

UNIVERSIDADE FEDERAL DO AMAZONAS
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM MATEMÁTICA

*DETECÇÃO DE ARESTAS COM O TENSOR DE
ESTRUTURA*

JOSÉ AMAURI SIQUEIRA DA SILVA

MANAUS

2009

UNIVERSIDADE FEDERAL DO AMAZONAS
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM MATEMÁTICA

JOSÉ AMAURI SIQUEIRA DA SILVA

*DETECÇÃO DE ARESTAS COM O TENSOR DE
ESTRUTURA*

Dissertação apresentada ao Programa de Pós-Graduação em Matemática da Universidade Federal do Amazonas como requisito parcial para obtenção do título de Mestre em Matemática, área de concentração em geometria Diferencial.

Orientador: Prof. Dr. Cícero Mota

MANAUS
2009

JOSÉ AMAURI SIQUEIRA DA SILVA

DETECÇÃO DE ARESTAS COM O TENSOR DE
ESTRUTURA

Dissertação apresentada ao Programa de Pós-Graduação em Matemática da Universidade Federal do Amazonas, como requisito parcial para obtenção do título de Mestre em Matemática, área de concentração em Geometria Diferencial.

Manaus, 20 de fevereiro de 2009.

BANCA EXAMINADORA

.....
Prof. Dr. Cícero Mota, Presidente
Universidade Federal do Amazonas.

.....
Prof Dr. Disney Douglas de Lima Oliveira,
Universidade Federal do Amazonas.

.....
Prof Dr. Romildo José da Silva,
Universidade Federal do Ceará.

AGRADECIMENTOS

A minha mãe o eterno apoio.

Agradeço aos professores a paciência e compreensão.

Aos meus amigos de mestrado, parceiros indispensáveis nessa conquista.

RESUMO

DETECÇÃO DE ARESTAS COM O TENSOR DE ESTRUTURA

Orientador: Prof. Dr. Cícero Mota.

Programa de Pós-Graduação em Matemática.

Ullrich Köthe propõe melhorias no cálculo do tensor de estrutura, que serão apresentadas neste trabalho. Primeira, a taxa de amostragem é alterada, pois o método padrão para cálculo desse tensor viola o teorema de amostragem de Shannon-Whittaker; segunda, usando um filtro não-linear para calcular as entradas do tensor de estrutura, a integração de informação oriunda de fora da aresta é minimizada.

ABSTRACT

THE EDGES DETECTION WITH STRUCTURE TENSOR

Ulrich Köthe proposes improvements in the calculation of the structure tensor, that will be presented in this work.

First, the tax of sampling is altered, therefore the standard method for calculation of that tensor violates the Shannon's sampling theorem; at second, using a non-linear filter to calculate the entrances of the structure tensor, the integration of information coming from out of the edge is minimized.

Sumário

1	Modelos Matemáticos Para Processamento de Imagens	1
1.1	Modelo Espacial de Sinais	1
1.1.1	Espaços de Sinais	1
1.2	Modelo Espectral de Sinais	2
1.2.1	Transformada de Fourier	2
1.2.2	Transformada de Fourier Discreta (DFT)	3
1.3	Filtros	4
1.3.1	Convolução	4
1.3.2	Convolução Bidimensional	4
1.3.3	Filtragem	5
1.3.4	Filtro Gaussiano	5
2	Derivadas	8
2.1	Derivadas Analógicas	8
2.1.1	Funções Diferenciáveis	9
2.1.2	Diferencial	10
2.1.3	O Vetor Gradiente	11
2.1.4	Transformadas de Fourier de Derivadas	11
2.1.5	Derivada da convolução	12
2.2	Derivadas Discretas	12
2.2.1	Derivada da Função Sinal	13
2.2.2	Gradiente Discreto	13
3	Teoria da Amostragem	15
3.1	Amostragem Pontual Uniforme.	15
3.2	Amostragem Pontual e Transformada de Fourier.	16
3.3	Teorema da Amostragem de Shannon-Whittaker.	19
3.4	Extensões do Teorema de Amostragem	22
3.5	Produto de Funções no Domínio Espacial e <i>Aliasing</i>	24

4	Detecção de Arestas	25
4.1	Imagem Digital	25
4.2	Definição de Arestas	26
4.3	Tensores	28
4.3.1	Tensor Q	28
4.3.2	Tensor de Estrutura	28
5	Mudanças Propostas	32
5.1	Amostragem Correta	32
5.2	Integração de Espaço	42
5.3	Filtro Hour-Glass	44
5.4	Detecção de Arestas com o Tensor de Estrutura	45
5.5	Resultados	46
5.6	Resultados	49

INTRODUÇÃO

O tensor de estrutura é uma ferramenta muito utilizada em processamento de imagens. Após os trabalhos de Förstner [6], Harris e Stevens [7], o tensor ganhou popularidade na detecção de cantos, na detecção de arestas [5], na análise de textura e no cálculo do fluxo óptico. Entretanto, o tensor de estrutura apresenta alguns problemas: primeiro, o método padrão de cálculo do tensor viola o teorema da amostragem de Shannon, portanto características pequenas podem ser perdidas devido ao aparecimento de *aliasing*; segundo, o cálculo das entradas deste tensor é feito usando uma integração espacial isotrópica, logo características próximas se fundem entre si; terceiro, estimativas de cantos derivadas do tensor de estrutura têm uma baixa precisão de localização.

Em uma imagem, as arestas e os cantos/junções são detectados de forma independente. As tentativas de derivar arestas e vértices do tensor de estrutura não foram bem sucedidas [5,7]. As dificuldades são causadas em parte pelos problemas acima citados.

Neste trabalho são apresentadas duas melhorias no tensor de estrutura propostas por Ullrich Köthe. A taxa de amostragem será aumentada para evitar o aparecimento de *aliasing*; um filtro não linear baseado na média espacial será utilizado para melhorar a localização das arestas e para impedir que características próximas se fundam.

O trabalho está dividido em seis capítulos: no primeiro capítulo apresentamos a teoria básica sobre processamento de imagens; no segundo capítulo, apresentamos os conceitos de derivadas; no terceiro, a teoria sobre amostragem; no quarto, vamos definir e detectar arestas em imagens digitais; no quinto, as mudanças propostas são apresentadas; e no sexto capítulo, os resultados obtidos são apresentados.

Capítulo 1

Modelos Matemáticos Para Processamento de Imagens

Para trabalharmos na detecção de arestas em imagem digital, objeto de estudo desta dissertação, precisamos conhecer a teoria matemática de processamento de imagens, que inclui técnicas de flitragen nos domínios espacial e da frequência. Neste capítulo apresentaremos as definições e resultados dessa teoria, necessários ao desenvolvimento do trabalho e fixaremos a notação a ser utilizada nos capítulos posteriores.

1.1 Modelo Espacial de Sinais

No modelo espacial de sinais, um sinal é representado por uma função $f : U \subset \mathbb{R}^m \rightarrow \mathbb{R}^n$, que mede a intensidade do sinal no domínio do espaço U . No caso do sinal imagem $U \subset \mathbb{R}^2$ e a função f associa a cada ponto p um vetor $f(p)$ que fornece a informação do nível de cinza em p .

1.1.1 Espaços de Sinais

Segundo [1] podemos representar o espaço de sinais por:

Definição 1.1.1. *Seja $f : U \subset \mathbb{R}^m \rightarrow \mathbb{R}^n$, U, m, n fixos, chamaremos de espaço de sinais a um subespaço $\{f : U \subset \mathbb{R}^m \rightarrow \mathbb{R}^n\}$ do espaço de funções, o sinal será representado pela função f . Um espaço de sinais é, portanto, um espaço vetorial onde valem:*

$$\begin{aligned}(f + g)(t) &= f(t) + g(t); \\ (\lambda f)(t) &= \lambda f(t), \quad \lambda \in \mathbb{R}.\end{aligned}$$

Seja um sinal representado por f vamos usar a seguinte notação para a função sinal.

função	domínio	contradomínio
analógica	contínuo	contínuo
com domínio amostrado	discreto	contínuo
com imagem quantizada	contínuo	discreto
sinal digital	discreto	discreto

1.2 Modelo Espectral de Sinais

Função que associa a cada frequência presente no sinal a amplitude e o ângulo de fase correspondentes. Dos métodos existentes vamos utilizar a transformada de Fourier para realizar essa medida.

1.2.1 Transformada de Fourier

A Transformada de Fourier (FT) é uma ferramenta muito empregada em processamento de imagens. Denominada assim em homenagem a Jean Baptiste Joseph Fourier (1768-1830), a FT decompõe um sinal em seus componentes elementares seno e cosseno. A FT aplicada a uma imagem no domínio espacial gera uma informação no domínio da frequência, em que cada ponto, definido por um vetor do tipo $(a.cosseno, b.seno)$, representa uma dada frequência contida no domínio espacial da imagem.

A Transformada de Fourier representa a soma de uma série de formas de ondas senoidais com diferentes amplitudes, fase e frequência. Ela é utilizada em processamento digital de imagens quando queremos conhecer frequências espaciais de um determinado padrão.

Definição 1.2.1. *Seja $f : R \rightarrow R$, definimos a Transformada de Fourier por:*

$$f \rightarrow \hat{f}(s) = \int_{-\infty}^{+\infty} f(t)e^{-2\pi its} dt = F(t).$$

Definição 1.2.2. *A Transformada Inversa de Fourier F^{-1} é dada por:*

$$f(t) = F^{-1}\hat{f}(s) = \int_{-\infty}^{+\infty} \hat{f}(s)e^{2\pi its} ds.$$

1.2.2 Transformada de Fourier Discreta (DFT)

A Transformada Discreta de Fourier (DFT) Bidimensional é de grande aplicabilidade em processamento digital de imagens (sinais bidimensionais), pois, muitas vezes, é conveniente a mudança do domínio espacial (x, y) para o domínio da frequência facilitando, assim, o seu processamento.

Definição 1.2.3. *Seja $f_0, f_1, \dots, f_{N-1}$ uma sequência, definimos $\hat{f}$ Transformada Discreta de Fourier ou DTF dessa sequência por:*

$$\hat{f}(j) = \frac{1}{N} \sum_{k=0}^{N-1} f(k) e^{-\frac{2\pi i k j}{N}}, j = 0, 1, \dots, N-1;$$

e a Transformada Inversa por:

$$f(k) = \sum_{j=0}^{N-1} \hat{f}(j) e^{\frac{2\pi i k j}{N}}, k = 0, 1, \dots, N-1.$$

Para sinais bidimensionais a extensão da Transformada de Fourier é feita de forma imediata:

$$F(k_1, k_2) = \frac{1}{N_1 N_2} \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} f(n_1, n_2) e^{-\frac{2\pi i k_1 n_1}{N_1}} e^{-\frac{2\pi i k_2 n_2}{N_2}};$$

e a Transformada Inversa é dada por:

$$f(n_1, n_2) = \sum_{k_1=0}^{N_1-1} \sum_{k_2=0}^{N_2-1} F(k_1, k_2) e^{\frac{2\pi i k_1 n_1}{N_1}} e^{\frac{2\pi i k_2 n_2}{N_2}}.$$

Um ponto importante da Transformada é o critério de Nyquist, que especifica que um sinal precisa ser amostrado pelo menos duas vezes em cada ciclo de variação, isto é, a frequência de amostragem (frequência de Nyquist) precisa ser no mínimo o dobro da maior frequência presente no sinal. Se este critério não for levado em consideração, os sinais de mais alta frequência serão erroneamente computados como de baixa frequência, fenômeno chamado de *ALIASING*.

1.3 Filtros

1.3.1 Convolução

Definição 1.3.1. *Seja f e g funções do espaço de sinais, definimos a operação convolução $f * g$ por:*

$$(f * g)(u) = \int_{-\infty}^{+\infty} f(t)g(u - t)dt.$$

Definição 1.3.2. *A convolução discreta para sinais digitais é dada por:*

$$(f * g)(u) = \sum_{t=-\infty}^{+\infty} f(t)g(u - t).$$

1.3.2 Convolução Bidimensional

Definição 1.3.3. *A convolução da imagem $f(k, j)$ com $g(j, k)$ é a imagem $f * g$ definida no pixel de coordenada (n, m) , pondo:*

$$(f * g)(n, m) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(k, j)g(n - k, n - j)dkdj.$$

A convolução é usada em processamento de imagem para operar diretamente no domínio espacial [1]. Consiste no realce de altas, médias ou baixas frequências de variação tonal da imagem, em detrimento das demais, alterando o valor de cada pixel da imagem com base no cálculo da média entre o pixel em questão e seus vizinhos, ponderada por valores pré-estabelecidos, dispostos em uma matriz, denominada máscara, *kernel* ou núcleo.

Vamos convencionar que dada uma imagem digital em sua representação matricial $f(i, j)$ o pixel de coordenadas $(0, 0)$ corresponde ao elemento do lado esquerdo superior da matriz.

Definição 1.3.4. *A convolução discreta de $f(k, j)$ com o núcleo $g(j, k)$ é dado por:*

$$(f * g)(n, m) = \sum_{k=-\infty}^{+\infty} \sum_{j=-\infty}^{+\infty} f(k, j)g(n - k, n - j).$$

Teorema 1.3.1. [Teorema da Convolução]. *Seja f e g sinais, então a Transformada de Fourier de f convolvida com g no domínio espacial é igual ao produto dos seus espectros no domínio da frequência, e a Transformada do produto de f por g no domínio espacial é igual a convolução no domínio da frequência.*

$$\begin{aligned} F(f(x) * g(x)) &= F(u).G(u); \\ F(f(x).g(x)) &= F(u) * G(u). \end{aligned}$$

A demonstração de Teorema da Convolução pode ser vista em [12].

1.3.3 Filtragem

A convolução de um sinal f por uma função g recebe comumente o nome de filtragem, em razão do teorema 1.3.1, visto que ela corresponde a uma modulação das frequências de f por g . Neste caso g é chamado de filtro e $f * g$ é chamado de imagem filtrada.

1.3.4 Filtro Gaussiano

A função gaussiana bidimensional tem grande aplicabilidade em processamento de imagem, sendo usada por exemplo para suavizar imagens (filtro passa baixa) e definir um espaço de escala.

Vamos enunciar algumas de suas propriedades sem demonstração. Para maiores detalhes consulte [12].

A gaussiana com desvio σ é dada por:

$$g_{\sigma}(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}.$$

A gaussiana possui várias propriedades importantes:

NORMALIZAÇÃO: A gaussiana é normalizada, isto é,

$$\int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} g_{\sigma}(x, y) dx dy = 1.$$

PROPRIEDADE DE SEMIGRUPO: A convolução de duas gaussianas de variâncias σ_1 e σ_2 é outra gaussiana de variância $\sigma_1 + \sigma_2$, isto é,

$$g_{\sigma_1+\sigma_2}(x, y) = g_{\sigma_1} * g_{\sigma_2}(x, y).$$

Logo um filtro gaussiano de variância $\sigma_1 + \sigma_2$ é equivalente a um filtro gaussiano de variância σ_1 em série com outro filtro gaussiano de variância σ_2 .

TRANSFORMADA DE FOURIER: A Transformada de Fourier de uma gaussiana bidimensional é outra gaussiana, ou seja, dado $g_\sigma(x, y)$ sua transformada é:

$$\hat{G}_\sigma(u, v) = e^{-2\pi^2(u^2+v^2)}.$$

SEPARABILIDADE E SIMETRIA ROTACIONAL: A gaussiana bidimensional é rotacionalmente simétrica e tem a propriedade de ser separável ($g_\sigma(x, y) = g_\sigma(x)g_\sigma(y)$ onde $g_\sigma(x)$ é a gaussiana unidimensional). Essa propriedade é útil para cálculo da convolução.

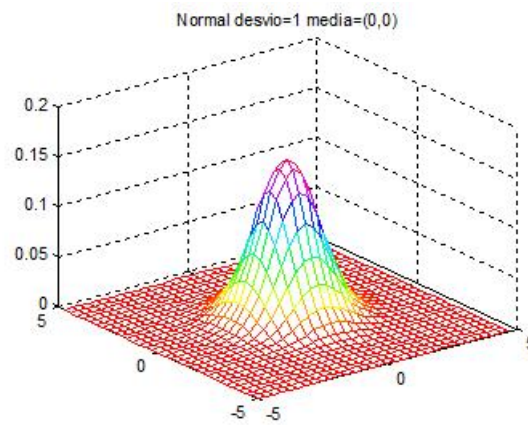


Figura 1.1: gaussiana: $g_{\sigma}(x, y)$.

