor (LEITHOLD, 2002), quanto maior a ordem da interpolação, menor o erro de aproximação.

Por exemplo, seja uma função discreta  $f[n] = a \cdot n + b$ , substituindo esta função na Equação 2.6, tem-se:

$$f'[n] = f[n] - f[n - 1] = (a \cdot n + b) - [a \cdot (n - 1) + b] \Rightarrow$$

$$f'[n] = a \cdot n + b - a \cdot n + a - b \Rightarrow$$

$$f'[n] = a$$

Observa-se que o resultado obtido é o mesmo para uma função contínua  $f(x) = a \cdot x + b$ , em que  $f'(x) = a$ .

Em seguida é mostrado outro exemplo, seja uma função discreta  $f[n] = a \cdot n^2 + b \cdot n + c$ , substituindo esta função na Equação 2.6, tem-se:

$$f'[n] = f[n] - f[n - 1] = (a \cdot n^2 + b \cdot n + c) - [a \cdot (n - 1)^2 + b \cdot (n - 1) + c] \Rightarrow$$

$$f'[n] = (a \cdot n^2 + b \cdot n + c) - [a \cdot n^2 - 2 \cdot n \cdot a + a + b \cdot n - b + c] \Rightarrow$$

$$f'[n] = 2 \cdot a \cdot n - a + b.$$

Neste caso, ocorre um erro entre a expressão obtida e o equivalente para a função contínua  $f(x) = a \cdot x^2 + b \cdot x + c$ , em que  $f'(x) = 2 \cdot a \cdot x + b$ .

Finalizando, observa-se então que a derivada numérica  $f'[n]$  pode ser vista como

produto escalar do vetor  $\begin{bmatrix} f[u] \\ f[u - 1] \end{bmatrix}$  com o vetor:

$$\mathbf{h} = \begin{bmatrix} 1 \\ -1 \end{bmatrix},$$

$$\text{ou seja } f'[n] = [f[u] f[u-1]] \cdot \begin{bmatrix} 1 \\ -1 \end{bmatrix}.$$

Pode-se ainda enunciar  $f'[n]$  como resultado de uma operação de convolução de entrada  $f[n]$  em um sistema  $h[n]$  composto de duas componentes:  $h[0] = 1$  e  $h[1] = -1$ .

O uso da média aritmética  $\bar{f}'[n]$  entre as derivadas anterior ( $f'_1[n] = f[n] - f[n-1]$ ) e posterior ( $f'_2[n] = f[n+1] - f[n]$ ) a posição  $n$ , fornece uma precisão maior à representação discreta da derivada da função em relação ao seu valor contínuo. Desta forma,  $\bar{f}'[n]$  pode ser representada por:

$$\bar{f}'[n] = \frac{f'_1[n] + f'_2[n]}{2}, \quad (2.7)$$

resolvendo a Equação 2.7 obtém-se:

$$\bar{f}'[n] = \frac{f[n+1] - f[n-1]}{2} \quad (2.8)$$

As funções crescentes ou decrescentes no trecho avaliado apresentam resultados numéricos melhor correlacionados com a derivada média entre os pontos imediatamente vizinhos a  $n$ . Vizinhos mais distantes tendem a não representar correlação forte com a aquele em  $n$ , o que poderia levar a uma ponderação mais fraca naqueles pontos na composição de uma média ponderada. Por outro lado, variações rápidas da função em torno de  $n$  podem levar a uma derivada média nula, o que provoca um erro importante na direção da função em torno de  $n$ .

Seja o exemplo de cálculo da derivada média com a função discreta  $f[n] = a \cdot n + b$ . Substituindo esta função na Equação 2.8, temos que:

$$\begin{aligned} f'[n] &= \frac{f[n-1] - f[n+1]}{2} = \frac{[a \cdot (n+1) + b] - [a \cdot (n-1) + b]}{2} \Rightarrow \\ f'[n] &= \frac{a \cdot n + a + b - a \cdot n + a - b}{2} \Rightarrow \\ f'[n] &= a \end{aligned}$$

Observa-se que o resultado obtido é o mesmo para a derivada da função contínua  $f(x) = a \cdot x + b$ . Em seguida é mostrado outro exemplo, em que a função discreta

$f[n] = a \cdot n^2 + b \cdot n + c$ , é derivada a partir da Equação 2.8, temos que:

$$f'[n] = \frac{f[n+1] - f[n-1]}{2} = \frac{[a \cdot (n+1)^2 + b \cdot (n+1) + c] - [a \cdot (n-1)^2 + b \cdot (n-1) + c]}{2} \Rightarrow$$

$$f'[n] = \frac{[a \cdot n^2 + 2 \cdot n \cdot a + a + b \cdot n + b + c] - [a \cdot n^2 - 2 \cdot n \cdot a + a + b \cdot n - b + c]}{2} \Rightarrow$$

$$f'[n] = \frac{4 \cdot a \cdot n + 2 \cdot b}{2} = 2 \cdot n + b.$$

Observa-se, assim, que os valores das derivadas obtidos, discreta e contínua, são iguais.

Analisando como convolução, com a nova abordagem, a derivada é obtida ao se convolver o sinal original  $f[n]$  com a resposta ao impulso  $h[n]$ , escrito na forma vetorial:

$$\underline{\mathbf{h}} = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} \quad (2.9)$$

Considerando agora os sinais bi-dimensionais (imagens), levando em conta a definição de gradiente apresentada, a aplicação do gradiente em uma imagem consiste na aplicação da derivada em cada eixo da mesma.

Fazendo uso da definição de *pixel*, descrita pela Equação 2.1 e da definição proposta por Mlsa e Rodrigues (2005), o gradiente de uma imagem  $I$  pode ser obtido pela convolução a seguir:

$$\nabla I(x, y) = I(x, y) * [H_x(x, y) \vec{a}_x + H_y(x, y) \vec{a}_y] \quad (2.10)$$

em que  $'*$ ' indica o operador convolução,  $I(x, y)$  representa um ponto genérico da imagem,  $\nabla$  representa o operador gradiente,  $\vec{a}_x$  e  $\vec{a}_y$  referem-se a cada uma das imagens geradas pela aplicação do gradiente, respectivamente, as componentes horizontal e vertical e  $\mathbf{H}_x(x, y)$  e  $\mathbf{H}_y(x, y)$  são as matrizes que convolvem com a imagem, gerando, respectivamente, as componentes horizontal e vertical do gradiente da imagem. Esta equação está de acordo com a metodologia de cálculo de derivadas apresentada anteriormente.

As matrizes  $\mathbf{H}_x(x, y)$  e  $\mathbf{H}_y(x, y)$  postas com dimensão 3x3, pois assim:

- ▶ obtém uma melhor aproximação, conforme foi mostrado anteriormente;
- ▶ possibilitam a influência dos *pixels* vizinhos no cálculo do gradiente, assim aumentando a correlação entre os pixels da imagem resultante.

### 2.3.3 Tipos de Gradientes para Imagens

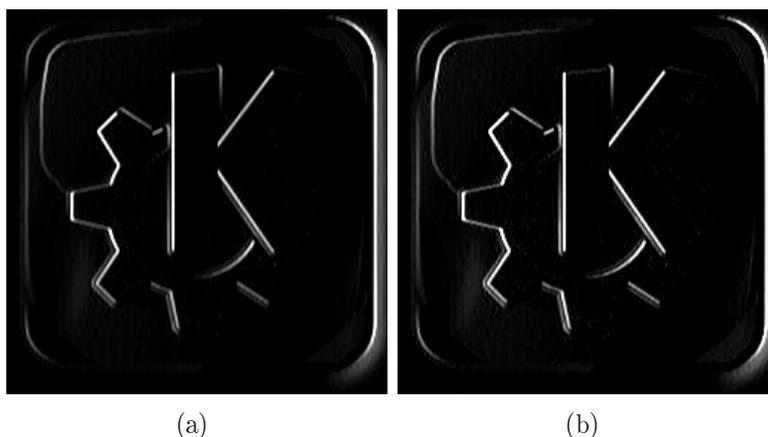
Os tipos de gradiente adotados nesta monografia são *Prewitt* e *Sobel* (GONZALEZ; WOODS, 2002; MLSA; RODRIGUES, 2005). As matrizes que convolvem com a imagem são dadas por:

$$\begin{array}{l}
 \textit{Sobel} \quad \mathbf{H}_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad \text{e} \quad \mathbf{H}_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \\
 \textit{Prewitt} \quad \mathbf{H}_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \quad \text{e} \quad \mathbf{H}_y = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}
 \end{array}$$

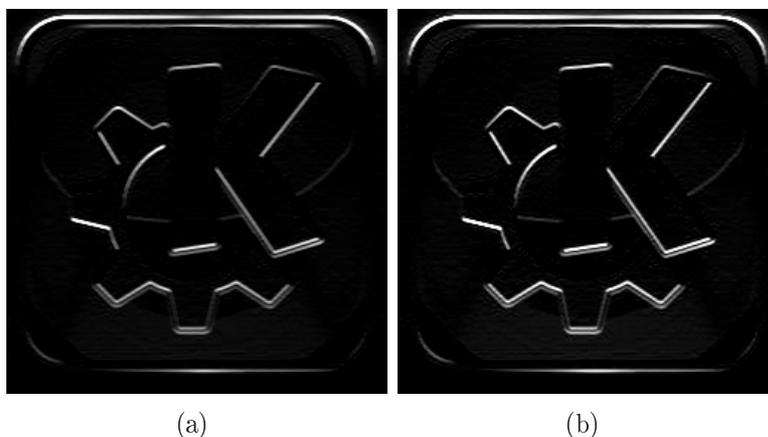
A seguir são apresentados os resultados da aplicação do gradiente em uma imagem (Figura 2.8), as imagens exibidas representam a intensidade do valor do gradiente obtido.



**Figura 2.8:** imagem original (demonstrativo do ambiente gráfico KDE).



**Figura 2.9:** componente do gradiente horizontal por a) *Prewitt* e b) *Sobel*.



**Figura 2.10:** componente do gradiente vertical por a) *Prewitt* e b) *Sobel*.

## 2.4 Transformada *Wavelet*

---

A Transformada *Wavelet* é uma ferramenta de análise espaço-freqüência que tem sido intensamente estudada durante as últimas décadas (COSTA, 2006).

No início, a Transformada *Wavelet* era usada apenas na matemática, porém, em poucos anos, esta demonstrou um grande potencial de aplicabilidade em diversos campos do conhecimento (VIDAKOVIC; MUELLER, 2003).

Muitos dos avanços obtidos nestes estudos foram desenvolvidos devido à cooperação de Ingrid Daubechies e Stephane Mallat. Daubechies (1988) desenvolveu uma família de *wavelets* com base compacta (*compact support*) e Mallat (MALLAT, 1989) pôs a Transformada *Wavelet* no conceito de decomposição multi-resolução de sinais, além da implementação da transformada rápida baseada em conceitos de filtragem (COSTA, 2006).

