

CENTRO UNIVERSITÁRIO DE ANÁPOLIS
MESTRADO EM SOCIEDADE, TECNOLOGIA E MEIO AMBIENTE

WAGNER OLIVEIRA DE ARAUJO

**ANÁLISE QUANTITATIVA DE SEMENTES DE MAMONA
EMPREGANDO MÉTODOS DE PROCESSAMENTO DE IMAGENS**

WAGNER OLIVEIRA DE ARAUJO

**ANÁLISE QUANTITATIVA DE SEMENTES DE MAMONA
EMPREGANDO MÉTODOS DE PROCESSAMENTO DE IMAGENS**

Dissertação de mestrado apresentada ao Programa de Pós-Graduação em Sociedade, Tecnologia e Meio Ambiente do Centro Universitário de Anápolis como parte dos requisitos para a obtenção do título de Mestre em Sociedade, Tecnologia e Meio Ambiente.

Área de concentração: Tecnologia e Meio Ambiente.

Orientador: Prof. Dr. Clarimar José Coelho

Anápolis
2011

CIP – CATALOGAÇÃO NA PUBLICAÇÃO

A658

Araujo, Wagner Oliveira de

Análise quantitativa de sementes de mamona empregando métodos de processamento de imagens / Wagner Oliveira de Araujo. – Anápolis: Mestrado em Sociedade, Tecnologia e Meio Ambiente da UniEVANGÉLICA, 2011.

87 f.: il.: 23 cm. color.

Dissertação (mestrado) – Centro Universitário de Anápolis. , Anápolis, BR–GO, 2011. Orientador: Clarimar José Coelho.

1. Biodiesel. 2. Sementes de Mamona. 3. Imagem fotográfica. 4. Análise de imagem. 5. Reconhecimento de padrões. I. Coelho, Clarimar José. II. Título.

CDU 504

CENTRO UNIVERSITÁRIO DE ANÁPOLIS

Reitor: Prof. Me. Carlos Hassel Mendes da Silva.

Pró-Reitor de Pós-Graduação, Pesquisa, Extensão.: Prof. Dr. Francisco Itami Campos.

Coordenador Programa de Mestrado Multidisciplinar: Prof. Dr. Roberto Prado de Moraes.

WAGNER OLIVEIRA DE ARAUJO

**ANÁLISE QUANTITATIVA DE SEMENTES DE MAMONA
EMPREGANDO MÉTODOS DE PROCESSAMENTO DE IMAGENS**

Esta Dissertação foi julgada adequada para a obtenção do título de Mestre em Sociedade, Tecnologia e Meio Ambiente e aprovada em sua forma final pelo Orientador e pela Banca Examinadora.


Orientador: _____

Prof. Dr. Clarimar José Coelho, UniEVANGÉLICA
Centro Universitário de Anápolis

Banca Examinadora:



Prof^a Dr^a Telma Woerle Lima Soares, UFG



Prof. Dr. Nilton Correia da Silva, UniEVANGÉLICA

Anápolis, 35 de fevereiro de 2014

Dedico este trabalho a minha avó *In memoriam* Maria José Barbosa, que sempre acreditou que eu podia ser e alcançar o que eu quisesse, e a minha mãe Neide Oliveira de Araujo, pela dedicação e apoio em todos os momentos difíceis.

AGRADECIMENTOS

Ao meu orientador Prof. Dr. Clarimar José Coelho pela confiança depositada, pelo constante apoio e pela orientação.

Ao meu co-orientador, Prof. Dr. Anderson da Silva Soares, pelos ensinamentos, atenção, colaboração em todas as etapas deste trabalho e que muito acrescentou em minha caminhada.

Ao Pesquisador, Prof. Doutorando Gustavo Teodoro Laureano pela colaboração, incentivo e compreensão do tempo despendido para a realização deste trabalho.

Agradeço ao Centro Nacional de Pesquisa de Algodão (CNPA) - Embrapa Algodão do Laboratório de Óleos Vegetais através do Dr. Everaldo Paulo de Medeiros pelas imagens gentilmente cedidas para esta pesquisa.

Eu nunca pude entender, Ilustríssimo Senhor, de onde se originou o fato de que aquilo que dos meus estudos achei conveniente publicar, para agradar ou servir aos outros, tenha encontrado em muitas pessoas uma certa animosidade em diminuir, defraudar e desprezar aquele pouco valor que, se não pela obra, ao menos pela minha intenção eu esperava merecer.

Galileu Galilei

RESUMO

Este trabalho faz a detecção de sementes de mamonas em imagens empregando o processamento digital de imagens. Inicialmente é introduzido um pequeno histórico sobre a evolução da produção de mamona no Brasil e seu contexto no programa de biodiesel brasileiro. É feita uma breve revisão dos principais conceitos sobre o processamento digital de imagens usados no trabalho. O problema da detecção de sementes de mamona é resolvido através de técnicas clássicas do processamento digital de imagens como: suavização em tons de cinza, suavização gaussiana, convolução com máscara, morfologia e abertura, histograma adaptativo e transformada de Hough circular. O processo de segmentação das imagens, empregando a transformada de Hough circular é capaz de identificar 42,74% das sementes presentes na imagem. Finalmente, os resultados são apresentados e discutidos.

Palavras-chave: Biodiesel. Sementes de Mamona. Imagem fotográfica. Análise de imagem. Reconhecimento de padrões.

ABSTRACT

This work is the detection of castor seeds in the images using digital image processing. Initially it is introduced a brief history on the evolution of the production of castor beans in Brazil and its context in the Brazilian biodiesel program. A brief review of the main concepts of digital image processing used in this paper. The problem of detection of castor seeds is solved by classical techniques of digital image processing such as: grayscale smoothing, gaussian smoothing, convolution with a mask, and aperture morphology, adaptive histogram and Hough transform circular. The process of image segmentation, using the circular Hough transform is able to identify 42.74 % of the seeds present in the image. Finally, the results are presented and discussed.

Keywords: Biodiesel. Seeds of Castor oil. Photographic image. Image analysis. Recognition of patterns.