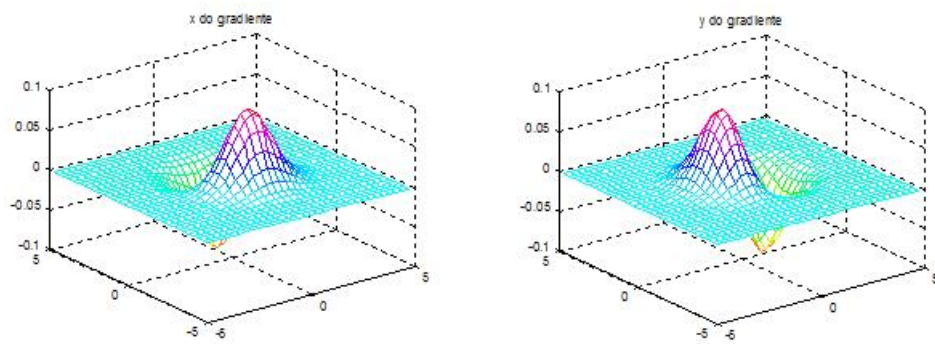


Figura 1.2: derivadas da gaussiana nas direções x : $g_{x,\sigma}(x, y)$
e y : $g_{y,\sigma}(x, y)$.

Capítulo 2

Derivadas

Neste capítulo vamos definir derivadas e gradiente de funções analógicas, mostrando a relação entre a transformada de Fourier e sua derivada.

2.1 Derivadas Analógicas

Definição 2.1.1. *Seja $f : U \rightarrow \mathbb{R}$ uma função real, definida num subconjunto aberto $U \subset \mathbb{R}^n$. Dado o ponto $a \in U$, a i -ésima derivada parcial de f no ponto a (onde $1 \leq i \leq n$) é o limite:*

$$\frac{\partial f}{\partial x_i}(a) = \lim_{t \rightarrow 0} \frac{f(a + te_i) - f(a)}{t},$$

quando tal limite existe.

Quando $U \subset \mathbb{R}^2$ é o que se chama uma "função analógica de duas variáveis reais" e escreve-se $f(x, y)$ para indicar seu valor no ponto $z = (x, y)$. Desta forma, as derivadas parciais de f num ponto $c = (a, b) \in U$ podem ser representadas por: $\frac{\partial f}{\partial x}(c)$ e $\frac{\partial f}{\partial y}(c)$ e aí teremos:

$$\frac{\partial f}{\partial x}(c) = \lim_{t \rightarrow 0} \frac{f(a + t, b) - f(a, b)}{t};$$

$$\frac{\partial f}{\partial y}(c) = \lim_{t \rightarrow 0} \frac{f(a, b + t) - f(a, b)}{t}.$$

O cálculo prático da i -ésima derivada parcial de uma função $f(x_1, \dots, x_n)$ se faz considerando todas as variáveis como se fossem constantes, exceto a i -ésima, e aplicando as regras usuais de derivação relativa a essa variável.

As derivadas parciais fornecem informações ao longo das retas paralelas aos eixos. Vamos estender este conceito de derivada a outras direções:

Definição 2.1.2. *Seja $f : U \rightarrow \mathbb{R}$ definida no aberto $U \subset \mathbb{R}^n$ a derivada direcional de f no ponto a , segundo o vetor v , é dada por:*

$$\frac{\partial f}{\partial v}(a) = \lim_{t \rightarrow 0} \frac{f(a + tv) - f(a)}{t},$$

quando tal limite existe.

As derivadas parciais tornam-se casos particulares das derivadas direcionais:

$$\frac{\partial f}{\partial x_i}(a) = \frac{\partial f}{\partial e_i}(a),$$

derivada direcional de f segundo o vetor e_i .

A derivada direcional $\frac{\partial f}{\partial v}(a)$ é a derivada no ponto $t = 0$, da função composta $f \circ \lambda : (-\epsilon, \epsilon) \rightarrow \mathbb{R}$ onde $\lambda : (-\epsilon, \epsilon) \rightarrow \mathbb{R}^n$ é o caminho retilíneo, $\lambda(t) = a + tv$, para o qual se tem $\lambda(0) = a$ e $\lambda'(t) = v$ para todo t . Aqui, $\epsilon > 0$ é escolhido tão pequeno que a imagem de λ esteja contida em U .

2.1.1 Funções Diferenciáveis

A noção de função diferenciável, que apresentaremos agora, é devida a Fréchet e Stolz. Ela constitui, para funções de n variáveis, a extensão adequada do conceito de funções derivável de uma só variável.

Definição 2.1.3. *Dada $f : U \rightarrow \mathbb{R}$, com $U \subset \mathbb{R}^n$, seja $a \in U$. Diremos que a função f é diferenciável no ponto a quando existirem constantes $A_1, \dots, A_n$ tais que, para todo vetor $v = (\alpha_1, \dots, \alpha_n) \in \mathbb{R}^n$, com $a + v \in U$, se tenha:*

$$f(a + v) = f(a) + A_1 \alpha_1 + \dots + A_n \alpha_n + r(v),$$

onde:

$$\lim_{v \rightarrow 0} \frac{r(v)}{|v|} = 0.$$

Quando f é diferenciável em todos os pontos de U , dizemos simplesmente que f é diferenciável. Se f é diferenciável no ponto a então, tomando $v = te_i$, temos: $\alpha_j = 0$ se $j \neq i$ e $\alpha_i = t$. Logo:

$$\frac{f(a + te_i) - f(a)}{t} = A_i + \frac{r(te_i)}{t} = A_i \pm \frac{r(te_i)}{|te_i|}.$$

Fazendo $t \rightarrow 0$, vemos que existe cada derivada parcial no ponto a , sendo $\frac{\partial f}{\partial x_i}(a) = A_i$. A definição abaixo é, portanto, equivalente à anterior.

Definição 2.1.4. *Diremos que a função $f : U \rightarrow R$ é diferenciável no ponto $a \in U$ quando existirem as derivadas parciais $\frac{\partial f}{\partial x_i}(a)$, ..., $\frac{\partial f}{\partial x_n}(a)$ e, além disso, para todo vetor $v = (\alpha_1, \dots, \alpha_n)$ tal que $a + v$, tivermos:*

$$f(a + v) = f(a) + \frac{\partial f}{\partial x_i}(a) \cdot \alpha_1 + \dots + \frac{\partial f}{\partial x_n}(a) \cdot \alpha_n + r(v),$$

onde

$$\lim_{v \rightarrow 0} \frac{r(v)}{|v|} = 0.$$

2.1.2 Diferencial

A diferencial de um caminho $f : R \rightarrow R^n$ é um vetor. Na situação dual, o papel de derivada de uma função $f : R^n \rightarrow R$ é desempenhado por um funcional linear, conforme mostraremos agora.

Seja $f : U \rightarrow R$ definido no aberto $U \subset R^n$, diferenciável no ponto $a \in U$. A diferencial de f no ponto a é o funcional linear $df(a) : R^n \rightarrow R$ cujo valor no vetor $v = (\alpha_1, \dots, \alpha_n)$ é dado por:

$$df(a) \cdot v = \frac{\partial f}{\partial v}(a) = \sum_{i=1}^n \frac{\partial f}{\partial x_i}(a) \cdot \alpha_i.$$

Como toda transformação linear $R^n \rightarrow R$, o funcional linear $df(a)$ possui uma matriz $1 \times n$ em relação à base canônica de R^n . Se identificarmos o funcional com sua matriz, teremos:

$$df(a) = \left(\frac{\partial f}{\partial x_1}(a), \dots, \frac{\partial f}{\partial x_n}(a) \right).$$

Quando $f : U \rightarrow R$ é diferenciável em todo ponto de U , obteremos uma aplicação $df : U \rightarrow (R^n)^* = \mathcal{L}(R^n; R)$, que associa a cada ponto $x \in U$ o funcional $df(x)$, cuja matriz é:

$$\left(\frac{\partial f}{\partial x_1}(x), \dots, \frac{\partial f}{\partial x_n}(x) \right).$$

A Aplicação df é contínua se, e somente se, cada uma das suas funções coordenadas $\frac{\partial f}{\partial x_i} : U \rightarrow R$ é contínua, isto é, se, e somente se, f é de classe C^1 .

2.1.3 O Vetor Gradiente

O produto interno natural induz a um isomorfismo entre R^n e seu dual R^{n*} [13]. Tal isomorfismo faz corresponder a cada vetor $v \in R^n$ o funcional $v^* \in (R^n)^*$ com $v^*(x) = \langle v, x \rangle$ para todo $x \in R^n$. Se $v = (\alpha_1, \dots, \alpha_n)$ então $v^*(e_1) = \alpha_1, \dots, v^*(e_n) = \alpha_n$, logo a matriz de v^* em relação à base canônica de R^n é $(\alpha_1, \dots, \alpha_n)$. A existência deste isomorfismo é responsável pelo fato de que no Cálculo Vetorial Clássico não ocorrem funcionais lineares. Em vez de um funcional, toma-se o produto interno $\langle v, x \rangle$ de um vetor fixo v por um vetor variável x .

Definição 2.1.5. Dada uma função $f : U \rightarrow R$, definida no aberto U do R^n definiremos o gradiente de f no ponto a como o vetor $\text{grad}f(a)$, que corresponde ao funcional $df(a)$ segundo o isomorfismo acima descrito. Isto significa, por definição, que:

$$\langle \text{grad}f(a), v \rangle = \frac{\partial f}{\partial v}(a) = df(a).v = \sum \frac{\partial f}{\partial x_i}(a)\alpha_i,$$

para todo $v = (\alpha_1, \dots, \alpha_n)$.

Destacaremos as três propriedades mais importantes do gradiente de uma função diferenciável f . Nessa discussão, fixaremos um ponto a e iremos supor que $\text{grad}f(a) \neq 0$. Então:

1º) O gradiente aponta para uma direção segundo a qual a função f é crescente;

2º) Dentre todas as direções ao longo das quais a função f cresce, a direção do gradiente é a de crescimento mais rápido;

3º) O gradiente de f no ponto a é perpendicular à superfície de nível de f que passa por esse ponto.

2.1.4 Transformadas de Fourier de Derivadas

Seja $f(t)$ a função sinal, sabemos que a Transformada Inversa de Fourier $F^{-1}\hat{f}(s) = f(t)$, então:

$$f(t) = F^{-1}\hat{f}(s) = \int_{-\infty}^{+\infty} \hat{f}(s)e^{2\pi its} ds.$$

Derivando $f(t)$:

$$\begin{aligned}
f'(t) &= \int_{-\infty}^{+\infty} \hat{f}(s) e^{2\pi i t s} \cdot 2\pi i s ds; \\
f'(t) &= 2\pi i s \int_{-\infty}^{+\infty} \hat{f}(s) e^{2\pi i t s} ds; \\
f'(t) &= 2\pi i s f(t);
\end{aligned}$$

portanto,

$$F(f'(t)) = 2\pi i s F(f(t)).$$

A transformada da derivada da função é igual a $2\pi i s$ vezes a transformada da própria função.

2.1.5 Derivada da convolução

Seja $f(t)$ a função sinal e $g(t)$ um filtro, então :

$$(f(t) * g(t))' = f'(t) * g(t) = f(t) * g'(t)$$

Esta propriedade deve-se ao fato de que as operações de derivação e integração podem ser realizadas através da convolução por funções específicas.

2.2 Derivadas Discretas

Definição 2.2.1. *Seja um espaço de escala gerado por g_σ , ou seja, $E = g_\sigma * f(x, y)$, podemos obter qualquer uma de suas derivadas parciais por:*

$$\frac{\partial E}{\partial x_i} = \frac{\partial (g_\sigma * f(x, y))}{\partial x_i} = \frac{\partial g_\sigma}{\partial x_i} * f(x, y) = g_\sigma * \frac{\partial f(x, y)}{\partial x_i} = f_{x_i}.$$

Observe que, mesmo que a função original não seja diferenciável, podemos calcular a derivada desde que o núcleo g_σ seja diferenciável.

Esta expressão mostra que $\frac{\partial E}{\partial x_i}$ também é um espaço de escala.

Só nos resta agora encontrar os operadores discretos para convolução com f , então analisando as séries de Taylor, temos que:

Dada uma função $g_\sigma(x)$ da qual se conhecem N amostras efetuadas em pontos equidistantes $x_1, x_2, \dots, x_n$, de modo que $x_i = x_{i-1} + h$ com $h > 0$. Representando a função $g_\sigma(x)$ nos pontos $x + h$ e $x - h$ através da fórmula de Taylor obtém-se:

$$g_\sigma(x + h) = g_\sigma(x) + hg'_\sigma(x) + \frac{1}{2}h^2g''_\sigma(x) + O(h^3);$$

$$g_\sigma(x - h) = g_\sigma(x) - hg'_\sigma(x) + \frac{1}{2}h^2g''_\sigma(x) + O(h^3);$$

Onde $O(h^3)$ representa o erro de truncamento. Subtraindo as equações apresentadas e isolando $g'_\sigma(x)$ obtém-se a equação para a primeira derivada de $g_\sigma(x)$:

$$g'_\sigma(x) = \frac{g_\sigma(x + h) - g_\sigma(x - h)}{2h}.$$

2.2.1 Derivada da Função Sinal

A derivada da função suavizada com uma gaussiana é a função convoluída com a derivada da gaussiana.

$$\frac{\partial(g_\sigma * f(x))}{\partial x} = \frac{\partial g_\sigma}{\partial x} * f(x).$$

2.2.2 Gradiente Discreto

Definição 2.2.2. *Seja g_σ uma gaussiana com desvio padrão σ e f a função sinal, as derivadas de f nas direções x e y são dadas por:*

$$f_x = g_{x,\sigma} * f;$$

$$f_y = g_{y,\sigma} * f.$$

Definição 2.2.3. Definimos o gradiente de uma função digital por:

$$\begin{aligned}\nabla f(x, y) &= f_x + f_y; \\ |\nabla f(x, y)| &= \sqrt{f_x^2 + f_y^2}; \\ \theta &= \arctan \frac{f_x}{f_y}.\end{aligned}$$

O gradiente desempenha papel importante em processamento de imagens, pois o módulo do gradiente retorna valores baixos em regiões homogêneas e valores maiores quando as intensidades variam mais bruscamente, indicando grande mudança nos níveis de cinza.

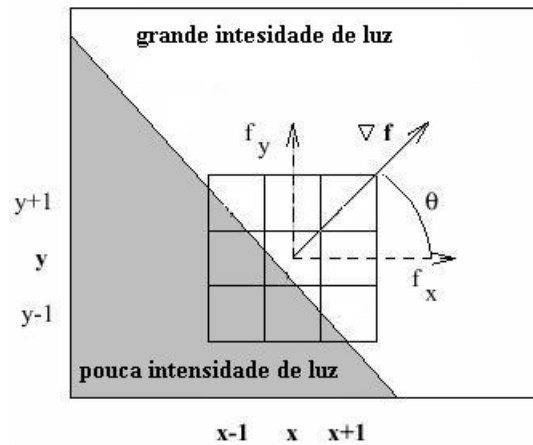


Figura 2.1:

Capítulo 3

Teoria da Amostragem

Mostraremos neste capítulo como se processa a amostragem de um sinal analógico, e em que circunstância podemos a partir de uma amostragem voltar ao sinal analógico sem perda de informações (*aliasing*) .

Definição 3.0.4. *Seja $f : U \subset \mathbb{R}^2 \rightarrow \mathbb{R}$ um sinal, e $\bar{U} = \{\dots, u_{-2}, u_{-1}, u_0, u_1, u_2, \dots\}$ um subconjunto enumerável de U . Uma amostragem pontual de f é a sequência*

$$(\dots, f(u_{-2}), f(u_{-1}), f(u_0), f(u_1), f(u_2), \dots).$$

Este processo é mostrado na figura abaixo, cada valor $f(u_i)$ é chamado de amostra do sinal. A sequência $(\dots, f(u_{-2}), f(u_{-1}), f(u_0), f(u_1), f(u_2), \dots)$ é chamada de sequência de amostras.

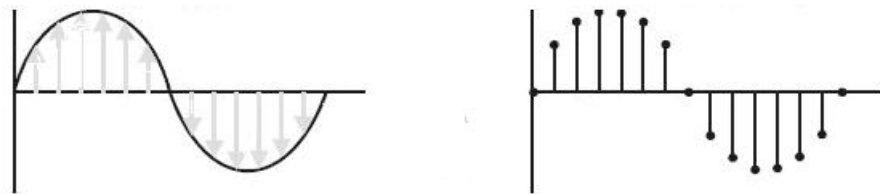


Figura 3.1: amostragem pontual de um sinal.