A definição matemática dos coeficientes da Transformada *Wavelet* de uma função  $f(x)$  é dada por:

$$c_{j_0}[k] = \int_{\langle D_\phi \rangle} f(x) \phi_{j_0,k} dx \quad (2.11)$$

$$d_j[k] = \int_{\langle D_\psi \rangle} f(x) \psi_{j,k} dx \quad (2.12)$$

em que a Equação 2.11 apresenta a propriedade de reescala da Transformada *Wavelet* e a Equação 2.12 apresenta a propriedade de extração de detalhes.

O termo  $j_0$  presente nas Equações 2.11 e 2.13 representa a reescala inicial da composição *Wavelet* (COSTA, 2006). Nesta monografia é usada apenas a informação relativa aos detalhes.

De modo inverso, a função original pode ser obtida por (VIDAKOVIC; MUELLER, 2003):

$$f(x) = \sum_k c_{j_0}[k] \phi_{j_0,k}(x) + \sum_{j=j_0}^{\infty} \sum_k c_{j_0}[k] \phi_{j_0,k}(x) \quad (2.13)$$

Observa-se que a Transformada *Wavelet* de um sinal é semelhante a Transformada de Fourier, porém a primeira é aplicada localmente no sinal. A aplicação local da Transformada *Wavelet* caracteriza-se pelos os núcleos da transformada  $\phi_{j_0,k}$ , e  $\psi_{j,k}$  serem definidos em um intervalo finito.

### 2.4.1 Análise de Fourier

A Transformada de Fourier de uma função  $f(t)$  é dada por (MANOLAKIS, 2007):

$$F(\omega) = \int_{-\infty}^{\infty} f(t) e^{-j\omega t} dt \quad (2.14)$$

Uma característica importante da Transformada de Fourier é sua análise global do sinal, de forma que sinais definidos por expressões diferentes em intervalos diferentes têm a mesma Transformada de Fourier quando os intervalos são trocados. Um exemplo é mostrado nas Figuras 2.11(a) a 2.12(b). Na Figura 2.11(a) é ilustra

o gráfico descrito por:

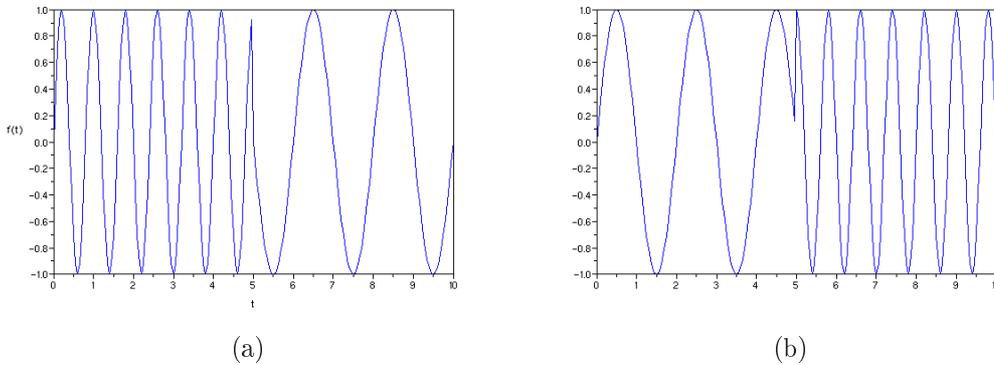
$$f(t) = \begin{cases} \sin(w_1 t), & 0 < t < t_0 \\ \sin(w_2 t), & t_0 < t < 2t_0 \\ 0, & t < 0 \text{ ou } t > 2t_0 \end{cases} \quad (2.15)$$

em que  $w_1 > w_2$ . Na Figura 2.12(a) é ilustrado o gráfico descrito por:

$$f(t) = \begin{cases} \sin(w_2 t), & 0 < t < t_0 \\ \sin(w_1 t), & t_0 < t < 2t_0 \\ 0, & t < 0 \text{ ou } t > 2t_0 \end{cases} \quad (2.16)$$

em que  $w_1 < w_2$ .

Pode-se observar pelas Figuras 2.11(b) e 2.12(b) que as Transformadas de Fourier obtidas são iguais, apesar da diferença entre as Equações 2.15 e 2.16. Este fato mostra que a Transformada de Fourier não extrai informações temporais do sinal.

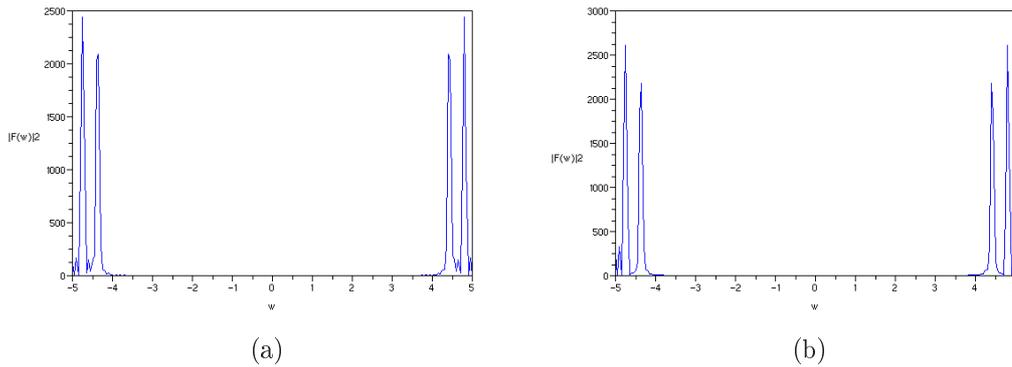


**Figura 2.11:** gráfico descrito pelas Equações a) 2.15 e b) 2.16.

Para o caso mostrado, a diferença quadrática média (entre as duas Transformadas de Fourier) é de  $9,207e-30$ . Tendo em vista que estes resultados foram obtidos em simulações numéricas, este pequeno erro evidencia que os espectros são iguais.

Justificando a observação: sejam  $g_1(t)$  e  $g_2(t)$  duas funções periódicas, de período divisor ( $t_0$ ), sendo que  $e^{j\omega t_0} = 1$ .

Vale ressaltar que a Transformada de Fourier de uma função periódica pode ser



**Figura 2.12:** Transformada de Fourier das Equações a) 2.15 e b) 2.16.

calculada levando em consideração apenas um intervalo múltiplo inteiro do período da função.

$$f(t) = \begin{cases} g_1(t), 0 < t < t_0 \\ g_2(t), t_0 < t < 2t_0 \\ 0, t < 0 \text{ ou } t > 2t_0 \end{cases} \Rightarrow$$

$$F(\omega) = \int_{-\infty}^{\infty} f(t)e^{-j\omega t} dt = \int_0^{2t_0} f(t)e^{-j\omega t} dt = \underbrace{\int_0^{t_0} g_1(t)e^{-j\omega t} dt}_{G_1(\omega)} + \underbrace{\int_{t_0}^{2t_0} g_2(t)e^{-j\omega t} dt}_{H(\omega)} \Rightarrow$$

$$H(\omega) = \int_{t_0}^{2t_0} g_2(t)e^{-j\omega t} dt = \int_0^{t_0} \underbrace{g_2(t-t_0)}_{g_2(t)} e^{-j\omega(t-t_0)} dt = \underbrace{e^{j\omega t_0}}_1 \underbrace{\int_0^{t_0} g_2(t)e^{-j\omega t} dt}_{G_2(\omega)} \Rightarrow$$

$$F(\omega) = G_1(\omega) + G_2(\omega).$$

Para:

$$f(t) = \begin{cases} g_1(t), 0 < t < t_0 \\ g_2(t), t_0 < t < 2t_0 \\ 0, t < 0 \text{ ou } t > 2t_0 \end{cases} \Rightarrow$$

$$\begin{aligned}
F(\omega) &= \int_{-\infty}^{\infty} f(t)e^{-j\omega t} dt = \int_0^{2t_0} f(t)e^{-j\omega t} dt = \underbrace{\int_0^{t_0} g_2(t)e^{-j\omega t} dt}_{G_2(\omega)} + \underbrace{\int_{t_0}^{2t_0} g_1(t)e^{-j\omega t} dt}_{H(\omega)} \Rightarrow \\
H(\omega) &= \int_{t_0}^{2t_0} g_1(t)e^{-j\omega t} dt = \int_0^{t_0} \underbrace{g_1(t-t_0)}_{g_1(t)} e^{-j\omega(t-t_0)} dt = \underbrace{e^{j\omega t_0}}_1 \underbrace{\int_0^{t_0} g_1(t)e^{-j\omega t} dt}_{G_1(\omega)} \Rightarrow \\
F(\omega) &= G_2(\omega) + G_1(\omega).
\end{aligned}$$

Assim mostrando que a Transformada de Fourier não extrai informações temporais do sinal.

### 2.4.2 Decomposição *Wavelet*

Uma função  $\phi(x)$ , definida em um domínio finito  $D_x$  e passível de ser um núcleo da Transformada *Wavelet* se esta obedecer as propriedades apresentadas a seguir (GONZALEZ; WOODS, 2002; VIDAKOVIC; MUELLER, 2003):

$$\int_{D_x} |\phi(x)|^2 dx < \infty \quad (2.17)$$

$$\int_{\langle D_\phi \rangle} \phi(x) dx = 0 \quad (2.18)$$

A Equação 2.17 mostra que a função  $\phi(x)$  é limitada e a Equação 2.18 mostra que a média temporal de  $\phi(x)$  é nula.

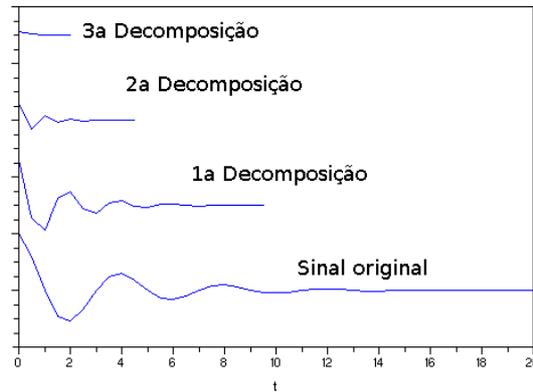
Outra característica freqüente com relação as funções do núcleo da Transformada *Wavelet* é que estas são reais e ortonormais (MLSA; RODRIGUES, 2005).

Uma vez definido que  $\phi(x)$  pode ser um núcleo de uma Transformada *Wavelet*, esta função deverá percorrer todo o domínio do sinal  $f(x)$  em questão e deverá ser capaz de identificar informações no tempo e na freqüência. Para atender estas exigências, define-se uma nova função  $\phi_{j,k}(x)$ :

$$\phi_{j,k}(x) = 2^{j/2} \phi(2^{-j}x - k) \quad (2.19)$$

em que os termos  $j$  e  $k$  realizam, respectivamente, as funções de re-escala (dilatação e compressão) e translação de  $\phi(x)$ .

Com relação a re-escala da decomposição *Wavelet*, esta pode ser interpretada como um algoritmo piramidal (GONZALEZ; WOODS, 2002), em que o sinal original é a base da pirâmide e cada nível da decomposição é um nível da pirâmide, conforme mostrado na Figura 2.13.



