

UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

Projeto de Filtros Digitais Para o Processamento
de Imagens

Marcelino Andrade
Pedro de Azevedo Berger

Prof. Francisco Assis de O. Nascimento

Brasília – DF
Junho – 2000

Resumo

Neste trabalho nós revisamos os problemas de projeto e implementação de filtros digitais para o processamento de imagens. Esta discussão envolve filtros FIR e IIR e com relação ao último, os aspectos de teste e condições de estabilidade são considerados também.

Abstract

The problem of designing and implementing digital filters for the processing of images, are reviewed and discussed. This discussion encompasses both FIR and IIR digital filters and with respect to the latter, the issues of stability testing are also considered.

Lista de Símbolos e Abreviações

DFT – Transformada de Fourier Discreta

FFT – Transforma de Fourier Rápida

FIR – Resposta ao impulso finita

IIR – Resposta ao impulso infinita

SLID – Sistema linear e invariante ao deslocamento

ROC – Região de Convergência

Lista de Figuras

1.1	Quantizador típico.....	4
1.2	Imagem em diferentes tamanho de níveis de cinza.....	5
2.1	Representação de um sistema bidimensional.....	8
2.2	SLID combinados em cascata.....	14
2.3	SLID combinados em paralelo.....	16
2.4	Resposta impulsional espacial e em frequência – Exemplo 2.3.....	18
2.5	ROC de uma seqüência de primeiro quadrante. Qualquer (z'_1, z'_2) na região pintada, também pertence à ROC.....	24
3.1	Filtro Passa-Baixa 2-D – Exemplo 3.2.....	31
3.2	Janela de Hamming 2-D e Resposta em Frequência do filtro 2-D – Exemplo 3.2.....	32
3.3	Janela de Hamming 1-D e Resposta em Frequência do filtro 2-D – Exemplo 3.2.....	33
3.4	Imagem Original, Imagem Filtrada e Resposta em Frequência do Filtro – Exemplo 3.3.....	35
3.5	Imagem Original, Imagem Filtrada e Resposta em Frequência do Filtro – Exemplo 3.4.....	36
3.6	Coeficientes do Filtro e Resposta em Frequência do Filtro – Exemplo 3.5.....	39
3.7	Resposta em Frequência do Filtro – Exemplo 3.5.....	40
3.8	Coeficientes do Filtro e Resposta em Frequência do Filtro – Exemplo 3.6.....	42
3.9	Resposta em Frequência do Filtro – Exemplo 3.6.....	43
3.10	Imagem Original, Imagem com Ruído e Imagem Filtrada – Exemplo 3.7.....	44

3.11	Imagem Original, Imagem Subamostrada e Pré-Filtrada e Imagem Subamostrada sem Pré-Filtragem – Exemplo 3.8.....	46
3.12	Isopotenciais da Função de Transformação – Exemplo 3.9.....	47
3.13	(a) mascara de saída e (b) mascara de entrada, que irão produzir um filtro $h(n_1, n_2)$ com suporte de primeiro quadrante.....	56

Lista de Tabelas

1.1	Tamanhos da figura 1.1 (a), (b), (c) e (d).....	6
1.2	Armazenamento em bits para arquivos de imagens com tamanhos e níveis de cinza variados.....	6

Índice

INTRODUÇÃO	1
1 A IMAGEM DIGITAL	2
1.1 MODELAMENTO SIMPLES DA IMAGEM.....	2
1.2 AMOSTRAGEM E QUANTIZAÇÃO.....	3
1.3 ARMAZENAMENTO, QUALIDADE E FILTRAGEM DA IMAGEM DIGITAL.....	5
2 FUNDAMENTOS MATEMÁTICOS	8
2.1 SISTEMAS MULTIMENSIONAIS.....	8
2.1.1 OPERADORES MATEMÁTICOS FUNDAMENTAIS.....	9
2.2 SISTEMA LINEAR E INVARIANTE AO DESLOCAMENTO.....	11
2.3 O PRINCÍPIO DA CONVOLUÇÃO.....	12
2.4 SISTEMAS EM CASCATA E PARALELO.....	14
2.5 ESTABILIDADE DE SLID.....	16
2.6 RESPOSTA EM FREQUÊNCIA DE UM SLID BIDIMENCIONAL.....	17
2.7 RESPOSTA IMPULSIONAL DERIVADA DA RESPOSTA EM FREQUÊNCIA.....	19

2.8	TRANSFORMADA DE FOURIER BIDIMENSIONAL.....	20
2.9	TEOREMA DE PARSEVAL.....	22
2.10	TRANSFORMA Z BIDIMENSIONAL.....	23
2.11	TEOREMA DE SHANKS.....	23
3	PROJETO DE FILTROS DIGITAIS	26
3.1	PROJETO DE FILTROS FIR.....	26
3.1.1	IMPLEMENTAÇÃO DO FILTRO FIR.....	27
3.1.2	PROJETO DE FILTRO FIR UTILIZANDO O MÉTODO DA JANELA.....	28
3.1.3	PROJETO DE FILTROS POR TRANSFORMAÇÃO.....	37
3.1.4	PROJETO DE FILTROS ÓTIMOS.....	49
3.2	PROJETO DE FILTROS IIR.....	53
3.2.1	IMPLEMENTAÇÃO DE FILTROS IIR.....	53
3.2.2	PROJETO DE FILTROS IIR USANDO EQUAÇÕES DA DIFERENÇA FORMA DIRETA.....	54
4	CONCLUSÕES.....	59

INTRODUÇÃO

As técnicas de processamento digital de imagens, em especial os filtros digitais que é escopo deste trabalho, decorrem principalmente da adequação da imagens para uma melhor interpretação humana e ou de máquinas digitais.

As aplicações de filtros digitais são inerentes as inúmeras áreas do processamento digital de imagens e são diversas as técnicas de projetos de filtros. O universo de estudo de filtros digitais para imagens é muito amplo, e é muitas vezes necessário restringir a abordagem em certos filtros que pertencem a determinadas classes de sistemas que possuem propriedades em comum. A classe dos sistema lineares invariantes ao deslocamento-SLID é a mais freqüentemente estudada em processamento de sinais para qualquer dimensão, esse sistema são de projeto e análise relativamente simples, além de suficientemente poderosos para resolverem muitos dos problemas práticos. Os SLID não pertencem a classe mais geral de sistemas que se pode estudar, mais certamente representam na maioria das aplicações práticas um razoável ponto de partida.

Tratando-se do filtro lineares e invariantes ao deslocamento é possível reuni-los em duas categorias: a primeira chamada de FIR (*Finite Impulse Response*) possuem resposta espacial finita a uma excitação impulsional, e a segunda conhecido com IIR (*Infinite Impulse Response*) é caracterizado por não possuir a resposta finita no espaço como os filtros FIR.

Os conceitos básicos da formação da imagem digital são apresentados no capítulo-1, onde os princípios da amostragem espacial e da quantização em amplitude são descritos na formação da imagem digital. No capítulo-2 são tratado os fundamentos matemáticos que descrevem o processo de filtragem, iniciando com a definição da seqüência bidimensional, de SLID e da convolução linear, e finalizando com os conceitos de estabilidade e resposta em freqüência. O último capítulo é dedicado às técnicas de projetos de filtros digitais, sendo primeiramente abordado os filtro FIR e em seguida os IIR, onde são apresentados diversas técnicas de projeto como: o método da janela, da transformação e ótimo para filtros FIR e o método direto para filtros IIR, além de exemplos de aplicação.

CAPÍTULO 1

A IMAGEM DIGITAL

Este capítulo aborda os principais fundamentos relacionados às imagens digitais monocromáticas. A primeira seção aborda um simples e intuitivo modelo matemático da imagem analógica. A Seção 1.2 apresenta os conceitos de amostragem e quantização na formação da imagem digital. E finalmente na Seção 1.3 é apresentado algumas considerações gerais relacionadas ao armazenamento da imagem digital.

1.1 MODELAMENTO SIMPLES DA IMAGEM

A imagem pode ser interpretada como uma função bidimensional $f(x, y)$, na qual cada coordenada espacial (x, y) está associada a um valor de amplitude positiva de intensidade luminosa, sendo a luz uma forma de energia, implica que $f(x, y)$ além de positiva é finita, ou seja,

$$0 < f(x, y) < \infty \quad (1.1)$$

A função $f(x, y)$ modela uma imagem analógica na qual todos os valores de “ x ” e “ y ” são possíveis e todos os valores de amplitude estão dentro da faixa dinâmica. Tratando-se da imagem monocromática, o valor zero e f_{\max} correspondem às cores branco e negro respectivamente, sendo os valores intermediários considerados níveis de cinza.

1.2 AMOSTRAGEM E QUANTIZAÇÃO

A construção da imagem digital requer necessariamente a digitalização das coordenadas espaciais (x, y) , denominada de amostragem espacial, e da função $f(x, y)$, denominada quantização.

Amostrar uma imagem contínua no espaço consiste em aproximá-la por amostras espaçadas e organizadas na forma matricial $N \times M$ como mostra a equação (1.2), no qual cada elemento é uma quantidade discreta que define a intensidade luminosa pontual:

$$f_d(x, y) = \begin{bmatrix} f(0,0) & f(0,1) & . & . & f(0, M-1) \\ f(1,0) & f(1,1) & . & . & f(1, M-1) \\ . & . & . & . & . \\ . & . & . & . & . \\ f(N-1,0) & f(N-1,1) & . & . & f(N-1, M-1) \end{bmatrix} \quad (1.2)$$

A função f_d é definida como imagem digital não quantizada em amplitude, e os componentes da matriz representam um elemento da imagem ou *pixel* (picture element). A implementação da função f_d é originária da multiplicação da função f contínua por uma matriz de impulsos.

A quantização de uma imagem que possui amplitudes contínuas é realizada pelo mapeamento da variável aleatória contínua de entrada X , intensidade luminosa, em outra variável aleatória $\hat{X} = Q[X]$ de saída que possui amplitudes discretas, a figura (1.1) mostra a representação de um quantizador uniforme com saturação de overflow a partir do quarto nível de quantização.

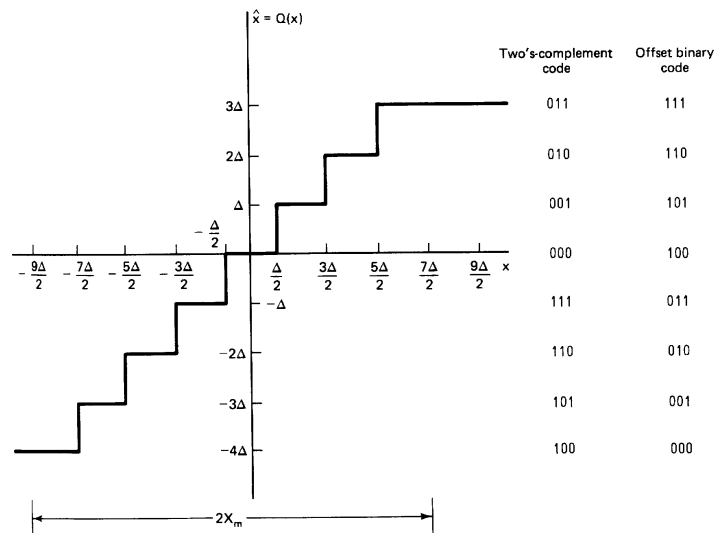


Figura 1.1 – Quantizador típico

A curva do quantizador apresentada é interpretada como a de um conversor A/D (analógico/Digital), no qual a entrada é um sinal com faixa dinâmica contínua $2 \cdot X_m$ e a saída é representada pelos códigos binário ou pelo complemento de dois.

A aplicação da amostragem e da quantização em uma imagem contínua no espaço e na amplitude define a imagem digital, a seguir é apresentado a figura da “lena” amostrada e quantizada em diferentes formatos:



(a) Tamanho 512 x 512 e 256 níveis de cinza (b) Tamanho 85 x 85 e 256 níveis de cinza



(c) Tamanho 512 x 512 e 16 níveis de cinza



(d) Tamanho 85 x 85 e 16 níveis de cinza

Figura 1.2 – Imagem em diferentes tamanho de níveis de cinza

Avaliando as imagens (a) e (b) verifica-se que a diminuição do tamanho da imagem destruiu praticamente todos os seus detalhes, ficando explícito que uma amostragem não adequada tende a rejeitar variações rápidas entre pixel vizinhos, comparando as imagens (a) e (c) observa-se a formação de contornos com a diminuição de níveis de cinza comprometendo relativamente a qualidade da imagem.

1.3 ARMAZENAMENTO, QUALIDADE E FILTRAGEM DA IMAGEM DIGITAL

O efeito da amostragem e da quantização de uma imagem influencia diretamente o tamanho do arquivo de armazenamento da mesma. Assumindo uma imagem de tamanho $N \times M$, G número de níveis de quantização e m o número de bits por pixel, além da consideração prática que estipula aos níveis de quantização o fato de serem múltiplos de dois, tem-se para a quantidade de bits de uma imagem digital sem compactação:

$$G = 2^m, \text{ níveis de quantização}$$

$$b = M \times N \times m, \text{ quantidade de bits da imagem,}$$

$$b = N^2 \times m, \text{ considerando uma imagem quadrada.} \quad (1.3)$$

Por exemplo, para a figura 1.2 que apresenta a mesma imagem com diferentes valores de quantização e amostragem espacial, os tamanhos de arquivo de armazenamento são resumidos na tabela 1.1:

Figura 1.2	Altura	Níveis de cinza	Bits/pixel	Armazenamento
(a)	512X512	256	8	2097152
(b)	85X85	256	8	57800
(c)	512X512	16	4	1048576
(d)	85X85	16	4	115600

Tabela 1.1 – Tamanhos da figura 1.1 (a), (b), (c) e (d)

A tabela 1.2 resume diversos tamanhos de armazenamento de uma imagem quadrada, submetida a variação do número de bits/pixel “m” e do tamanho “N”.

N	NÚMERO DE BITS/PIXEL							
	1	2	3	4	5	6	7	8
32	1.024	2.048	3.072	4.096	5.120	6.144	7.168	8.192
64	4.096	8.192	12.288	16.384	20.480	24.576	28.672	32.768
128	16.384	32.768	49.152	65.536	81.920	98.304	114.688	131.072
256	65.536	131.072	196.608	262.144	327.680	393.216	458.752	524.288
512	262.144	524.288	786.432	1.048.576	1.310.720	1.572.864	1.835.008	2.097.152
1024	1.048.576	2.097.152	3.145.728	4.194.304	5.242.880	6.291.456	7.340.032	8.388.608

Tabela 1.2 – Armazenamento em bits para arquivos de imagens com tamanhos e níveis de cinza variados

O efeito da variação de “N” e “m” sobre o tamanho do arquivo da imagem é modelado pela equação (1.3) e apresenta natureza totalmente objetiva. Ou seja, um aumento de “N” ou “m” implica necessariamente no aumento de bits de armazenamento ou resolução além de esforço computacional. Considerando a qualidade da imagem as características objetivas devem conviver com as subjetivas,

sendo difícil definir o que é uma “boa” imagem sem definir os requisitos de sua aplicação.

Esse capítulo explicitou fundamentos importantes para as abordagens subsequente. O entendimento da imagem digital monocromática apresentados na Seção 1.2, modelada por uma matriz na qual cada elemento representa um pixel, cuja amplitude é a representação da intensidade luminosa, é fundamental para a compreensão das técnicas de filtragem abordadas no capítulo 3. E finalmente o efeito da amostragem e da quantização da imagem possibilitou, como mostrado na Seção 1.3, a visualização da influência no tamanho do arquivo imagem.