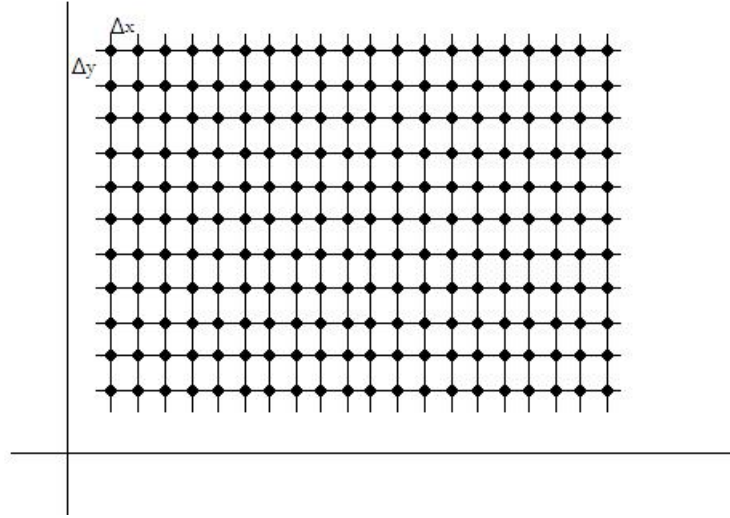
3.1 Amostragem Pontual Uniforme.

A amostragem pontual uniforme é bastante comum e importante na teoria de sinais. Além de permitir um tratamento matemático relativamente simples, ela é utilizada nos diversos dispositivos que manipulam sinais digitais. Inicialmente precisamos definir o conceito de **reticulado**.

Definição 3.1.1. *Seja x_i um eixo de R^2 e um segmento Δx_i não nulo. Um reticulado no espaço R^2 é o conjunto dos pontos de R^2 dados por:*

$$R_{\Delta} = R_{\Delta x_1, \Delta x_2} = \{m_1 \Delta x_1, m_2 \Delta x_2\}; m_i \in Z\}.$$

A figura 3.2 mostra um reticulado no plano R^2 . Cada ponto do reticulado é chamado de vértice do reticulado. É comum nos referirmos ao reticulado através do vetor $\Delta = (\Delta x_1, \Delta x_2)$ que o define.



A amostragem pontual é dita uniforme quando o conjunto de pontos u_i de $(..., f(u_{-2}), f(u_{-1}), f(u_0), f(u_1), f(u_2...))$ são vértices de um reticulado do espaço. O número de amostras, por unidade de espaço, é chamado de taxa de amostragem, ou frequência de amostragem. Em geral, há perda no processo de amostragem pontual. Nestas condições, estamos interessados em saber em que condições podemos evitar perda no processo de amostragem pontual e como reconstruir o sinal original a partir de suas amostras.

3.2 Amostragem Pontual e Transformada de Fourier.

A amostragem pontual do sinal impulso, δ de Dirac, é a função δ de Kronecker [1], definido por:

$$\delta(p) = \begin{cases} 1 & \text{se } p=0 \\ 0 & \text{se } p \neq 0. \end{cases}$$

Definição 3.2.1. Chamamos de função pente associada a um reticulado R_Δ , onde Δ é o vetor $(\Delta x_1, \Delta x_2)$, ao sinal definido por

$$\text{pent}_\Delta = \sum_{k, i_k \in \mathbb{Z}} \delta(x - (i_1 \Delta x_1, i_2 \Delta x_2)), x \in \mathbb{R}^2.$$

Na figura a seguir, ilustramos a função pente, para o caso unidimensional e bidimensional :

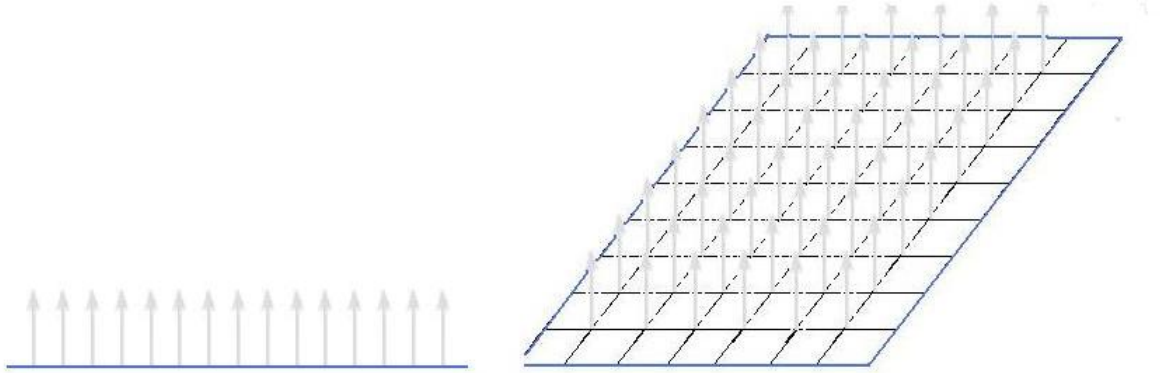


Figura 3.3: função pente unidimensional e função pente bidimensional.

A importância da função pente está no fato de que fazer uma amostragem pontual uniforme de um sinal f no reticulado $R_{\Delta x_1, \Delta x_2}$ é equivalente a multiplicar f pela função pente associada ao reticulado. No caso unidimensional, temos:

$$f \cdot \text{pent}_{\Delta t} = (f(k\Delta))_{k \in \mathbb{Z}},$$

conforme ilustrado na figura abaixo,

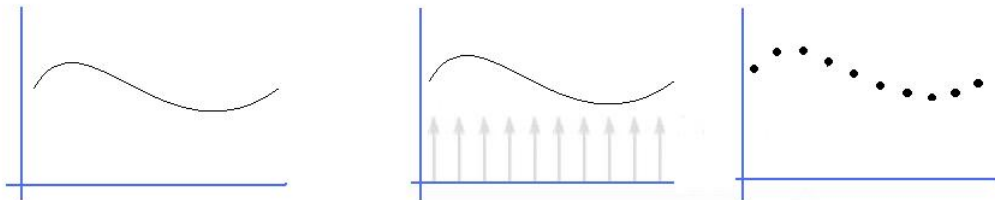


Figura 3.4: amostragem pontual e função pente.

por esta razão a função pente é também chamada de sinal de amostragem.

Um outro resultado importante é que a Transformada de Fourier de uma função pente é outra função pente [14]. Mais precisamente, temos:

$$F(\text{pent}_{\Delta t}(t)) = \frac{1}{\Delta t} \text{pent}_{\frac{1}{\Delta t}}(s).$$

Finalmente, tome $f(x) = \int_{-\infty}^{+\infty} f(t)\delta(x-t)dt$ segue-se que:

$$f * \delta_{t_0}(t) = f(t - t_0).$$

Esse resultado pode ser generalizado para se obter o produto de convolução de um sinal f por uma função pente $\text{pent}_{\Delta t}(t)$.

Vamos ilustrar este resultado no caso unidimensional.

$$(f * \text{pent}_{\Delta t}) = \sum_{-\infty}^{+\infty} f(t - k\Delta t).$$

ou seja, o sinal f é transladado para os vértices do reticulado e o resultado é obtido somando todos os transladados do sinal. Esse resultado é ilustrado na figura abaixo. É claro que esse resultado é valido para sinais multidimensionais.

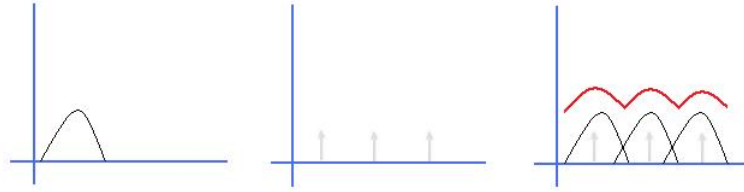


Figura 3.5: Convolução de um sinal f com uma função pente.

Usando os resultados acima podemos finalmente obter informações sobre o modelo espectral de um sinal amostrado (Figura 3.6). No domínio do espaço a amostragem corresponde ao produto com a função pente; do domínio da frequência esse produto corresponde ao produto de convolução com a transformada da função pente, obtida utilizando a equação $(f * \text{pent}_{\Delta t}) = \sum_{-\infty}^{+\infty} f(t - k\Delta t)$ na mesma Figura podemos observar o sinal amostrado e sua Transformada de Fourier.

Portanto, o modelo espectral de sinal amostrado em um reticulado definido por Δt é formado a partir do modelo espectral do sinal original transladado por intervalos de comprimento $\frac{1}{\Delta t}$ na figura 3.7 temos um sinal amostrado

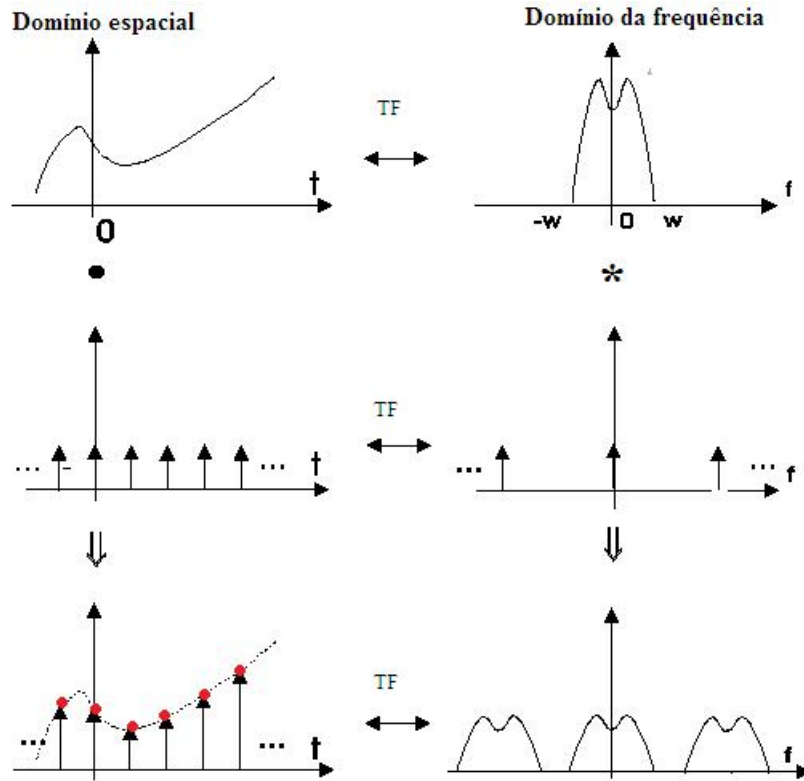


Figura 3.6: Amostragem de sinal no domínio da frequência.

onde ocorre superposição de frequências. Mais precisamente, altas frequências do sinal aparecem como baixas frequências no espectro do sinal amostrado. No espectro final, essas altas frequências se perdem tornando impossível a recuperação da informação correta das frequências presentes no sinal original.

3.3 Teorema da Amostragem de Shannon-Whittaker.

Considere agora um sinal f com banda limitada no domínio $[-\Omega, +\Omega]$ do espaço de frequências, então se:

$$\frac{1}{\Delta t} > 2\Omega, \text{ ou seja, } \Delta t < \frac{1}{2\Omega}.$$

No modelo espectral do sinal amostrado não haverá superposição de frequências, portanto o modelo espectral do sinal original pode ser obtido

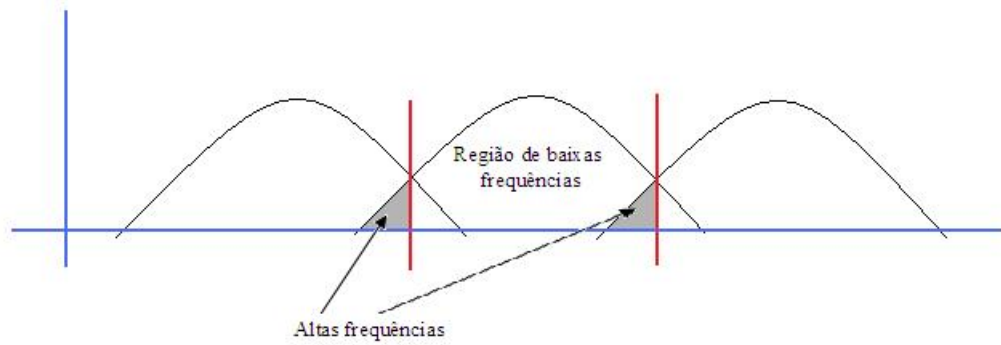


Figura 3.7: Amostragem de sinal, efeito *aliasing*.

usando um filtro de passa baixa com ganho unitário, cuja função de transferência é a função pulso $p_{\Omega}(s)$ (figura 3.8).

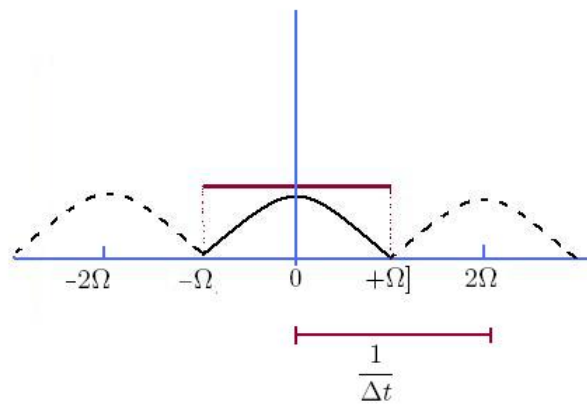


Figura 3.8: Amostragem ideal de sinal.

O sinal original pode então ser obtido usando a Transformada de Fourier Inversa após o processo de filtragem. Com esses argumentos podemos enunciar:

Teorema 3.3.1. [Teorema da Amostragem de Shannon-Whittaker]. Sejam f um sinal de banda limitada, e Ω o menor valor de frequência tal que $\text{supp}(\hat{f}) \subset [-\Omega, +\Omega]$. O sinal f pode ser reconstruído de forma exata a partir de um conjunto uniforme de amostras $\{m\Delta t; m \in \mathbb{Z}\}$, se $\Delta t < \frac{1}{2\Omega}$.

prova: Seja f um sinal banda limitada conforme acima e sua sequência de amostragem $f_{disc} = (f(k\Delta t))_{k \in \mathbb{Z}}$, temos que:

$$f(k\Delta t) = f(t) \cdot \text{pent}_{\Delta t}(t);$$

Tendo em vista que $F(f \cdot g) = F(f) * F(g)$, usando a Transformada de Fourier na equação acima, temos:

$$f_{disc}(s) = F[f(t) \cdot \text{pent}_{\Delta t}(t)];$$

$$f_{disc}(s) = f(s) * F[\text{pent}_{\Delta t}(t)];$$

$$f_{disc}(s) = f(s) * \frac{1}{\Delta t} \text{pent}_{\frac{1}{\Delta t}}(t);$$

$$f_{disc}(s) = \frac{1}{\Delta t} f(s) * \text{pent}_{\frac{1}{\Delta t}}(t).$$

Temos que $f_{disc}(s)$ é o modelo discreto de f no domínio de Fourier. Observando que o sinal $f_{disc}(s)$ é composto por $f(s)$ transladados com comprimento $\frac{1}{\Delta t}$ e por hipótese $\Delta t < \frac{1}{2\Omega}$, podemos então passar um filtro passa baixa cuja função de transferência é a função pulso $P_{\Omega}(s)$ e teremos $f(s)$ a partir de $f_{disc}(s)$ ou seja:

$$f(s) = \left[\frac{1}{\Delta t} f(s) * \text{pent}_{\frac{1}{\Delta t}} \right] \cdot P_{\Omega}(s).$$

Para voltar ao modelo espacial usamos a Transformada Inversa de Fourier:

$$f(t) = F^{-1} \left[\frac{1}{\Delta t} f(s) * \text{pent}_{\frac{1}{\Delta t}} \right] * F^{-1}[P_{\Omega}(s)];$$

Como $F(f * g) = F(f) \cdot F(g)$ e a Transformada inversa da função pulso é a função sinc a equação acima fica:

$$f(t) = f(t) \cdot \text{pent}_{\Delta t} t * 2\Omega \text{sinc}(2\pi\Omega t);$$

Como $f(t) \cdot \text{pent}_{\Delta t}(t) = f(k\Delta t)_{k \in \mathbb{Z}}$ temos então:

$$f(t) = f(k\Delta t) * 2\Omega \text{sinc}(2\pi\Omega t);$$

portanto,

$$f(t) = \sum_{k=-\infty}^{k=+\infty} f(k\Delta t) \cdot 2\Omega \text{sinc}(2\pi\Omega t - k\Delta t).$$

A equação $f(t) = \sum_{k=-\infty}^{k=+\infty} f(k\Delta t) \cdot 2\Omega \text{sinc}(2\pi\Omega t - k\Delta t)$ é a expressão de reconstrução exata do sinal f a partir de suas amostras $f(k)$.

O teorema Shannon-Whittaker garante que se o sinal f é limitado com uma banda de 0 a Ω ciclos por unidade de tempo, então ele fica completamente determinado por uma amostragem pontual uniforme em um reticulado de frequência $\Delta t < \frac{1}{2\Omega}$. Isso significa que devemos pegar pelo menos uma amostra em cada meio ciclo do sinal.