LISTA DE ILUSTRAÇÕES

Figura 2.1:	Imagem obtida com a câmara <i>cyber-shot</i> DSC-P32.	21
Figura 2.2:	Etapas de aplicação dos métodos para a análise da imagem.	22
Figura 3.1:	Imagem em níveis de cinza.	26
Figura 3.2:	Plano de discretização.	26
Figura 3.3:	Discretização espacial.	26
Figura 3.4:	Discretização em amplitude.	26
Figura 3.5:	Cálculo entre a matriz imagem e matriz máscara.	29
Figura 3.6:	Resultado de um célula de convolução.	29
Figura 3.7:	Subárea de uma imagem.	31
Figura 3.8:	Máscara (3×3).	31
Figura 3.9:	Forma da distribuição gaussiana com média zero e desvio padrão σ em 1D. . .	32
Figura 3.10:	Forma da distribuição gaussiana com média zero e desvio padrão σ em 2D. . .	33
Figura 3.11:	Disco.	35
Figura 3.12:	Quadrado.	35
Figura 3.13:	Linha.	35
Figura 3.14:	Conjunto A	36
Figura 3.15:	Conjunto A transladado.	36
Figura 3.16:	Conjunto B	36
Figura 3.17:	Reflexão de B	36
Figura 3.18:	Conjunto A e seu complemento.	36
Figura 3.19:	A diferença entre dois conjuntos.	36
Figura 3.20:	Imagem original.	37
Figura 3.21:	Resultado da dilatação.	37
Figura 3.22:	Imagem original.	38
Figura 3.23:	Resultado da erosão.	38
Figura 3.24:	Imagem original.	39
Figura 3.25:	Resultado da abertura.	39
Figura 3.26:	Imagem original.	40
Figura 3.27:	Resultado do histograma adaptativo.	40
Figura 3.28:	Pontos no espaço imagem.	42
Figura 3.29:	Mapeamento dos pontos no espaço de Hough.	42
Figura 3.30:	Detecção das propriedades de colinearidade.	42
Figura 3.31:	Propriedade de reflexão.	42
Figura 3.32:	Exemplo de um arranjo acumulador (espaço de Hough).	43
Figura 3.33:	Exemplo de um círculo perfeito.	44
Figura 3.34:	Exemplo de uma tangente t e um vetor gradiente \mathbf{g}	44
Figura 3.35:	Circunferência de raio r e centro x_c e y_c	47

Figura 3.36: Plano imagem com 5 <i>pixels</i>	48
Figura 3.37: Circunferência mostrando os <i>pixels</i> no plano de parâmetros.	48
Figura 3.38: Plano de parâmetros de centro x_c, y_c para circunferências.	49
Figura 4.1: Imagem de entrada.	53
Figura 4.2: Suavização em níveis de cinza.	54
Figura 4.3: Bordas detectadas pelo método de <i>Canny</i>	54
Figura 4.4: Segmentação usando <i>watershed</i> morfológicas.	55
Figura 4.5: Filtro espacial com gaussiana, máscara (3×3) e convolução circular.	55
Figura 4.6: Morfologia de abertura.	56
Figura 4.7: Histograma adaptativo.	56
Figura 4.8: Matriz de acumulação de votos com dimensão de 1282×962	57
Figura 4.9: Representação 2D da matriz de acumulação de votos.	58
Figura 4.10: Imagem com círculos localizados (centros x_c, y_c).	58
Figura 4.11: Círculos secantes.	61
Figura 4.12: Círculos tangenciais externos.	61
Figura 4.13: Círculos concêntricos.	61
Figura 4.14: Recorte da Figura 4.10 dos pontos localizados pela transformada de Hough circular que apresentam anomalia.	61
Figura 4.15: Detecção com marcação através de numeração nas sementes de mamona.	62

LISTA DE TABELAS

Tabela 1:	Acurácia do Experimento.	62
-----------	----------------------------------	----

LISTA DE QUADROS

- 1 Aproximação discreta de uma distribuição gaussiana com desvio padrão $\sigma = 1,4$. 33
- 2 Máscara em 1D utilizada para calcular a convolução em 2D mais eficientemente. . 34

LISTA DE ABREVIATURAS

1D	Imagem de uma dimensão
2D	Imagem de duas dimensões
COS	Coseno de um ângulo qualquer
DNA	<i>Deoxyribonucleic Acid</i> (ADN, em português: ácido desoxirribonucleico)
JPEG	<i>Joint Photographic Experts Group</i>
MATLAB	<i>MATrix LABORatory</i> (Plataforma de computação científica adotada neste trabalho)
PIXEL	<i>Picture versus Element</i>
SEN	Seno de um ângulo qualquer
TG	Tangente de um ângulo qualquer
TH	Transformada de Hough
USB	<i>Universal Serial Bus</i>

LISTA DE SÍMBOLOS

Escalares são representados em itálico; vetores e sequências em minúsculo e negrito e matrizes, em maiúsculo e negrito.

$(\hat{B})_x$	Reflexão do conjunto B e translação dessa reflexão por x
$(A)_x$	Translação de A por $x = (x_1, x_2)$
Δ_1	Círculo perfeito
\hat{B}	Reflexão do conjunto B
\mathbb{R}	Conjunto numérico dos números reais
\mathbb{Z}	Conjunto numérico dos números inteiros
\mathbf{R}	Matriz de resultados da transformada de Hough circular dos raios
\mathbf{XY}	Matriz de resultados da transformada de Hough circular dos pontos x e y
$\bar{\mu}$	Nível de cinza médio
$\bar{\mu}_g$	Média global
ψ	Fator de ganho local
ρ	Ângulo formado entre a tangente t e o eixo horizontal
σ	Desvio padrão computados em uma vizinhança
\mathbf{g}	Vetor gradiente
θ	Coefficiente angular que o segmento de reta r faz com o eixo x da abscissa
A	Conjunto de elementos
a	Elemento de um conjunto
A^c	O complemento do conjunto A
B	Conjunto B ou elemento estruturante
b	Elemento de um conjunto
B^c	O complemento do conjunto B
c	Dimensão de um vetor
c'	Dimensão de um vetor
d	Distância entre dois pontos
e	Exponencial

$f(x, y)$	Função representativa de imagem digital x, y coordenadas no plano cartesiano
$G(x)$	Suavização gaussiana aplicada em uma dimensão
$G(x, y)$	Suavização gaussiana aplicada em duas dimensões
i	Número unário
I	Representação da luminosidade monocromática
k	Constante pertencente ao intervalo
L	Denota o número de níveis de cinza
m	Número de linhas de uma matriz
m'	Número de linhas de uma matriz
n	Número de colunas de uma matriz
n'	Número de colunas de uma matriz
n_k	É o k -ésimo número de <i>pixels</i> com intensidade r_k
NM	Representação de arco da circunferência
O	Centro da circunferência
$p(r_k)$	Função discreta de estimativa de probabilidade de ocorrência do nível de cinza
r	Raio do círculo
r_k	Representa o k -ésimo nível de cinza
rad	Denota radiano
T	Período de uma matriz
t	Período de uma matriz
T'	Período de uma matriz
t'	Período de uma matriz
w_i	É o i -ésimo coeficiente genérico de uma máscara de convolução
x	Eixo da abscissa
x_c	Parâmetros correspondentes aos pontos centrais de um circunferência
y	Eixo da ordenada
y_c	Parâmetros correspondentes aos pontos centrais de um circunferência
z_i	É o i -ésimo nível de cinza de <i>pixel</i> sob a máscara em qualquer posição