**Figura 2.13:** exemplo de re-escala do sinal  $x(t) = e^{-t} \cos(\omega t)$  em 3 níveis

A mesma análise pode ser estendida para 2 dimensões.

### 2.4.3 Transformada *Wavelet* 2D

Uma determinada Transformada *Wavelet* 2D é obtida por uma extensão da equivalente 1D. A aplicação da Transformada *Wavelet* 2D necessita de 4 funções de 2 variáveis, considerando-se a propriedade de separabilidade de variáveis, dadas a seguir (GONZALEZ; WOODS, 2002):

$$\phi(x, y) = \psi(x)\psi(y). \quad (2.20)$$

$$\begin{cases} \psi_H(x, y) = \phi(x)\psi(y) \\ \psi_V(x, y) = \psi(x)\phi(y) \\ \psi_D(x, y) = \phi(x)\phi(y) \end{cases} \quad (2.21)$$

Em que a equação 2.20 representa a propriedade de reescala e as demais (Equação 2.21) têm a característica de extrair informações relativas a detalhes da

imagem original, em que  $\psi_H(x, y)$  fornece detalhes horizontais,  $\psi_V(x, y)$  fornece detalhes verticais e  $\psi_D(x, y)$  fornece detalhes diagonais, nas Figuras 2.14 e 2.15 são ilustradas a imagem original e os resultados da aplicação da Transformada *Wavelet*.



Figura 2.14: imagem original.

Existem diversas funções que satisfazem as restrições necessárias para serem *Wavelets*. Dentre elas destacam-se: Haar e Daubechies, sendo estas formadas por uma família de funções.

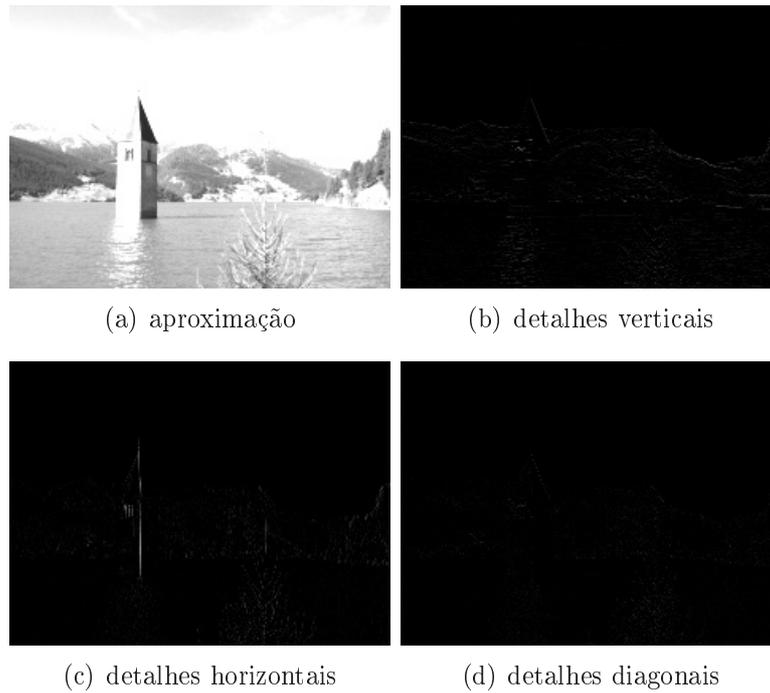
#### 2.4.4 Transformada *Wavelet* Haar

A Transformada *Wavelet* Haar foi a primeira a ser estudada e é a mais comum em aplicações, principalmente envolvendo imagens. As funções que compõem a Transformada *Wavelet* Haar são descritas nas equações a seguir:

$$\phi(x) = \begin{cases} 1 & p/0 < x < 1 \\ 0 & \text{caso contrário} \end{cases} \quad (2.22)$$

$$\psi(x) = \begin{cases} 1 & p/0 < x < 0.5 \\ -1 & p/0.5 < x < 1 \\ 0 & \text{caso contrário} \end{cases} \quad (2.23)$$

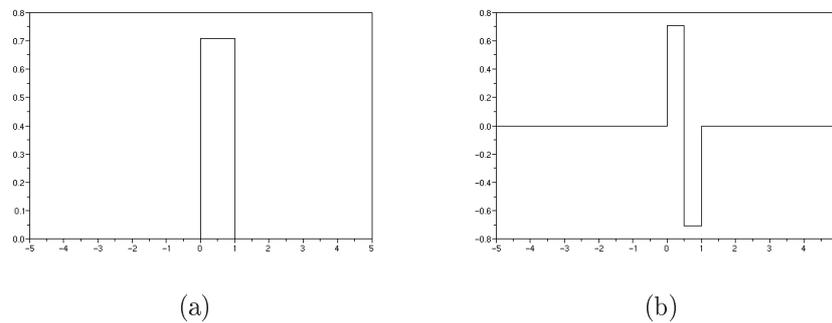
É observado nas equações apresentadas, que a re-escala do sinal de entrada



**Figura 2.15:** características extraídas da imagem na Figura 2.14 pelas *Wavelets*.

pela função  $\phi(x)$  (função de reescala) pode ser interpretada como a média do sinal no intervalo. A parte seguinte, proporcionada pela função  $\psi(x)$  (função mãe), é interpretada como a diferença entre a média da primeira metade do intervalo e a segunda metade do intervalo (LIMA, 2003).

Na Figura 2.16 são apresentas as funções de reescala e mãe da Transformada *Wavelet* Haar:



**Figura 2.16:** a) função de reescala e b) função mãe da Transformada *Wavelet* Haar.

### 2.4.5 Transformada *Wavelet* Daubechies

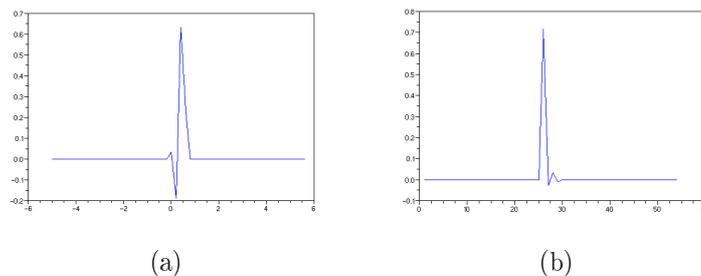
As funções que compõem as Transformadas *Wavelets* Daubechies foram estudadas por Ingrid Daubechies, sendo fortes disseminadoras do uso e estudo das *Wavelets*. As funções que determinam as Transformadas *Wavelets* Daubechies  $\phi$  e  $\psi$  possuem as seguintes características:

$$\int_{-\infty}^{\infty} \phi(x) dx = 1 \quad (2.24)$$

$$\int_{-\infty}^{\infty} x^l \phi(x) dx = 0, l \in [0, 1, \dots, N] \quad (2.25)$$

$$\int_{-\infty}^{\infty} x^l \psi(x) dx = 0, l \in [0, 1, \dots, N] \quad (2.26)$$

A Transformada *Wavelet* Haar corresponde à Transformada *Wavelet* Daubechies de ordem 1. As Figuras 2.17(a) à 2.19(b) apresentam, respectivamente, a função de reescala e a função mãe da Transformada *Wavelet* Daubechies 4 (Db4), 6 (Db6) e 8 (Db8).



**Figura 2.17:** a) função de reescala e b) função mãe da Transformada *Wavelet* Db4.

## 2.5 Pontos Críticos

Os pontos críticos são definidos no trabalho de (ALEXANDROV, 2004) como pontos da imagem que apresentam curvatura acentuada, assim como no trabalho de (JÚNIOR, 2007). Nesta seção, os pontos críticos são apresentados como pontos que apresentam características peculiaridades .

Na Figura 2.20 são mostrados os pontos críticos de três formas elementares presentes em imagens binárias em que estes foram localizados manualmente:

Nas situações apresentadas, os pontos são considerados críticos se representarem cantos (ponto com mudança brusca na curvatura) ou cruzamentos de retas.

### 2.5.1 Técnicas de Localização de Pontos Críticos

A aplicação de uma técnica de localização de pontos críticos proposto por Júnior (2007) segue os passos descritos na Figura 2.21.

O algoritmo de localização dos pontos críticos inicia o processo após o pré-processamento com a segmentação da imagem. A partir da imagem segmentada, o contorno parametrizado é calculado e a partir da função dos picos de curvatura são localizados os cantos (JÚNIOR, 2007).

Um outro método tradicional da literatura, proposto por Alexandrov (2004) realiza a detecção de pontos críticos de formas baseados na conectividade de pontos da imagem segmentada, que são utilizadas como base para o reconhecimento de caracteres (OCR - *Optical Character Recognition*).

O método da conectividade analisa a imagem segmentada através de uma janela de dimensão 3x3 procurando por um padrão pré-estabelecido.

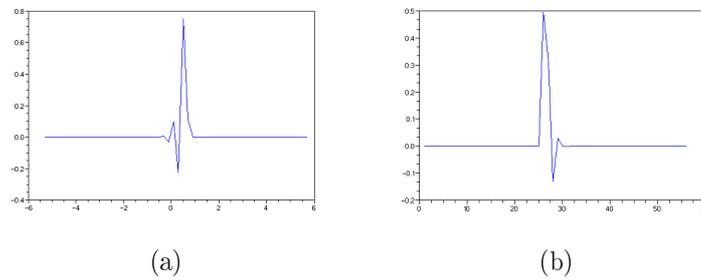
Nas Figuras 2.22(a) à 2.24(p) admite-se que o objeto segmentado corresponda aos pontos brancos e o fundo corresponda aos pontos pretos. Outro aspecto do texto original é que este lida apenas com os padrões apresentados nas Figuras 2.22 e 2.23.

Observa-se que a possibilidade de pontos é muito alta, tendo em vista que são apresentados apenas padrões em matrizes 3x3. Para ilustrar a técnica citada, são apresentadas nas Figuras 2.25(a) à 2.25(c), a imagens de objetos segmentados e nas Figuras 2.26(a) à 2.26(c) os pontos críticos encontrados.

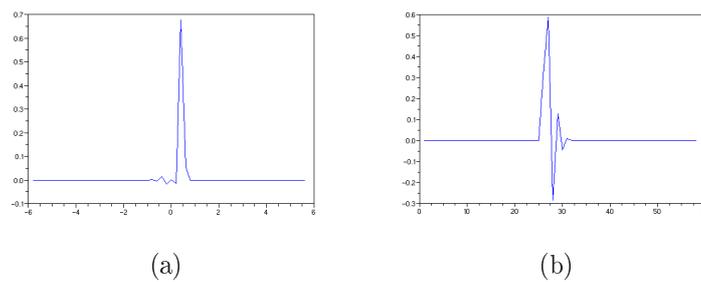
Nas Figuras 2.26(a) à 2.26(c) foram usados na busca os padrões relativos as Figuras 2.22(a) a 2.22(d) e 2.23(d) à 2.24(g).