A série $f(t) = \sum_{k=-\infty}^{k=+\infty} f(k\Delta t) 2\Omega \text{sinc}[2\pi\Omega(t - k\Delta t)]$ converge absolutamente e a convergência é uniforme nas partes compactas do domínio. Usando a notação da demonstração acima, o Teorema da amostragem de Shannon-Whittaker pode ser enunciando do seguinte modo:

Teorema 3.3.2. [Teorema da reconstrução ideal]. *Se um sinal f tem banda limitada, $\text{supp } f \subset [-\Omega, \Omega]$, e f é amostrado pontualmente dentro do limite de Nyquist, então a sequência de amostras $(f(k\Delta t))$ é uma representação exata do sinal e o sinal f pode ser reconstruído a partir de suas amostras.*

3.4 Extensões do Teorema de Amostragem

O Teorema da Amostragem de Shannon-Wittaker foi obtido com uso de diversas hipóteses. Dentre elas, três são bastante restritivas: o sinal é unidimensional, a amostragem deve ser uniforme e o sinal deve possuir banda limitada.

A extensão do teorema para incluir sinais de banda não-limitada, bem como a extensão para amostragem pontual não uniforme, não serão tratadas aqui.

A extensão para sinais bidimensionais é de grande importância devido ao fato de que o sinal imagem, nosso principal objeto de estudo, tem dimensão 2.

Existem diversas possibilidades de estender o teorema para sinais bidimensionais. Uma das dificuldades é a geometria e a topologia do domínio do sinal. Uma extensão bastante natural, e suficiente para nossos propósitos, será descrita a seguir

Considere o sinal definido em um domínio U retangular bidimensional

$$U = \prod_{i=1}^n [-a_i, b_i],$$

do espaço. A amostragem é feita em um reticulado uniforme $\Delta = (\Delta x_1, \Delta x_2)$, bidimensional.

Já vimos anteriormente que a Transformada de Fourier se estende naturalmente. Portanto a extensão do conceito de banda limitada é imediata. O limite de Nyquist deve ser satisfeito considerando a restrição do sinal a cada coordenada x_i , $i = 1, 2$ do espaço, ou seja, existe um vetor $\Omega = (\Omega_1, \Omega_2)$ de forma que:

$$\Delta x_1 < \frac{1}{2\Omega_1}, \Delta x_2 < \frac{1}{2\Omega_2}.$$

A extensão da função sinc é feita por separabilidade,

$$\text{sinc}(x_1, x_2) = \text{sinc}(x_1) \cdot \text{sinc}(x_2).$$

Nas condições acima, a extensão do teorema é imediata:

Seja um sinal $f : U \subset \mathbb{R}^2 \rightarrow \mathbb{R}$ que possui banda limitada, e sua frequência de amostragem obedece o limite de Nyquist. Então f fica completamente determinado pelo valor de suas amostras nos vértices do reticulado. Mais precisamente, temos:

$$f(x_1, x_2) = \sum_{k_1=-\infty}^{+\infty} \sum_{k_2=-\infty}^{+\infty} f(k_1\Delta x_1, k_2\Delta x_2) \text{sinc}[2\pi\Omega(X - K\Delta X)],$$

onde:

$$\text{sinc}[2\pi\Omega(X - K\Delta X)] = \text{sinc}[2\pi\Omega_1(x_1 - k_1\Delta x_1), 2\pi\Omega_2(x_2 - k_2\Delta x_2)].$$

Então a demonstração dessa versão bidimensional do Teorema da Amostragem de Shannon-Whitaker é análoga à demonstração do caso unidimensional.

3.5 Produto de Funções no Domínio Espacial e *Aliasing*

Suponha uma operação feita ponto a ponto no domínio espacial, no qual temos uma imagem f e um filtro g , supondo ainda que as transformadas F e G tenham banda limitada e mesma largura de banda ω .

Para o produto $f(x).g(x)$ no domínio espacial, teremos no domínio da frequência $F * G$ com 2ω de banda, como tal operação dobra a largura de banda, devemos dobrar a taxa de amostragem sob pena de perdemos informações podendo ocorrer *aliasing*.

Capítulo 4

Detecção de Arestas

Neste capítulo vamos definir imagem digital, arestas e como detectá-las usando o gradiente da imagem.

4.1 Imagem Digital

O termo **imagem digital** refere-se à função bidimensional de intensidade da luz $f(x, y)$, onde x e y denotam as coordenadas espaciais. E o valor f em qualquer ponto (x, y) é proporcional ao nível de cinza da imagem naquele ponto [2]. Uma imagem digital é discretizada nas coordenadas e no nível de cinza. Podemos olhar uma imagem digital como sendo uma matriz cujos índices de linhas e de colunas identificam um ponto na imagem, e o valor desse elemento da matriz identifica o nível de cinza naquele ponto. Os elementos dessa matriz digital são chamados de **pixel**.

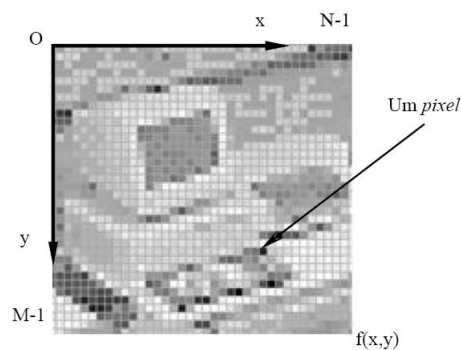


Figura 4.1: Imagem digital.

4.2 Definição de Arestas

Definição 4.2.1. Para uma função unidimensional $f : R \rightarrow R$, arestas de f são o conjunto de pontos $x \in R$ onde $|f'(x)|$ é máximo.

Para estender o conceito acima para funções $f : R^2 \rightarrow R$, vamos definir aresta de f como o conjunto de pontos onde o módulo do gradiente é máximo na direção do gradiente, ou seja:

Definição 4.2.2. Aresta de $f = \{ (x, y) \in R^2, |\nabla f(x, y)| \text{ é máximo na direção de } \nabla f(x, y) \}$.

A Figura 4.4 ilustra o conceito de aresta.

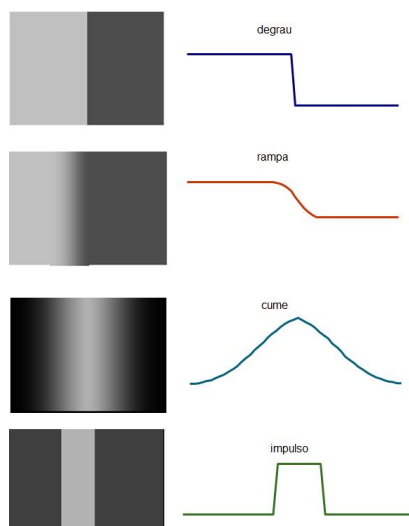


Figura 4.2: tipos de arestas

Nas imagens a seguir, temos:

4.5 imagem original.

4.6 derivada na direção x , $f_x = g_{x,\sigma} * f$.

4.7 derivada na direção y , $f_y = g_{y,\sigma} * f$.

4.8 norma do gradiente.



Figura 4.3: fotografia digital de uma placa.

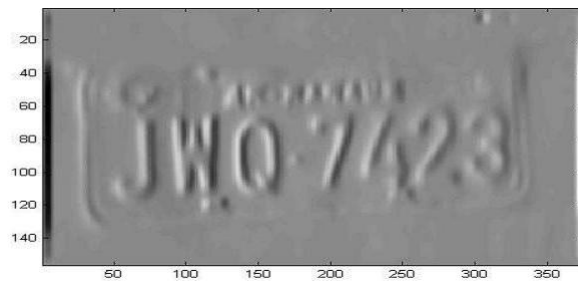


Figura 4.4: derivada na direção x.

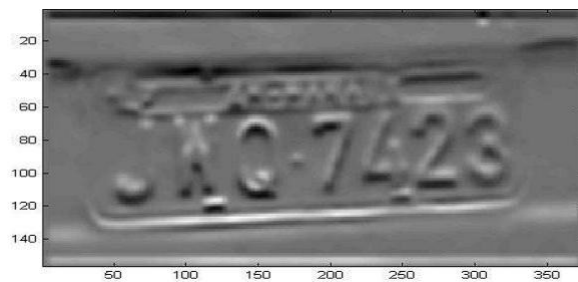


Figura 4.5: derivada na direção y.



Figura 4.6: norma do gradiente.

4.3 Tensores

Neste capítulo vamos definir o Tensor Q e o Tensor de Estrutura, mostraremos como é feita a detecção de arestas usando o Tensor de Estrutura e algumas implementações.

4.3.1 Tensor Q

Definição 4.3.1. *Em cada ponto da imagem o Tensor Q é obtido pelo produto $\nabla f(x, y) \nabla f(x, y)^T$, onde as derivadas de f são obtidas através da convolução das derivadas da gaussiana bidimensional com a função imagem, $f_x = g_{x,\sigma} * f$ e $f_y = g_{y,\sigma} * f$:*

$$Q_\sigma = \begin{bmatrix} q_{11} & q_{12} \\ q_{21} & q_{22} \end{bmatrix} = \begin{bmatrix} f_x^2 & f_x f_y \\ f_x f_y & f_y^2 \end{bmatrix}.$$

Um cálculo simples mostra que o Tensor Q tem apenas um autovalor não nulo que representa características unidimensionais (aresta, fig 2.1):

$$Av = \lambda v;$$

$$[A - \lambda I]v = 0;$$

$$\det(A - \lambda I) = 0;$$

$$f_x^2 f_y^2 - \lambda f_x^2 - \lambda f_y^2 + \lambda^2 - (f_x f_y)^2 = 0;$$

$$\lambda[\lambda - (f_x^2 + f_y^2)] = 0;$$

$$\lambda_1 = 0;$$

$$\lambda_2 = f_x^2 + f_y^2.$$

4.3.2 Tensor de Estrutura

O Tensor de Estrutura é baseado no Tensor Q , onde em geral as entradas deste tensor são convoluídas com uma gaussiana com desvio σ . O cálculo da média espacial distribui informações das vizinhanças, e cada ponto que recebe contribuições de arestas com direções diferentes terão 2 autovalores positivos, que permite serem reconhecidas como característica intrínseca 2D.

O maior autovalor será usado para detectar as arestas; e o menor autovalor tem informações de pontos nas proximidades de junções/cantos, como

não temos uma boa definição de junções, cantos ou vértices, não faremos uso do segundo autovalor.

$$S_{\sigma',\sigma} = \begin{bmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{bmatrix} = \begin{bmatrix} g_{\sigma'} * f_x^2 & g_{\sigma'} * f_x f_y \\ g_{\sigma'} * f_x f_y & g_{\sigma'} * f_y^2 \end{bmatrix} = \begin{bmatrix} g_{\sigma'} * q_{11} & g_{\sigma'} * q_{12} \\ g_{\sigma'} * q_{21} & g_{\sigma'} * q_{22} \end{bmatrix}$$

Nas imagens a seguir temos que:

- 1- Para cada pixel da imagem foi calculado o tensor de estrutura.
- 2- O maior autovalor deste tensor é armazenados em uma matriz K_1 .
- 3- Usaremos posteriormente K_1 para obter arestas.

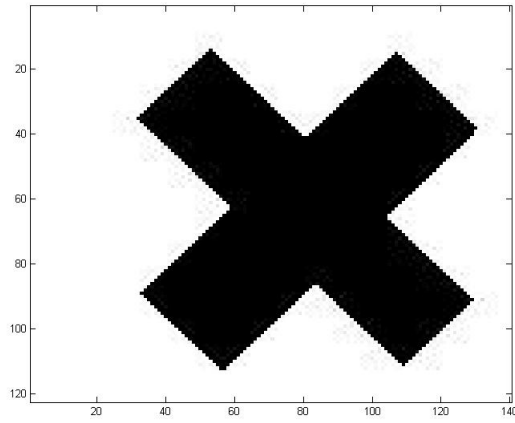


Figura 4.7: imagem original usada no algoritmo.

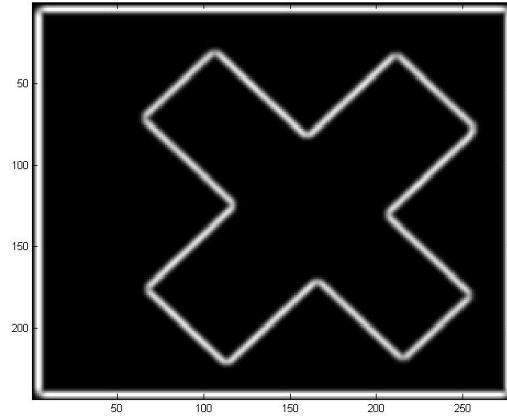


Figura 4.8: matriz de autovalores k_1 do tensor, usada para detectar arestas.

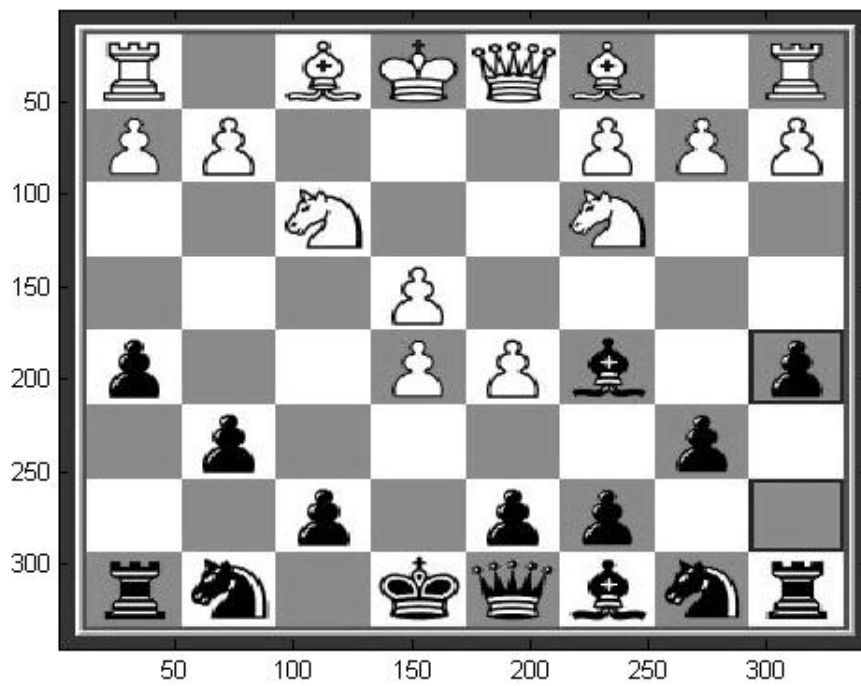


Figura 4.9: imagem original.

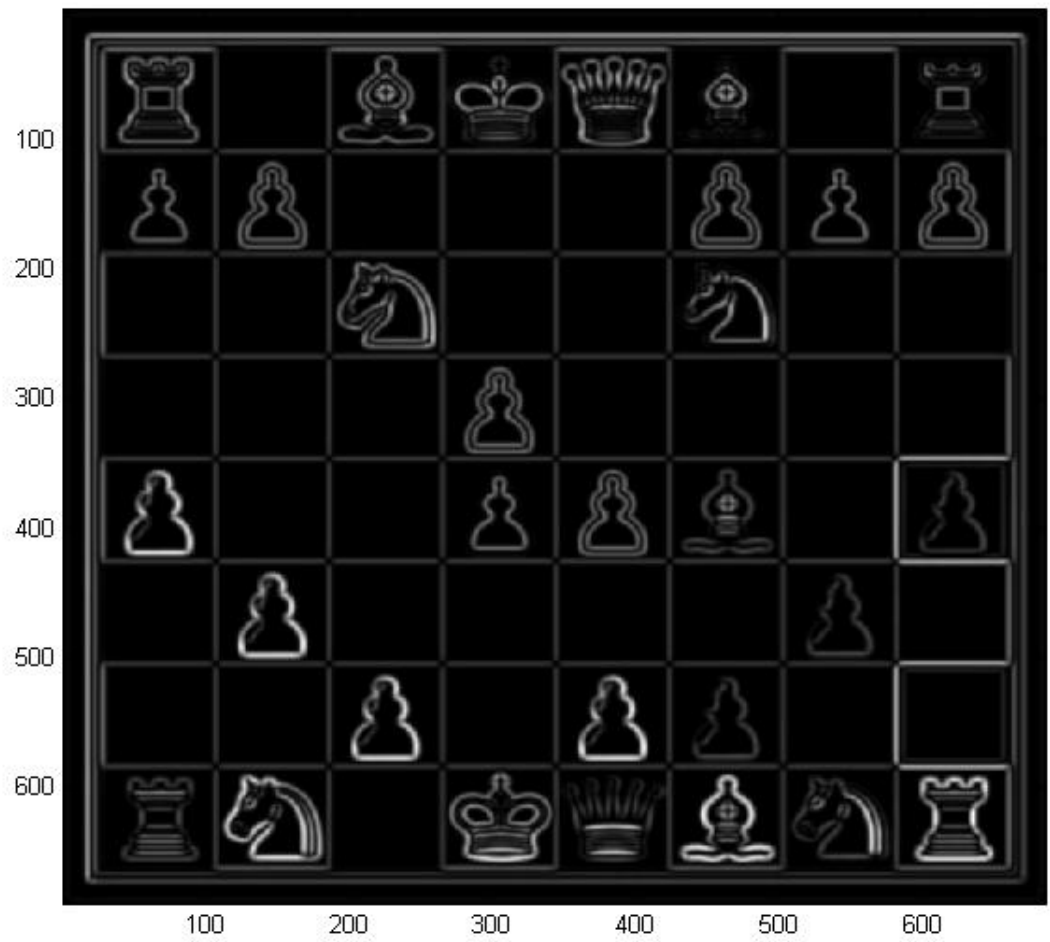


Figura 4.10: matriz de autovalores K_1 do tensor, usada para detectar arestas.