SUMÁRIO

1	INTRODUÇÃO	17
2	MATERIAIS E MÉTODOS	21
3	PROCESSAMENTO DIGITAL DE IMAGEM	24
3.1	Representação matemática de imagem digital	25
3.2	Amostragem e quantização	25
3.3	Operações de convolução com máscaras	27
3.3.1	Convolução unidimensional	27
3.3.2	Convolução bidimensional	28
3.4	O processamento da vizinhança	30
3.5	Processamento no domínio espacial	32
3.5.1	Filtro gaussiano	32
3.6	Morfologia matemática	34
3.6.1	Definições básicas	35
3.6.2	Dilatação	36
3.6.3	Erosão	37
3.6.4	Abertura	38
3.7	Histograma adaptativo	39
3.8	Transformada de Hough	41
3.8.1	Transformada de Hough para retas	41
3.8.2	Transformada de Hough para círculos	43
3.8.3	<i>Backmapping</i>	51
4	RESULTADOS	53
5	CONCLUSÃO	64
	REFERÊNCIAS	66
	GLOSSÁRIO	69
	APÊNDICE A IMPLEMENTAÇÃO DO CÓDIGO FONTE	70
	ANEXO A FUNÇÕES UTILIZADAS NO CÓDIGO FONTE	74
A.1	Função CircularHough_Grd	74
A.2	Função DrawCircle	85

1 INTRODUÇÃO

A utilização do óleo da mamoneira como matéria-prima para a produção de biodiesel¹ tem algumas vantagens sobre os demais óleos (FILHO; VILLANO, 1984) entre elas, destaca-se, por ser o único óleo conhecido na natureza que é solúvel em álcool quando submetido a baixas temperaturas (MACÊDO BELTRÃO et al., 2004).

A produção de agroenergia em larga escala, além da redução de custos em relação ao petróleo, apresenta vantagens ambientais e gera emprego e renda no setor rural. Em outras palavras, promove o desenvolvimento sustentável do interior do Brasil, em especial nas regiões mais remotas, lembrando que o Brasil é um país que reúne o quantitativo de vantagens comparativas para liderar a agricultura de energia. A vantagem comparativa que se destaca é a perspectiva de incorporação de áreas à agricultura de energia, sem competição com a agricultura de alimentos e com impactos ambientais circunscritos ao socialmente aceito. Outro aspecto a ser considerado é a possibilidade de múltiplos cultivos dentro do ano calendário (BRASIL, 2008).

Uma das principais metas do programa de biodiesel produzido a partir do óleo de mamona reside na possibilidade de estimular a agricultura familiar, incentivos a esse tipo de agricultura, criando emprego no campo de modo que o agricultor possa sobreviver e se manter no seu local de origem e diminuindo o êxodo desordenado para as grandes cidades (PIMENTEL; NUNES, 2008).

O governo do Brasil inclui o biodiesel na matriz energética como medida para equalizar os problemas econômicos e ambientais que, o declínio do petróleo poderá proporcionar. O biodiesel foi lançado através da lei no 11.097 de 13 de janeiro de 2005 (BRASIL, 2005a). A lei torna obrigatória a adição de 2% de biodiesel no diesel fóssil a partir de janeiro de 2008 e de 3% em julho do mesmo ano. Para tanto é necessário um incremento na produção agrícola de oleaginosas em todo o país (LUCENA, 2009).

Em curto prazo, a principal força propulsora do crescimento da demanda por energia será a pressão social pela substituição de combustíveis fósseis. Considere-se que a concentração de CO_2 atmosférico teve um aumento de 31% nos últimos 250 anos, atingindo provavelmente o nível

¹Combustível originado a partir de óleos vegetais. A concentração de biodiesel é informada por meio de uma nomenclatura específica, definida por *BX*, onde X refere-se à percentagem em volume do biodiesel o qual é misturado ao diesel do petróleo. Assim, B2, B3 e B100 referem-se, respectivamente, às misturas de biodiesel/diesel contendo 2%, 3% e 100% de biodiesel.

mais alto observado nos últimos 20 milhões de anos. Os valores tendem a aumentar significativamente se as fontes emissoras de gases de efeito estufa não forem controladas, como, por exemplo, a queima de combustíveis fósseis e a produção de cimento, responsáveis pela produção de cerca de 75% destes gases (BRASIL, 2005b).

A conscientização da população mundial vem crescendo em relação ao aquecimento global e a diminuição dos recursos naturais do planeta. Atualmente têm surgido alternativas sustentáveis baseadas na substituição dos recursos não renováveis. A matriz energética é um dos maiores desafios dos governos, entidades e cidadãos com o envolvimento em questões ambientais (LUCENA, 2009).

A iniciativa de investir em um combustível de menor agressividade ambiental contribui sensivelmente para o desenvolvimento de novas fontes de energia. O biodiesel da mamona com a fórmula B100 (uso de 100% de combustível renovável) é uma alternativa viável de substituição de produtos não-renováveis como o petróleo por energia proveniente de biomassa que, além de não conter enxofre, ser biodegradável e não corrosivo, contribui pouco para o aumento do efeito estufa.

A mamoneira pertence à classe *Dicotyledoneae*, ordem *Geraniales*, família *Euforbiácea*, gênero *Ricinus* e espécie *R. communis* L. No Brasil, recebe as denominações de mamoneira, rícino, carrapateira e palma-de-cristo. Na Inglaterra e nos Estados Unidos, é conhecida pelos nomes de *castor beans* e *castor oil*. Entre seus parentes mais próximos estão a mandioca, a borracha e o pinhão (WEISS, 2000).

Os gregos denominavam essa planta de *aporano* ou de *cróton*, e os latinos de *ricinus* que significam em grego e em latim, respectivamente, carrapato. Essa designação é atribuída devido à semelhança da forma de suas sementes com a forma do animal pertencente ao grupo dos ácaros. Sua existência é relatada há mais de 4.000 anos. São encontradas sementes dessa espécie em urnas funerárias de múmias egípcias principalmente em múmias de sacerdotes (RODRIGUES; OLIVEIRA; FONSECA, 2002).

A pesquisa com cultivares no Brasil começou no Estado de São Paulo, em 1937, com trabalhos desenvolvidos pelo Instituto Agrônomo de Campinas (IAC) e com o lançamento de plano de trabalho para o melhoramento genético desta espécie. Várias instituições nacionais trabalharam em melhoramentos da mamona, como a Universidade Federal de Viçosa (UFV), Escola Superior de Agronomia Luiz de Queiroz (ESALQ - USP), Instituto de Pesquisas Agronômicas (IPA) de Pernambuco, Empresa de Pesquisa Agropecuária de Minas Gerais (EPAMIG). O IAC não participou do lançamento das cultivares, porém participou ativamente das redes de competição de cultivares de portes anão, médio e alto, que culminou com o lançamento das cultivares BRS 149 Nordestina e BRS 188 Paraguaçu, respectivamente, em 1998 e 1999 para toda a área zoneada para a mamona no Nordeste e Norte de Minas Gerais (MACÊDO BELTRÃO, 2004).