A Figura 2.26(a) mostra que na Figura 2.25(a) foram encontrados os 4 pontos críticos. Contudo, A Figura 2.26(b) mostra que na Figura 2.25(b) não foi encontrado nenhum ponto crítico e na imagem da secção da caixa torácica foram encontrados muitos pontos.

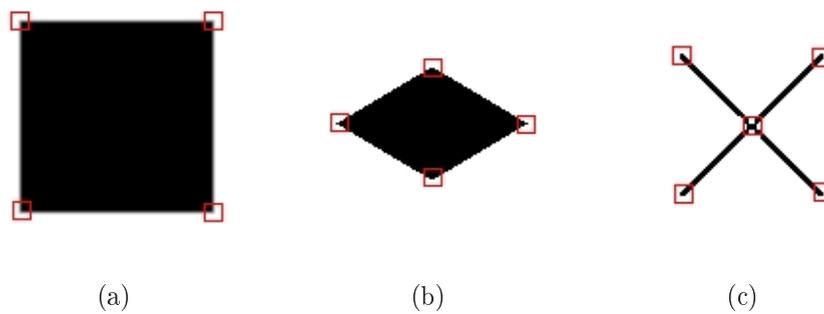
Este resultado mostra que este método necessita da previsão de todos os tipos de



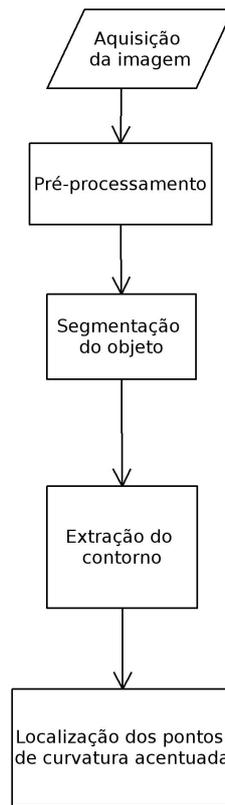
**Figura 2.18:** a) função de reescala e b) função mãe da Transformada *Wavelet* Db6.



**Figura 2.19:** a) função de reescala e b) função mãe da Transformada *Wavelet* Db8.



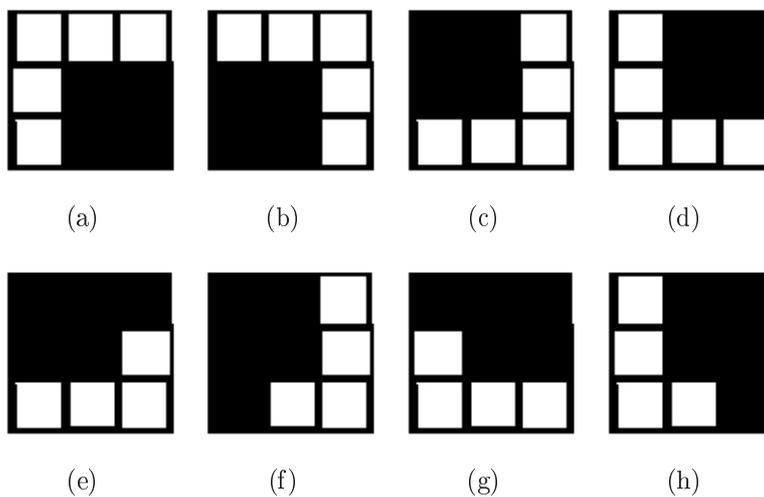
**Figura 2.20:** pontos críticos em a) um quadrado, b) losango e c) retas concorrentes.



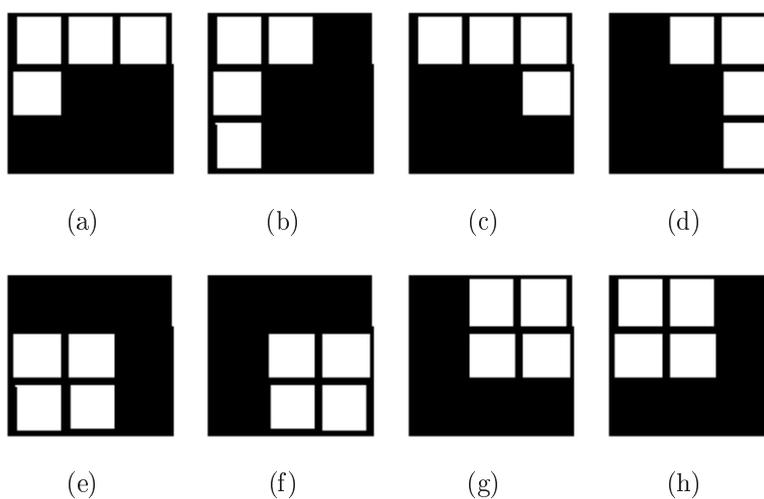
**Figura 2.21:** fluxograma para localização de pontos críticos

pontos críticos possíveis em uma imagem. A abordagem por conectividade tem como limitadores o conhecimento a priori dos pontos críticos e o custo computacional, visto que o algoritmo consiste na aplicação de convoluções entre a imagem original e os padrões procurados.

No uso de matrizes  $3 \times 3$ , seriam possíveis 512 combinações o que inviabiliza essa abordagem para localização de pontos críticos relativos a padrões cujas matrizes são de ordens superiores. O uso de matrizes de ordem baixa pode produzir falsos pontos críticos devido ao ruído presente na imagem (como apresentado na Figura 2.26(c)).



**Figura 2.22:** possíveis formas de conectividades.



**Figura 2.23:** possíveis formas de conectividades.

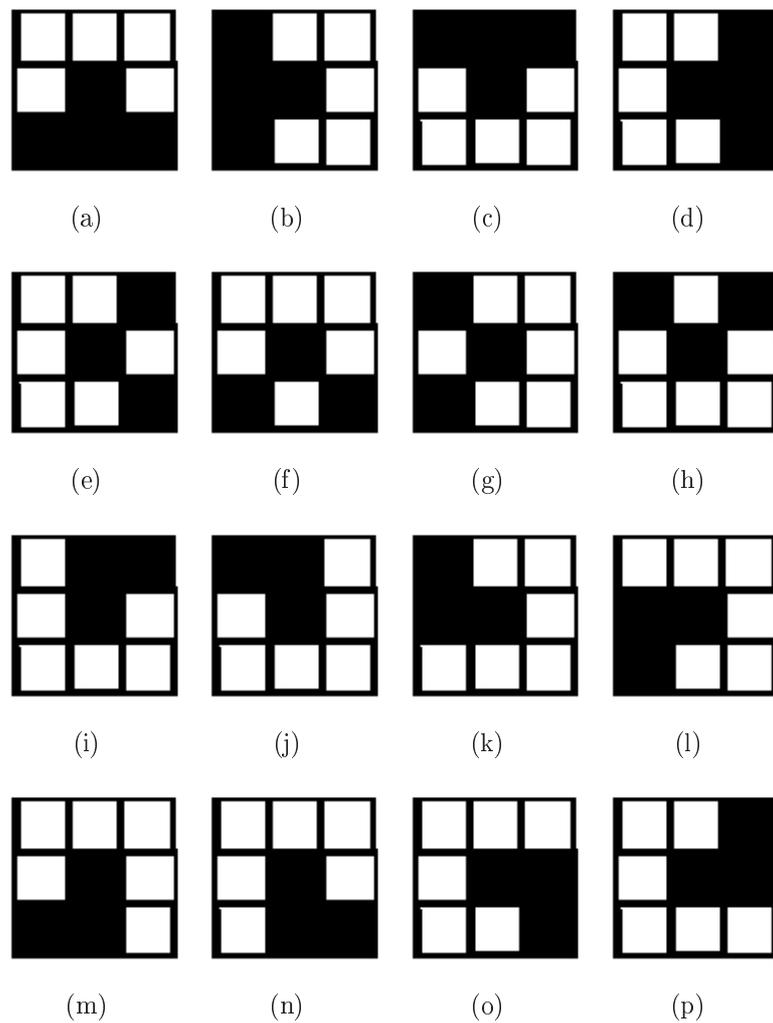


Figura 2.24: possíveis formas de conectividades.

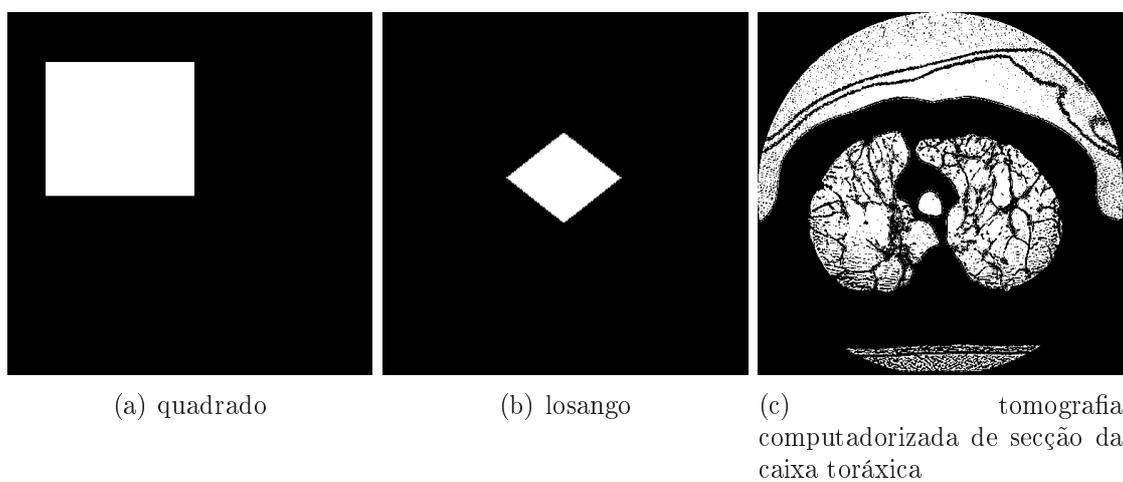
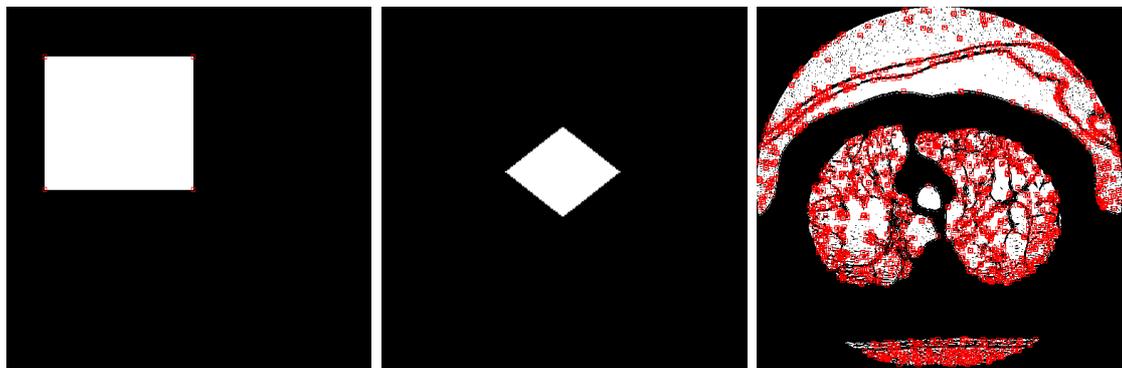


Figura 2.25: imagens usadas no algoritmo de conectividade.