Capítulo 5

Mudanças Propostas

Este capítulo é dedicado às mudanças no cálculo do tensor de estrutura, são apresentadas também as melhorias na detecção das arestas.

5.1 Amostragem Correta

Suponha que a imagem original $f(x, y)$ foi amostrada corretamente na taxa de Nyquist. Fixando a distância $\lambda N = 1$ no domínio espacial, então f tem banda limitada $\omega N = \pi$ no domínio da frequência, portanto se $|\omega_1| |\omega_2| \geq \pi \Rightarrow F(\omega_1, \omega_2) = 0$, onde F é a Transformada de Fourier de f e ω_1 e ω_2 são as coordenadas da frequência. Pelo teorema da convolução, a convolução de f com filtros derivados gaussianos corresponde ao produto de F com a Transformada de Fourier de G_x e G_y dos filtros.

$$f * g_x = F \cdot G_x = f_x$$

$$f * g_y = F \cdot G_y = f_y.$$

Suponha que as derivadas da função $G(x, y)$ não tenham banda limitada, as derivadas f_x e f_y obtidas pelo produto da imagem com G_x e G_y terão ainda banda limitada com largura de banda $\omega N = \pi$

Para obter o tensor Q multiplicamos o vetor gradiente $\nabla f(x, y)$ com seu transposto $\nabla f(x, y)^T$.

$$Q = \nabla f(x, y) \nabla f(x, y)^T.$$

A multiplicação ponto a ponto de duas funções no domínio espacial corresponde à convolução no domínio da frequência

$$f_1 f_2 \simeq F_1 * F_2.$$

A convolução de dois espectros com mesma largura de banda dobra a largura de banda. Então, para evitar perda de informações e o *aliasing* devemos representar os elementos do tensor Q com metade da distância da amostra da imagem original. Este pequeno detalhe conduz a melhorias significativas na qualidade das arestas obtidas mais tarde, sendo assim, dobramos a amostragem durante o cálculo das imagens derivadas.

Seja a convolução de uma imagem discreta f com o núcleo de um filtro analógico g :

$$(f * g)(x, y) = \sum_{i, j} f(i, j) \cdot g(x - i, y - j).$$

Apesar de f ser discreto, o lado direito desta equação é uma função analógica que pode ser avaliada em pontos arbitrários (x, y)

Nós podemos dobrar a taxa de amostragem diretamente no cálculo da imagem derivada $f * g_x$ e $f * g_y$. O problema da densidade insuficiente das amostras não se aplica aos métodos baseados no Tensor de Estrutura, pois ele afeta todos os algoritmos que fazem exame de produtos de derivadas.

Nas figuras a seguir, comparamos a detecção pelo método de Canny original e Canny com amostragem dobrada, usamos imagens sintéticas e fotografias digitais, em ambos os casos as diferenças são claramente visíveis.

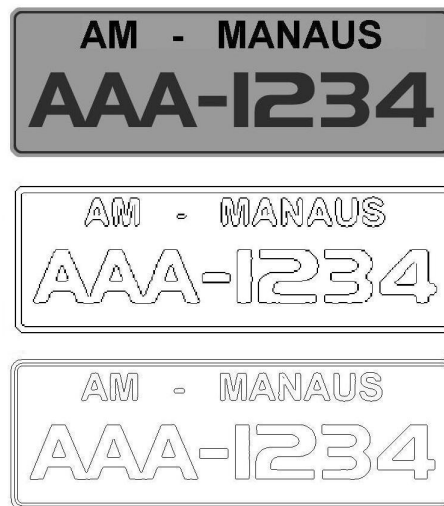


Figura 5.1: Imagem sintética.



Figura 5.2: Fotografia digital, Gabrielle.



Figura 5.3: Detecção usando método de Canny.



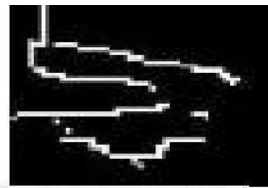
Figura 5.4: Detecção usando método de Canny dobrando a amostragem.



Figura 5.5: Detecção usando método de Canny dobrando a amostragem.



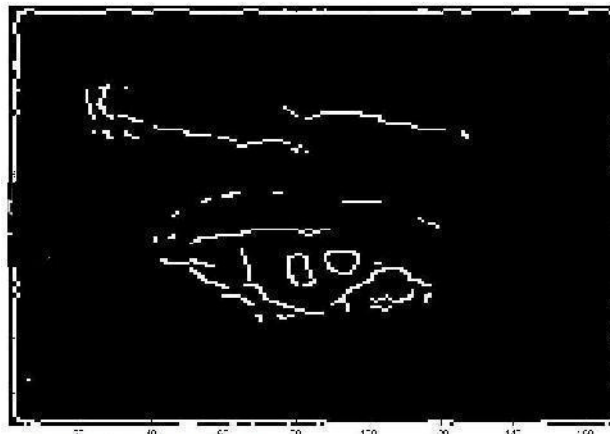
Figura 5.6: Detecção com o tensor de estrutura dobrando a amostragem.



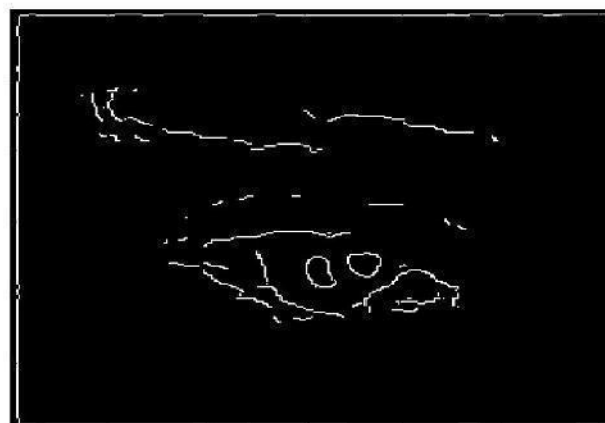
a) amostragem simples



b) detalhe da imagem



c) amostragem dobrada



d) tensor de estrutura

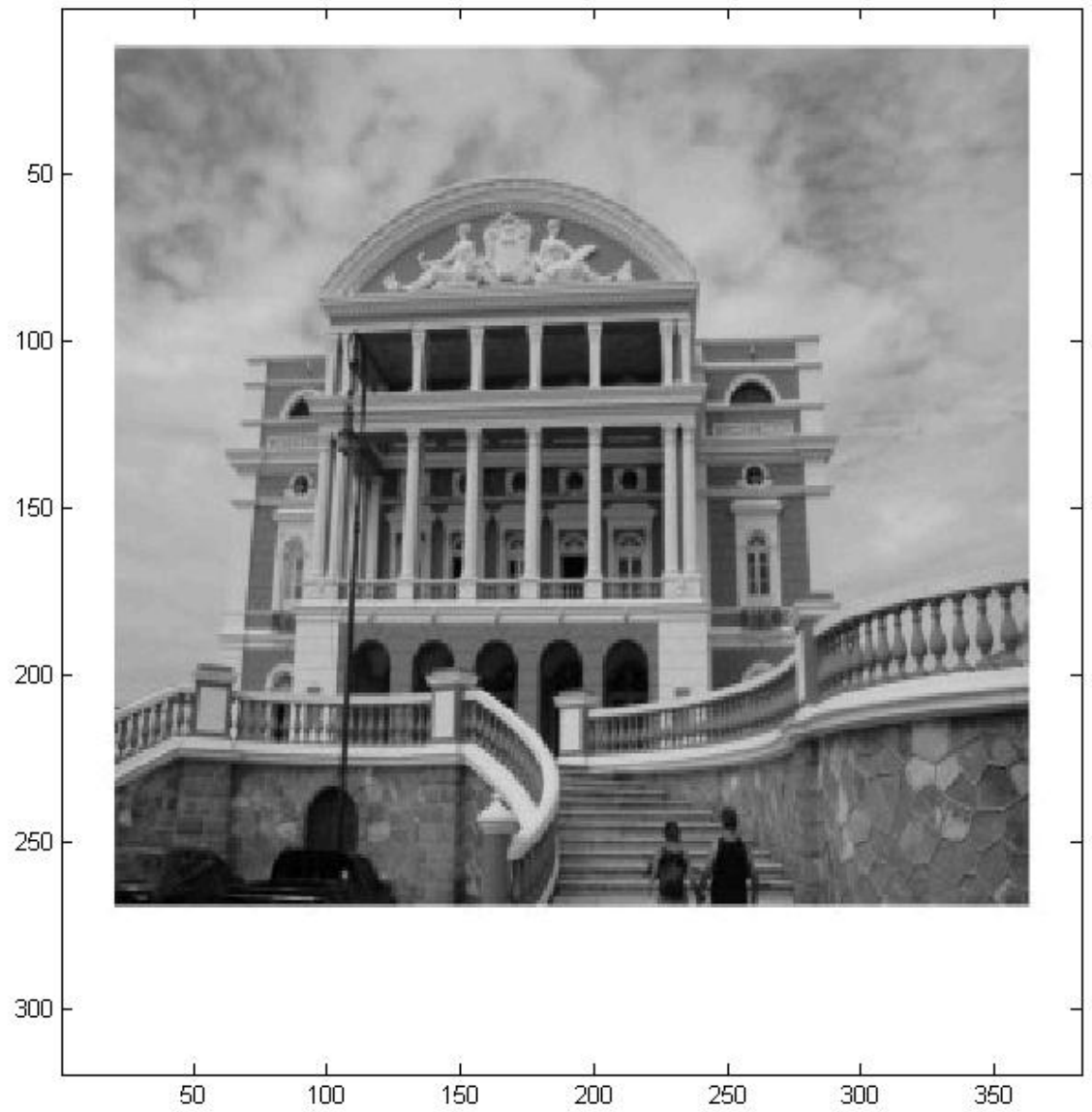


Figura 5.7: Fotografia digital, Teatro Amazonas.

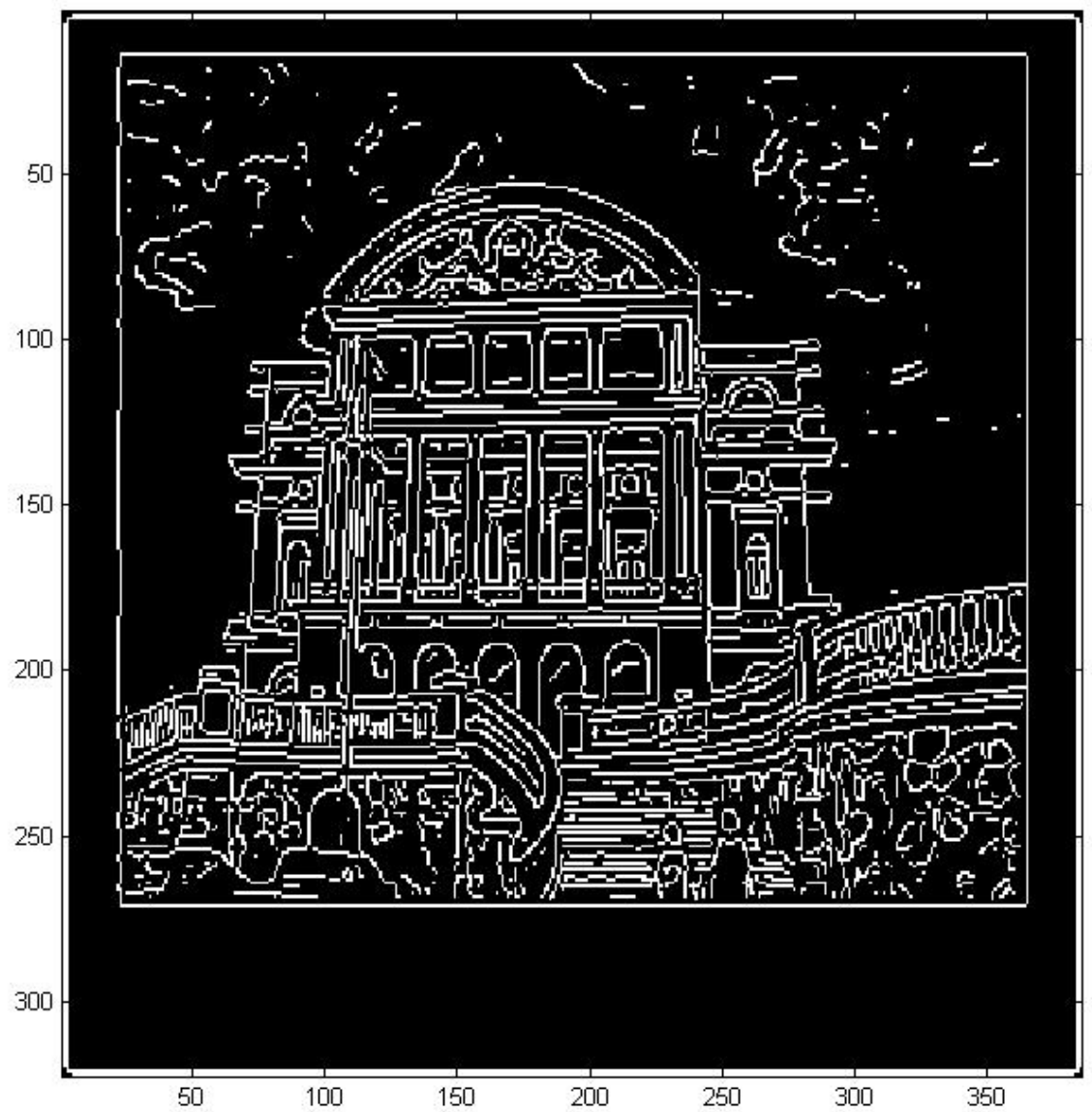


Figura 5.8: Detecção usando método de Canny.