A mamona (*Ricinus communis* L.), no cenário de oleaginosas utilizadas na produção de biodiesel, tem se destacado por ser de fácil cultivo, apresentar precocidade na produção e possuir alto teor de óleo em suas sementes (entre 40% e 50%). Essa cultura ocorre espontaneamente e

de forma asselvajada em várias regiões do Brasil, chegando a ser confundida como planta nativa. Sua origem ainda é controversa havendo relatos de sua procedência tanto no continente africano como no asiático. Entretanto sabe-se que é originária de um país de clima tropical e foi introduzida no Brasil com a colonização portuguesa (LUCENA, 2009).

No Brasil é observado um declínio na competitividade dessa cultura que pode ser explicado pelo baixo nível tecnológico do agricultor, escassez de sementes melhoradas ou mesmo melhores sistemas de plantio e colheita (FILHO, 1999). Entretanto, em condições semi-áridas, com a utilização de cultivares melhoradas, a exemplo da BRS 149 Nordestina e BRS 188 Paraguaçu cultivadas seguindo as recomendações adequadas (época de plantio, zoneamento, manejo), é possível obter produtividade superior a média nacional que, de acordo com a Companhia Nacional de Abastecimento (CONAB) (CONAB, 2009), entre os anos de 2008 e 2009 foi de 789 kg de mamona por hectare ².

A delimitação do ponto ideal de colheita é um aspecto muito importante, pois a colheita tardia pode causar perda das sementes principalmente quando a cultivar plantada possui algum grau de deiscência³ e a colheita precoce pode ser refletida na qualidade da semente.

Informações que contemplem a qualidade das sementes em função do estágio⁴ de maturação dos frutos no momento da colheita é de fundamental importância para avaliar e quantificar a influência da maturação dos frutos da mamona sobre a qualidade das sementes das cultivares e seus efeitos na produtividade, aspectos econômicos e armazenamento.

Atualmente, esses tipos de informações acerca da qualidade das sementes de acordo com o período de maturação dos frutos no momento da colheita são praticamente inexistentes. Nesse sentido, este trabalho introduz o uso do processamento de imagens digitais baseado em imagens fotográficas digitais para apoiar a tomada de decisão sobre quando a colheita deve ocorrer visando a uma maior produtividade das sementes. O custo da análise baseada em imagens fotográficas é muito menor quando comparado com a análise por DNA⁵ e com a análise de dados de sementes provenientes de um espectrofotômetro (LUCENA, 2009).

A pretensão em métodos de processamento de imagens digitais decorre de duas áreas principais de aplicação: melhoria da informação visual para a interpretação humana e o processamento de dados de cenas para percepção automática através de máquinas (GONZALEZ; WOODS, 2002).

A primeira aplicação da área de processamento de imagens foi na década de 1920 na tentativa de aprimorar imagens de jornal digitalizadas transmitidas entre Londres e Nova Iorque. O tempo necessário para essa transmissão era de uma semana. O sistema *Bartlane* de transmissão de imagens por cabo submarino conseguiu reduzir a transmissão para três horas. Avanços

²Um hectare (conhecido também como hectômetro quadrado, símbolo: hm^2) é uma unidade de medida de área equivalente a 100 ares (unidade de medida de área) ou a um quadrado cujo lado é igual a cem metros.

³Abertura espontânea, por maturidade.

⁴Período.

⁵*DeoxyriboNucleic Acid* (ADN, em português: ácido desoxirribonucleico).

expressivos na área vieram apenas com o advento dos computadores digitais trinta décadas mais tarde (GONZALEZ; WOODS, 2002).

Em 1964, procurava-se utilizar técnicas de aprimoramento de imagens para corrigir vários tipos de distorção presentes nas imagens da lua transmitidas pela sonda espacial Ranger 7⁶. Essas técnicas serviram de base para métodos de aprimoramento de realce e restauração de imagens de posteriores programas espaciais como as expedições tripuladas da série Apollo (CASTLEMAN, 1995).

Nas décadas seguintes, surgiram mais atividades que necessitariam dos adventos proporcionados por esta área de conhecimento. Entre elas estão a medicina, a biologia, a meteorologia, a astronomia, a agricultura, o sensoriamento remoto, a cartografia e a indústria petrolífera. Percebe-se dessa forma que o processamento de imagens digitais é essencialmente uma área multidisciplinar, tanto na atuação quanto na derivação, pois suas técnicas baseiam-se na física, neurofisiologia, engenharia elétrica e ciência da computação. Paralelamente, com o avanço dos recursos computacionais, permitiu-se alcançar um desenvolvimento mais eficiente e robusto do processamento de imagens digitais (FILHO; NETO, 1999; GONZALES; WOODS, 2009; RAMOS, 2010).

No contexto da produção agrícola, o crescimento do agronegócio vem contribuindo para o aumento das pesquisas de processamento de imagens na agricultura. Na tentativa de distribuição de laranjas (RAMALHO; MEDEIROS, 2003), soja (PADILHA; KHATCHATOURIAN, 2007) e outros produtos perecíveis, busca-se a criação de sistemas de inspeção visual automática para seleção dos alimentos, direcionando a classificação para os diferentes mercados a que o produtor esteja voltado.

Os maiores problemas relacionados com as imagens fotográficas de sementes de mamonas usadas neste trabalho são a aproximação entre as sementes, iluminação intensa e contraste do fundo preto e a coloração dos grãos de tonalidade próxima ao preto. A falta de espaçamento entre as sementes provoca o problema de detecção de extremidades e o reconhecimento do objeto de interesse o que dificulta a detecção de extremidade por métodos tradicionais como *Sobel*, *Prewitt*, *Canny*. Uma maneira de superar é utilizando morfologia de abertura para a suavização do contorno da semente, delimitando e isolando cada grão.

Este trabalho tem como objetivo fazer a detecção automática das sementes de mamonas presentes em uma imagem fotográfica digital empregando a transformada Hough circular.

O texto está organizado em 5 seções. A Seção 2 descreve as imagens e os métodos utilizados para a detecção das sementes na imagem. A Seção 3 apresenta a revisão dos principais conceitos envolvendo o processamento digital de imagens. A Seção 4 apresenta e discute os resultados obtidos. A Seção 5 conclui o trabalho e a apresenta os trabalhos futuros. O Apêndice apresenta os programas desenvolvidos e um Glossário de termos.

⁶Sonda espacial do Instituto de Tecnologia da Califórnia, nos Estados Unidos da América, do *Jet Propulsion Laboratory* (JPL) da *National Aeronautics and Space Administration* (NASA).