(a) pontos localizados no quadrado  
(b) pontos localizados no losango  
(c) pontos localizados na tomografia computadorizada de secção da caixa torácica

**Figura 2.26:** pontos críticos localizados pelo algoritmo de conectividade.

# Capítulo 3

## Metodologia

Neste capítulo é descrito a metodologia proposta para realizar a localização de pontos críticos em imagens digitais. Inicialmente é descrito o ambiente de desenvolvimento, na Seção 3.1.1 são descritas as imagens para testes dos algoritmos desenvolvidos. Por fim, a Seção 3.2 descreve o algoritmo proposto para detecção de pontos críticos em imagens em níveis de cinza e colorido.

### 3.1 Ambiente de Desenvolvimento

---

A implementação e execução dos algoritmos e elaboração desta monografia foi realizada usando um *notebook* (Figura 3.1) Acer modelo *Aspire 3100* (<http://www.acer.com/>), o qual apresenta um processador AMD® *Sempron* 1.8GHz, 256KB L2 *Cache* (<http://www.amd.com/>) e 512MB de memória RAM DDR2.

O sistema operacional utilizado foi o *GNU/Linux* (<http://www.linux.org/>), distribuição *Ubuntu 7.10 - The Gutsy Gibbon* (<http://www.ubuntu.org/>). As simulações computacionais foram feitas no *Scilab 4.0* (<http://www.scilab.org/>) utilizando as toolboxes *Scilab Imaging and Video Package - SVIP 0.4.3* (<http://sivp.sourceforge.net/>) e *Scilab Wavelets Toolbox - SWT 0.1.0* (<http://scwt.sourceforge.net/>). O SVIP foi utilizado para aquisição e processamento de imagens e o SWT foi utilizado para análises *Wavelets*.

Em alguns momentos foi necessário usar aplicativos de análise e processamento de imagens mais refinados, nestes casos foram usado o *GNU Image Manipulation Program - GIMP 2.2.13* (<http://www.gimp.org/>). Os fluxogramas foram



**Figura 3.1:** computador usado na elaboração da monografia.

desenvolvidos com o *Kivio 1.6.2* (<http://www.koffice.org/kivio/>).

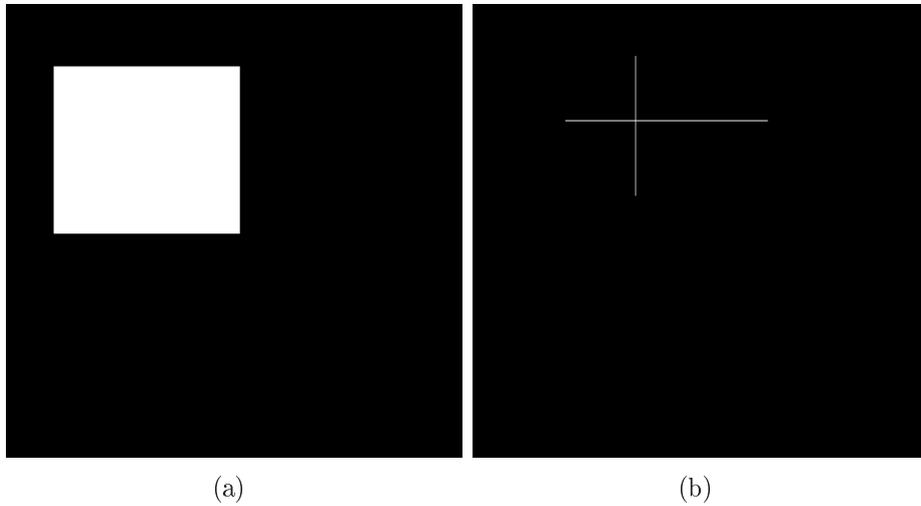
### 3.1.1 Imagens Utilizadas

Nesta seção são apresentadas as imagens digitais utilizadas durante o desenvolvimento desta monografia. As diferentes imagens são utilizadas para verificar o desempenho do algoritmo em diversas condições de complexidade. Foram escolhidas três tipos de imagens dependendo do seu nível de complexidade (simples, moderada ou complexa).

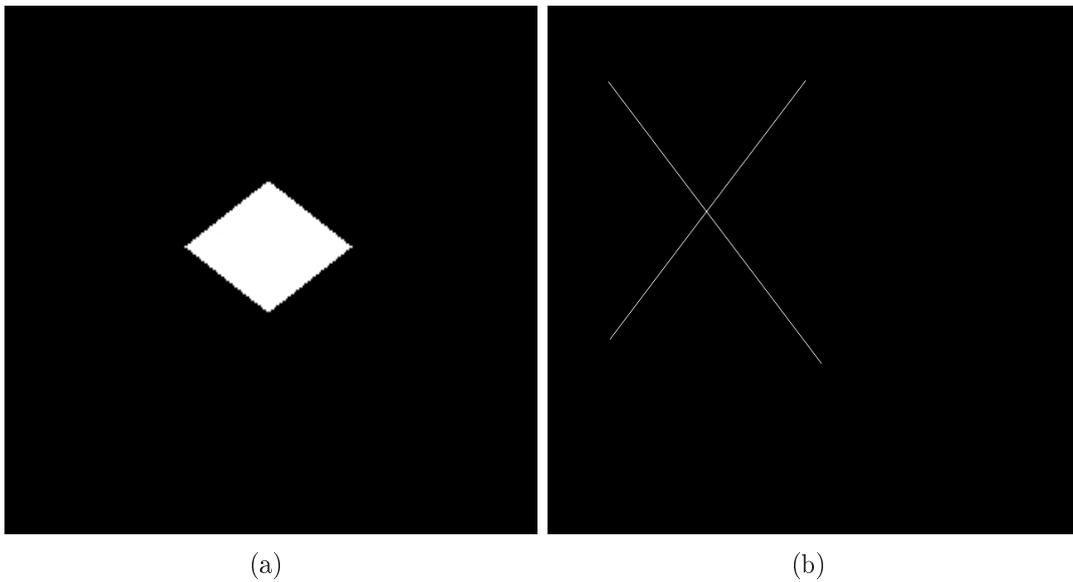
As imagens simples são ilustradas na Figura 3.2. Estas imagens foram produzidas sinteticamente e possuem baixa complexidade através do alto contraste entre os objetos, em que os cantos são formados pelo encontro de retas horizontais e verticais.

As imagens de complexidade moderada são imagens sintéticas mostradas na Figura 3.3. Estas figuras possuem alto contraste, em que os cantos são formados pelo encontro de retas diagonais.

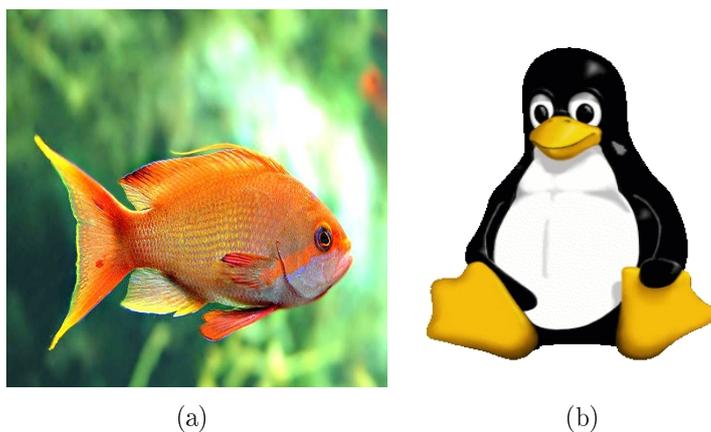
As imagens complexas são mostradas na Figura 3.4. Estas figuras são imagens reais, obtidas na *internet*, com excessão da radiografia do tórax que foi proveniente de um projeto de mestrado em processamento digital de imagens médicas. Sua complexidade encontra-se na coloração das imagens e, principalmente, nos contornos irregulares.



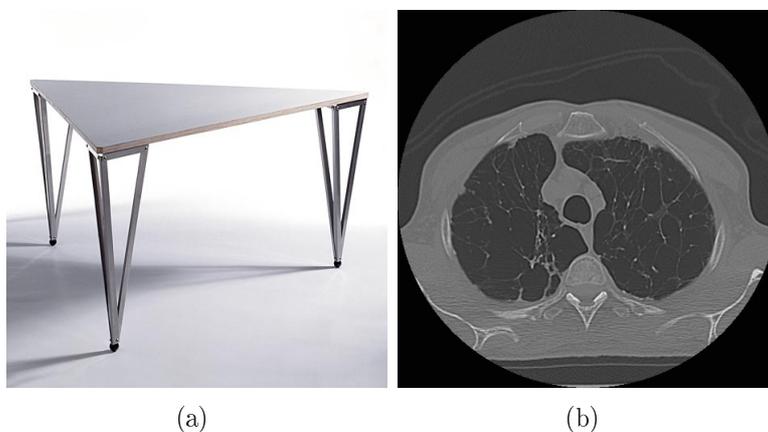
**Figura 3.2:** imagem de baixa complexidade a) quadrado e b) retas coordenadas.



**Figura 3.3:** imagem de complexidade moderada a) quadrado e b) retas coordenadas.



**Figura 3.4:** imagem de alta complexidade a) peixe e b) Tux, mascote do Linux.



**Figura 3.5:** imagem de alta complexidade a) mesa e b) tomografia computadorizada de secção do tórax.

## 3.2 Método Proposto para Detecção de Pontos Críticos

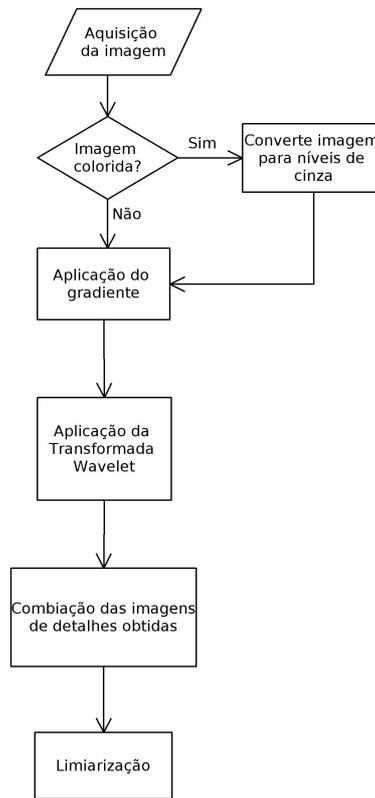
---

Neste trabalho é proposto o algoritmo ilustrado na Figura 3.6.

O algoritmo inicia através da aquisição da imagem. Em seguida, esta é convertida para níveis de cinza, caso esta seja originalmente colorida.