CAPÍTULO 2

FUNDAMENTOS MATEMÁTICOS

Os fundamentos matemáticos abordados nesse capítulo são básicos para o entendimento operacional dos filtros tratados no capítulo 3. A primeira seção caracteriza um sistema bidimensional, apresentando os operadores matemáticos fundamentais e uma representação de um sinal bidimensional. A Seção 2.2 e 2.3 descrevem os Sistemas Lineares e Invariantes ao Deslocamento - SLID e o princípio da convolução, sendo importante relatar que os filtros tratados nesse trabalho são todos SLID. As Seções 2.4 e 2.5 abordam as vantagens de se trabalhar com os SLID e define uma condição de estabilidade para esse sistemas. E finalmente nas Seções 2.6 a 2.9 é realizada uma abordagem em frequência dos SLID.

2.1 SISTEMAS MULTIDIMENSIONAIS

Um sistema multidimensional é caracterizado por um operador que efetua o mapeamento de um sinal de entrada x , com mais de uma dimensão, em outro de saída y . A figura 2.1 ilustra esse efeito para uma sistema em duas dimensões, no qual um sinal $x(n_1, n_2)$ de entrada sofre uma transformação $T[\cdot]$ e origina como saída um sinal $y(n_1, n_2)$.

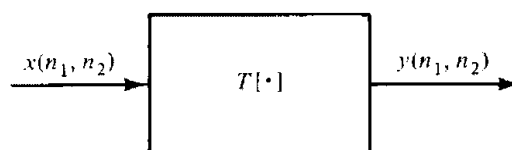


Fig. 2.1 Representação de um sistema bidimensional

O operador $T[\cdot]$ representa a regra ou um conjunto de regras de mapeamento do sinal de entrada no sinal de saída. Podendo por exemplo ser a representação de um filtro passa-baixas aplicado em Processamento Digital de Imagens - PDI, onde $x(n_1, n_2)$ é a imagem original, $y(n_1, n_2)$ a imagem filtrada e $T[\cdot]$ o operador do filtro .

2.1.1 OPERADORES MATEMÁTICOS FUNDAMENTAIS

Uma imagem digital pode ser combinada e alterada de diversas formas com o uso de operadores matemáticos. Essa subseção apresenta operadores matemáticos fundamentais utilizados em PDI e uma representação da seqüência bidimensional.

a) Operador de Adição

Seja $w(n_1, n_2)$ e $x(n_1, n_2)$ as representações de duas imagens ou sinais bidimensionais. A operação de adição é definida pela soma, amostra por amostra, de w e x com a formação de y , a equação (2.1) modela o operador adição.

$$y(n_1, n_2) = x(n_1, n_2) + w(n_1, n_2) \quad (2.1)$$

Uma aplicação usual em imagens desse operador é a extração do valor médio, no qual sendo x a imagem original e w o valor médio negativo de x , tem-se como resultado da soma de x e w um sinal bidimensional y com média zero. Cabe salientar que y não é uma imagem digital e certamente apresentaria valores com amplitudes negativas para uma imagem x com detalhes.

b) Operador Multiplicação

Uma outra operação fundamental é a operação de multiplicação, na qual sendo c uma constante e $x(n_1, n_2)$ uma imagem ou sinal bidimensional, pode-se definir $y(n_1, n_2)$ com a aplicação da multiplicação de cada amostra de x pelo valor constante c , como:

$$y(n_1, n_2) = c \cdot x(n_1, n_2) \quad (2.2)$$

Generalizando o operador multiplicação c , esse pode ser uma seqüência $c(n_1, n_2)$ que possui valores variáveis com n_1 e n_2 .

Uma aplicação interessante do operador multiplicação é realizada quando se deseja visualizar o erro entre duas imagens a e b muito semelhantes. Para isso basta primeiramente efetuar a operação de adição $a + (-b)$ resultando em um sinal bidimensional y com valores próximo de zero, aplicar o operador multiplicação para aumentar a faixa dinâmica de y e por fim somar um valor constante positivo para eliminar os valores negativos.

c) Operador de Deslocamento

Uma imagem 2-D ou sinal bidimensional pode também ser linearmente deslocada no espaço. O operador deslocamento visa esse objetivo e a sua representação matemática é definida na equação (2.3).

$$y(n_1, n_2) = x(n_1 - m_1, n_2 - m_2) \quad (2.3)$$

Utilizando as operações fundamentais de adição, multiplicação e deslocamento é possível decompor qualquer seqüência 2-D pelo somatório infinito apresentado na equação (2.4):

$$x(n_1, n_2) = \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} x(k_1, k_2) \delta(n_1 - k_1, n_2 - k_2) \quad (2.4)$$

Onde, $\delta(n_1 - k_1, n_2 - k_2)$ representa o impulso unitário deslocado com resposta diferente de zero em (k_1, k_2) .

2.2 SISTEMA LINEAR E INVARIANTE AO DESLOCAMENTO

O universo de estudo de processamento de imagens é muito amplo, e é muitas vezes necessário restringir a abordagem em certas classes de operadores que possuem propriedades em comum. A classe dos sistemas lineares invariantes ao deslocamento SLID é a mais freqüentemente estudada em processamento de sinais para qualquer dimensão, esses sistemas são de projeto e análise relativamente simples, além de suficientemente poderosos para resolverem muitos dos problemas práticos. Os SLID não pertencem a classe mais geral de sistemas que se pode estudar, mais certamente representam, na maioria das aplicações práticas, um razoável ponto de partida.

a) SISTEMA LINEAR

Um sistema é chamado linear se e somente forem satisfeita duas condições: se o sinal de entrada é a soma de duas seqüências, o sinal de saída deverá ser a soma das saídas individuais correspondentes, e multiplicando-se a entrada por um valor escalar qualquer, deverá ser agregado à saída esse valor escalar multiplicado. Assumindo $L[\cdot]$ um operador linear tem-se:

$$y_1 = L[x_1]; \quad y_2 = L[x_2];$$

logo,

$$ay_1 + by_2 = L[ax_1 + bx_2] \quad (2.5)$$

b) SISTEMA INVARIANTE AO DESLOCAMENTO

Um sistema invariante ao deslocamento é aquele que quando submetido a uma seqüência de entrada deslocada de n amostras, possui como saída uma

seqüência também deslocada de n amostras, que deve ser igual a um seqüência de saída não deslocada para uma seqüência de entrada não deslocada, ou seja:

$$y(n_1, n_2) = T[x(n_1, n_2)]$$

O sistema $T[\cdot]$ é invariante ao deslocamento se e somente se:

$$y(n_1 - m_1, n_2 - m_2) = T[x(n_1 - m_1, n_2 - m_2)] \quad (2.6)$$

Para toda a seqüência x e qualquer valor de deslocamento (m_1, m_2) .

Linearidade e invariância ao deslocamento são propriedades independentes, dessa forma é possível a existência de sistemas que obedecem a linearidade sem obedecer a invariância ao deslocamento ou o contrário. Os sistemas que possuem as duas propriedades são chamados de Sistemas Lineares e Invariante ao Deslocamento SLID. Os filtros tratados no capítulos 3 são todos pertencentes a essa classe de sistemas.

2.3 O PRINCÍPIO DA CONVOLUÇÃO

A equação (2.4) representa uma seqüência 2-D genérica, aplicando-se um operador linear nessa seqüência tem-se:

$$y(n_1, n_2) = L \left[\sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} x(k_1, k_2) \delta(n_1 - k_1, n_2 - k_2) \right] \quad (2.7)$$

Explorando o princípio da linearidade

$$y(n_1, n_2) = \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} x(k_1, k_2) L[\delta(n_1 - k_1, n_2 - k_2)] \quad \text{ou}$$

$$y(n_1, n_2) = \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} x(k_1, k_2) h_{k_1 k_2}(n_1, n_2) \quad (2.8)$$

Onde $h_{k_1 k_2}(n_1, n_2)$ é a resposta do sistema para um impulso unitário localizado em (k_1, k_2) , que pode ser variante ou invariante ao deslocamento, se o sistema é invariante ao deslocamento algumas simplificações são efetuadas. Seja a resposta espacial impulsional definida por:

$$h_{k_1 k_2}(n_1, n_2) \therefore L[\delta(n_1 - k_1, n_2 - k_2)] \quad (2.9)$$

Para $k_1 = k_2 = 0$, tem-se:

$$h_{00}(n_1, n_2) = L[\delta(n_1, n_2)] \quad (2.10)$$

Aplicando o principio da invariância ao deslocamento definido na equação (2.6)

$$h_{k_1 k_2}(n_1, n_2) = h_{00}(n_1 - k_1, n_2 - k_2) \quad (2.8)$$

Definindo $h(n_1, n_2) = h_{00}(n_1, n_2)$, pode-se escrever a seqüência de saída como:

$$y(n_1, n_2) = \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} x(k_1, k_2) h(n_1 - k_1, n_2 - k_2) \quad (2.9.1)$$

ou fazendo, $n_1 - k_1 = l_1$ e $n_2 - k_2 = l_2$

$$y(n_1, n_2) = \sum_{l_1=-\infty}^{\infty} \sum_{l_2=-\infty}^{\infty} h(l_1, l_2) x(n_1 - l_1, n_2 - l_2) \quad (2.9.2)$$

Essa relação é conhecida como a convolução bidimensional e define que qualquer sistema SLID pode ser complemente caracterizado pela sua resposta espacial impulsional invariante $h(n_1, n_2)$.

2.4 SISTEMAS EM CASCATA E PARALELO

Uma das vantagens de utilização de sistemas lineares invariantes ao deslocamento é a facilidade de análise quando encontram-se combinados na forma em cascata, paralela ou mista. Essa facilidade é oriunda de algumas propriedades da convolução descrita a seguir:

PROPRIEDADES:

- ✓ Comutação, $x ** h = h ** x$
- ✓ Associação, $(x ** h) ** g = x ** (h ** g)$
- ✓ Distribuição, $x ** (h + g) = (x ** h) + (x ** g)$

Exemplo 2.1 – Demostre que os sistemas SLID nas configurações em cascata a e b são equivalentes.

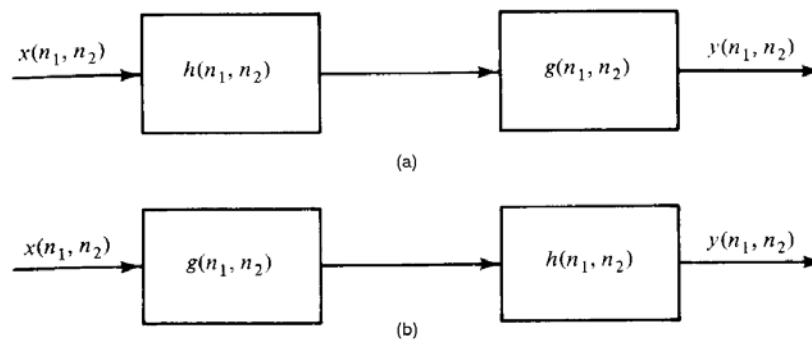


Fig 2.2 – SLID combinados em cascata

Solução:

Para (a)

$$w = x ** h \text{ e } y = w ** g = (x ** h) ** g ,$$

utilizando a propriedade associativa:

$$y = x ** (h ** g) ,$$

a resposta impulsional equivalente de (a) é:

$$h_{equiv} = h ** g$$

Para (b)

$$w' = x ** g \text{ e } y = w' ** h = (x ** g) ** h ,$$

utilizando a propriedade associativa

$$y = x ** (g ** h) ,$$

utilizando a propriedade comutativa

$$y = x ** (h ** g) ,$$

a resposta impulsional equivalente de (b) é:

$$h_{equiv} = h ** g ,$$

Portanto os sistemas são equivalentes

Exemplo 2.2 – Encontre a resposta impulsional equivalente dos sistemas h e g em paralelo.

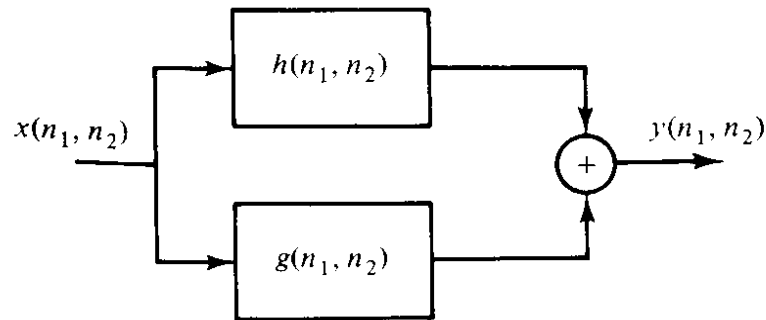


Fig 2.3 – SLID combinados em paralelo

Solução

A saída do sistema é dada por:

$$y = (x ** h) + (x ** g),$$

aplicando a propriedade distributiva

$$y = x ** (h + g),$$

a resposta impulsional equivalente é dada por:

$$h_{equiv} = h + g .$$

2.5 ESTABILIDADE DE SLID

A condição de estabilidade assumida nesse trabalho é a conhecida como BIBO (bounded input, bounded output). Para um sistema BIBO estável, quando a entrada do sistema é limitada $|x(n_1, n_2)| < B$, a saída também deve ser limitada $|y(n_1, n_2)| < B'$ para todo (n_1, n_2) .

A condição necessária e suficiente para o SLID ser BIBO estável, é dada pelo somatório finito da resposta impulsional.

$$\sum_{n_1=-\infty}^{\infty} \sum_{n_2=-\infty}^{\infty} |h(n_1, n_2)| = S_1 < \infty \quad (2.10)$$

2.6 RESPOSTA EM FREQUÊNCIA DE UM SLID BIDIMENCIONAL

Considere um sistema bidimensional com resposta impulsional $h(n_1, n_2)$ e com entrada exponencial complexa.

$$x(n_1, n_2) = \exp(j\omega_1 n_1 + j\omega_2 n_2) \quad (2.11)$$

onde ω_1 e ω_2 são chamadas de frequências verticais e horizontais respectivamente. A saída do sistema pode ser determinada pela convolução de $x(n_1, n_2)$ com $h(n_1, n_2)$

$$\begin{aligned} y(n_1, n_2) &= \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} \exp[j\omega_1(n_1 - k_1) + j\omega_2(n_2 - k_2)] h(k_1, k_2) \\ &= \exp(j\omega_1 n_1 + j\omega_2 n_2) \cdot \left[\sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} h(k_1, k_2) \exp[-j\omega_1 k_1 - j\omega_2 k_2] \right] \\ &= \exp(j\omega_1 n_1 + j\omega_2 n_2) \cdot H(\omega_1, \omega_2) \end{aligned} \quad (2.12)$$

A saída é uma exponencial complexa, com a faixa de frequência igual a do sinal de entrada, mas com a amplitude e a fase alterada pelo ganho complexo $H(\omega_1, \omega_2)$. Esse ganho é conhecido como resposta em frequência do sistema, por definição:

$$H(\omega_1, \omega_2) \triangleq \sum_{n_1} \sum_{n_2} h(n_1, n_2) \exp[-j\omega_1 n_1 - j\omega_2 n_2] \quad (2.13)$$

Exemplo 2.3 Determine a resposta em frequência do SLID que possui resposta impulsional:

$$h(n_1, n_2) = \delta(n_1 + 1, n_2) + \delta(n_1 - 1, n_2) + \delta(n_1, n_2) + \delta(n_1, n_2 + 1) + \delta(n_1, n_2 - 1).$$

Solução:

$$H(\omega_1, \omega_2) = \sum_{n_1} \sum_{n_2} h(n_1, n_2) \exp[-j\omega_1 n_1 - j\omega_2 n_2] = 2 \cdot [\cos(\omega_1) + \cos(\omega_2)] + 1.$$