2 MATERIAIS E MÉTODOS

As imagens de sementes de mamonas utilizadas neste trabalho são fotografadas por uma câmera digital *Cyber-shot* DSC-P32 da marca *Sony*, adotando-se um ângulo de captação superior reto de 90° com resolução fina de 1280 × 960 *pixels* sob luz natural solar. A câmera é conectada a uma interface de saída *Universal Serial Bus* (USB) que, por sua vez, é ligada ao computador para onde as imagens são transferidas no formato *Joint Photographic Experts Group* (JPEG). Todas as imagens, num total de oito, são do ano de 2007 e são cedidas pela casa de vegetação da Embrapa Algodão de Campina Grande - PB. A Figura 2.1 mostra um dos tipos de imagem que é usada para a detecção de sementes neste trabalho.



Figura 2.1: Imagem obtida com a câmera *cyber-shot* DSC-P32.

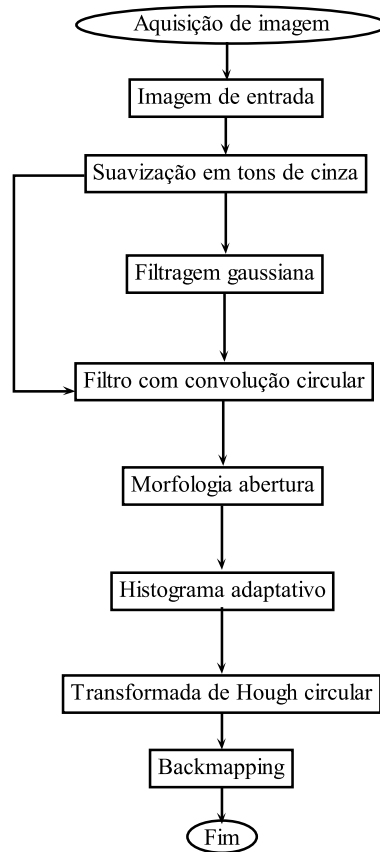


Figura 2.2: Etapas de aplicação dos métodos para a análise da imagem.

As etapas empregadas para a detecção de sementes (análise de imagem) são mostradas no fluxograma da Figura 2.2 e são as seguintes: suavização em tons de cinza, filtragem gaussiana, filtragem com a convolução circular, morfologia/abertura, histograma adaptativo, transformada de Hough circular e *backmapping*.

Na etapa de suavização, a imagem de entrada (imagem da Figura 2.1, por exemplo) é convertida em tons de cinza. A conversão de imagens coloridas para tons de cinza é realizada eliminando as informações de matiz e saturação enquanto mantém a luminosidade.

Em termos práticos, trabalha-se com uma representação para a luminosidade monocromática I em proporções dos componentes de cor vermelha (R), verde (G) e azul (B), através da Fórmula (1) definida por (STAUGAARD, 1987)

$$I = 0,3 \times R + 0,59 \times G + 0,11 \times B. \quad (1)$$

Após esta etapa, a dimensão da imagem é 1280×960 *pixels* e o número L de níveis de cinza da imagem é 256 níveis. Isto significa que cada *pixel* pode ter um valor de cinza entre 0 a 255 que requer no máximo 8 bits para o armazenamento na memória do computador. A partir desta fase, todas as funções implementadas foram em tons de cinza.

A filtragem gaussiana é feita usando uma máscara (3×3) para a remoção de detalhes

e ruídos. A filtragem com a convolução circular é aplicada para evitar o *aliasing*¹. No processamento morfológico, é aplicada a abertura com elemento estruturante em disco de 5 *pixels* na tentativa de separar as sementes.

Para redistribuir os tons de cinza da imagem é aplicado um histograma adaptativo com o objetivo de melhorar o contraste. A transformada de Hough circular é usada na fase de segmentação para a localização de cada semente. A transformada de Hough circular produz falsos picos para solucionar este problema é aplicado o *backmapping* para a redução destas anomalias.

Os programas implementados para cada uma das técnicas são feitos no ambiente MATLAB 7.0 para *Windows* devido a sua adequação para a solução do problema.

¹É o efeito em forma de serra que se cria ao desenhar uma reta em diagonal.

3 PROCESSAMENTO DIGITAL DE IMAGEM

O processamento de imagens digitais pode ser definido como qualquer condição de processamento de dados em que a entrada e a saída são imagens como fotografias ou quadros de vídeo. Indo além do tratamento de imagens, que preocupa-se somente com manipulação de figuras para sua representação final, o processamento de imagens serve para novos processamentos de dados como aprendizagem de máquina ou reconhecimento de padrões (GOMES; VELHO, 2002; PRATT, 2007).

As diversas técnicas que envolvem o tratamento da imagem como um sinal bi-dimensional, empregadas, na prática, a padrões de processamento de sinal, eram, há cerca de duas décadas, feitas, em sua maior parte, de forma analógica, através de dispositivos ópticos. Contudo, em razão do grande aumento da frequência de processamento dos computadores, tais técnicas foram gradualmente substituídas por processos digitais (BLANCHET; CHARBIT, 2006; GONZALEZ; WOODS, 2002).

A área do processamento de imagem é objeto de crescente interesse porque viabiliza grande número de aplicações em duas categorias bem distintas: (a) o aprimoramento de informações pictóricas para interpretação humana; e (b) análise de informações extraídas de uma cena de forma automática por computador (FILHO; NETO, 1999). A análise de imagens é uma tarefa difícil de ser implementada e o que há de fato disponível até o momento é um conjunto de técnicas de processamento de imagens que pode ser utilizado para obter informações sobre os objetos de interesse de uma cena.

Relacionam-se abaixo algumas das diversas aplicações que utilizam processamento de imagens:

- a) Cartografia: identificação, interpretação de objetos presentes nas imagens;
- b) Medicina: técnicas de tomografia computadorizada, ressonância magnética e radiografia baseada em Raio-X;
- c) Visão Computacional: consiste na automatização de tarefas realizadas pelos seres humanos.

Nas seções seguintes, serão apresentados os principais conceitos matemáticos e técnicas que apóiam o desenvolvimento dos métodos necessários para a identificação de sementes nas

imagens.

3.1 Representação matemática de imagem digital

Suponha que a imagem contínua $f(x, y)$ é aproximada por amostras igualmente espaçadas, arranjadas na forma de uma matriz $m \times n$ como mostra a Relação (2). Cada coordenada da matriz, $m \times n$, representa uma quantidade discreta ou elemento da imagem digital (YOUNG; GERBRANDS; VLIET, 1998; PRATT, 2007). O lado direito da Relação (2) representa o que é normalmente denominado de imagem digital.

$$f(x, y) \approx \begin{bmatrix} f(0, 0) & f(0, 1) & \dots & f(0, n-1) \\ f(0, 0) & f(0, 0) & \dots & f(1, n-1) \\ \vdots & \vdots & & \\ f(m-1, 0) & f(m-1, 1) & \dots & f(m-1, n-1) \end{bmatrix} \quad (2)$$

Sob uma abordagem matemática formal, o processo de amostragem pode ser visto como uma divisão do plano cartesiano (x, y) em uma grade. As coordenadas do centro de cada grade é uma dupla de elementos do produto $\mathbb{Z} \times \mathbb{Z}$ (também escrito \mathbb{Z}^2) (GONZALEZ; WOODS, 2002; ACHARYAM; RAY, 2005).