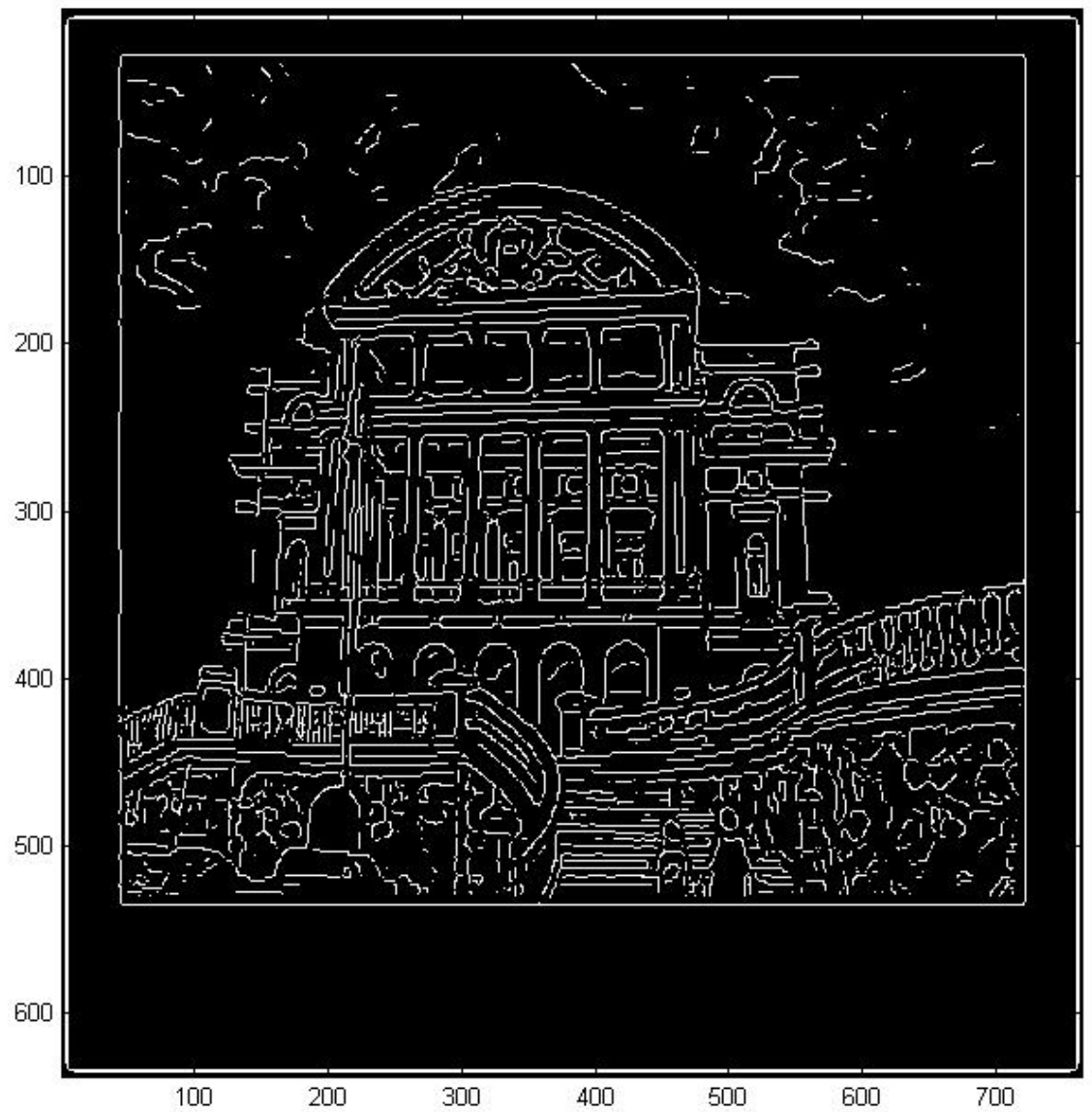
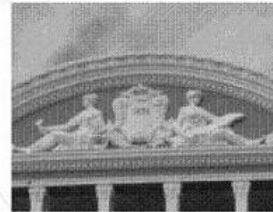
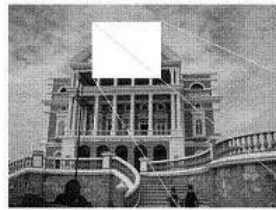
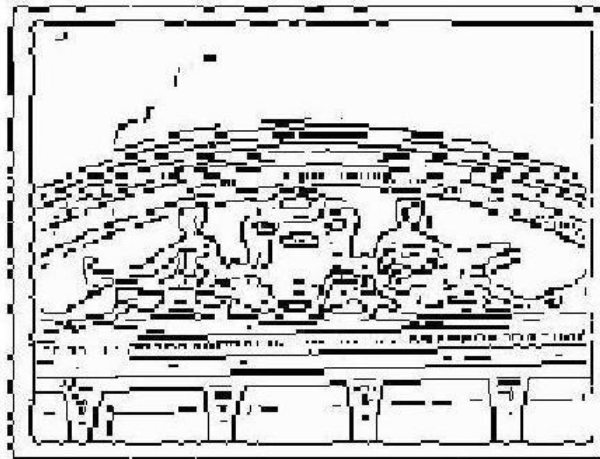


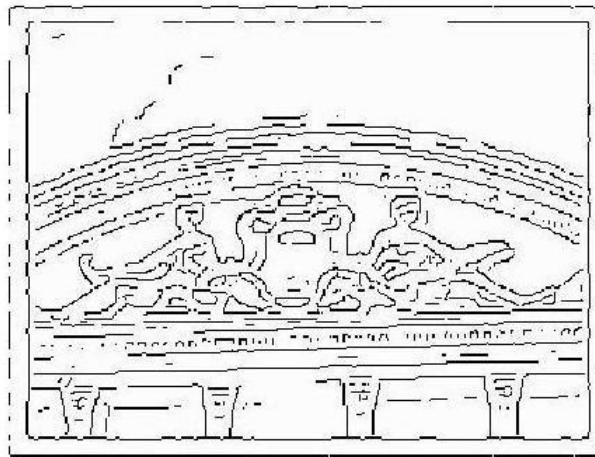
Figura 5.9: Detecção usando método de Canny dobrando a amostragem.



a) detalhe da imagem



b) amostragem simples



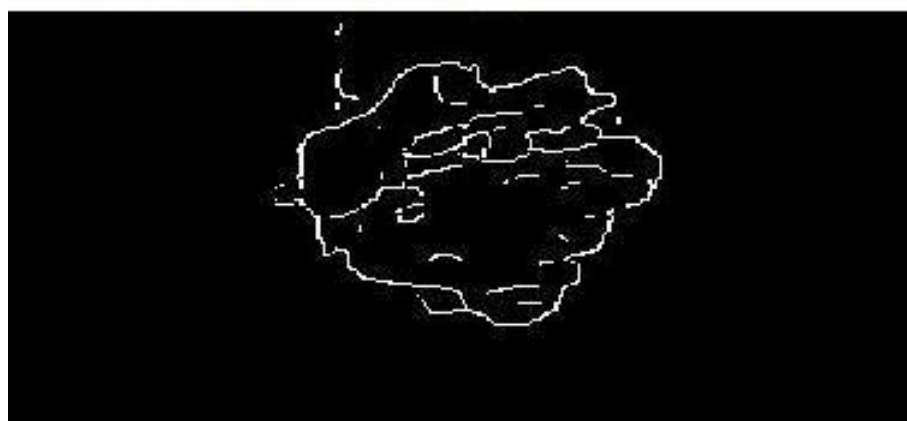
c) amostragem dobrada



a) imagem original.



b) amostragem simples.



c) amostragem dobrada.

Figura 5.10: Imagem médica.

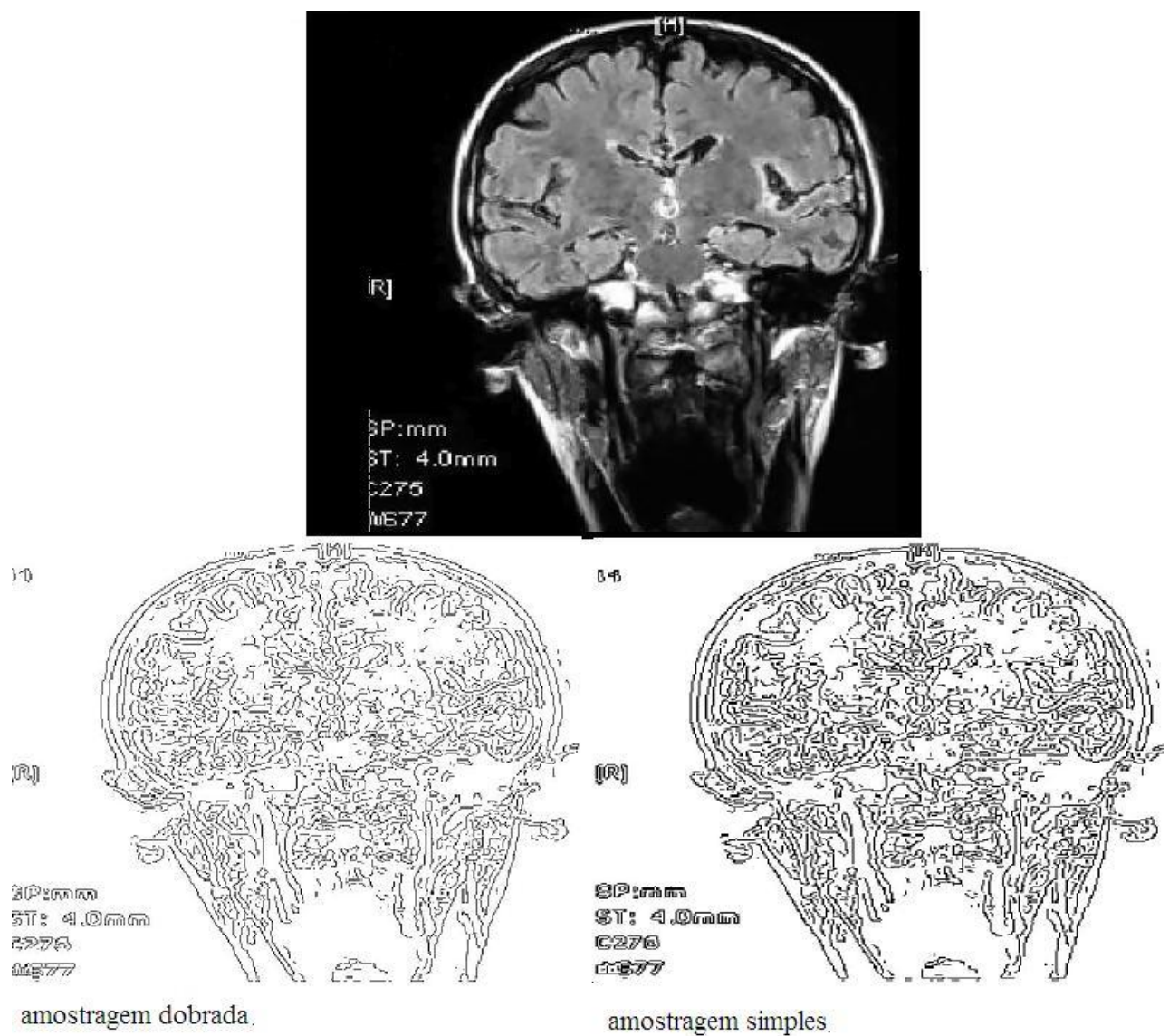


Figura 5.11: Imagem médica.

5.2 Integração de Espaço

O Tensor Q tem somente um autovalor diferente de zero que vai representar característica intrínseca unidimensional, o cálculo da média espacial é

feito geralmente com filtros gaussianos que distribui essas informações sobre uma vizinhança. O problema desses filtros é que ele borra a imagem, logo as informações da estrutura ficam comprometidas. Por exemplo, se duas arestas paralelas estiverem próximas, elas se fundirão em uma única aresta e as informações em torno delas serão perdidas. Pequenas arestas circulares se fundirão formando pequenos blocos e as informações do centro serão perdidas também.

A falha na etapa de integração encontra-se na natureza linear de cálculo da média. A média rotacional simétrica é aplicada em todos os pontos que conduz ao erro, as informações da estrutura devem ser distribuídas nas arestas e não perpendicular a elas. Daqui o natural é usar filtros não lineares baseado na difusão unisotrópica [11] ou na gaussiana unisotrópica [10] baseado no sentido local da aresta, entretanto os filtros elípticos não conduzem a melhorias significativas sobre a integração isotrópica tradicional. Então propomos usar um filtro não-linear orientado do tipo *hour-glass* (figura 5.4), podemos interpretar esse filtro como codificador de prováveis continu-ações de uma aresta.

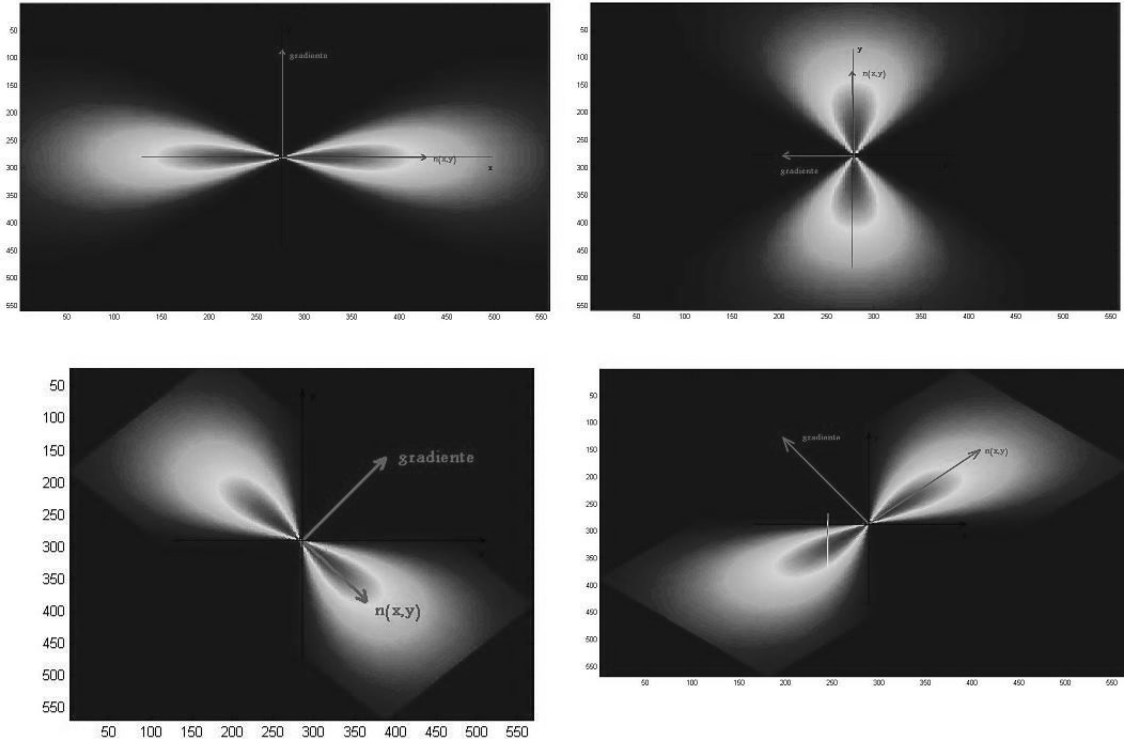


Figura 5.12: Filtro Hour-Glass.

5.3 Filtro Hour-Glass

Vamos definir nossa máscara não-linear com funções separáveis polares, onde a parte radial é um filtro gaussiano e a parte angular modula a gaussiana de modo que a função seja nula nos pontos perpendiculares à aresta no sentido (x, y) .

$$h_{\sigma', \rho}(r, \phi, \phi_0) = \frac{1}{N} e^{-\frac{r^2}{2\sigma'^2}} e^{-\frac{\tan(\phi - \phi_0)^2}{2\rho^2}}$$

Onde ρ mede a força de orientação, e N a constante de normalização que faz a máscara integrar a unidade. Em cada ponto na imagem, a máscara é girada de acordo com a orientação local da aresta definida pelo vetor $n(x, y) = (\cos\phi_0, \sin\phi_0)^t$ que é perpendicular ao gradiente.

Seja p, q o sistema de coordenadas locais, então:

$$r^2 = p^2 + q^2$$

$$\tan(\theta - \theta_0) = \frac{p}{q}$$

quando $r = 0$ faça $\phi := 0$

n, p e q podem ser calculadas a partir das coordenadas globais $X = (x, y)$ onde

$$p = n^\top \cdot x$$

$$q = n_\perp^\top \cdot x$$

a máscara então ficará :

$$h_{\sigma', \rho}(x, n) = \frac{1}{N} \cdot e^{-\frac{x^\top x}{2\sigma'^2} - \frac{1}{2\rho^2} \cdot \left(\frac{n^\top x}{n_\perp^\top x}\right)^2} \text{ se } n^\top x \neq 0$$

$$h_{\sigma', \rho}(x, n) = 0 \text{ se } n^\top x = 0 \text{ e } n_\perp^\top x \neq 0$$

$$h_{\sigma', \rho}(x, n) = \frac{1}{N} \text{ nos outros casos.}$$

O operador linear é definido por:

$$T_{\sigma, \sigma', \rho} = \tau_{\sigma', \rho}[Q_\sigma]$$

Os elementos do operador são dados por:

$$t_{i,j} = \sum_{x', y'} h_{\sigma', \rho}(x - x', y - y', n(x', y')) q_{i,j}(x', y')$$

5.4 Detecção de Arestas com o Tensor de Estrutura

As melhorias feitas no tensor da estrutura permitem detecção de arestas com maior precisão. Nosso tensor T de ordem 2 é positivo e simétrico

$$T = \begin{bmatrix} t_{11} & t_{12} \\ t_{12} & t_{22} \end{bmatrix}$$

Logo possui dois autovalores dados por:

$$\lambda_{1,2} = \frac{1}{2}(t_{11} + t_{22} \pm \sqrt{(t_{11} - t_{22})^2 + 4t_{12}^2})$$

Os quais codificam os valores de interesse dados pelos autovetores ortogonais n_1 e n_2 , onde n_1 é dado por:

$$n_1 = \begin{pmatrix} \cos\phi_1 \\ \sin\phi_1 \end{pmatrix}$$

a orientação de n_1 é dada por:

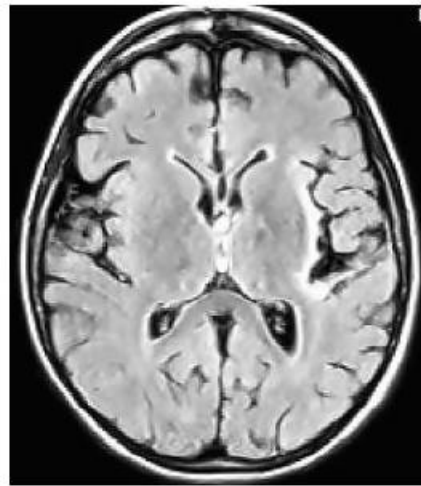
$$\phi_1 = \frac{1}{2} \arctan\left(\frac{2t_{12}}{t_{11} - t_{22}}\right)$$

T pode ser decomposto na seguinte soma de tensores:

$$T_{aresta} + T_{\lambda_2} = (\lambda_1 - \lambda_2)n_1n_1^\top + \lambda_2 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$\lambda_1 - \lambda_2$ pode ser interpretado como uma força de aresta. Formalmente, estas quantidades são obtidas pela contração do T_{aresta} e do T_{λ_2} ao longo da diagonal principal, isto é, pelo traço da matriz, que é rotacionalmente invariante, então usamos a força de aresta T_{aresta} , $\sqrt{\lambda_1 - \lambda_2}.n_1$, que é implementado no algoritmo de Canny em vez do gradiente normal.