Código no Matlab:

```

» h(1:11,1:11)=zeros;
» h(5:7,5:7)=ones;
» subplot(2,1,1);
» mesh(-5:5,-5:5,h);
» title('Domínio do Espaço');
» ylabel('Amostras (-5 a 5)');
» xlabel('Amostras (-5 a 5)');
» zlabel('Amplitude');
» w1=-pi:.1:pi;
» w2=-pi:.1:pi;
» [w1,w2] = meshgrid(w1,w2);H=2*(cos(w1)+cos(w2));
» subplot(2,1,2); mesh(w1,w2,H./max(max(H)));
» title('Domínio da Frequência'); ylabel('w2 (-pi a pi)'); xlabel('w1 (-pi a pi)');
» zlabel('Amplitude')

```

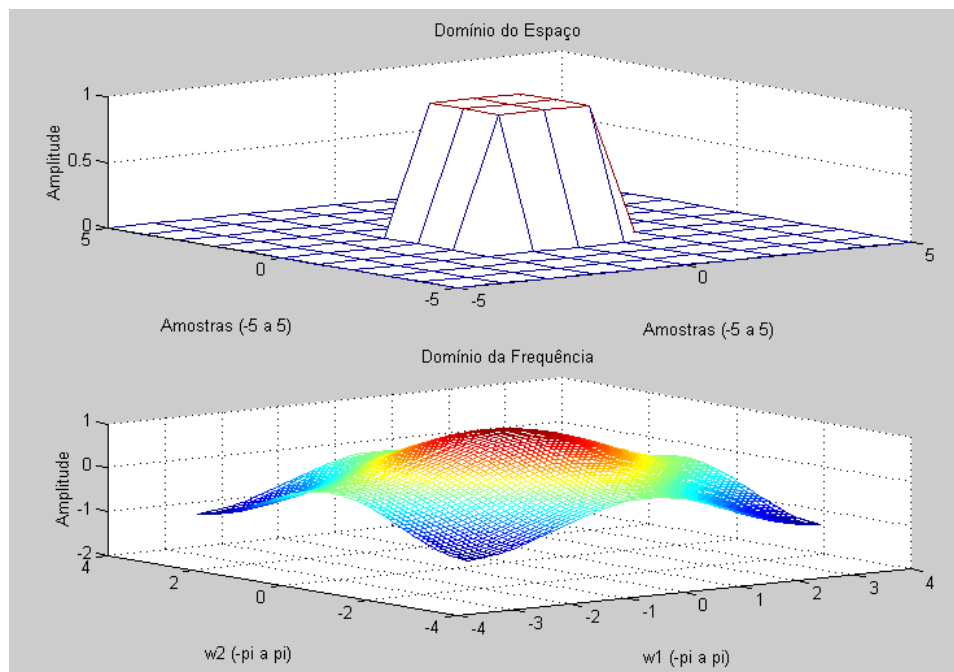


Fig. 2.4 – Resposta impulsional espacial e em frequência – Exemplo 2.3

2.7 RESPOSTA IMPULSIONAL DERIVADA DA RESPOSTA EM FREQUÊNCIA

A resposta em frequência de um SLID 2-D é uma função contínua bidimensional e periódica que pode ser expressa como uma combinação linear de exponenciais complexa como definido na equação (2.13). Os coeficientes dessas exponenciais complexas representam a resposta impulsional do sistema, que são extraídos pela manipulação matemática seguinte:

Multiplica-se ambos os termos da equação (2.13) pela exponenciais complexas $\exp(j\omega_1 k_1 + j\omega_2 k_2)$ e aplica-se a integral em um período completo $-\pi$ a π .

$$\begin{aligned} & \frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} H(\omega_1, \omega_2) \cdot \exp(j\omega_1 k_1 + j\omega_2 k_2) d\omega_1 d\omega_2 \\ &= \frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \sum_{n_1} \sum_{n_2} h(n_1, n_2) \exp(-j\omega_1 n_1 - j\omega_2 n_2) \cdot \exp(j\omega_1 k_1 + j\omega_2 k_2) d\omega_1 d\omega_2 \\ &= \sum_{n_1} \sum_{n_2} h(n_1, n_2) \left\{ \frac{1}{2\pi} \int_{-\pi}^{\pi} \exp[-j\omega_1 (n_1 - k_1)] d\omega_1 \right\} \left\{ \frac{1}{2\pi} \int_{-\pi}^{\pi} \exp[-j\omega_2 (n_2 - k_2)] d\omega_2 \right\} \end{aligned}$$

Efetando a simplificação:

$$\frac{1}{2\pi} \int_{-\pi}^{\pi} \exp[-j\omega(n-k)] d\omega = \delta(n-k), \text{ e escrevendo o resultado em termos de}$$

(n_1, n_2) :

$$h(n_1, n_2) = \frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} H(\omega_1, \omega_2) \cdot \exp(j\omega_1 n_1 + j\omega_2 n_2) d\omega_1 d\omega_2 \quad (2.14)$$

Exemplo 2.4 Determine a resposta impulsional para um filtro passa-baixa ideal, especificado pela resposta em frequência:

$$H(\omega_1, \omega_2) = \begin{cases} 1, & |\omega_1| \leq a < \pi, \quad |\omega_2| \leq b < \pi \\ 0, & \text{otherwise} \end{cases}$$

solução:

$$\begin{aligned} h(n_1, n_2) &= \frac{1}{4\pi^2} \int_{-a}^a \int_{-b}^b \exp(j\omega_1 n_1 + j\omega_2 n_2) d\omega_2 d\omega_1 \\ &= \frac{1}{2\pi} \int_{-a}^a \exp(j\omega_1 n_1) d\omega_1 \\ &\quad \frac{1}{2\pi} \int_{-b}^b \exp(j\omega_2 n_2) d\omega_2 \\ &= \frac{\sin an_1}{\pi n_1} \frac{\sin bn_2}{\pi n_2} \end{aligned}$$

2.8 TRANSFORMADA DE FOURIER BIDIMENSIONAL

A saída espacial de um SLID para uma entrada exponencial complexa foi demonstrado na Seção 2.6 e a resposta espacial $h(n_1, n_2)$ a partir da resposta em frequência foi apresentada na Seção 2.7. Similarmente a (2.14), pode-se representar a entrada $x(n_1, n_2)$ de um sistema pela equação:

$$x(n_1, n_2) = \frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} X(\omega_1, \omega_2) \cdot \exp(j\omega_1 n_1 + j\omega_2 n_2) d\omega_1 d\omega_2 \quad (2.15)$$

onde, $X(\omega_1, \omega_2)$ é uma função complexa de $x(n_1, n_2)$, que pode ser expressa por:

$$X(\omega_1, \omega_2) = \sum_{n_1} \sum_{n_2} x(n_1, n_2) \exp[-j\omega_1 n_1 - j\omega_2 n_2] \quad (2.16)$$

As equações 2.15 e 2.16 são chamadas de transformadas inversa e direta de Fourier respectivamente.

Logo, se aplicarmos uma entrada exponencial complexa em um operador linear $L[\cdot]$ que possui resposta impulsional espacial $h(n_1, n_2)$ e frequencial $H(\omega_1, \omega_2)$, é conhecido da equação 2.12 que:

$$L[\exp(j\omega_1 n_1 + j\omega_2 n_2)] = H(\omega_1, \omega_2) \cdot \exp(j\omega_1 n_1 + j\omega_2 n_2) \quad (2.17)$$

Utilizando a representação de $x(n_1, n_2)$ como uma integral dupla de exponenciais ponderada, equação 2.15, pode-se escrever:

$$y(n_1, n_2) = L[x(n_1, n_2)] = L\left[\frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} X(\omega_1, \omega_2) \cdot \exp(j\omega_1 n_1 + j\omega_2 n_2) d\omega_1 d\omega_2\right]$$

ou ainda, aplicando o princípio da linearidade

$$= \frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} X(\omega_1, \omega_2) \cdot L[\exp(j\omega_1 n_1 + j\omega_2 n_2)] d\omega_1 d\omega_2 \quad (2.18)$$

E finalmente utilizando a equação 2.17, tem-se:

$$\begin{aligned} y(n_1, n_2) &= \frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} X(\omega_1, \omega_2) \cdot H(\omega_1, \omega_2) \cdot \exp(j\omega_1 n_1 + j\omega_2 n_2) d\omega_1 d\omega_2 \\ &= \frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} Y(\omega_1, \omega_2) \cdot \exp(j\omega_1 n_1 + j\omega_2 n_2) d\omega_1 d\omega_2 \end{aligned} \quad (2.19)$$

onde

$$Y(\omega_1, \omega_2) = X(\omega_1, \omega_2) \cdot H(\omega_1, \omega_2) \quad (2.20)$$

O resultado expresso em (2.20) é extremamente importante, e revela que uma saída $y(n_1, n_2)$ no domínio do espaço, representada pela convolução de $x(n_1, n_2)$ e

a resposta impulsional $h(n_1, n_2)$ de um SLID, é representada no domínio das freqüência como um produto entre $X(\omega_1, \omega_2)$ e $H(\omega_1, \omega_2)$.

2.9 TEOREMA DE PARSEVAL

O Teorema de Parseval é extremamente importante em Processamento Digital de Sinais e em especial em PDI, sendo definido para os sistemas bidimensionais como a igualdade descrita na equação (2.21):

Se, $x(n_1, n_2) \leftrightarrow X(\omega_1, \omega_2)$ e $w(n_1, n_2) \leftrightarrow W(\omega_1, \omega_2)$

Então,

$$\sum_{n_1} \sum_{n_2} x(n_1, n_2) w^*(n_1, n_2) = \frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} X(\omega_1, \omega_2) W^*(\omega_1, \omega_2) d\omega_1 d\omega_2 \quad (2.21)$$

Essa equação descreve que o produto interno de duas seqüência 2-D no domínios espacial é igual ao produto interno dessa seqüência no domínio da transformada de Fourier. A equação (2.21) é reduzida à convolução quando

$$w(n_1, n_2) = h^*(m_1 - n_1, m_2 - n_2).$$

Um outro importante conceito extraído do teorema é verificado quando $w(n_1, n_2) = x(n_1, n_2)$, dessa forma tem-se:

$$\sum_{n_1} \sum_{n_2} |x(n_1, n_2)|^2 = \frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} |X(\omega_1, \omega_2)|^2 d\omega_1 d\omega_2 \quad (2.22)$$

A equação 2.22 descreve que a energia total do sinal $x(n_1, n_2)$ é igual a densidade espectral de potência de $X(\omega_1, \omega_2)$, ou seja, a energia de um sinal é totalmente conservada na transformação entre os domínios do espaço e da freqüência.

O objetivo principal deste capítulo foi apresentar os fundamentos matemáticos que visam auxiliar o pleno entendimento do processo de filtragem descrito no capítulo seguinte. O domínio de conceitos como: SLID, resposta impulsional, convolução, representação em frequência e outros tratados neste capítulo, além de fundamentais em PDI, são amplamente utilizados na área de conhecimento de processamento digital de sinais em qualquer dimensão.

2.10 TRANSFORMADA Z BIDIMENSIONAL

Podemos formalmente definir a Transformada Z 2-D de um sinal de entrada discreto $x(n_1, n_2)$ como:

$$X(z_1, z_2) = \sum_{n_2=-\infty}^{\infty} \sum_{n_1=-\infty}^{\infty} x(n_1, n_2) z_2^{-n_2} z_1^{-n_1} \quad (2.34)$$

A soma da equação (2.34) pode não convergir para alguns valores das variáveis complexas z_1 e z_2 . Aqueles valores de z_1 e z_2 , para os quais a soma (2.34) converge, formam a região de convergência (ROC) no hiperplano (z_1, z_2) .

Podemos fazer a transformação inversa usando as integrais da equação (2.35).

$$x(n_1, n_2) = \left(\frac{1}{2\pi} \right)^2 \oint_{c_1} \oint_{c_2} X(z_1, z_2) z_2^{n_2-1} z_1^{n_1-1} dz_2 dz_1 \quad (2.35)$$

Os contornos de ambas as integrais devem estar contidos dentro da região de convergência hiperplano (z_1, z_2) . Eles devem ser fechados e devem envolver a origem no sentido contrário aos ponteiros do relógio, no plano de suas respectivas variáveis.

2.11 TEOREMA DE SHANKS

A estabilidade é uma preocupação inevitável no projeto de filtros IIR, uma vez que estes filtros possuem pólos (ao contrário dos filtros FIR). Portanto devemos estabelecer alguma restrição à resposta ao impulso dos filtros IIR que desejamos projetar, de forma que a estabilidade destes filtros estejam sempre garantidas. Esta restrição pode ser encontrada a partir do teorema de Shanks.

Suponha um sistema cuja transformada Z seja da forma:

$$H(z_1, z_2) = \frac{1}{A(z_1, z_2)} \quad (2.36)$$

O teorema de Shanks estabelece que o sistema, na forma da equação (2.36), será estável, se e somente se, $A(z_1, z_2) \neq 0$ para qualquer $|z_1|, |z_2| \geq 1$. Deste teorema podemos concluir que um filtro IIR com uma região de suporte de primeiro quadrante será sempre estável. Isto porque, a ROC de um sistema de primeiro quadrante é como mostrada na figura (2.1).

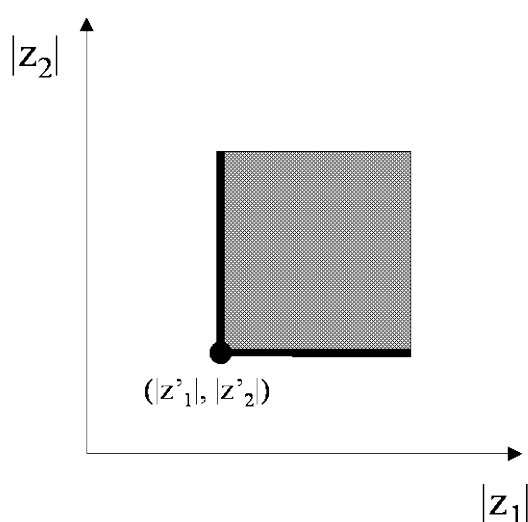


Figura 2.5 – ROC de uma seqüência de primeiro quadrante. Qualquer (z'_1, z'_2) na região pintada, também pertence à ROC.

Observe que a ROC, deste sistema, inclui a superfície unitária. Se aliarmos este fato à propriedade de que a ROC não pode conter superfícies de pólos,

concluimos que os sistemas com resposta ao impulso de primeiro quadrante são sempre estáveis. Portanto, neste trabalho iremos restringir a resposta ao impulso dos filtros IIR a serem projetados como tendo região de suporte de primeiro quadrante. Caso contrário teríamos que incluir na técnica de projeto, critérios para a estabilidade de filtros IIR.

CAPÍTULO 3

PROJETO DE FILTROS DIGITAIS

Esse capítulo descreve diversas técnicas de implementação de filtros para imagens, subdivididas em projetos de filtros FIR e IIR. A Seção 3.1 descreve as técnicas de projeto de filtros digitais FIR (Resposta ao Impulso Finita) e seção 3.2 as técnicas de projeto de filtros IIR (Resposta ao Impulso Infinito).

3.1 PROJETO DE FILTROS FIR

Os filtros FIR possuem resposta impulsional limitada a um número finito de amostras diferentes de zero e de amplitude finita, por consequência são filtros que originam saídas absolutamente estáveis quando a entrada é um impulso.

É freqüentemente desejado em PDI que o filtro aplicado possua uma característica conhecida como fase-zero, pois a resposta de um filtro não fase-zero tende a destruir linhas e bordas da imagem. O entendimento desse efeito é perfeitamente visualizado pela transformada de Fourier da imagem, que descreve a imagem no domínio das freqüências como um somatório de componentes senoidais complexas. Um filtro linear e invariante ao deslocamento com resposta em freqüência não trivial atenua ou amplifica algumas dessas senoides além de atrasar algumas componentes das outras, sendo esse último efeito nocivo às bordas e linhas.