O produto cartesiano forma o conjunto de todos os pares ordenados dos elementos a, b , onde a e b pertencem a \mathbb{Z} (conjunto dos inteiros). Assim, $f(x, y)$ é uma imagem digital, se (x, y) são números inteiros de $\mathbb{Z} \times \mathbb{Z}$ e f é uma função que atribui um valor de nível de cinza (isto é, um número real do conjunto de números reais \mathbb{R} para cada par distinto de coordenadas). Isto é, f é o resultado do processo de quantização descrito a seguir.

Se os níveis de cinza resultantes forem também números inteiros (como geralmente é o caso), \mathbb{Z} substitui \mathbb{R} e uma imagem digital então se torna uma função bidimensional cujas coordenadas e valores de amplitude são números inteiros (GONZALEZ; WOODS, 2002; FILHO; NETO, 1999).

3.2 Amostragem e quantização

A amostragem pode ser definida como o processo de medição instantânea de valores de um sinal analógico em intervalos regulares. O intervalo entre as amostras é determinado por um pulso de sincronismo e a sua frequência é chamada de taxa de amostragem. O processo de amostragem converte a imagem analógica em uma matriz de x por y pontos denominados *pixel* ou elemento da imagem (FILHO; NETO, 1999).

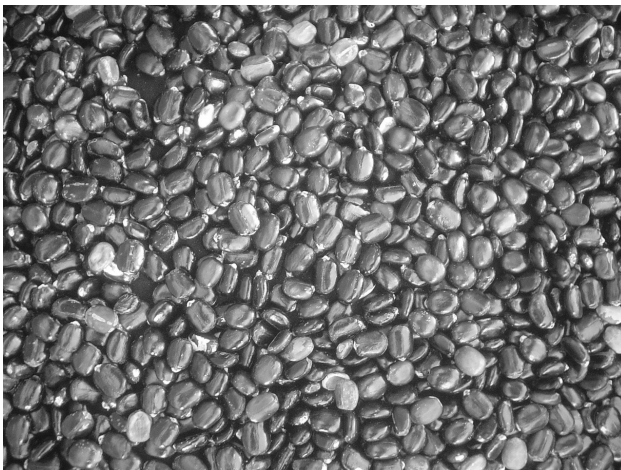


Figura 3.1: Imagem em níveis de cinza.

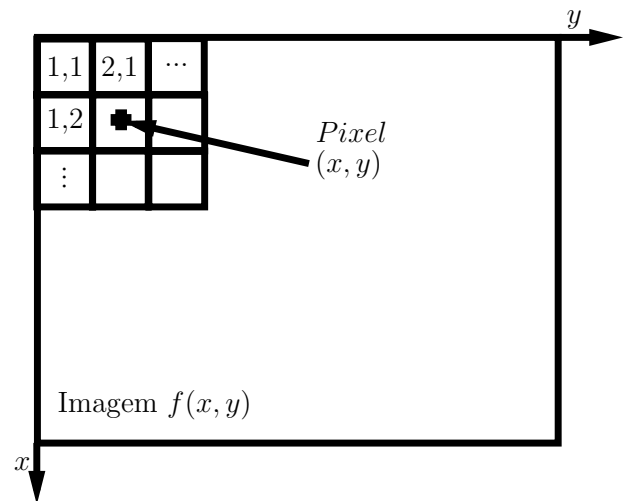


Figura 3.2: Plano de discretização.

A Figura 3.2 mostra a convenção de discretização da Figura 3.1 em que o eixo x (abscissa) e y (ordenada) são apresentados em posições opostas no tradicional plano cartesiano (x, y) .

Para ser adequada ao processamento computacional, uma função $f(x, y)$ precisa ser digitalizada, tanto espacialmente quanto em amplitude. A digitalização das coordenadas espaciais (x, y) é denominada amostragem da imagem e a digitalização da amplitude é chamada quantização em níveis de cinza (GONZALEZ; WOODS, 2002).

$f(1,1)f(2,1)f(3,1)f(4,1)f(5,1)f(6,1)f(7,1)f(8,1)f(9,1)f(10,1)f(11,1)$
 $f(1,2)f(2,2)f(3,2)f(4,2)f(5,2)f(6,2)f(7,2)f(8,2)f(9,2)f(10,2)f(11,2)$
 $f(1,3)f(2,3)f(3,3)f(4,3)f(5,3)f(6,3)f(7,3)f(8,3)f(9,3)f(10,3)f(11,3)$
 $f(1,4)f(2,4)f(3,4)f(4,4)f(5,4)f(6,4)f(7,4)f(8,4)f(9,4)f(10,4)f(11,4)$
 $f(1,5)f(2,5)f(3,5)f(4,5)f(5,5)f(6,5)f(7,5)f(8,5)f(9,5)f(10,5)f(11,5)$
 $f(1,6)f(2,6)f(3,6)f(4,6)f(5,6)f(6,6)f(7,6)f(8,6)f(9,6)f(10,6)f(11,6)$
 $f(1,7)f(2,7)f(3,7)f(4,7)f(5,7)f(6,7)f(7,7)f(8,7)f(9,7)f(10,7)f(11,7)$

Figura 3.3: Discretização espacial.

28	24	24	28	28	25	28	34	36	32	29
42	37	35	36	34	29	30	34	33	30	27
27	24	23	26	26	24	25	29	30	27	26
26	25	25	27	30	31	31	32	28	26	25
33	31	29	28	29	29	26	22	27	26	26
27	27	25	24	27	29	26	20	27	28	28
30	32	31	31	37	43	40	33	27	29	31
29	26	29	25	29	27	29	18	27	29	29

Figura 3.4: Discretização em amplitude.

Pode-se observar a Figura 3.3 uma discretização espacial das coordenadas de cada amplitude de tamanho (11×7) apresentada através dos níveis de cinza na Figura 3.4.

A quantização é o processo de atribuição de valores discretos em que cada *pixel* assume um valor inteiro na faixa de 0 a $(2^k - 1)$. Quanto maior o valor de k maior o número de níveis de cinza presentes na imagem digitalizada e maiores valores de m e n implicam uma imagem de maior resolução.

A digitalização eletrônica consiste na conversão analógica-digital (A/D) em que o número de amostras do sinal contínuo por unidade de tempo indica a taxa de amostragem.

O número de bits do conversor utilizado determina o número de tons de cinza resultantes na imagem digitalizada. Os tons de cinza variam de (0 . . . 255) totalizando 256 tonalidades entre preto (0) e branco (255) para uma imagem de 8 bits (FILHO; NETO, 1999).