A partir da imagem em níveis de cinza é calculado o módulo do gradiente da imagem a partir das componentes do gradiente da mesma, em que uma das componentes possuem os detalhes verticais e a outra possui os detalhes horizontais da imagem original, conforme descrito na Seção 2.3.3.

Em seguida, a Transformada *Wavelet* é aplicada sobre o módulo do gradiente. A aplicação da Transformada *Wavelet* gera 4 (quatro) imagens, conforme descrito na Seção 2.4.3 e ilustrado nas Figuras 3.7(a) e 3.7(b).



**Figura 3.6:** esquema geral do processamento envolvido na localização de pontos críticos.

A partir dos coeficientes de detalhes (partes da direita e inferiores da Figura 3.7(b)) são combinadas da seguinte forma:

$$Im_{detalhes} = \sum_{i=1}^{n(C_{Im})} S_i(C_{Im})^{\frac{1}{i}} \quad (3.1)$$

em que:

- ▶  $n(C_{Im})$  é o número de elementos do conjunto  $C_{Im}$ ;
- ▶  $C_{Im} = \{Im_A, Im_B, Im_C, Im_D\}$ ;
- ▶  $S_i(C_{Im})^{\frac{1}{i}}$  denota a soma do produto interno elevado a  $\frac{1}{i}$  dos elementos de todos os sub-conjuntos de  $C_{Im}$  com  $i$  elementos;

A combinação das imagens tem como objetivo obter a maior quantidade possível de informação sobre os detalhes da imagem, ilustrada na Figura 3.8.

Descrição dos termos pertencentes a Equação 3.1:



**Figura 3.7:** a) imagem original e b) imagens geradas pela Transformada *Wavelet* Haar.

- ▶  $Im_A$  é a imagem relativa aos detalhes verticais;
- ▶  $Im_B$  é a imagem relativa aos detalhes horizontais;
- ▶  $Im_C$  é a imagem relativa aos detalhes diagonais;
- ▶  $Im_D$  é a imagem relativa aos detalhes diagonais obtida pela aplicação a Transformada *Wavelet* na imagem do módulo do gradiente rotacionada de 90 graus.



**Figura 3.8:** imagem de detalhes da imagem da Figura 3.7.

A partir da imagem de detalhes é calculado o seu histograma, apresentado na Figura 3.9.

Através de uma análise manual é feita a detecção dos pontos críticos da imagem aplicando uma limiarização da imagem de detalhes.

Observa-se que a obtenção dos resultados são independentes do nível de

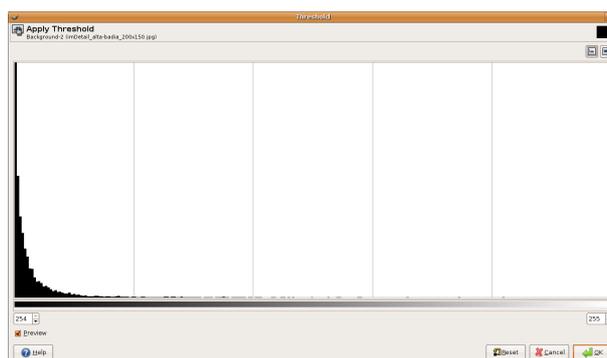


Figura 3.9: histograma da imagem de detalhes (Figura 3.8).

sensibilidade<sup>1</sup>, ou seja, o usuário da técnica proposta tem a liberdade de escolher o nível de sensibilidade do algoritmo.

A automatização na escolha do limiar é possível quando uma aplicação for especificada, pois assim é possível determinar o nível de sensibilidade que o algoritmo deve ter.

A seguir são listadas algumas sugestões de formas para escolha do nível de sensibilidade:

- ▶ Empírica, após a definição o valor do limiar é constante;
- ▶ Uso de técnicas estatísticas, tais como redes neurais ou algoritmos genéticos;
- ▶ Uso de funções que incorporem elementos de iluminação e ruído presentes.

---

<sup>1</sup>Nível de sensibilidade é tido como o valor escolhido como limiar para a imagem de detalhes, de modo que quanto menor o nível de sensibilidade mais peculiar o ponto precisa ser para o algoritmo tê-lo como crítico

# Capítulo 4

## Resultados

Neste capítulo são descritos os resultados obtidos em cada uma das etapas do método proposto descrito na Seção 3.2.

Inicialmente é realizada a comparação entre os resultados obtidos utilizando os dois detectores de bordas estudados (*Prewitt* e *Sobel*). Em seguida são mostrados as comparação da Transformada *Wavelet* utilizada para formação da imagem de detalhes.

Por fim, é realizado a comparação dos resultados obtidos na detecção de cantos.

### 4.1 Comparação entre Métodos de Gradiente

---

Nesta seção são apresentados os resultados relativos aos detalhes obtidos através da combinação dos coeficientes de detalhes de algumas imagens utilizando a Transformada *Wavelet* Haar.



(a) resultado obtido pelo gradiente de *Prewitt*      (b) resultado obtido pelo gradiente de *Sobel*

**Figura 4.1:** imagens de detalhes da Figura 3.2(a).

Para as imagens usadas nesta monografia, a combinação que mostrou-se mais



(a) resultado obtido pelo gradiente de *Prewitt*      (b) resultado obtido pelo gradiente de *Sobel*

**Figura 4.2:** imagens de detalhes da Figura 3.2(b).



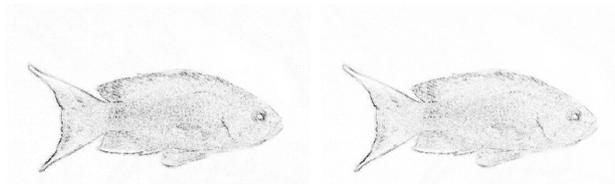
(a) resultado obtido pelo gradiente de *Prewitt*      (b) resultado obtido pelo gradiente de *Sobel*

**Figura 4.3:** imagens de detalhes da Figura 3.3(a).



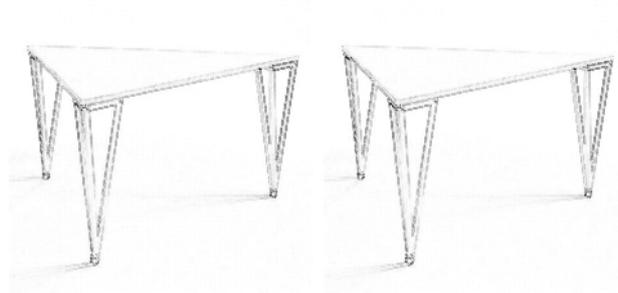
(a) resultado obtido pelo gradiente de *Prewitt*      (b) resultado obtido pelo gradiente de *Sobel*

**Figura 4.4:** imagens de detalhes da Figura 3.3(b).



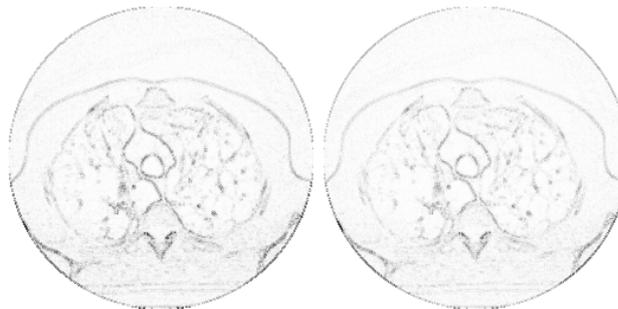
(a) resultado obtido pelo gradiente de *Prewitt*      (b) resultado obtido pelo gradiente de *Sobel*

**Figura 4.5:** imagens de detalhes da Figura 3.4(a).



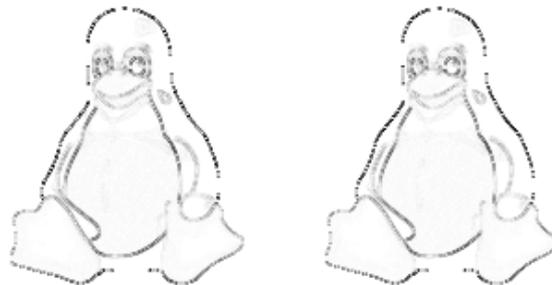
(a) resultado obtido pelo gradiente de *Prewitt*      (b) resultado obtido pelo gradiente de *Sobel*

**Figura 4.6:** imagens de detalhes da Figura 3.4(b).



(a) resultado obtido pelo gradiente de *Prewitt*      (b) resultado obtido pelo gradiente de *Sobel*

**Figura 4.7:** imagens de detalhes da Figura 3.5(a).



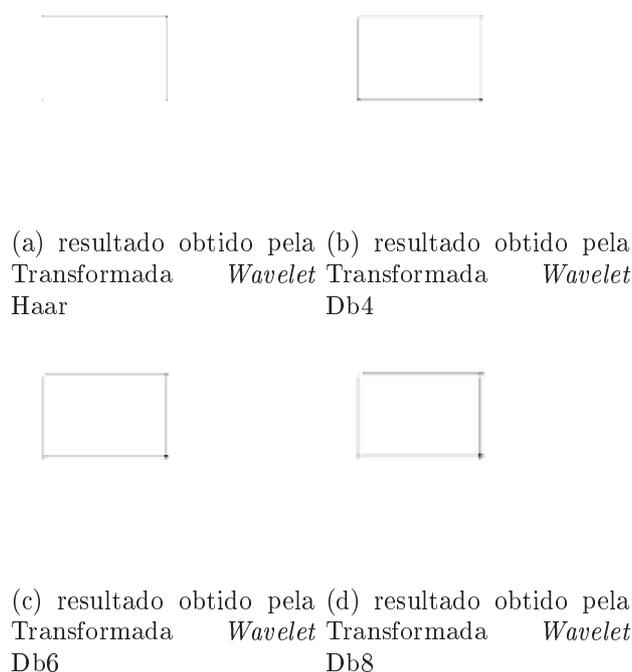
(a) resultado obtido pelo gradiente de *Prewitt*      (b) resultado obtido pelo gradiente de *Sobel*

**Figura 4.8:** imagens de detalhes da Figura 3.5(b).

eficiente é o gradiente de Sobel, isto porque, nas imagens sintéticas e binárias (principalmente) percebe-se que as bordas e cantos ficam mais evidenciados.