A condição no domínio da freqüência para o filtro possuir fase-zero é expressa por:

$$H(\omega_1, \omega_2) = H^*(\omega_1, \omega_2) \quad (3.1)$$

Calculando a transformada inversa de Fourier da equação (3.1) tem-se a representação do filtro de fase-zero no domínio do espaço, o que traz como consequência uma resposta espacial impulsional perfeitamente simétrica:

$$h(n_1, n_2) = h^*(-n_1, -n_2) \quad (3.2)$$

Cabe salientar que um filtro caracterizado com fase-zero, possui no domínio das frequências uma resposta puramente real e que a simetria no domínio espacial pode ser utilizada para a redução do numero de multiplicações no processo de filtragem.

3.1.1 IMPLEMENTAÇÃO DO FILTRO FIR

A abordagem matemática expressa no capítulo 2, definiu que a saída de um sistema SLID é determinada pela convolução da entrada com a resposta impulsional do sistema, sendo $h(k_1, k_2)$ o filtro FIR e SLID no domínio do espaço e $x(n_1, n_2)$ a representação da imagem original. A saída $y(n_1, n_2)$ é descrita pela convolução expressa na equação (3.3) e representa a imagem filtrada:

$$y(n_1, n_2) = \sum_{k_1} \sum_{k_2} h(k_1, k_2) \cdot x(n_1 - k_1, n_2 - k_2) \quad (3.3)$$

Exemplo 3.1 – A saída de um filtro FIR e SLID localizados na região $\{(n_1, n_2) : 0 \leq n_1 < N_1; 0 \leq n_2 < N_2\}$ pode ser calculada usando:

$$y(n_1, n_2) = \sum_{k_1} \sum_{k_2}^{N_1-1, N_2-1} h(k_1, k_2) \cdot x(n_1 - k_1, n_2 - k_2) \quad (3.4)$$

Obs.: A convolução descrita possui $N_1 N_2$ multiplicações e $N_1 N_2 - 1$ adições para cada amostra $y(n_1, n_2)$.

3.1.2 PROJETO DE FILTRO FIR UTILIZANDO O MÉTODO DA JANELA

O projeto de filtros com janela 2-D é conceitualmente igual ao projeto de filtroS FIR em 1-D, sendo caracterizado como um método no domínio espacial, pois a resposta espacial impulsional é aproximada antes que a resposta em frequência ideal.

O método consiste em definir uma resposta impulsional ideal $i(n_1, n_2)$, quase sempre infinita, que é submetida a um truncamento por via da multiplicação da resposta ideal pela função janela $w(n_1, n_2)$, por fim tem-se a função de transferência com características FIR como verifica-se a seguir:

Seja:

$i(n_1, n_2) \text{ e } I(\omega_1, \omega_2) \rightarrow$ Resposta impulsional e freqüencial do filtro ideal;

$h(n_1, n_2) \text{ e } H(\omega_1, \omega_2) \rightarrow$ Resposta impulsional e freqüencial do filtro FIR a ser projetado;

Logo,

$$h(n_1, n_2) = i(n_1, n_2) w(n_1, n_2) \quad (3.5)$$

onde, $w(n_1, n_2)$ é definida em uma região R finita que limitará a resposta impulsional $h(n_1, n_2)$.

A representação em frequência do filtro é definida pela convolução no domínio transformado:

$$H(\omega_1, \omega_2) = \frac{1}{(2 \cdot \pi)^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} I(\Omega_1, \Omega_2) \cdot W(\omega_1 - \Omega_1, \omega_2 - \Omega_2) d\Omega_1 d\Omega_2 \quad (3.6)$$

onde $W(\omega_1, \omega_2)$ é a transformada de Fourier da função janela.

O truncamento do filtro ideal no espaço implica além da eliminação de amostras fora do domínio da janela em uma suavização do filtro no domínio da frequência

Os parâmetros necessários para a escolha da janela apropriada são inúmeros, entretanto três deles são obrigatórios em todos os projetos de filtros FIR de fase-zero pelo método de janelamento.

- i. A janela deve possuir uma região R finita;
- ii. Para $H(\omega_1, \omega_2)$ aproximar-se de $I(\omega_1, \omega_2)$, $W(\omega_1, \omega_2)$ deve se aproximar-se de uma função impulso 2-D;
- iii. Para $h(n_1, n_2)$ possuir fase-zero, a janela $w(n_1, n_2)$ tem que satisfazer a relação $w(n_1, n_2) = w^*(n_1, n_2)$;

Uma forma interessante de construção da janela 2-D é efetuada a partir da janela em 1-D, sendo os dois métodos clássicos descritos a seguir:

1º Método - Formação da janela 2-D quadrada ou retangular a partir do produto de duas janelas 1-D

$$W(n_1, n_2) = W(n_1) \cdot W(n_2) \quad (3.7)$$

2º Método - Efetuar uma rotação circular na janela 1-D

$$W(n_1, n_2) = W(\sqrt{n_1^2 + n_2^2}) \quad (3.8)$$

Entre os tipos mais utilizados de janelas 1-D, são listados alguns a seguir:

Rectangular

$$w[n] = \begin{cases} 1, & 0 \leq n \leq M, \\ 0, & \text{otherwise} \end{cases}$$

Bartlett (triangular)

$$w[n] = \begin{cases} 2n/M, & 0 \leq n \leq M/2, \\ 2 - 2n/M, & M/2 < n \leq M, \\ 0, & \text{otherwise} \end{cases}$$

Hanning

$$w[n] = \begin{cases} 0.5 - 0.5 \cos(2\pi n/M), & 0 \leq n \leq M, \\ 0, & \text{otherwise} \end{cases}$$

Hamming

$$w[n] = \begin{cases} 0.54 - 0.46 \cos(2\pi n/M), & 0 \leq n \leq M, \\ 0, & \text{otherwise} \end{cases}$$

Exemplo 3.2 – utilizamos a função fwind1 do Matlab para projetar um filtro simétrico circular passa-banda com banda entre 0.2 e 0.7 (frequência normalizada, onde 1.0 corresponde metade da frequência de amostragem) .

1. Implementamos a matriz Hd com a resposta em frequência do filtro ideal. A função freqspace define o espaço em frequência bidimensional f1 e f2.

Código do Matlab:

```
» [f1,f2] = freqspace(21,'meshgrid');
» Hd = ones(21);
» r = sqrt(f1.^2 + f2.^2);
» Hd((r<0.2)|(r>0.7)) = 0;
» mesh(f1,f2,Hd)
```

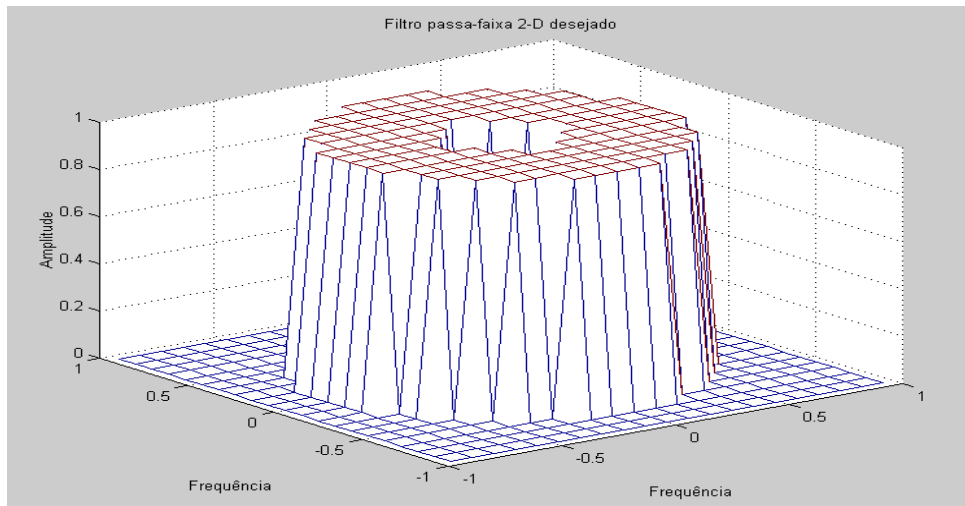


Figura 3.1 – Filtro Passa-Baixa 2-D – Exemplo 3.2

2. Projeto do filtro FIR utilizando a janela de Hamming em duas dimensões

Código do Matlab

```

» subplot(2,1,1)
» hamming2=hamming(21)* hamming(21)';
» mesh(-10:10,-10:10,hamming2)
» title('Janela de Hamming em 2-D')
» xlabel('Amostras')
» ylabel('Amostras')
» zlabel('Amplitude')
» subplot(2,1,2)
» h = fwind2(Hd,hamming2);
» freqz2(h)
» title('Resposta em frequência do filtro 2-D')

```

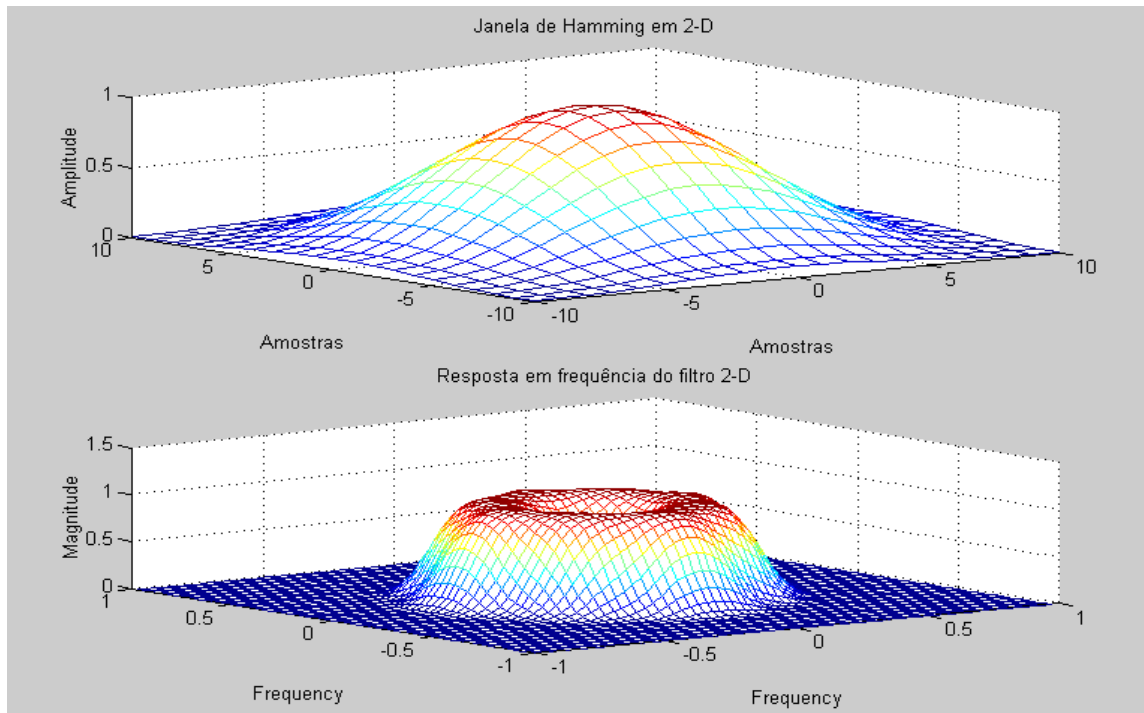



Figura 3.2 – Janela de Hamming 2-D e Resposta em Frequência do filtro 2-D
– Exemplo 3.2

3. Projeto do filtro FIR utilizando a janela de Hamming em 1-D

Código do Matlab:

```

» subplot(2,1,1)
» plot(-10:10,hamming(21))
» grid
» title('Janela de Hamming em 1-D')
» xlabel('Amostras')
» ylabel('Amplitude')
» subplot(2,1,2)
» h = fwind1(Hd,hamming(21));
» freqz(h)
» title('Resposta em frequência do filtro 2-D')

```

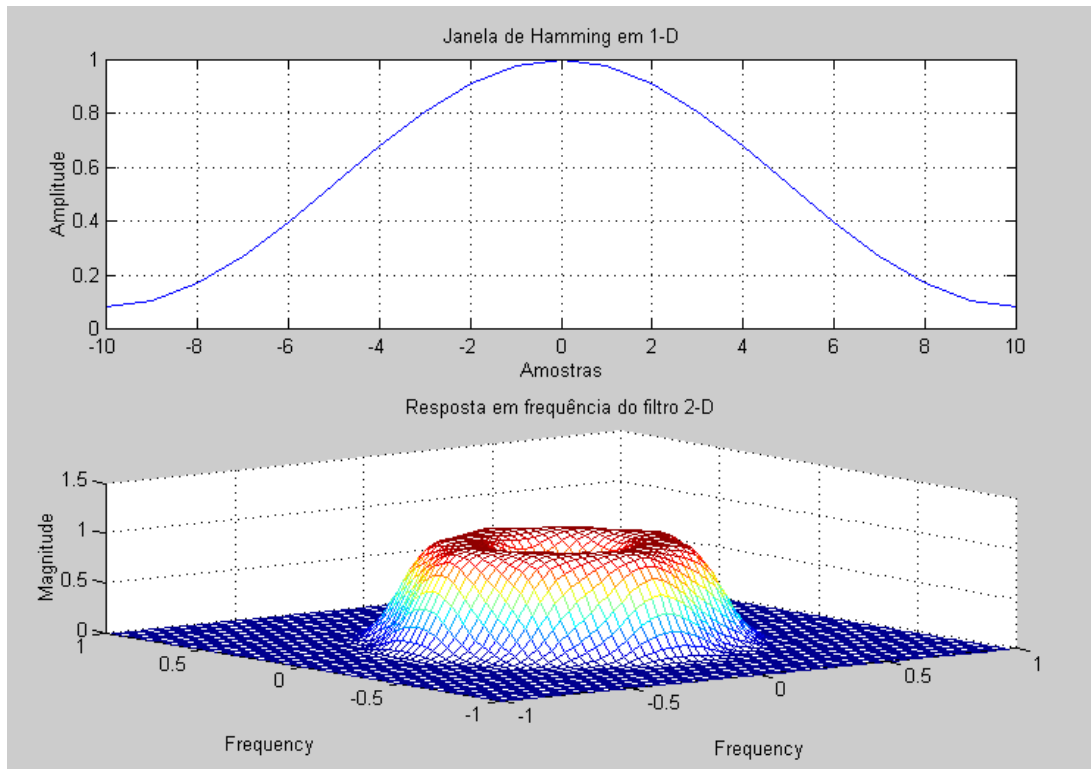


Figura 3.3 – Janela de Hamming 1-D e Resposta em Frequência do filtro 2-D
– Exemplo 3.2

ALGORITIMO

fwind2 calcula h usando a transformada inversa de Fourier de $H_d(\omega_1, \omega_2)$ e multiplicando-o por uma janela bidimensional $w(n_1, n_2)$.

$$h_d(n_1, n_2) = \frac{1}{(2\pi)^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} H_d(\omega_1, \omega_2) e^{j\omega_1 n_1} e^{j\omega_2 n_2} d\omega_1 d\omega_2$$

$$h(n_1, n_2) = h_d(n_1, n_2) w(n_1, n_2)$$

fwind1 utiliza a janela em uma dimensão especificada e a rotaciona formando uma janela bidimensional simétrica (método de Huang).

$$w(n_1, n_2) = w(t) \Big|_{t = \sqrt{n_1^2 + n_2^2}}$$

Utilizando duas janela, fwind1 implementa a janela bidimensional pela multiplicação:

$$w(n_1, n_2) = w_1(n_1)w_2(n_2)$$

finalmente, fwind1 chama fwind2 com Hd e a janela em duas dimensões implementada

Exemplo 3.3 – utilizamos a função fwind2 do Matlab para implementar um filtro simétrico (12 x 12) passa-baixa.