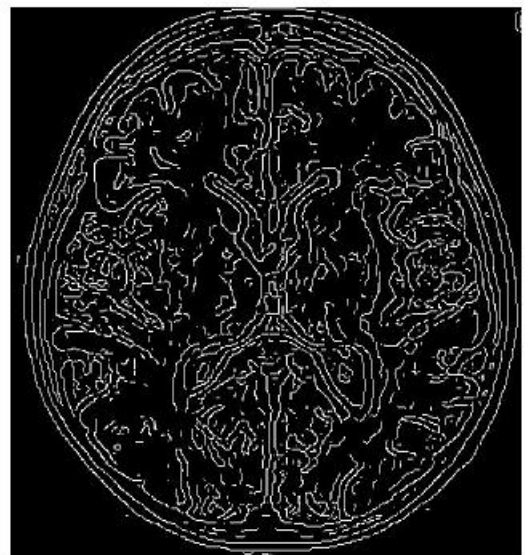
5.5 Resultados



a) imagem original



b) amostragem simples



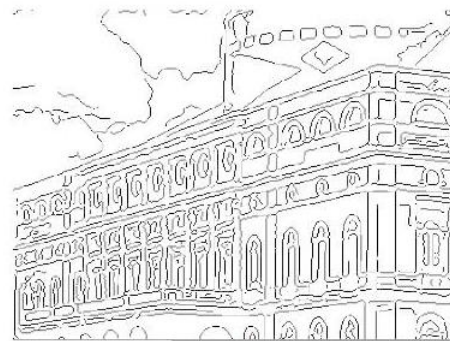
c) amostragem dobrada

Figura 5.13: Imagem médica.

Imagens da implementação



canny com amostragem simples



canny com amostragem dobrada

Figura 5.14: amostragem simples e dobrada.

Imagens da implementação

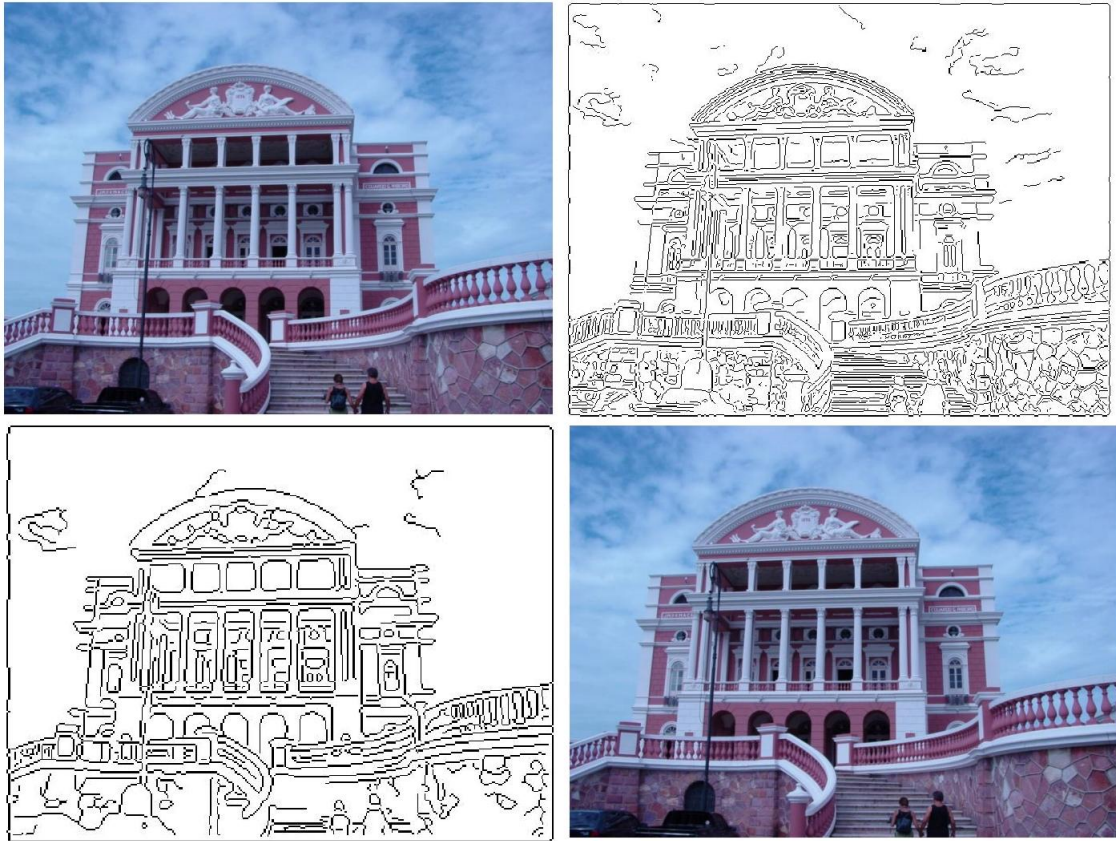


Figura 5.15: amostragem simples e dobrada.

5.6 Resultados

Imagens da implementação

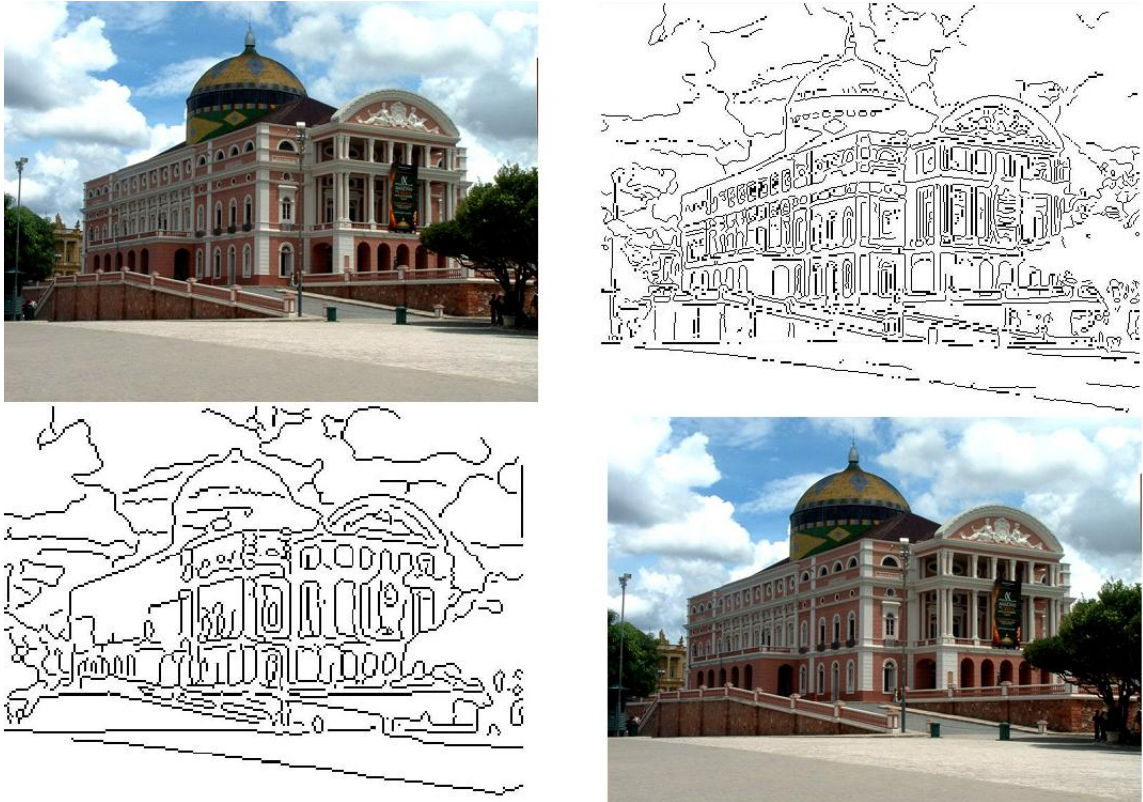


Figura 5.16: imagem original



Figura 5.17: imagem original

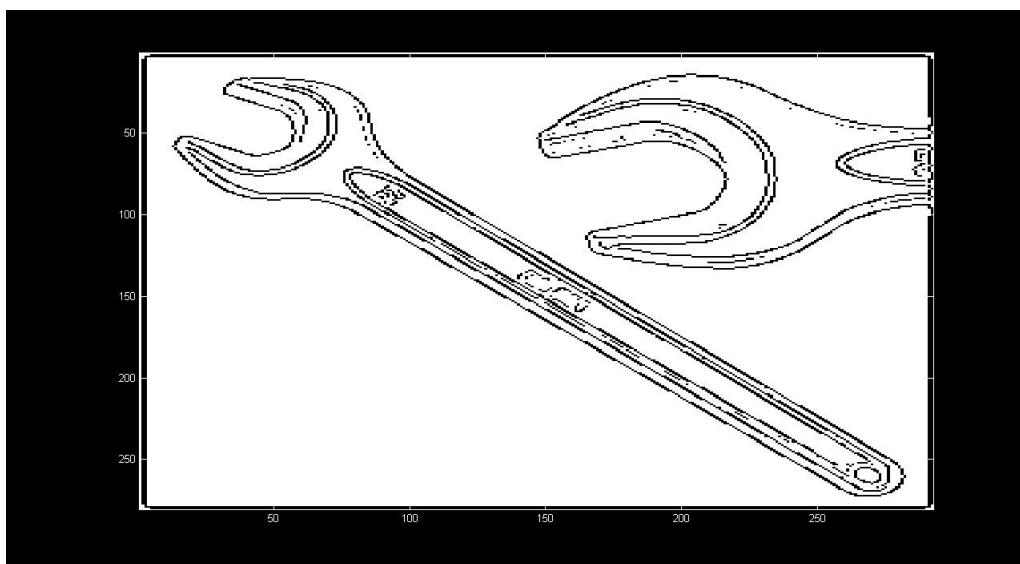


Figura 5.18: detecção com gradiente, amostragem simples

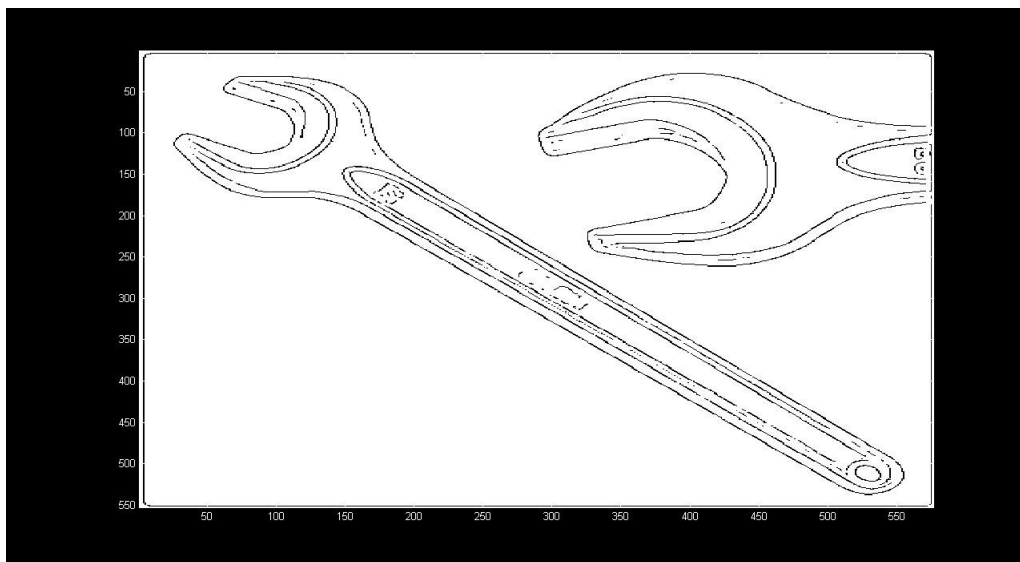


Figura 5.19: detecção com gradiente, amostragem dobrada

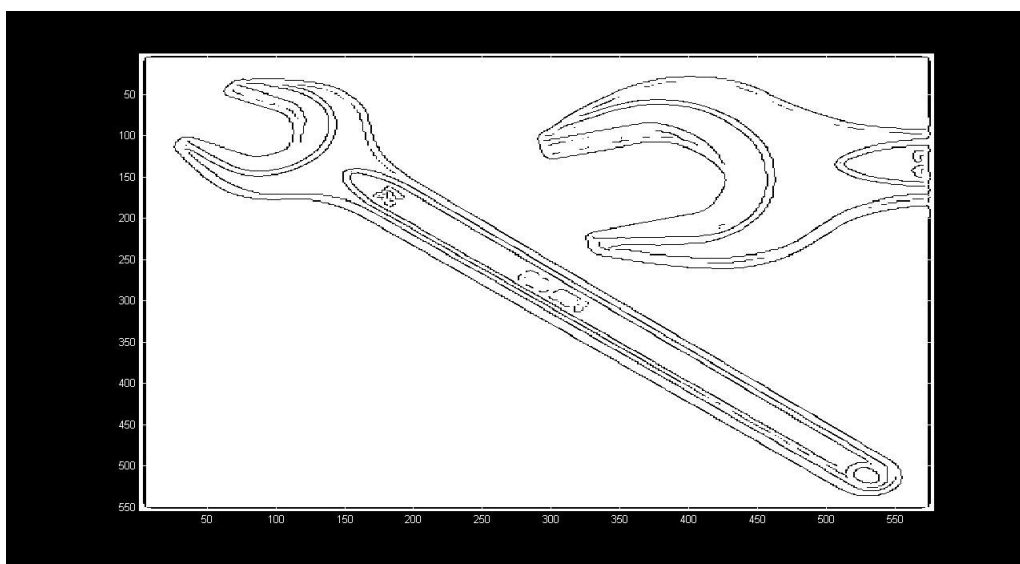
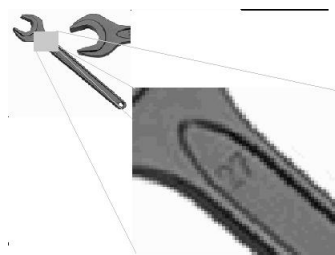
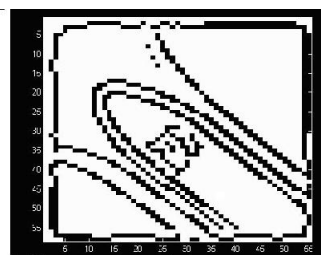


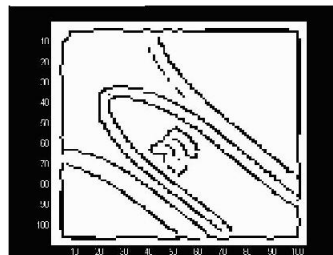
Figura 5.20: detecção com Tensor de Estrutura



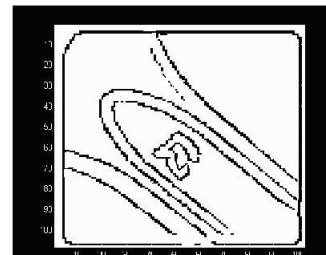
a) detalhe da imagem original



b) detecção amostragem simples



c) Amostragem dobrada



d) tensor de estrutura com filtro não linear

Referências Bibliográficas

- [1] GOMES J., VELHO L., *Computação Gráfica:Imagem. Série de Computação e matemática. IMPA/SBM, Rio de janeiro, 424p, 2002.*
- [2] GONZALEZ R.C., WOODS R.E., *Processamento de Imagens Digitais, editora Blucher, São Paulo, 2000*
- [3] KÖTHE, U., *Edge and Junction Detection an Improved Structure Tensor, Cognitive Systems Lab, University of Hanburg, Germany, 2003*
- [4] KÖTHE U., *Integrated Edge and Junction Detection with the boundary tensor, Cognitive Systems Lab, University of Hamburg, Germany 2003*
- [5] FÖRSTNER W., *A Framework for Low Level Feature Extraction, in: J.-O. Eklundh(Ed.) : ComputerVision – ECCV'94, Vol.II.Springer LNCS801, pp.383 – 394, 1994*
- [6] FÖRSTNER W., *A Feature Based Corresponding Algorithm for Image Matching, Intl. Arch. of Photogrammetry and Remote Sensing, vol. 26, pp. 150-166, 1986*
- [7] HARRIS C., STEVENS M., *A Combined Corner and Edge Detector. Proc. of 4th Alvey Vision Conference, 1988*
- [8] DINIZ P.S.R, SILVA E.A.B., NETTO S.L., *Processamento Digital de Sinais - Porto Alegre, Bookman, 590p, 2004.*
- [9] QUEVEDO C.P, *Cálculo Avançado - Ed.Interciência ltda, 174 p, 2000.*
- [10] H.-H. NAGEL, A. GEHRKE., *Spatiotemporally adaptive estimation and segmentation of OF-Leds; in: H. Burkhardt and B. Neumann (Eds.) : Computer Vision - ECCV'98, Springer LNCS 1407, pp. 86102, 1998*
- [11] J. WEICKERT, T. BROX., *Difusion and Regularization of Vector- and Matrix-Valued Images in: M. Z. Nashed, O. Scherzer (eds.), Inverse*

Problems, Image Analysis, and Medical Imaging. Contemporary Mathematics, Vol. 313, AMS, 2002

- [12] TEIXEIRA, RALPH COSTA, *Introdução aos espaços de escala (Edps em processamento de imagem)* - Rio de Janeiro : IMPA, 2001
- [13] LIMA, ELON LAGES, *Curso de análise, volume 2* - IMPA, CNPQ, 2000.
- [14] BRACEWELL, R. N., *Fourier Transform and Its Applications.* - McGraw-Hill, New York, second edition, 1986.