## 4.2 Comparação entre Transformadas *Wavelets*

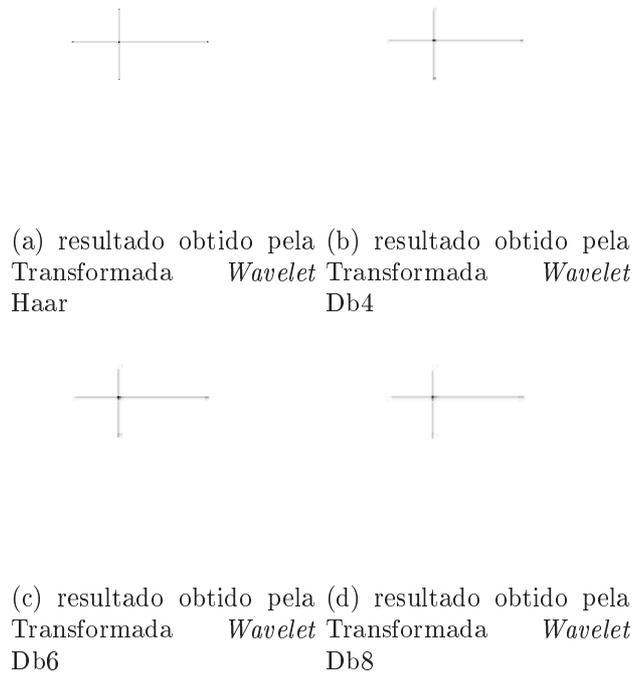
Devido ao bom desempenho do gradiente de *Sobel*, este será usado nesta seção para mostrar a comparação dos resultados obtidos para a formação da imagem de detalhes utilizando diferentes implementações da Transformada *Wavelet* (Haar, db4, db6 e db8). Os resultados são mostrados nas Figuras 4.9 à 4.16.



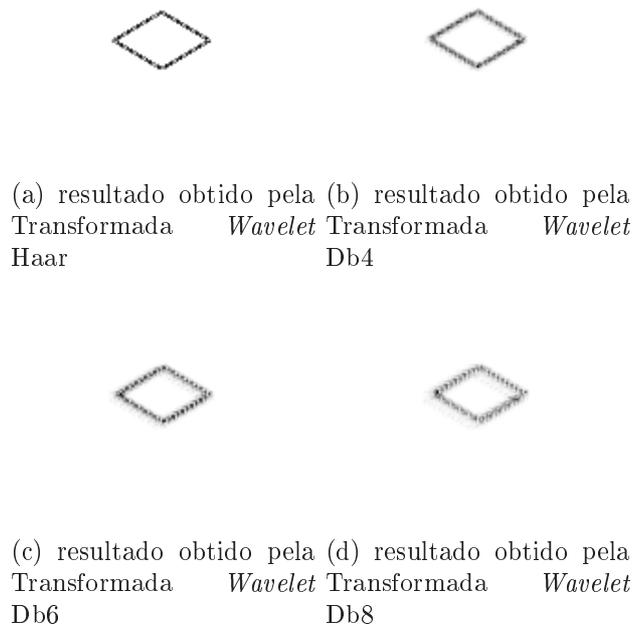
**Figura 4.9:** imagens de detalhes da Figura 3.2(a).

Observando-se as Figuras 4.9 a 4.16, é fácil perceber que a aplicação da Transformada *Wavelet* Haar mostrou-se mais sensível às bordas do que as demais, evitando borramento nas imagens obtidas. Outra observação é que quanto maior a ordem da Transformada *Wavelet* Daubechies, maior o borramento.

A Transformada *Wavelet* Haar aparenta ter maior uniformidade com relação aos pontos, possibilitando uma limiarização mais direta, enquanto que as demais Transformadas *Wavelet* Daubechies apresentam regiões de transição mais largas, conforme apresentado nas Figuras 2.16, 2.17, 2.18 e 2.19 onde foram abordadas as características das Transformadas *Wavelet*.



**Figura 4.10:** imagens de detalhes da Figura 3.2(b).



**Figura 4.11:** imagens de detalhes da Figura 3.3(a).



(a) resultado obtido pela Transformada Haar



(b) resultado obtido pela Transformada *Wavelet* Db4

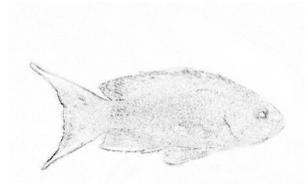


(c) resultado obtido pela Transformada Db6



(d) resultado obtido pela Transformada *Wavelet* Db8

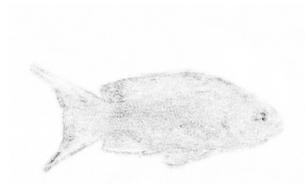
**Figura 4.12:** imagens de detalhes da Figura 3.3(b).



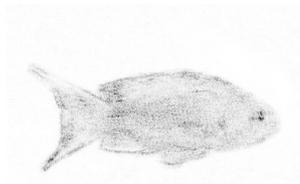
(a) resultado obtido pela Transformada Haar



(b) resultado obtido pela Transformada *Wavelet* Db4

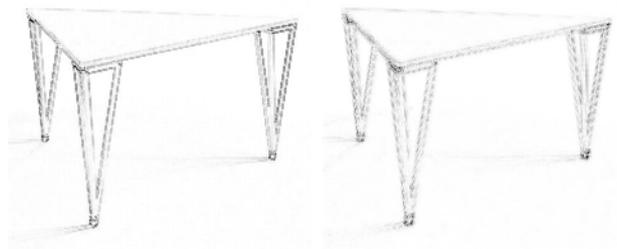


(c) resultado obtido pela Transformada Db6

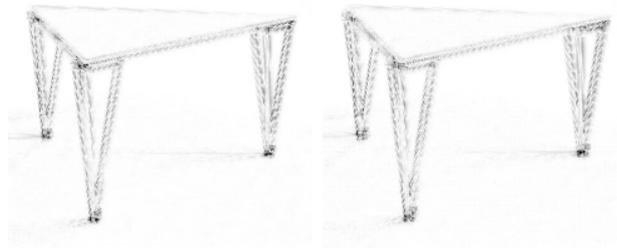


(d) resultado obtido pela Transformada *Wavelet* Db8

**Figura 4.13:** imagens de detalhes da Figura 3.4(a).

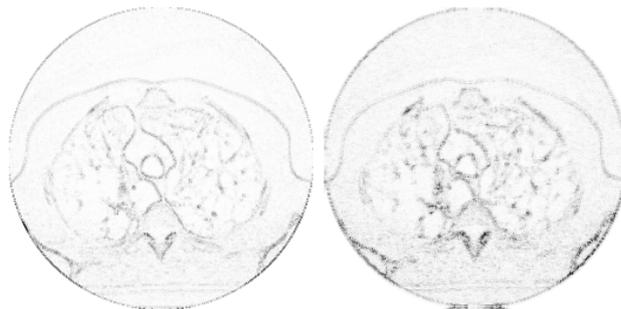


(a) resultado obtido pela Transformada *Wavelet* Haar (b) resultado obtido pela Transformada *Wavelet* Db4

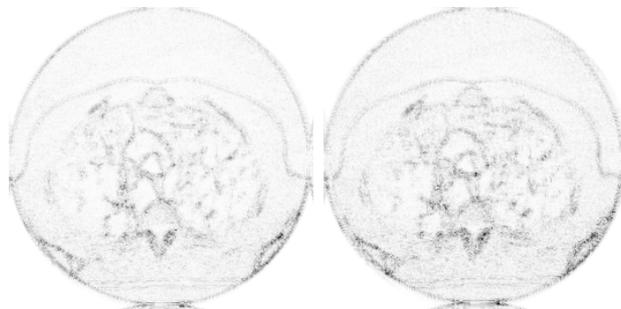


(c) resultado obtido pela Transformada *Wavelet* Db6 (d) resultado obtido pela Transformada *Wavelet* Db8

**Figura 4.14:** imagens de detalhes da Figura 3.4(b).

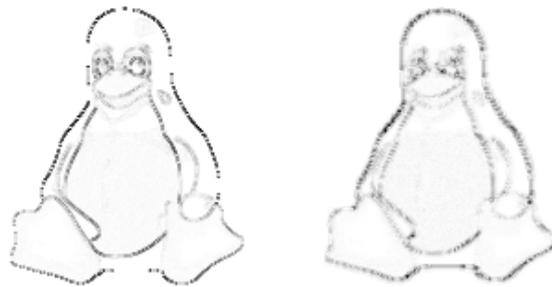


(a) resultado obtido pela Transformada *Wavelet* Haar (b) resultado obtido pela Transformada *Wavelet* Db4

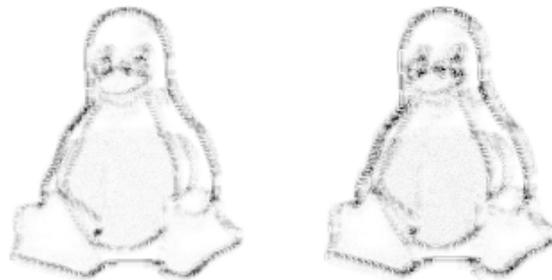


(c) resultado obtido pela Transformada *Wavelet* Db6 (d) resultado obtido pela Transformada *Wavelet* Db8

**Figura 4.15:** imagens de detalhes da Figura 3.5(a).



(a) resultado obtido pela Transformada *Wavelet* Haar (b) resultado obtido pela Transformada *Wavelet* Db4

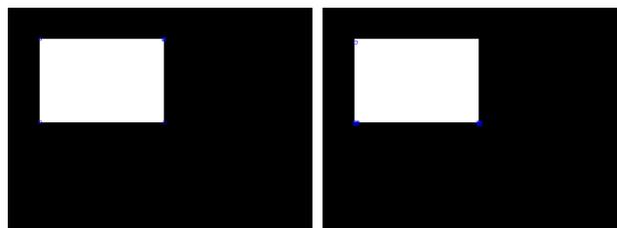


(c) resultado obtido pela Transformada *Wavelet* Db6 (d) resultado obtido pela Transformada *Wavelet* Db8

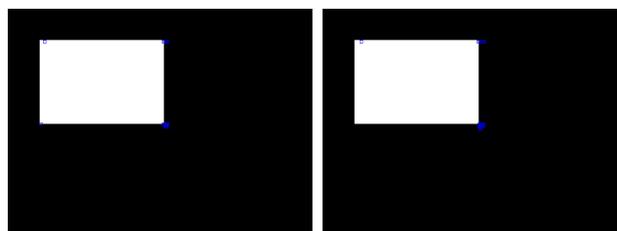
**Figura 4.16:** imagens de detalhes da Figura 3.5(b).

### 4.3 Localização de Pontos Críticos

Nesta seção são mostrados os diferentes tipos de pontos críticos localizados utilizando o método proposto (Figura 3.6).



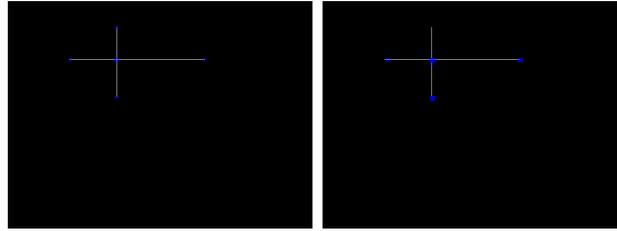
(a) pontos localizados com a Transformada *Wavelet Haar* (b) pontos localizados com a Transformada *Wavelet Db4*



(c) pontos localizados com a Transformada *Wavelet Db6* (d) pontos localizados com a Transformada *Wavelet Db8*

**Figura 4.17:** pontos críticos localizados na Figura 3.2(a).

Para as imagens usadas nesta monografia, a combinação que mostrou-se mais eficiente é o gradiente de Sobel seguido da Transformada *Wavelet Haar*, isto porque, nas imagens sintéticas e binárias (principalmente), é observado que os pontos críticos, que são facilmente percebidos visualmente, são localizados com melhor exatidão, atribui-se este resultado a descontinuidade da função mãe da Transformada *Wavelet Haar*, a qual não é percebida nas funções das Transformadas *Wavelet Daubechies*.