Código do Matlab:

```
» [f1,f2] = freqspace(21,'meshgrid');
» r = sqrt(f1.^2 + f2.^2);
» Hd=ones(21);
» Hd(r>0.1)=0;
» h=fwind2(Hd,boxcar(12)*boxcar(12)');
» B=(filter2(h,lena,'valid'));
» subplot(2,2,1)
» imshow(lena)
» title('Imagem original')
» subplot(2,2,2)
» imshow(uint8(B))
» title('Imagem filtrada')
» subplot(2,1,2)
» freqz2(h)
» title('Resposta em frequência do filtro passa-baixa 12 x 12')
```

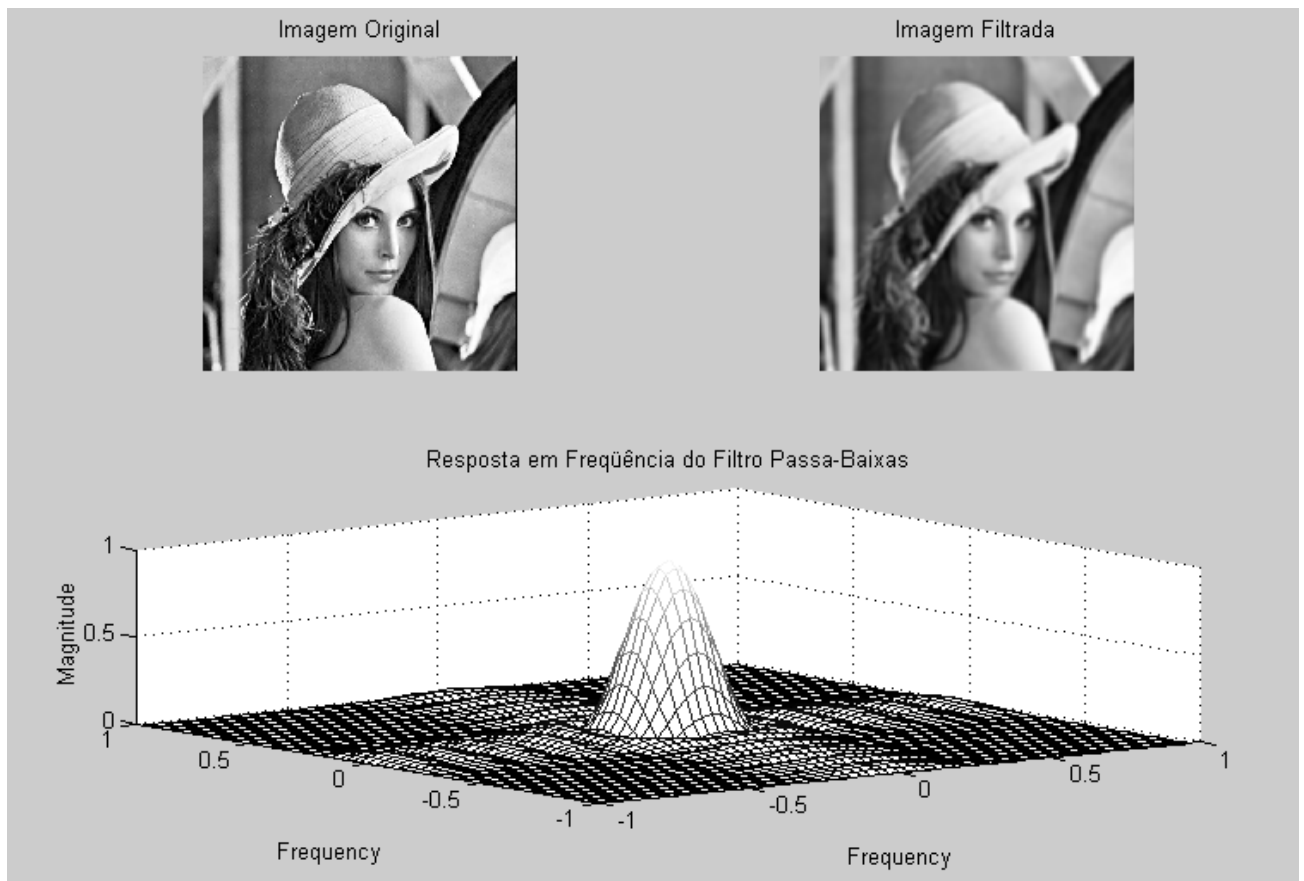


Figura 3.4 – Imagem Original, Imagem Filtrada e Resposta em Freqüência do Filtro
– Exemplo 3.3

Exemplo 3.4 – Utilizamos a função `fwind2` para implementar uma máscara passa-altas simétrica (8 x 8), com frequência de corte $fc=40\%$ (fc corresponde ao percentual da metade da frequência de amostragem).

Código no Matlab

```
» A=imread('Borboleta.bmp');
» [f1,f2] = freqspace(21,'meshgrid');
» r = sqrt(f1.^2 + f2.^2);
» Hd=ones(21);Hd(r<0.4)=0;
» h=fwind2(Hd, hanning (8)* hanning (8)');
» B=(filter2(h,A,'valid'));
» imshow(uint8(B))
```

```

» subplot(2,2,1);
» imshow(A);
» title('Imagem Original')
» subplot(2,2,2);
» imshow(uint8(B));
» title('Imagem Filtrada')
» subplot(2,1,2);
» freqz2(h);
» title('Resposta em Freqüência do Filtro Passa-Altas')

```

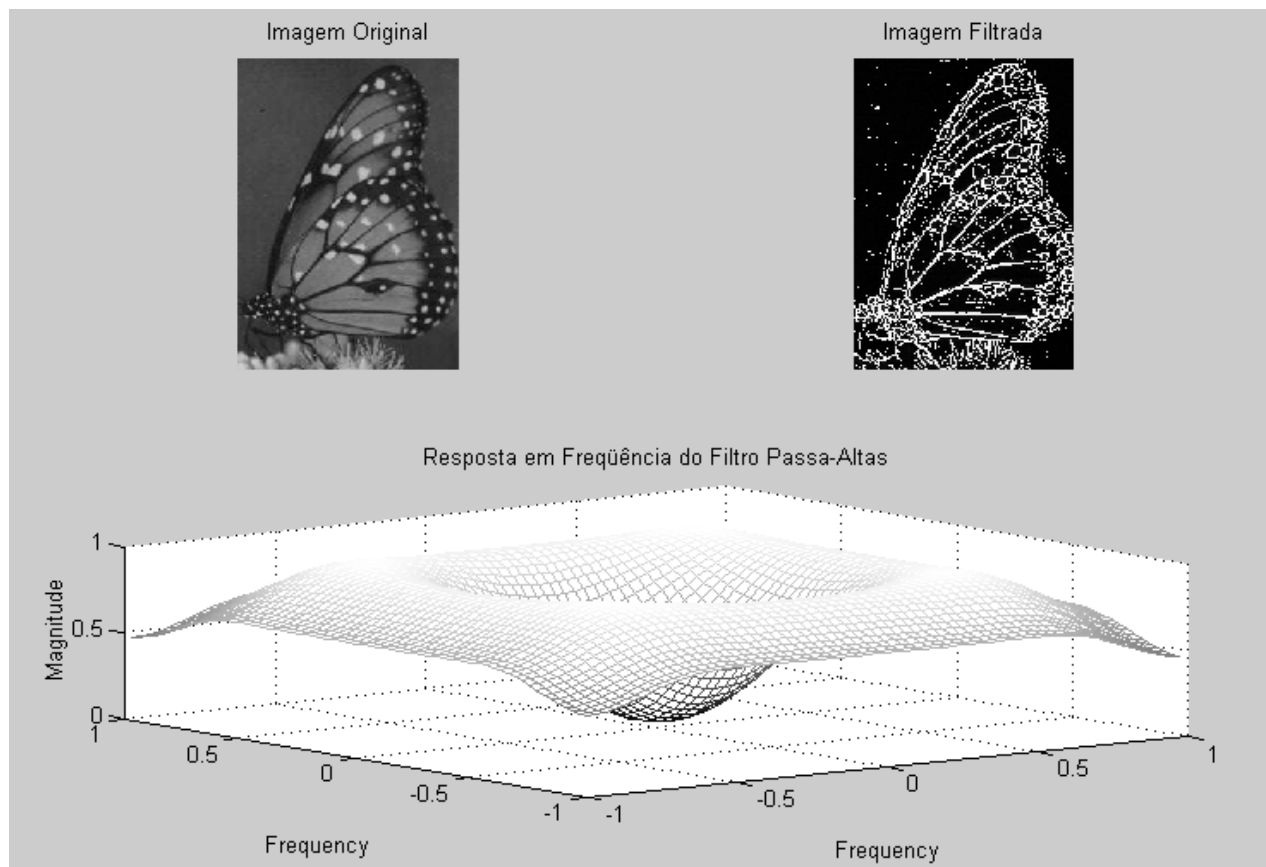


Figura 3.5 – Imagem Original, Imagem Filtrada e Resposta em Freqüência do Filtro
– Exemplo 3.4

3.1.3 PROJETO DE FILTROS POR TRANSFORMAÇÃO

Este método tem como princípio a transformação de um filtro 1-D não recursivo em um filtro 2-D de fase-zero através de uma substituição de variáveis. Desta forma o projeto de um filtro 2-D é subdividido em um projeto de um filtro 1-D e uma transformação. Esta técnica foi introduzida por McClellan (McClellan, 1972), mas tem sido extensamente refinada e modificada nos anos subseqüentes. Uma limitação deste método é que ele só se aplica a projeto de filtros de fase-zero. Entretanto esta limitação não é muito severa, uma vez que a maioria das aplicações requerem filtros de fase-zero. Se a condição de fase-zero não for desejada, então o método de Janela pode ser usado.

A resposta ao impulso de um filtro 1-D de fase zero é hermitianamente simétrica, ou seja:

$$h[n] = h^*[-n] \quad (3.9)$$

Isto implica que este filtro deve ser centrado na origem e ter um número de amostras ímpares. Assuma que o suporte deste filtro seja $[-N, N]$ e sua resposta em frequência seja $H(\omega)$, então podemos escrever:

$$\begin{aligned} H(\omega) &= h(0) + \sum_{n=1}^N h(n) (e^{-j\omega n} + e^{j\omega n}) \\ &= \sum_{n=0}^N a(n) \cos \omega n \end{aligned} \quad (3.10)$$

onde,

$$\begin{cases} a(n) = h(0), & n = 0 \\ a(n) = 2h(n), & n > 0 \end{cases} \quad (3.11)$$

Observe que podemos usar os polinômios de Chebyshev $\{T_n[x]\}$ para expressar $\cos \omega n$ como um polinômio de grau n em $\cos \omega$. Então:

$$\cos \omega n = T_n[\cos \omega] \quad (3.12)$$

e

$$H(\omega) = \sum_{n=0}^N a(n) T_n[\cos \omega] \quad (3.13)$$

Neste ponto realizamos a transformação da resposta em frequência do filtro 1-D de forma a obter a resposta em frequência do filtro 2-D. Isto é feito substituindo $\cos \omega$ por uma função de duas variáveis, chamada função de transformação. Se representarmos a função de transformação por $F(\omega_1, \omega_2)$, então a resposta em frequência do filtro 2-D $H(\omega_1, \omega_2)$ será:

$$H(\omega_1, \omega_2) = \sum_{n=0}^N a(n) T_n[F(\omega_1, \omega_2)] \quad (3.14)$$

Para que o desenvolvimento realizado até agora tenha sentido, precisamos estabelecer as condições para uma boa escolha da função de transformação $F(\omega_1, \omega_2)$. Primeiro $F(\omega_1, \omega_2)$ deve ser a resposta em frequência de um filtro 2-D, para garantir que $H(\omega_1, \omega_2)$ seja também a resposta em frequência de um filtro 2-D. Depois, o comportamento de $H(\omega)$ e $F(\omega_1, \omega_2)$ devem predizer o comportamento de $H(\omega_1, \omega_2)$. E finalmente, deve existir um procedimento para uma escolha particular de $F(\omega_1, \omega_2)$ e $H(\omega)$ para satisfazer o conjunto de especificações em frequência contidas em $H(\omega_1, \omega_2)$. Uma escolha simples para $F(\omega_1, \omega_2)$ proposta por (McClellan, 1972) é:

$$F(\omega_1, \omega_2) = A + B \cos \omega_1 + C \cos \omega_2 + D \cos(\omega_1 - \omega_2) + E \cos(\omega_1 + \omega_2) \quad (3.15)$$

onde A, B, C, D e E são parâmetros livres.

Exemplo 3.5 – utilizamos a função `ftrans2` do Matlab para projetar um filtro simétrico circular passa-baixa com frequência de corte de 0,4 (frequência normalizada, onde 1.0 corresponde metade da frequência de amostragem).

1. Projeto do filtro FIR de fase zero 1-D usado como protótipo

Código do Matlab:

```
» h=fir1(100,0.4);  
» figure(1);  
» subplot(2,1,1);  
» stem(h,'filled');  
» title('Coeficientes do Protótipo Passa-Baixa 1-D');  
» xlabel('n');  
» ylabel('h[n]');  
» H=fft(h);  
» H=fftshift(H);  
» subplot(2,2,2);  
» plot(-pi:pi/50:pi,abs(H));  
» title('Resposta em Frequência do Protótipo Passa-Baixa 1-D');  
» ylabel('Amplitude');  
» xlabel('Frequência');
```

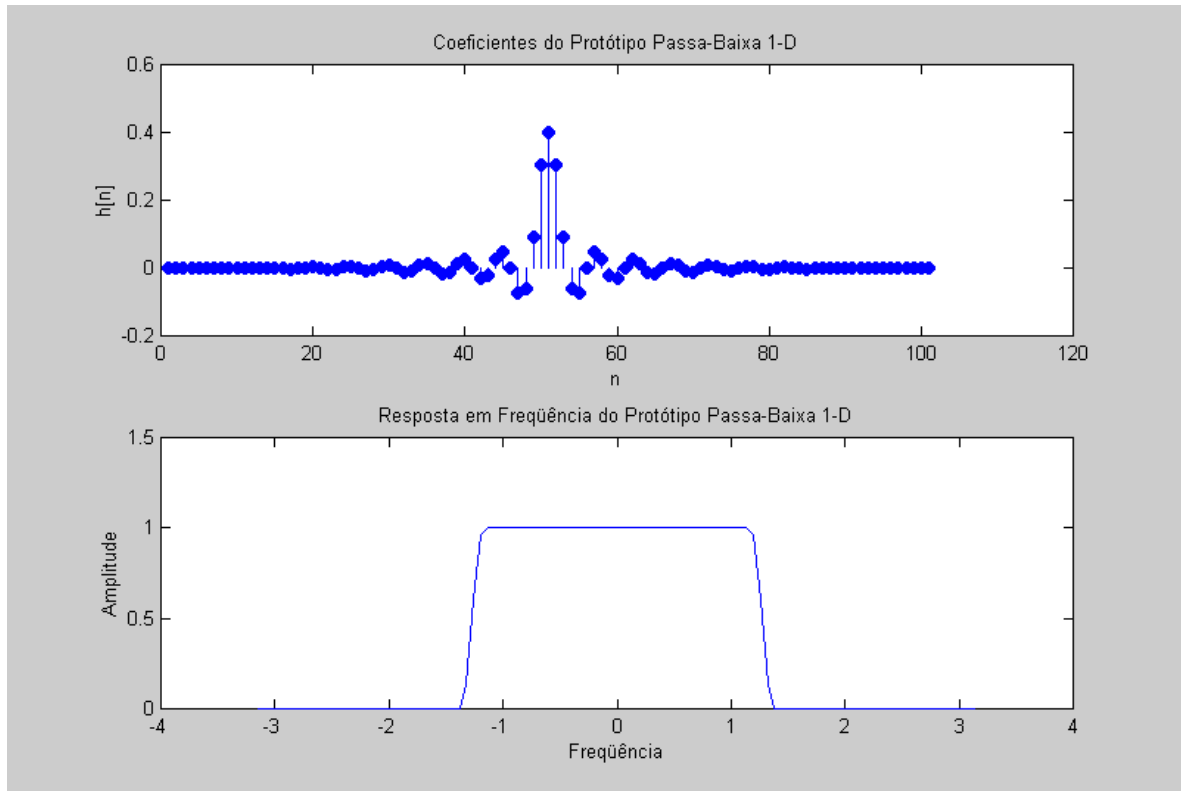


Figura 3.6 – Coeficientes do Filtro e Resposta em Frequência do Filtro – Exemplo

2. Projeto do filtro 2-D baseado no protótipo

Código do Matlab:

```
» f=ftrans2(h);  
» F=fft2(f);  
» F=fftshift(F);  
» figure(2);  
» mesh(-pi:pi/50:pi,-pi:pi/50:pi,abs(F));  
» title('Filtro Passa-Baixa 2-D');  
» zlabel('Amplitude');  
» ylabel('Frequência');  
» xlabel('Frequência');
```