A Figura 3.4 mostra a discretização em amplitude da Figura 3.1. Cada número inteiro na Figura 3.4 representa os tons de cinza na Figura 3.1.

3.3 Operações de convolução com máscaras

3.3.1 Convolução unidimensional

A convolução unidimensional é uma operação matemática que tem como objetivo a redução de informações de alta e baixa frequência conforme o tamanho do filtro. A convolução de duas funções $f(x)$ e $g(x)$, denotada por $f(x) * g(x)$, definida pela integral

$$f(x) * g(x) = \int_{-\infty}^{\infty} f(\alpha)g(x - \alpha)d\alpha \quad (3)$$

em que α é uma variável de integração.

A importância da convolução na análise do domínio da frequência reside no fato de que $f(x) * g(x)$ e $F(u)G(u)$ constituem um par de transformadas de Fourier. Em outras palavras, se $f(x)$ tem como transformada de Fourier $F(u)$ e $g(x)$ tem como transformada de Fourier $G(u)$, então $f(x) * g(x)$ tem como transformada de Fourier $F(u)G(u)$. Este resultado é enunciado formalmente como

$$f(x) * g(x) \Leftrightarrow F(u)G(u) \quad (4)$$

este cálculo indica que a convolução no domínio x pode também ser obtida como a transformada de Fourier inversa do produto $F(u)G(u)$. Maiores detalhes sobre a transformada de Fourier (JAHNE, 2002; YOUNG; GERBRANDS; VLIET, 1998; PRATT, 2007). O resultado análogo é a convolução no domínio da frequência que pode ser calculado a partir do produto no domínio x ; tal como,

$$f(x)g(x) \Leftrightarrow F(u) * G(u). \quad (5)$$

Esses dois resultados das Equações 4 e 5 são comumente denominados como *teorema da convolução*.

Suponha que, em vez de serem contínuas $f(x)$ e $g(x)$ sejam discretizadas em vetores de tamanho c e c' , respectivamente: $\{f(0), f(1), f(2), \dots, f(c - 1)\}$ e $\{g(0), g(1), g(2), \dots, g(c' - 1)\}$. A formulação do teorema da convolução discreta consiste com a periodicidade assume que as funções discretas $f(x)$ e $g(x)$ são periódicas com algum período T (GONZALEZ; WOODS, 2002). A convolução resultante então será periódica e de mesmo período. O problema é como escolher um valor para T . Pode-se mostrar que, a menos que

$$T \geq c + c' - 1 \quad (6)$$

seja obedecida, os períodos individuais da convolução serão sobrepostos, um fenômeno comumente denominado *erro de revestimento*. Se $T = c + c' - 1$, os períodos são adjacentes. Se $T > c + c' - 1$, os períodos são separados, sendo o grau de separação definido pela diferença T e $c + c' - 1$. Como o período assumido deve ser maior do que ambos c e c' , o comprimento das sequências de amostragem deve ser aumentado de modo que ambos sejam de tamanho T . Sequências estendidas podem ser formadas adicionando zeros às amostras

$$f_e(x) = \begin{cases} f(x) & 0 \leq x \leq c - 1 \\ 0 & c \leq x \leq T - 1 \end{cases} \quad (7)$$

e

$$g_e(x) = \begin{cases} g(x) & 0 \leq x \leq c' - 1 \\ 0 & c' \leq x \leq T - 1. \end{cases} \quad (8)$$

Com base nas extensões definidas em 7 e 8 a convolução discreta de $f_e(x)$ e $g_e(x)$ é definida pela expressão

$$f_e(x) * g_e(x) = \frac{1}{T} \sum_{t=0}^{T-1} f_e(t)g_e(x - t) \quad (9)$$

onde (x) é a dimensão da imagem f_e e (t) é a dimensão do filtro g_e .

3.3.2 Convolução bidimensional

A convolução bidimensional de duas funções $f(x, y)$ e $g(x, y)$, denotada por $f(x, y) * g(x, y)$, é definida pelas integrais

$$f(x, y) * g(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\alpha, \beta)g(x - \alpha, y - \beta)d\alpha d\beta. \quad (10)$$

O teorema da convolução em duas dimensões, então, é expresso pelas relações

$$f(x, y) * g(x, y) \Leftrightarrow F(u, v)G(u, v), \quad (11)$$

$$f(x, y)g(x, y) \Leftrightarrow F(u, v) * G(u, v). \quad (12)$$

A convolução discreta 2D é formulada representando-se $f(x, y)$ e $g(x, y)$ como matrizes discretas de dimensão $(m \times n)$ e $(m' \times n')$, respectivamente. As matrizes devem ser periódicas com algum período T e T' nas direções x e y , respectivamente. O erro de revestimento nos períodos individuais da convolução é evitado pela escolha

$$T \geq m + m' - 1, \quad (13)$$

$$T' \geq n + n' - 1. \quad (14)$$

As sequências periódicas são formadas ao se estender $f(x, y)$ e $g(x, y)$ como segue

$$f_e(x, y) = \begin{cases} f(x, y) & 0 \leq x \leq m - 1 \quad e \quad 0 \leq y \leq n - 1 \\ 0 & m \leq x \leq T - 1 \quad ou \quad n \leq y \leq T - 1 \end{cases} \quad (15)$$

e

$$g_e(x, y) = \begin{cases} g(x, y) & 0 \leq x \leq m' - 1 \quad e \quad 0 \leq y \leq n' - 1 \\ 0 & m' \leq x \leq T - 1 \quad ou \quad n' \leq y \leq T' - 1. \end{cases} \quad (16)$$

A convolução 2D de $f_e(x, y)$ e $g_e(x, y)$ é definida pela relação

$$f_e(x, y) * g_e(x, y) = \frac{1}{TT'} \sum_{t=0}^{T-1} \sum_{t'=0}^{T'-1} f_e(t, t') g_e(x - t, y - t') \quad (17)$$

onde (x, y) é a dimensão da imagem f_e e (t, t') são as dimensões do filtro g_e .

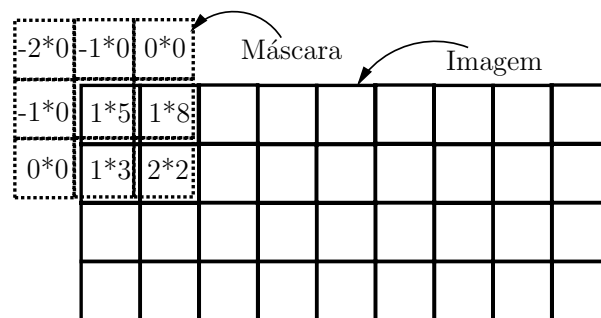


Figura 3.5: Cálculo entre a matriz imagem e matriz máscara.

$$(-2*0) + (-1*0) + (0*0) + (-1*0) + (1*5) + (1*8) + (0*0) + (1*3) + (2*2)$$

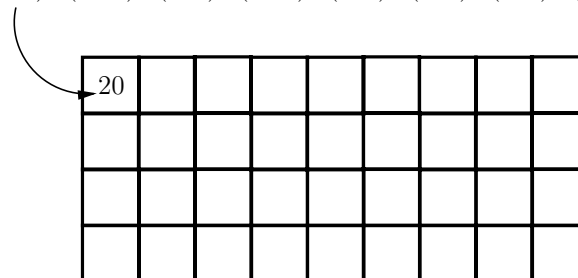


Figura 3.6: Resultado de um célula de convolução.