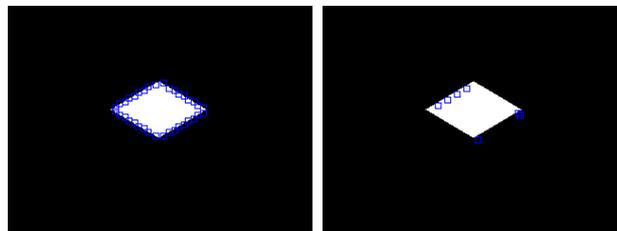


(a) pontos localizados com a Transformada *Wavelet Haar* (b) pontos localizados com a Transformada *Wavelet Db4*

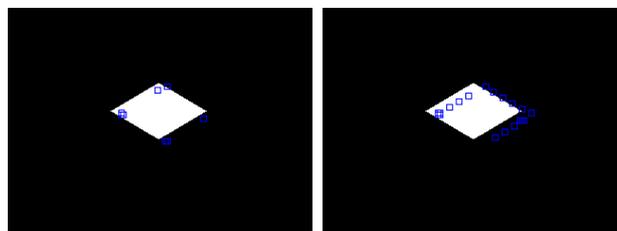


(c) pontos localizados com a Transformada *Wavelet Db6* (d) pontos localizados com a Transformada *Wavelet Db8*

**Figura 4.18:** pontos críticos localizados na Figura 3.2(b).

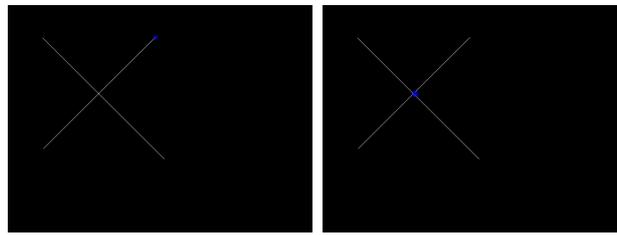


(a) pontos localizados com a Transformada *Wavelet Haar* (b) pontos localizados com a Transformada *Wavelet Db4*

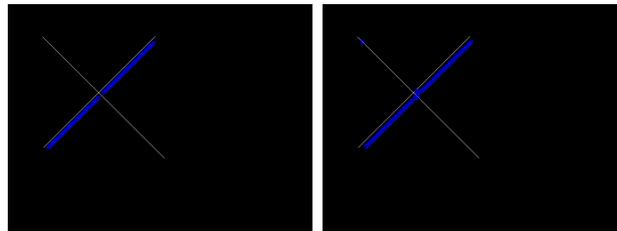


(c) pontos localizados com a Transformada *Wavelet Db6* (d) pontos localizados com a Transformada *Wavelet Db8*

**Figura 4.19:** pontos críticos localizados na Figura 3.3(a).



(a) pontos localizados com a Transformada *Wavelet Haar* (b) pontos localizados com a Transformada *Wavelet Db4*



(c) pontos localizados com a Transformada *Wavelet Db6* (d) pontos localizados com a Transformada *Wavelet Db8*

**Figura 4.20:** pontos críticos localizados na Figura 3.3(b).



(a) pontos localizados com a Transformada *Wavelet Haar* (b) pontos localizados com a Transformada *Wavelet Db4*



(c) pontos localizados com a Transformada *Wavelet Db6* (d) pontos localizados com a Transformada *Wavelet Db8*

**Figura 4.21:** pontos críticos localizados na Figura 3.4(a).

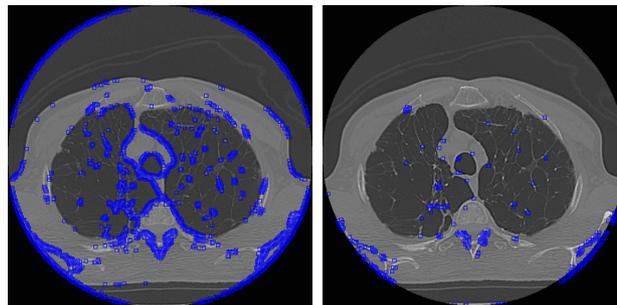


(a) pontos localizados com a Transformada *Wavelet Haar* (b) pontos localizados com a Transformada *Wavelet Db4*

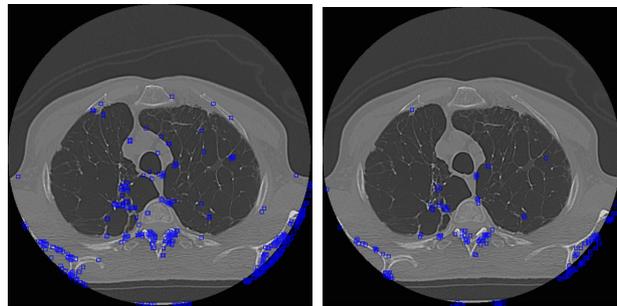


(c) pontos localizados com a Transformada *Wavelet Db6* (d) pontos localizados com a Transformada *Wavelet Db8*

**Figura 4.22:** pontos críticos localizados na Figura 3.4(b).

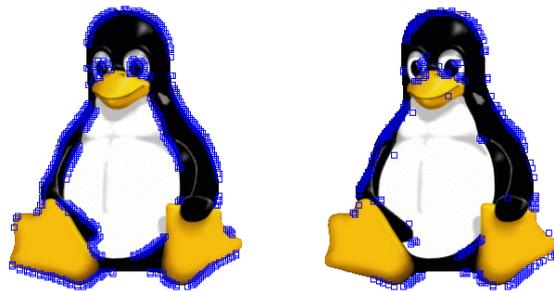


(a) pontos localizados com a Transformada *Wavelet Haar* (b) pontos localizados com a Transformada *Wavelet Db4*

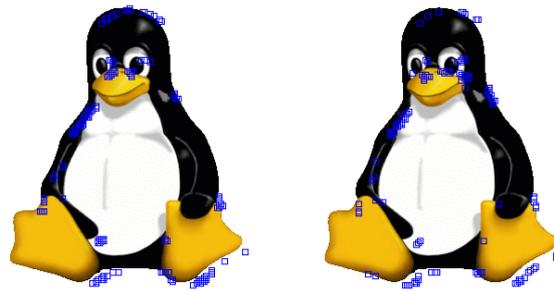


(c) pontos localizados com a Transformada *Wavelet Db6* (d) pontos localizados com a Transformada *Wavelet Db8*

**Figura 4.23:** pontos críticos localizados na Figura 3.5(a).



(a) pontos localizados com a Transformada *Wavelet Haar* (b) pontos localizados com a Transformada *Wavelet Db4*



(c) pontos localizados com a Transformada *Wavelet Db6* (d) pontos localizados com a Transformada *Wavelet Db8*

**Figura 4.24:** pontos críticos localizados na Figura 3.5(b).

## Conclusões e Perspectivas Futuras

Nesta monografia é apresentado um método de identificação e localização de pontos críticos em imagens digitais, através de um conjunto de algoritmos baseado no realce e extração dos detalhes presentes na imagem.

A metodologia proposta é eficiente pois não demanda custo computacional que possa comprometer uma aplicação em tempo real, sendo este tipo de sistema uma aplicação promissora para o método proposto.

Percebe-se que o algoritmo, conforme é concebido, não é de total eficácia; pois alguns detalhes são muito dependentes da estimativa do parâmetro de limiarização que é feito após a aplicação da Transformada *Wavelet* ou, como no caso de retas diagonais, ocorre confusão entre contorno com curvatura suave e pontos críticos.

### 5.1 Perspectivas Futuras

---

Seguem sugestões de atividades futuras, baseadas na presente monografia:

- ▶ realizar análises quantitativas de desempenho e eficácia do algoritmo;
- ▶ elaborar métodos de automatizar o nível de sensibilidade;
- ▶ aplicação conjunta de algoritmos de rastreamento para fins de monitoramento;
- ▶ uso de classificadores para reconhecimento de objetos presentes na imagem;
- ▶ uso de técnicas de reconhecimento de padrões para identificar características relativas aos pontos críticos dos objetos presentes na imagem;

- ▶ desenvolvimento de um algoritmo para classificar os pontos críticos, de modo que isto possa identificar características sutis de objetos;
- ▶ implementar os métodos em linguagens de médio e alto níveis (por exemplo C/C++, Java e Python).

# Referências Bibliográficas

ALEXANDROV, V. Using critical points in contours for segmentation os touching characters. In: *International Coference On Computer Systems*. [S.l.: s.n.], 2004.

APOSTOL, T. M. *Cálculo*. [S.l.: s.n.], 1998.

COSTA, R. C. S. *Inspeção Automática de Laranjas Destinadas Produção de Suco, Utilizando Técnicas de Processamento Digital de Imagens*. 77 p. Monografia (Graduação) — CEFET-CE, Fortaleza, set 2006.

DAUBECHIES, I. Orthonormal bases of compactly supported wavelets. *Comm. Pure Appl. Math.*, v. 41, n. 7, p. 909–996, mar 1988.

GOMES, J. P. *Reconhecimento Automático de Alvos em Imagens SAR*. Dissertação (mestrado) — Instituto Tecnológico da Aeronáutica, São José dos Campos, SP, Brasil, 2006.

GONZALEZ, R. C.; WOODS, R. E. *Digital Image Processing*. 2nd. ed. New Jersey: Prentice Hall, 2002. ISBN 0-201-18075-8.

JÚNIOR, I. C. de P. *Abordagem Wavelet para Detecção de Cantos em Formas*. Dissertação (mestrado) — Universidade Federal do Ceará, Fortaleza, CE, 2007.

LARSON, R. E. *et al. Cálculo y Geometría Analítica*. 6. ed. [S.l.]: Mc Graw Hill, 1998.

LEITHOLD, L. *Cálculo com Geometria Analítica*. 3. ed. [S.l.]: Harbra, 2002.

LIMA, P. C. de. Wavelets: Uma introdução. *ICEX - UFMG*, p. 25, 2003.

MALLAT, S. G. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 11, n. 7, p. 674–693, jul 1989.

MANOLAKIS, J. G. P. e D. G. *Digital Signal Processing*. [S.l.]: Prentice Hall, 2007.

MLSA, P. A.; RODRIGUES, J. J. *Handbook of Image & Video Processing*. [S.l.]: University of Texas at Austin, 2005.

MOTA, C. *Processamento Geométrico de Imagens*. Tese (doutorado) — Instituto Nacional de Matemática Pura e Aplicada, 1999.

VIDAKOVIC, B.; MUELLER, P. *Wavelets for Kids*. [S.l.], 2003.