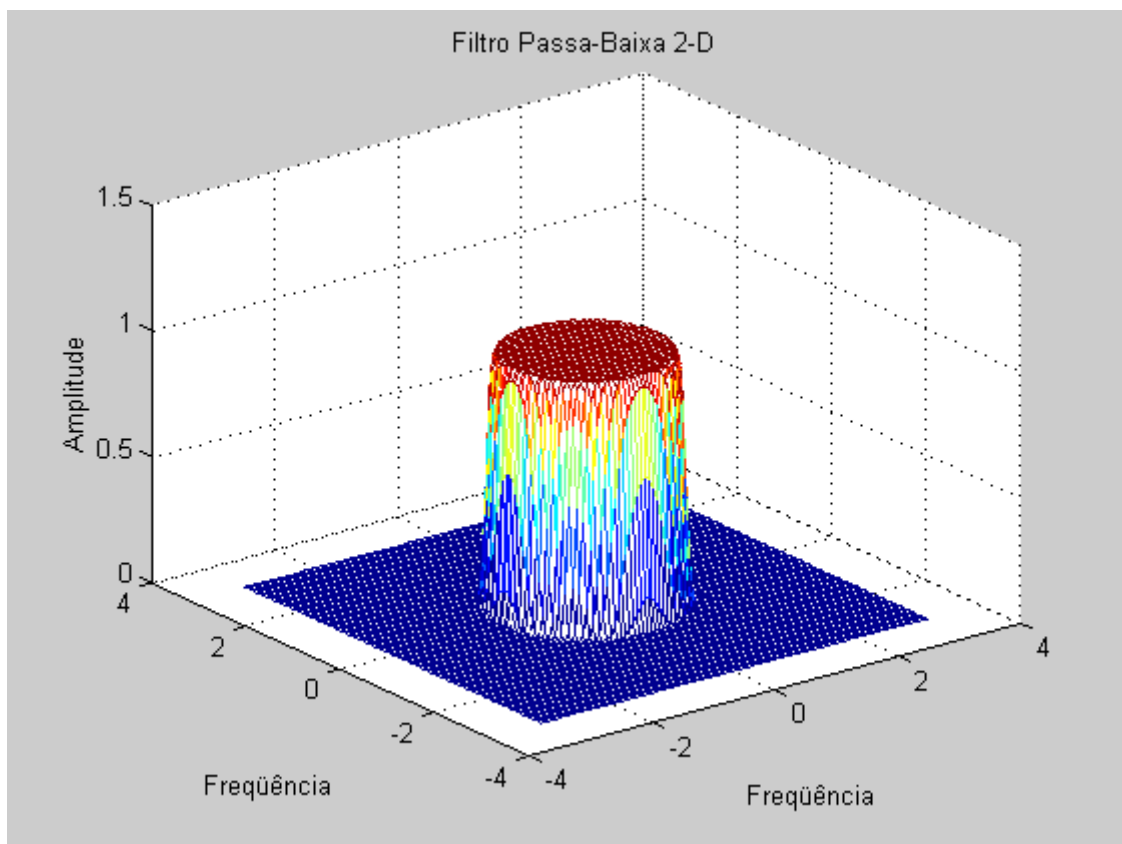


Figura 3.7 – Resposta em Frequência do Filtro – Exemplo 3.5

Exemplo 3.6 – utilizamos a função ftrans2 do Matlab para projetar um filtro simétrico circular passa-banda com banda entre 0.2 e 0.6 (frequência normalizada, onde 1.0 corresponde metade da frequência de amostragem)

1. Projeto do filtro FIR de fase zero 1-D usado como protótipo

Código do Matlab:

```
» h=fir1(100,[0.2 0.6]);
» figure(1);
» subplot(2,1,1);
» stem(h,'filled');
» title('Coeficientes do Protótipo Passa-Banda 1-D');
» xlabel('n');
» ylabel('h[n]');
» H=fft(h);
» H=fftshift(H);
» subplot(2,2,2);
» plot(-pi:pi/50:pi,abs(H));
» title('Resposta em Frequência do Protótipo Passa-Banda 1-D');
» ylabel('Amplitude');
» xlabel('Frequência');
```

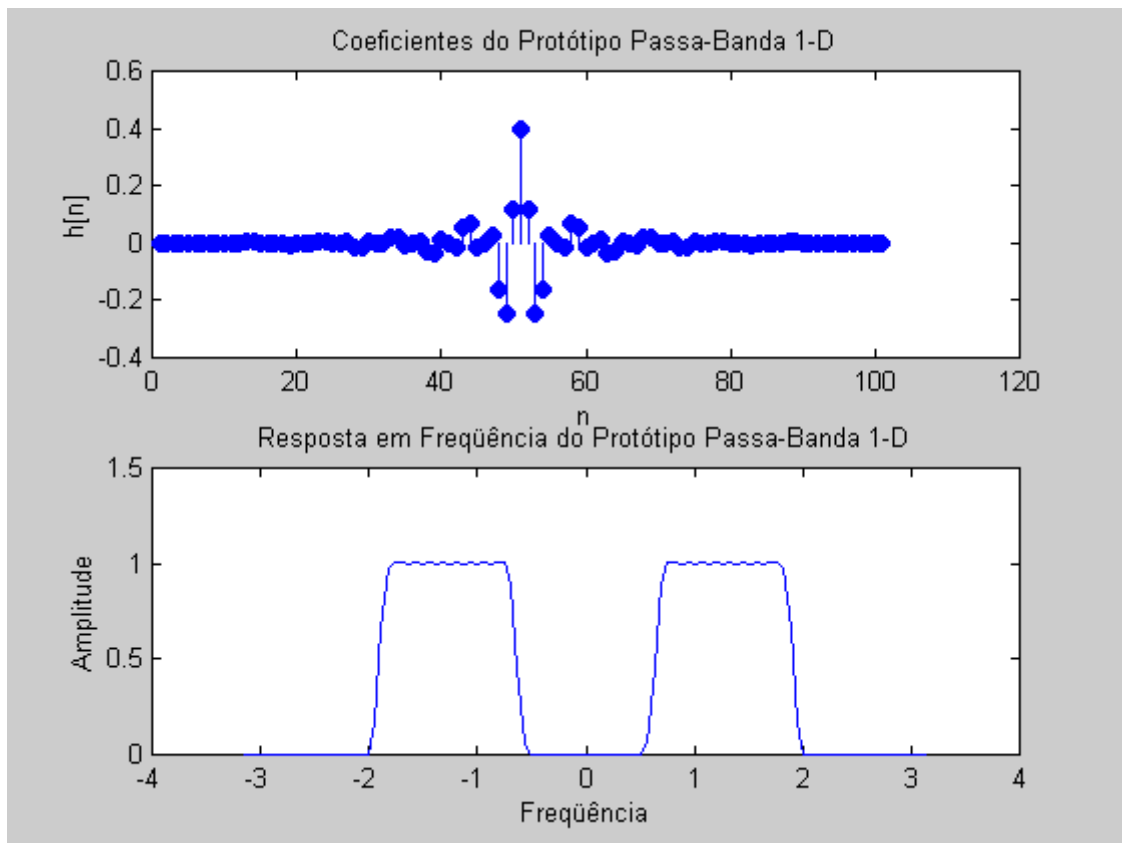


Figura 3.8 – Coeficientes do Filtro e Resposta em Frequência do Filtro – Exemplo 3.6

2. Projeto do filtro 2-D baseado no protótipo

Código do Matlab:

```

» f=ftrans2(h);
» F=fft2(f);
» F=fftshift(F);
» figure(2);
» mesh(-pi:pi/50:pi,-pi:pi/50:pi,abs(F));
» title('Filtro Passa-Baixa 2-D');
» zlabel('Amplitude');
» ylabel('Frequência');
» xlabel('Frequência');

```

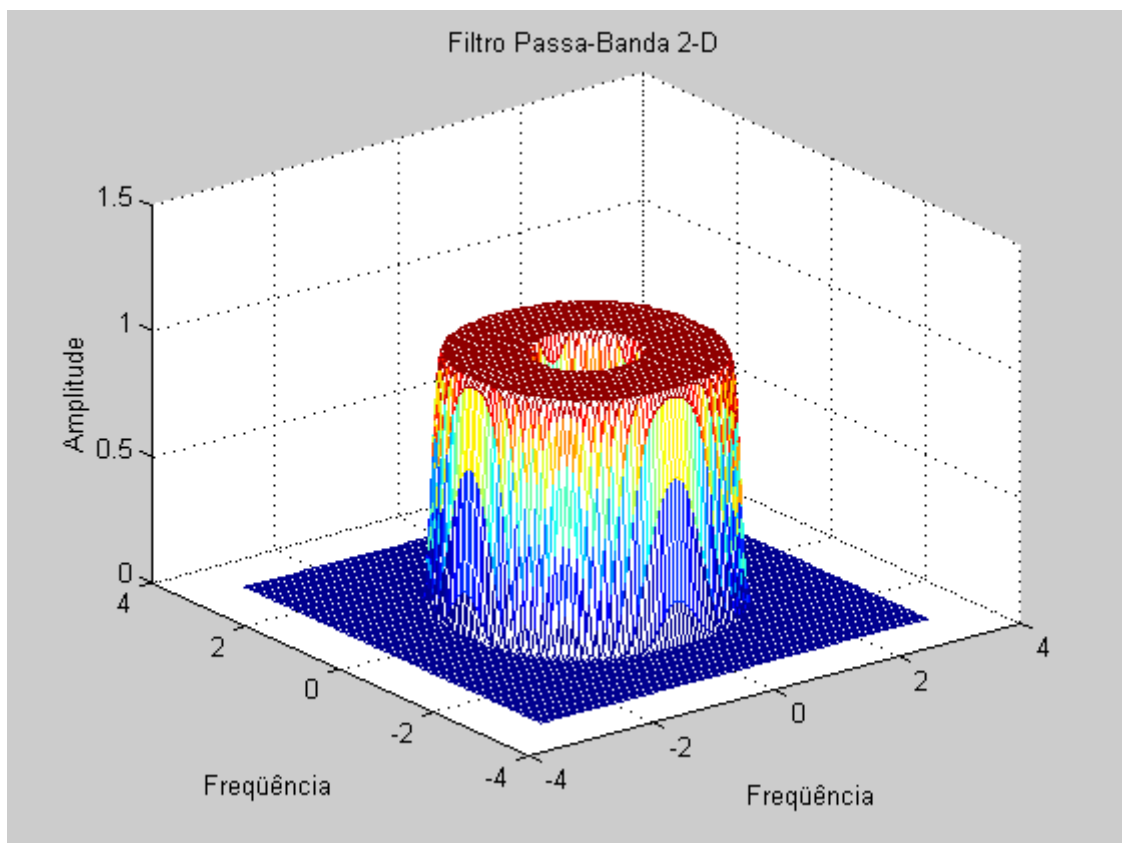


Figura 3.9 – Resposta em Frequência do Filtro – Exemplo 3.6

ALGORITIMO

fir1 implementa um filtro FIR de fase zero usando o método da janela.

ftrans2 transforma o protótipo produzido por fir1, através da transformação mostrada na equação (3.14), usando uma função de transformação $F(\omega_1, \omega_2)$ do tipo mostrada na equação (3.15) com os seguintes parâmetros: $A=B=-1/2$, $C=D=-1/2$ e $E=1/4$.

Exemplo 3.7 – Utilizamos o filtro do exemplo 3.5 para filtrar uma imagem a qual adicionamos um ruído impulsivo.

Código do Matlab

```
» clear;
```

```

» I=imread('lenam','bmp');
» figure(1);
» I=im2double(I);
» J=imnoise(I,'salt & pepper',0.2);
» subplot(2,2,1);
» imshow(I);
» title('Imagem original');
» subplot(2,2,2);
» imshow(J);
» title('Imagem com ruído impulsivo');
» h=fir1(12,0.2);
» f=ftrans2(h);
» e=filter2(f,J,'same');
» subplot(2,1,2);
» imshow(e);
» title('Imagem filtrada');

```

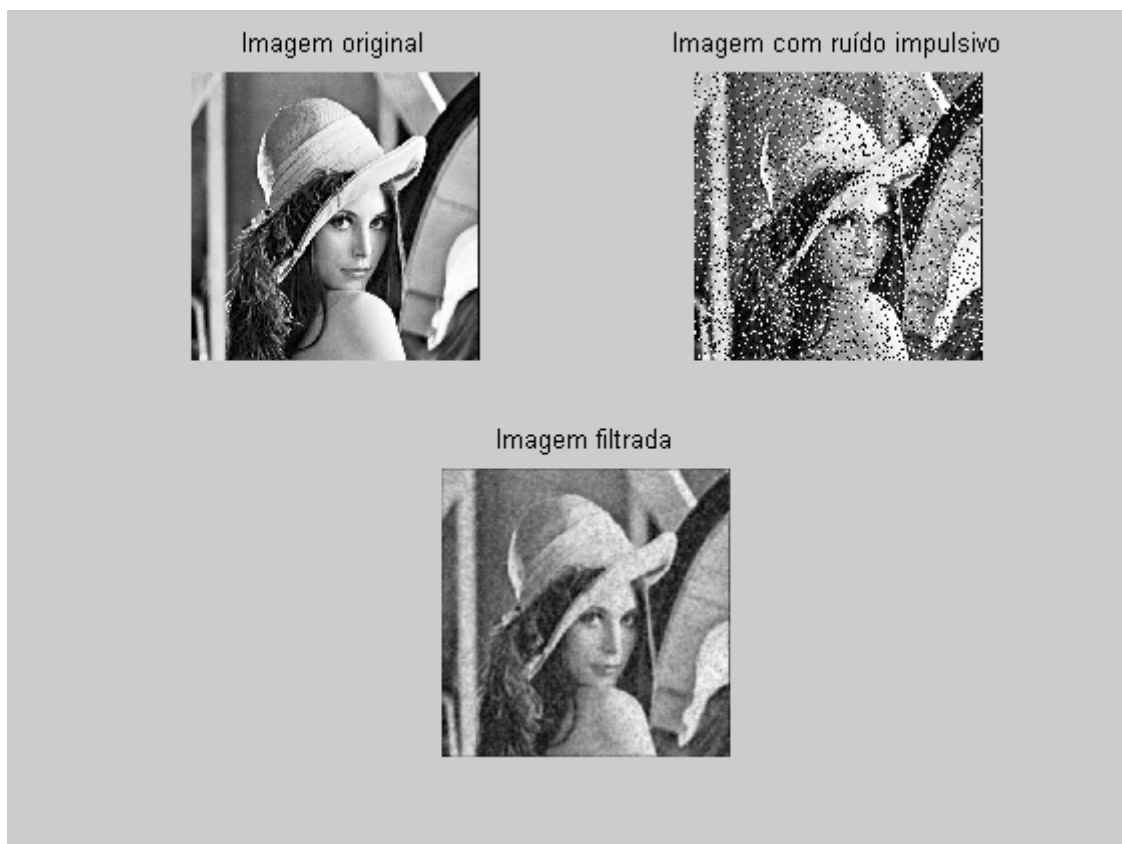


Figura 3.10 – Imagem Original, Imagem com Ruído e Imagem Filtrada– Exemplo 3.7

Exemplo 3.8 – Utilizamos um filtro projetado da mesma forma do exemplo 3.5 com frequência de corte de 0,12 (frequência normalizada, onde 1.0 corresponde metade da frequência de amostragem). Depois subamostramos uma imagem com uma pré-filtragem usando este filtro e comparamos com a subamostragem da mesma imagem sem pré-filtragem.