APÊNDICE

```
%- -%  
%- -----  
%- -%  
%-----algoritmo para detecção simultânea de bordas e-----  
%- -%  
%-----cantos usando o tensor de estrutura com-----  
%- -%  
%-----filtro não linear.-----  
%- -%  
%- -----  
%- -%
```

clear

```
gauss=(1/16)*[1 2 1;2 4 2;1 2 1];  
%I = imread('c:\amazonasp.jpg'); % carrega a imagem em I  
%I = imread('c:\xadrez_.jpg'); % carrega a imagem em I  
%I = imread('c:\placa1.jpg'); % carrega a imagem em I  
%I = imread('c:\x.jpg'); % carrega a imagem em I %  
%I = imread('c:\retangulo.jpg'); % carrega a imagem em I %  
I = imread('c:\cruz.jpg'); % carrega a imagem em I %  
%I = imread('c:\placaPQNA.jpg'); % carrega a imagem em I  
%I = imread('c:\placag.jpg'); % carrega a imagem em I  
%I = imread('c:\letraf.jpg'); % carrega a imagem em I  
subplot(3,2,1); colormap 'gray'  
imagesc(I); %mostra a imagem original%  
title('imagem original');  
%----- metodo implementado-----  
dx= [-1,0,+1;-1,0,+1;-1,0,+1]; % mascara de convolução para  
%encontrar a derivada na direção x%
```

```

gx=conv2(double(dx),double(gauss));
fx = conv2(double(gx),double(I)); % fx= derivada de I na
%direção x%
subplot(3,2,2); imagesc(fx); title('derivada em x');
%-----
dy= [-1,-1,-1;0,0,0;+1,+1,+1]; % mascara de convolução para
%encontrar a derivada na direção y%
gy=conv2(double(dy),double(gauss));
fy = conv2(double(gy),double(I));% fy= derivada de I na
%direção y%
subplot(3,2,3); imagesc(fy); title('derivada em y');
%-----calculo da norma do gradiente da imagem-----
NVI=sqrt(fx.*fx+fy.*fy); subplot(3,2,4);
imagesc(NVI);%imagem da norma do gradiente%
title('Norma do Gradiente');

% -----Thresholding NVI-----

I_max=max(max(NVI)); I_min=min(min(NVI));
level=0.05*(I_max-I_min)+I_min; subplot(3,2,5);
Ibw=max(NVI,level.*ones(size(NVI))); imagesc(Ibw); title('Após o
Thresholding NVI');

%-----Usando interpolação para encontrar os pixels onde as
%normas do-----Gradiente são máximo local -----
[n,m]=size(Ibw); for i=2:n-1, for j=2:m-1,
    if Ibw(i,j) > level,
        X=[-1,0,+1;-1,0,+1;-1,0,+1];
        Y=[-1,-1,-1;0,0,0;+1,+1,+1];
        Z=[Ibw(i-1,j-1),Ibw(i-1,j),Ibw(i-1,j+1);
            Ibw(i,j-1),Ibw(i,j),Ibw(i,j+1);
            Ibw(i+1,j-1),Ibw(i+1,j),Ibw(i+1,j+1)];
        XI=[fx(i,j)/NVI(i,j), -fx(i,j)/NVI(i,j)];%vetor unitario
        YI=[fy(i,j)/NVI(i,j), -fy(i,j)/NVI(i,j)];
        ZI=interp2(X,Y,Z,XI,YI);
        if Ibw(i,j) >= ZI(1) & Ibw(i,j) >= ZI(2);

```

```

        I_temp(i,j)=I_max;
    else
        I_temp(i,j)=I_min;
    end
else
    I_temp(i,j)=I_min;
end
end end subplot(3,2,6); imagesc(I_temp); title('detecção
implementada usando gradiente'); colormap(gray);

%-----dobrando a amostragem nas matrizes fx e fy derivadas-----%
[x,y]=size(I);
fx2=imresize(fx,[2*x,2*y],'bilinear'); %interpolação bilinear.
fy2=imresize(fy,[2*x,2*y],'bilinear'); %interpolação bilinear.
NVI2=sqrt(fx2.*fx2+fy2.*fy2); pause
% -----Thresholding NVI2-----
I_max=max(max(NVI2)); I_min=min(min(NVI2));
level=0.05*(I_max-I_min)+I_min; subplot(3,2,1);
Ibw=max(NVI2,level.*ones(size(NVI2))); imagesc(Ibw); title('Após o
Thresholding NVI2 gradiente definição dobrada');

%-----Usando interpolação para encontrar os pixels onde as
%normas do-----Gradiente são máximo local -----

[n,m]=size(Ibw); for i=2:n-1, for j=2:m-1,
    if Ibw(i,j) > level,
        X=[-1,0,+1;-1,0,+1;-1,0,+1];
        Y=[-1,-1,-1;0,0,0;+1,+1,+1];
        Z=[Ibw(i-1,j-1),Ibw(i-1,j),Ibw(i-1,j+1);
            Ibw(i,j-1),Ibw(i,j),Ibw(i,j+1);
            Ibw(i+1,j-1),Ibw(i+1,j),Ibw(i+1,j+1)];
        XI=[fx2(i,j)/NVI2(i,j), -fx2(i,j)/NVI2(i,j)];%vetor unitario
        YI=[fy2(i,j)/NVI2(i,j), -fy2(i,j)/NVI2(i,j)];
        ZI=interp2(X,Y,Z,XI,YI);
        if Ibw(i,j) >= ZI(1) & Ibw(i,j) >= ZI(2);

```

```

        I_temp2(i,j)=I_max;
    else
        I_temp2(i,j)=I_min;
    end
else
    I_temp2(i,j)=I_min;
end
end end

subplot(3,2,2); imagesc(I_temp2); title('detecção implementada NVI2
gradiente definição dobrada'); colormap(gray);
% -----tensor gradiente

TG11= fx2.*fx2; TG12= fx2.*fy2; TG21= fy2.*fx2; TG22= fy2.*fy2;
% -----tensor de estrutura calculo do filtro não linear

[n,m]=size(TG11); for x=2:n-1;
    for y=2:m-1;

        N=1;
        if fx2(x-1,y-1)==0 &  fy2(x-1,y-1)>0;
            phi1=90;
        end
        if fx2(x-1,y-1)==0 &  fy2(x-1,y-1)<0;
            phi1=270;
        end

        if fx2(x-1,y-1)~=0
            phi1=atand(fy2(x-1,y-1)/fx2(x-1,y-1));
            %phi= angulo do vetor gradiente%
        end
        %-----

        if fx2(x,y-1)==0 &  fy2(x,y-1)>0;
            phi2=90;
        end
        if fx2(x,y-1)==0 &  fy2(x,y-1)<0;

```

```

        phi2=270;

end
    if fx2(x,y-1)~=0
        phi2=atand(fy2(x,y-1)/fx2(x,y-1));
        %phi= angulo do vetor gradiente%
    end
    %-----

if fx2(x+1,y-1)==0 &  fy2(x+1,y-1)>0;
    phi3=90;
end
if fx2(x+1,y-1)==0 &  fy2(x+1,y-1)<0;
    phi3=270;

end
    if fx2(x+1,y-1)~=0
        phi3=atand(fy2(x+1,y-1)/fx2(x+1,y-1));
        %phi= angulo do vetor gradiente%
    end

    %-----

if fx2(x-1,y)==0 &  fy2(x-1,y)>0;
    phi4=90;
end
if fx2(x-1,y)==0 &  fy2(x-1,y)<0;
    phi4=270;

end
    if fx2(x-1,y)~=0
        phi4=atand(fy2(x-1,y)/fx2(x-1,y));
        %phi= angulo do vetor gradiente%
    end
end

```

```

%-----

if fx2(x,y)==0 & fy2(x,y)>0;
    phi5=90;
end
if fx2(x,y)==0 & fy2(x,y)<0;
    phi5=270;
end
if fx2(x,y)~=0
    phi5=atand(fy2(x,y)/fx2(x,y));
    %phi= angulo do vetor gradiente%
end

%-----

if fx2(x+1,y)==0 & fy2(x+1,y)>0;
    phi6=90;
end
if fx2(x+1,y)==0 & fy2(x+1,y)<0;
    phi6=270;
end
if fx2(x+1,y)~=0
    phi6=atand(fy2(x+1,y)/fx2(x+1,y));
    %phi= angulo do vetor gradiente%
end

%-----

if fx2(x-1,y+1)==0 & fy2(x-1,y+1)>0;
    phi7=90;
end
if fx2(x-1,y+1)==0 & fy2(x-1,y+1)<0;
    phi7=270;
end
end

```

```

    if fx2(x-1,y+1)~=0
        phi7=atand(fy2(x-1,y+1)/fx2(x-1,y+1));
        %phi= angulo do vetor gradiente%
    end

%-----

    if fx2(x,y+1)==0 & fy2(x,y+1)>0;
        phi8=90;
    end
    if fx2(x,y+1)==0 & fy2(x,y+1)<0;
        phi8=270;
    end
    if fx2(x,y+1)~=0;
        phi8=atand(fy2(x,y+1)/fx2(x,y+1));
        %phi= angulo do vetor gradiente%
    end

%-----

    phi9=0;
    if fx2(x+1,y+1)==0 & fy2(x+1,y+1)>0;
        phi9=90;
    end
    if fx2(x+1,y+1)==0 & fy2(x+1,y+1)<0;
        phi9=270;
    end
    if fx2(x+1,y+1)~=0;
        phi9=atand(fy2(x+1,y+1)/fx2(x+1,y+1));
        %phi= angulo do vetor gradiente%
    end

%-----

```

```

phi=[phi1;phi2;phi3;phi4;phi5;phi6;phi7;phi8;phi9];%

for b=-1:1;
    for a=-1:1;
        c=1;
    if fx2(x,y)==0 & fy2(x,y)==0;
        ng=[0;0];
        ng2=[0;0];
    else

ng=[-sind(phi(c,1));cosd(phi1(c,1))]
;%vetor n perpendicular ao gradiente

ng2=[cosd(phi1(c,1));sind(phi1(c,1))]
;%vetor n unitario perpendicular ao ng

    end

    %hg=filtro hour-glass

    p=[x+a y+b]*ng;
    q=[x+a y+b]*ng2;

hg=1;

    if p==0 & q~=0;
        hg=0;
    end
    if p~=0;
        hg=(1/N)*exp((-0.5*(x^2+y^2)-0.5*(q/p)^2));
    end
end

```

```

if hg==1
    hg=0;
else
    hg=1;
end

```

```

Hg(a+2,b+2)=hg;
c=c+1;
end

```

```

end

```

```

%-----

```

```

TS1=1/9.*(Hg.*[TG11(x-1,y-1) TG11(x,y-1) TG11(x+1,y-1);
TG11(x-1,y) TG11(x,y) TG11(x+1,y);TG11(x-1,y+1)
TG11(x,y+1) TG11(x+1,y+1)]));

```

```

TS2=1/9.*(Hg.*[TG12(x-1,y-1) TG12(x,y-1) TG12(x+1,y-1);
TG12(x-1,y) TG12(x,y) TG12(x+1,y);TG12(x-1,y+1)
TG12(x,y+1) TG12(x+1,y+1)]));

```

```

TS3=1/9.*(Hg.*[TG22(x-1,y-1) TG22(x,y-1) TG22(x+1,y-1);
TG22(x-1,y) TG22(x,y) TG22(x+1,y);TG22(x-1,y+1)
TG22(x,y+1) TG22(x+1,y+1)]));

```

```

TS11(x,y)=TS1(1,1)+TS1(1,2)+TS1(1,3)+TS1(2,1)+TS1(2,2)
+TS1(2,3)+TS1(3,1)+TS1(3,2)+TS1(3,3);

```

```

TS12(x,y)=TS2(1,1)+TS2(1,2)+TS2(1,3)+TS2(2,1)+TS2(2,2)

```

```

+TS2(2,3)+TS2(3,1)+TS2(3,2)+TS2(3,3);

TS21=TS12;

TS22(x,y)=TS3(1,1)+TS3(1,2)+TS3(1,3)+TS3(2,1)+TS3(2,2)
+TS3(2,3)+TS3(3,1)+TS3(3,2)+TS3(3,3);

%hg=filtro hour-glass valor maximo permitido 765 e valor minino -765

TS=[TS11(x,y) TS12(x,y);TS21(x,y) TS22(x,y)];
[S,D]= eig(TS);

        K1(x,y)=D(1,1);
        K2(x,y)=D(2,2);
    end
end

subplot(3,2,3); K3=sqrt(K1); imagesc(K3);

title('matriz de autovalores sqrt(K1) associado do tensor');
%-----
subplot(3,2,4); K4=sqrt (K2-K1); imagesc(K4); title('matriz de
autovalores sqrt(K2-k1) do tensor'); pause
% -----Thresholding na matriz de autovalores k4=k2-k1 -----
I_max=max(max(K4)); I_min=min(min(K4)); level=0.05*(I_max-I_min);

```

```

subplot(3,2,5); Ibw=max(K4,level.*ones(size(K4))); imagesc(Ibw);
title('Após o Thresholding K4'); fx3 = conv2(double(dx),double(K4))
fy3 = conv2(double(dy),double(K4)) NVI3=sqrt(fx3.*fx3+fy3.*fy3);

%-----Usando interpolação para encontrar os pixels onde as normas do---
%-----Gradiente são máximo local -----
subplot(3,2,6); [n,m]=size(Ibw); for i=2:n-1, for j=2:m-1,
    if Ibw(i,j) > level,
        X=[-1,0,+1;-1,0,+1;-1,0,+1];
        Y=[-1,-1,-1;0,0,0;+1,+1,+1];
        Z=[Ibw(i-1,j-1),Ibw(i-1,j),Ibw(i-1,j+1);
            Ibw(i,j-1),Ibw(i,j),Ibw(i,j+1);
            Ibw(i+1,j-1),Ibw(i+1,j),Ibw(i+1,j+1)];
        XI=[fx3(i,j)/NVI3(i,j), -fx3(i,j)/NVI3(i,j)];%vetor unitario
        YI=[fy3(i,j)/NVI3(i,j), -fy3(i,j)/NVI3(i,j)];
        ZI=interp2(X,Y,Z,XI,YI);
        if Ibw(i,j) >= ZI(1) & Ibw(i,j) >= ZI(2);
            I_temp3(i,j)=I_max;
        else
            I_temp3(i,j)=I_min;
        end
    else
        I_temp3(i,j)=I_min;
    end
end end

subplot(3,2,6); imagesc(I_temp3); title('detecção implementada k4');
colormap(gray);
%-----
%
%
%
%
%-----
% pause
%K5=TS11.*TS22-TS21.*TS12 -0,004.*(TS11+TS22);
%clc
%imagesc(K1);
%title('matriz de autovalores K1 associado do tensor');
%pause

```

```

%imagesc(K2);
%title('imagem da matriz de autovalores K2 do tensor');
%pause
%imagesc(NVI);
%title('gradiente da imagem ');
%pause
%colormap 'gray'
%imagesc(K5);
%title(' matriz de cantos detectados metodo de harris det-0,004*traço A ');

\begin{flushleft}

\end{flushleft}

```