Código no Matlab

```
» clear;
» I=imread('lenam','bmp');
» figure(1);
» subplot(2,1,2);
» I=im2double(I);
» imshow(I);
» title('Imagem original(512x512)');
» h=fir1(30,0.12);
» f=ftrans2(h);
» J1=imresize(I,0.15,'bicubic',0);
» subplot(2,2,1);
» imshow(J1);
» title('Imagem filtrada e depois subamostrada (76x76)');
» J2=imresize(I,0.15,'bicubic',h);
» subplot(2,2,2);
» imshow(J2);
» title('Imagem subamostrada sem pré-filtragem (76x76)');
```



Figura 3.11 – Imagem Original, Imagem Subamostrada e Pré-Filtrada e Imagem Subamostrada sem Pré-Filtragem – Exemplo 3.8

Exemplo 3.9 – Fizemos no Matlab o gráfico das isopotenciais (contornos) de $F(\omega_1, \omega_2)$.

Código do Matlab:

```

»clear;
»A=-.5;
»B=.5;
»C=.5;
»D=.5;
»warning off;
»for w=-2*pi:.1*pi:2*pi,
»u=-pi:.001*pi:pi;
»v=acos((cos(w)-A-B*cos(u))./(C+D*cos(u)));
»plot(v,u,'k');

```

```

»hold on;
»plot(-v,-u,'k');
» hold on;
»end
»plot(u,pi,'k');
»plot(u,-pi,'k');
»plot(u,0,'k');
»title('Isopotenciais de F(w1,w2)=cosw, para w=0,...,pi');
»xlabel('w1');
»ylabel('w2');

```

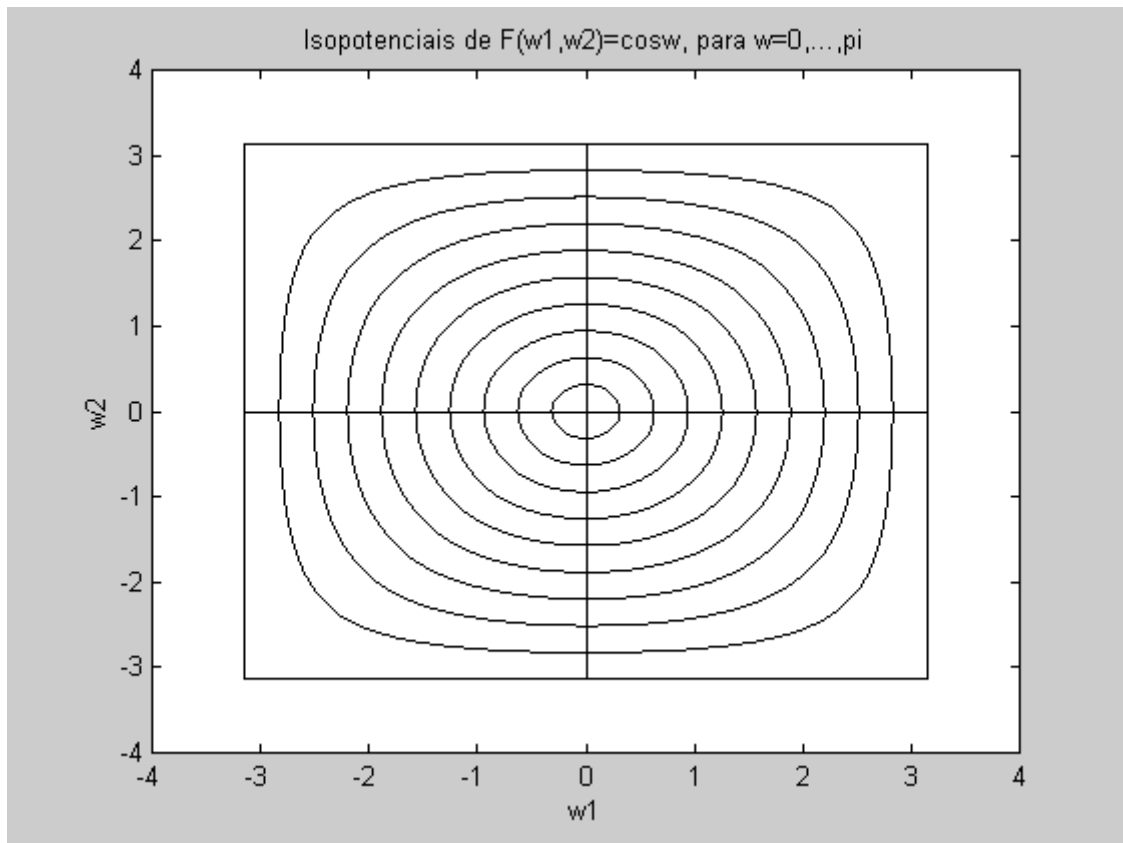


Figura 3.12 – Isopotências da Função de Transformação – Exemplo 3.9

Para obter isopotenciais de $F(\omega_1, \omega_2)$ neste exemplo tomamos a equação (3.15) e igualamos esta equação a $\cos \omega$, com ω variando de 0 a π em passos de

0.1 π . Calculamos ω_2 variando ω_1 de $-\pi$ a π e plotamos (ω_1, ω_2) para os diferentes valores de ω .

Observe que da equação (3.14) a resposta em freqüência do filtro 2-D $H(\omega_1, \omega_2)$ deve ser constante nos contornos de $F(\omega_1, \omega_2)$. E valor que $H(\omega_1, \omega_2)$ irá assumir em cada isopotencial dependerá do protótipo 1-D através dos coeficientes $\{a(n)\}$.

Ainda nos resta o problema de encontrar uma escolha particular de $H(\omega)$ e $F(\omega_1, \omega_2)$ que satisfaça as especificações de uma determinada resposta em freqüência $H(\omega_1, \omega_2)$. Da equação (3.14) podemos escrever:

$$\begin{aligned} H(\omega_1, 0) &= \sum_{n=0}^N a(n) \mathcal{I}_n[F(\omega_1, 0)] \\ &= \sum_{n=0}^N a(n) \mathcal{I}_n[\cos \omega_1] = H(\omega_1) \end{aligned} \quad (3.16)$$

Portanto um corte perpendicular ao plano (ω_1, ω_2) na resposta em freqüência do filtro 2-D deve ser igual à resposta em freqüência do filtro 1-D. E a escolha de $H(\omega)$ a partir das especificações em freqüência do filtro 2-D é direta.

A escolha de $F(\omega_1, \omega_2)$ é um pouco mais complexa e depende de um processo de otimização. Vamos descrever este método através de um exemplo. Suponha que desejamos projetar um filtro passa-banda 2-D com sua banda passante elíptica. Então a borda da banda passante deve aproximar o contorno:

$$\left(\frac{\omega_1}{\omega_{10}} \right)^2 + \left(\frac{\omega_2}{\omega_{20}} \right)^2 = 1 \quad (3.18)$$

Devemos então especificar uma transformação de tal forma que se ela for exata e atravessarmos as isopotenciais de $F(\omega_1, \omega_2)$ transversalmente ao contorno elíptico (3.18), então $F(\omega_1, \omega_2)$ deverá permanecer constante. Logo podemos definir uma função de erro representando a variação de $F(\omega_1, \omega_2)$, de forma que:

$$E(\omega_1) = \cos \omega_0 - F \left(\omega_1, \omega_{20} \sqrt{1 - \left(\frac{\omega_1}{\omega_{10}} \right)^2} \right) \quad (3.19)$$

O erro $E(\omega_1)$ é uma função linear de A,B,C,D e $\cos \omega_0$. Portanto, podemos por exemplo, através de programação linear, determinar estes parâmetros de forma a minimizar o erro quadrático médio.

Ainda existe uma condição para a escolha de $F(\omega_1, \omega_2)$ que negligenciamos até agora. Uma vez que $F(\omega_1, \omega_2)$ substitui $\cos \omega$ na resposta em frequência do filtro 1-D, então ela deve atender:

$$-1 \leq F(\omega_1, \omega_2) \leq 1, \quad 0 \leq \omega_1 \leq \pi, \quad 0 \leq \omega_2 \leq \pi \quad (3.20)$$

Felizmente esta restrição não compromete o desenvolvimento realizado até agora, já que podemos sempre fazer:

$$\begin{aligned} F(\omega_1, \omega_2) &= C_1 F(\omega_1, \omega_2) - C_2, \\ C_1 &= \frac{2}{F_{\max} - F_{\min}}, \\ C_2 &= C_1 F_{\max} - 1 \end{aligned} \quad (3.21)$$

e desta forma atender as restrições.

3.1.4 PROJETO DE FILTROS ÓTIMOS

Sabemos que filtros com suporte temporal finito têm resposta em frequência que somente se aproximam das especificações em frequência ideais. Ou seja, freqüentemente teremos que lidar com um erro entre a resposta em frequência do filtro ideal $I(\omega_1, \omega_2)$ e a do filtro projetado $H(\omega_1, \omega_2)$ que é dado por:

$$E(\omega_1, \omega_2) = H(\omega_1, \omega_2) - I(\omega_1, \omega_2) \quad (3.22)$$

Desta forma, um método ótimo de projeto de filtros seria tentar minimizar alguma função de erro, como por exemplo, o erro médio quadrático:

$$E_2 = \frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} |E(\omega_1, \omega_2)|^2 d\omega_1 d\omega_2 \quad (3.23)$$

Através da relação de Parseval E_2 pode ser expressado no domínio espacial como:

$$\begin{aligned} E_2 &= \sum_{n_1} \sum_{n_2} [h(n_1, n_2) - i(n_1, n_2)]^2 \\ &= \sum_{(n_1, n_2) \in R} [h(n_1, n_2) - i(n_1, n_2)]^2 + \sum_{(n_1, n_2) \notin R} [h(n_1, n_2) - i(n_1, n_2)]^2 \quad (3.24) \\ &= \sum_{(n_1, n_2) \in R} [h(n_1, n_2) - i(n_1, n_2)]^2 + \sum_{(n_1, n_2) \notin R} [i(n_1, n_2)]^2 \end{aligned}$$

De (3.24) podemos concluir que os coeficientes que minimizam E_2 são:

$$h(n_1, n_2) = \begin{cases} i(n_1, n_2), & (n_1, n_2) \in R \\ 0, & (n_1, n_2) \notin R \end{cases} \quad (3.25)$$

que é o filtro que resulta do método de Janela com uma janela constante sobre R .

Num desenvolvimento mais genérico podemos definir $H(\omega_1, \omega_2)$ como :

$$\begin{aligned} H(\omega_1, \omega_2) &= \sum_{i=1}^N a(i) \phi_i(\omega_1, \omega_2), \\ a(i) &= h(n, m) \\ e \\ \phi_i(\omega_1, \omega_2) &= \begin{cases} 2 \cos(\omega_1 n + \omega_2 m), & (n, m) \neq 0 \\ 1, & (n, m) = 0 \end{cases} \end{aligned} \quad (3.26)$$

e minimizar o E_2 tomando suas derivadas com relação a todos os $a(k)$ e igualando-as a zero e resolvendo o sistema de equações. Já que todas as derivadas parciais

são funções lineares dos coeficientes desconhecidos, o problema se resume à solução de N equações lineares. Estas equações podem ser escritas como:

$$\sum_{i=1}^N a(i) \phi_{ik} = I_k, \quad k = 1, 2, \dots, N$$

$$\phi_{ik} = \frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \phi_i(\omega_1, \omega_2) \phi_k(\omega_1, \omega_2) d\omega_1 d\omega_2 \quad (3.27)$$

$$I_k = \frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} I(\omega_1, \omega_2) \phi_k(\omega_1, \omega_2) d\omega_1 d\omega_2$$

Geralmente $\{\phi_i(\omega_1, \omega_2)\}$ são ortogonais, e a solução do sistema de equações lineares se reduz a $a(i) = I_i / \phi_{ii}$. As integrais das equações em (3.27) são normalmente difíceis de serem resolvidas, para contornar este problema podemos usar a abordagem a seguir.

O processo de filtragem no domínio da frequência é baseado na equação (2.20), repetida abaixo:

$$Y(\omega_1, \omega_2) = H(\omega_1, \omega_2) \cdot X(\omega_1, \omega_2) \quad (3.28)$$

onde $X(\omega_1, \omega_2)$ e $Y(\omega_1, \omega_2)$ são as transformadas de Fourier das imagens de entrada e saída, respectivamente, e $H(\omega_1, \omega_2)$ é a função de transferência do filtro ideal que desejamos aproximar. Utilizando o teorema da convolução (Seção 2.3), a equação (3.28) pode ser expressa no domínio espacial por:

$$y(n_1, n_2) = \sum_{l_1=0}^{N-1} \sum_{l_2=0}^{N-1} h(l_1, l_2) x(n_1 - l_1, n_2 - l_2) \quad (3.29)$$

com l_1 e $l_2 = 0, 1, 2, \dots, N-1$.

Como H é a transformada de Fourier de h , segue da equação (2.16) que

$$H(\omega_1, \omega_2) = \sum_{n_1} \sum_{n_2} h(n_1, n_2) \exp[-j\omega_1 n_1 - j\omega_2 n_2] \quad (3.30)$$

utilizando a transformação

$$H(k_1, k_2) = H(\omega_1, \omega_2) \Big|_{\omega_1=2\pi k_1/N_1, \omega_2=2\pi k_2/N_2} \quad (3.31)$$

com k_1 e $k_2=1,2,...N-1$. A equação (3.31) é chamada de transformada discreta de Fourier

$$H(k_1, k_2) = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} h(n_1, n_2) \exp[-j \frac{2\pi}{N_1} n_1 k_1 - j \frac{2\pi}{N_2} n_2 k_2] \quad (3.32)$$

fazendo $h(n_1, n_2)$ igual a zero para $n_1 > \alpha$ e $n_2 > \alpha$ com $\alpha < N$, teremos uma máscara de convolução \hat{h} de tamanho $\alpha \times \alpha$ e transformada discreta de Fourier:

$$\hat{H}(k_1, k_2) = \sum_{n_1=0}^{\alpha-1} \sum_{n_2=0}^{\alpha-1} \hat{h}(n_1, n_2) \exp[-j \frac{2\pi}{N_1} n_1 k_1 - j \frac{2\pi}{N_2} n_2 k_2] \quad (3.33)$$

A geração de máscaras espaciais a partir da especificação em frequência ideal é dada pela equação (3.34), que define os coeficientes ótimos, segundo o critério do erro médio quadrático, para $\hat{h}(n_1, n_2)$.

$$e^2 = \sum_{k_1=0}^{N_1-1} \sum_{k_2=0}^{N_2-1} \left| \hat{H}(k_1, k_2) - H(k_1, k_2) \right|^2 \quad (3.34)$$

A equação (3.33) pode ser expressa na forma matricial:

$$\hat{H} = C \hat{h} \quad (3.35)$$

onde \hat{H} é um vetor coluna de ordem N^2 contendo os elementos de $\hat{H}(k_1, k_2)$, \hat{h} é um vetor coluna de ordem α^2 contendo os elementos de $\hat{h}(n_1, n_2)$, e C uma matriz de ordem $N^2 \times \alpha^2$ de termos exponenciais.

Um procedimento para formação dos vetores \hat{H} , \hat{h} e da matriz C é descrito abaixo.

Seja,

$$\begin{aligned}\hat{H}(k_1, k_2) &\Rightarrow \hat{H}(i), \text{ com } i = 0, 1, 2, \dots, N^2 - 1 \\ \hat{h}(n_1, n_2) &\Rightarrow \hat{h}(j), \text{ com } j = 0, 1, 2, \dots, \alpha^2 - 1\end{aligned}\quad (3.36)$$

com $i = k_1 N + k_2$, para $k_1, k_2 = 0, 1, \dots, N - 1$ e $j = n_1 \alpha + n_2$, para $n_1, n_2 = 0, 1, \dots, \alpha - 1$, são formados os vetores $\hat{H}(i)$ e $\hat{h}(j)$.

Finalmente, os elementos correspondentes à matriz C, denotados por C(i,j), são gerados a partir dos termos exponenciais

$$\frac{1}{N} \exp[-j2\pi(k_1 n_1 + k_2 n_2) / N] \Rightarrow C(i, j) \quad (3.37)$$

com $i = k_1 N + k_2$, para $k_1, k_2 = 0, 1, \dots, N - 1$ e $j = n_1 \alpha + n_2$, para $n_1, n_2 = 0, 1, \dots, \alpha - 1$

Na notação matricial, a equação (3.34) torna-se:

$$\begin{aligned}e^2 &= (\hat{\mathbf{H}} - \mathbf{H})^* (\hat{\mathbf{H}} - \mathbf{H}) \\ &= \|\hat{\mathbf{H}} - \mathbf{H}\|^2 \\ &= \|\mathbf{C}\hat{\mathbf{h}} - \mathbf{H}\|^2\end{aligned}\quad (3.38)$$

onde $*$ é a transposta conjugada e $\|\cdot\|$ é a norma Euclidiana Complexa. Tomando-se a derivada parcial em relação a \hat{h} e igualando-a ao vetor nulo, defini-se o mínimo de e^2 com relação a \hat{h} .

$$\begin{aligned}\frac{\partial e^2}{\partial \hat{h}} &= 2C^* (\hat{C}h - H) = 0 \\ \hat{h} &= (C^* C)^{-1} C^* H \\ &= C^\# H\end{aligned}\tag{3.39}$$

a matriz $C^\# = (C^* C)^{-1} C^*$ é conhecida como a pseudo inversa. A equação (3.39) origina a máscara $\hat{h}(n_1, n_2)$ de ordem $n \times n$ com os coeficientes de erro mínimo da equação (3.34). Os coeficiente de $\hat{h}(n_1, n_2)$ são em geral complexos. Entretanto se o filtro especificado no domínio da freqüência for real e simétrico, $\hat{h}(n_1, n_2)$ também será real e simétrico.

Uma outra função de erro que é freqüentemente usada no projeto de filtros ótimos é a norma de Chebyshev:

$$E_K = \max_{(\omega_1, \omega_2) \in K} |H(\omega_1, \omega_2) - I(\omega_1, \omega_2)|\tag{3.40}$$

onde, K é um subconjunto compacto do plano de freqüência 2-D.

Este tipo de filtro é chamado de “equiripple” e pode ser mostrado que ele minimiza o “ripple” na banda passante e de parada. O projeto deste tipo de filtro pode ser realizado pelo método da transformação sob algumas condições. O método da transformação só pode ser ótimo no sentido da norma de Chebyshev, e isto acontece quando: os contornos da banda de passagem e de parada são isopotenciais da função de transformação, a transformação é de primeira ordem e o protótipo 1-D é ótimo no sentido da norma de Chebyshev.

3.2. PROJETO DE FILTROS IIR

3.2.1. IMPLEMENTAÇÃO DE FILTROS IIR

Na implementação de filtros IIR 1-D, os métodos padrões são a forma direta, forma em cascata e a forma em paralelo. Na implementação de filtros 2-D o único método que pode ser usado recursivamente é a forma direta (a não ser que os outros métodos sejam especificamente inseridos na técnica de projeto) . Na forma direta a equação da diferença que representa o filtro é obtida da função do sistema e é usada recursivamente para calcular a saída do filtro dada uma entrada específica.

3.2.2. PROJETO DE FILTROS IIR USANDO EQUAÇÕES DA DIFERENÇA (FORMA DIRETA)

Um filtro de resposta ao impulso infinita arbitrária $h(n_1, n_2)$ tem sua transformada Z da seguinte forma:

$$H(z_1, z_2) = \frac{\sum_{(k_1, k_2) \in R_B} b(k_1, k_2) z_1^{-k_1} z_2^{-k_2}}{\sum_{(k_1, k_2) \in R_A} a(k_1, k_2) z_1^{-k_1} z_2^{-k_2}} \quad (3.41)$$

No caso 1-D existem dois métodos padrões para o projeto de filtros IIR. O primeiro método é projetar o filtro a partir de um protótipo analógico e o segundo é projetá-lo diretamente. O primeiro método é geralmente muito mais simples do que o segundo. Usando um protótipo analógico elíptico e o método da transformação bilinear, por exemplo, filtros IIR ótimos podem ser projetados seguindo um número finito e fixo de passos. Infelizmente, este método não é útil no caso 2-D, uma vez que o teorema fundamental da álgebra nem sempre é válido e não existem métodos práticos para realizar a Transformada Z Inversa. Portanto, só nos resta explorar o método de projeto direto.

Suponha uma resposta ao impulso $h_d(n_1, n_2)$ e uma resposta em frequência $H_d(\omega_1, \omega_2)$ ideais. podemos estimar os coeficientes $H(z_1, z_2)$ de forma a aproximar em algum sentido $h(n_1, n_2)$ de $h_d(n_1, n_2)$ ou $H(\omega_1, \omega_2)$ de $H_d(\omega_1, \omega_2)$. O critério de erro normalmente usado é o erro médio quadrático, dado por:

$$E_2 = \sum_{n_1=-\infty}^{\infty} \sum_{n_2=-\infty}^{\infty} |h(n_1, n_2) - h_d(n_1, n_2)|^2$$

ou,

$$E_2 = \int_{\omega_1=-\pi}^{\pi} \int_{\omega_2=-\pi}^{\pi} W(\omega_1, \omega_2) \left(|H(\omega_1, \omega_2)| - |H_d(\omega_1, \omega_2)| \right)^2 d\omega_1 d\omega_2$$

onde $W(\omega_1, \omega_2)$ é alguma função de ponderação.

Suponha agora que $h_d(n_1, n_2)$ é dado e desejamos estimar os coeficientes de $H(z_1, z_2)$. Suponha também, que $h_d(n_1, n_2)$ é uma seqüência de primeiro quadrante (condição necessária para a estabilidade do filtro, como foi especificado na seção 2.11). A escolha óbvia para a região de suporte de $h(n_1, n_2)$ é que ela seja também de primeiro quadrante similar á região de suporte de $h_d(n_1, n_2)$. As máscaras de entrada e de saída mostradas na figura (3.13) produzem um filtro $h(n_1, n_2)$ com região de suporte de primeiro quadrante. A escolha das mascaras de entrada e saída de $h(n_1, n_2)$ determinam R_A e R_B na equação (3.41).

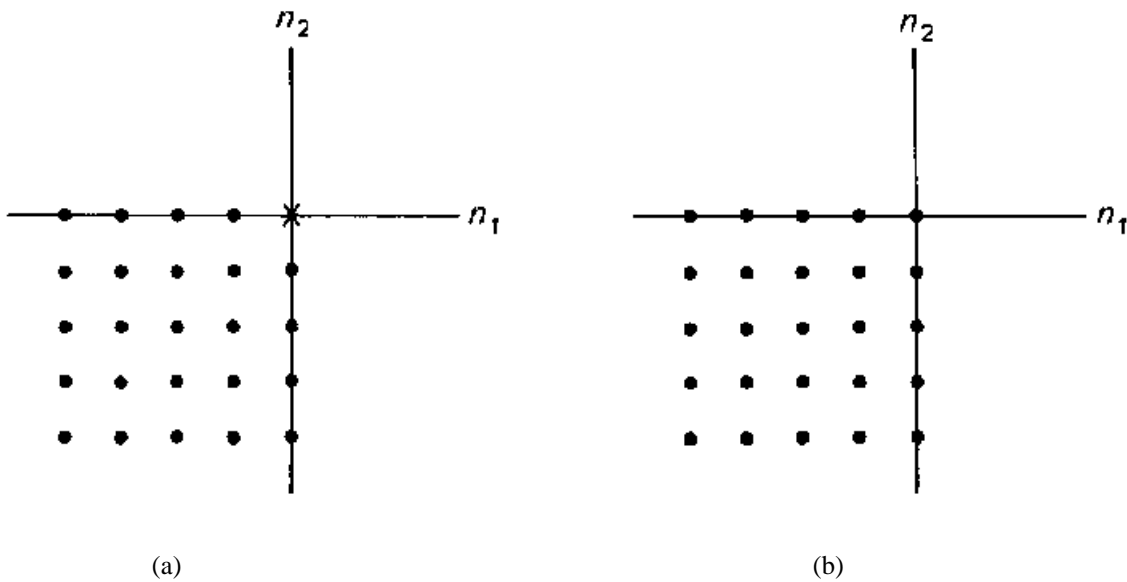


Figura 3.13 – (a) mascara de saída e (b) mascara de entrada, que irão produzir um filtro $h(n_1, n_2)$ com suporte de primeiro quadrante.

A equação da diferença que corresponde a equação (3.41) é dada por:

$$\sum_{(k_1, k_2) \in R_A} a(k_1, k_2) y(n_1 - k_1, n_2 - k_2) = \sum_{(k_1, k_2) \in R_B} b(k_1, k_2) x(n_1 - k_1, n_2 - k_2) \quad (3.43)$$

se fizermos $x(n_1, n_2) = \delta(n_1, n_2)$ e $y(n_1, n_2) = h(n_1, n_2)$ na equação (3.32), teremos:

$$\sum_{(k_1, k_2) \in R_A} a(k_1, k_2) h(n_1 - k_1, n_2 - k_2) = \sum_{(k_1, k_2) \in R_B} b(k_1, k_2) \delta(n_1 - k_1, n_2 - k_2) \quad (3.44)$$

e se substituirmos $h(n_1, n_2)$ por $h_d(n_1, n_2)$ na equação (3.44), não podemos esperar que o lado esquerdo da equação seja igual ao lado direito. Já que desejamos ter $h(n_1, n_2)$ o mais próximo possível de $h_d(n_1, n_2)$, podemos usar o critério de erro:

$$E_2 = \left[\sum_{(k_1, k_2) \in R_A} a(k_1, k_2) h_d(n_1 - k_1, n_2 - k_2) - \sum_{(k_1, k_2) \in R_B} b(k_1, k_2) \delta(n_1 - k_1, n_2 - k_2) \right]^2 \quad (3.45)$$

A solução da equação (3.45) analiticamente é geralmente muito complicada. Uma alternativa à solução analítica é o método de otimização apresentado a seguir, chamado de Steepest Descent. Vamos definir o vetor de parâmetros p formado pelos coeficientes do filtro $\{a(n_1, n_2), b(n_1, n_2)\}$. Definimos também Δp , como sendo perturbações do vetor de parâmetros p . Geralmente algoritmos de otimização são por natureza iterativos. A cada interação eles buscam reduzir o erro E_2 . Nós podemos expressar a dependência do erro com o vetor de parâmetros como $E_2(p)$. Então a cada interação buscamos incrementar o vetor Δp de forma que:

$$\{E_2(p + \Delta p)\}_{k+1} < \{E_2(p)\}_k \quad (3.46)$$

Neste algoritmo. para achar o valor mínimo do erro E_2 , seguimos o procedimento a seguir:

1. Começamos com um valor inicial p_0 , para o vetor de coeficientes, que fornece um chute inicial de onde o ponto de mínimo está localizado na superfície.
2. Usando este chute inicial, calculamos o vetor gradiente do erro médio quadrático $E_2(p)$.
3. Então, o próximo chute é calculado fazendo-se uma mudança no chute inicial na direção oposta do gradiente de erro, calculado no passo anterior. Ou seja, devemos encontrar o vetor de perturbações Δp que somado ao vetor de parâmetros p , resultará num novo vetor de parâmetros que apresenta um erro E_2 menor do que o anterior. Como é mostrado nas equações (3.47) e (3.48):

$$p_{k+1} = p_k + \Delta p = p_k + \frac{1}{2} \mu \left[-\nabla E_2(p) \right] \quad (3.47)$$

logo,

$$\begin{aligned} \Delta p &= -\nabla E_2(p) \\ &= \left[-\frac{\partial E_2}{\partial a(n_1, n_2)}, \dots, -\frac{\partial E_2}{\partial b(n_1, n_2)}, \dots \right]^T \end{aligned} \quad (3.48)$$

onde μ , é um fator de passo usado para calcular a convergência.

Voltamos ao passo 2, com o novo vetor de parâmetro p_{k+1} e começamos tudo de novo

CONCLUSÕES

Neste trabalho tentamos apresentar as técnicas de projeto de filtros digitais para imagens, bem como as ferramentas matemáticas necessárias para a compreensão dessas técnicas. Devido a grande variedade de técnicas e trabalhos que abordam este assunto, não foi possível cobrir todas elas. Mas buscamos relacionar as técnicas mais simples e mais usadas, de forma que elas possam servir de base para o estudo de técnicas mais complexas, como por exemplo, técnicas de filtragem adaptativa. Uma vez que dividimos as técnicas de projeto de filtros em técnicas para filtros FIR e para filtros IIR, é conveniente fazer uma comparação entre as técnicas para esses dois tipos de filtros.

Filtros FIR têm muitas vantagens sobre filtros IIR. Estabilidade não é uma preocupação no projeto e na implementação de filtros FIR. Para filtros IIR, entretanto, testar a estabilidade e estabilizar filtros não estáveis, sem afetar a magnitude da resposta em frequência, é uma tarefa muito trabalhosa. A condição de fase-zero é extremamente fácil de se alcançar em filtros FIR. Projetar filtros IIR de fase-zero é possível, mas é mais complicada do que fazê-lo para filtros FIR. Além disso projetar filtros FIR, como podemos perceber, é bem mais simples do que projetar filtros IIR.

A principal vantagem de um filtro IIR sobre filtros FIR é a redução do número de operações aritméticas, quando este é implementado de forma direta. Porém quando o filtro FIR é implementado usando algoritmos de FFT, o que normalmente acontece, esta vantagem desaparece. Por causa dessas vantagens no projeto e na implementação de filtros FIR, estes são normalmente mais usados do que filtros IIR.

BIBLIOGRAFIA

- Dudgeon, Dan E. "Multidimensional digital signal processing". Prentice-Hall, New Jersey, 1984.
- James H. McClellan, "The Design of Two-Dimensional Digital Filters by Transformation". Proc. 7th Annual Princeton Conf. Information Sciences and Systems (1973) 247-51.
- Oppenheim, Alan V., Schafer, Ronald W. "Discrete-time signal processing". Prentice-Hall, New Jarsey1999.
- Haykin, Simon. "Adaptive filter theory". Prentice-Hall, New Jersey, 1996.
- Huang,T.S. "Two-Dimensional Digital Signal Processing I". Springer-Verlag, New York, 1980.
- Oppenheim A.V., and Jae S. L. "Advanced Topics in Signal Processing". Prentice-Hall, New Jersey, 1988.
- Mersereau R.M., and Dudgeon D.E. "Two-Dimensional Digital Filtering". Proc. IEEE. Vol.63, pp. 610-623, April 1975.