A operação de convolução bidimensional entre duas matrizes \mathbf{A} e \mathbf{B} , denotada $(\mathbf{A} * \mathbf{B})$, pode ser entendida como um conjunto de somas de produtos entre os valores de \mathbf{A} (imagem) e \mathbf{B} (máscara¹), sendo que, inicialmente, a máscara \mathbf{B} é espelhada e, após cada soma de produtos, é deslocada espacialmente de uma posição. Veja as Figuras 3.5 e 3.6.

¹Os termos *template*, *janela* e *filtro* são também frequentemente usados para denominar uma máscara.

A máscara, após ter sido espelhada tanto na horizontal quanto na vertical, percorrerá todos os pontos da imagem, deslocando-se ao longo de cada linha e entre as várias linhas, da direita para a esquerda, de cima para baixo, até ter processado o último elemento da matriz imagem. O resultado será armazenado na matriz **C** de mesmas dimensões que a imagem original (FARID, 2007; PRATT, 2007). Seja a matriz **A** (imagem) dada por

$$\mathbf{A} = \begin{bmatrix} 5 & 8 & 3 & 4 & 6 & 2 & 3 & 7 \\ 3 & 2 & 1 & 1 & 9 & 5 & 1 & 0 \\ 0 & 9 & 5 & 3 & 0 & 4 & 8 & 3 \\ 4 & 2 & 7 & 2 & 1 & 9 & 0 & 6 \\ 9 & 7 & 9 & 8 & 4 & 1 & 0 & 9 \\ 5 & 2 & 1 & 8 & 4 & 1 & 0 & 9 \\ 1 & 8 & 5 & 4 & 9 & 2 & 3 & 8 \\ 3 & 7 & 1 & 2 & 3 & 4 & 4 & 6 \end{bmatrix}, \quad (18)$$

e seja a matriz **B** (máscara) a seguir

$$\mathbf{B} = \begin{bmatrix} 2 & 1 & 0 \\ 1 & 1 & -1 \\ 0 & -1 & -2 \end{bmatrix}, \quad (19)$$

espelhamento da matriz **B** (máscara)

$$\mathbf{B} = \begin{bmatrix} -2 & -1 & 0 \\ -1 & 1 & 1 \\ 0 & 1 & 2 \end{bmatrix}. \quad (20)$$

A operação de convolução bidimensional produzirá como resultado a matriz

$$\mathbf{C} = \begin{bmatrix} 20 & 10 & 2 & 26 & 23 & 6 & 9 & 4 \\ 18 & 1 & -8 & 2 & 7 & 3 & 3 & -11 \\ 14 & 22 & 5 & -1 & 9 & -2 & 8 & -1 \\ 29 & 21 & 9 & -9 & 10 & 12 & -9 & -9 \\ 21 & 1 & 16 & -1 & -3 & -4 & 2 & 5 \\ 15 & -9 & -3 & 7 & -6 & 1 & 17 & 9 \\ 21 & 9 & 1 & 6 & -2 & -1 & 23 & 2 \\ -9 & -5 & -25 & -10 & -12 & -15 & -1 & -12 \end{bmatrix}. \quad (21)$$

3.4 O processamento da vizinhança

A operação aritmética orientada a vizinhança utiliza o conceito de convolução com máscaras, que é introduzido e detalhado na Seção 3.3.

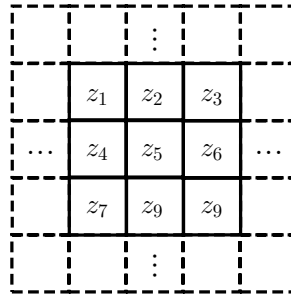


Figura 3.7: Subárea de uma imagem.

O processamento da vizinhança é tipicamente formulado no contexto das operações por máscara. Essa operação modifica o valor de um *pixel* em função do seu nível de cinza próprio e o de seus vizinhos (BLANCHET; CHARBIT, 2006), considerando-se a área da subimagem mostrada na Figura 3.7.

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

Figura 3.8: Máscara (3×3).

Suponha que se deseja trocar o valor de z_5 pela média dos valores dos *pixels* de uma região (3×3) definido pela máscara da Figura 3.8 centrada no *pixel* de valor z_5 . Para isto, é necessário fazer uma operação aritmética definida por

$$z = \frac{1}{9}(z_1 + z_2 + \dots + z_9) = \frac{1}{9} \sum_{i=1}^9 z_i \quad (22)$$

em que o valor calculado para z é atribuído a z_5 na Figura 3.7.

Em referência à máscara, mostrada na Figura 3.8, a mesma operação pode ser obtida em termos mais gerais, centrando-se a máscara em z_5 , multiplicando-se cada *pixel* sob a máscara pelo coeficiente correspondente, e adicionando esses resultados. Isto é,

$$z = (w_1 z_1 + w_2 z_2 + \dots + w_9 z_9) = \sum_{i=1}^9 w_i z_i. \quad (23)$$

Sendo $w_i = \frac{1}{9}$, $i = 1, 2, \dots, 9$, os valores desta operação produz o mesmo resultado que o procedimento de média apresentado na Equação (22). A Equação (23) é amplamente usada em processamento de imagens (MCANDREW, 2004).

3.5 Processamento no domínio espacial

As abordagens que envolvem métodos no domínio espacial referem-se ao próprio plano da imagem e as abordagens nessa categoria são baseadas na manipulação direta sobre os *pixels* da imagem. O uso de máscaras espaciais para o processamento de imagens é usualmente chamado de filtragem espacial e as máscaras são denominadas filtros espaciais.

Os filtros espaciais mais utilizados são classificados em passa-baixa e passa-alta. Os filtros passa-baixa ou de suavização eliminam os componentes de alta frequência, deixando *passar* as baixas frequências. Esse tipo de filtro causa um *borramento* acentuado na imagem devido, fundamentalmente, a caracterização dos componentes de alta frequência, que evidenciam as bordas e outros detalhes finos da imagem. Dessa forma, a imagem perde detalhes, ou seja, a informação de alta frequência.

O filtro Gaussiano é um exemplo de filtro linear passa-baixa, ele é utilizado para eliminação de informação de *alta frequência* (ruído). A imagem é suavizada pela convolução da imagem com o núcleo Gaussiano. No entanto, o operador de suavização Gaussiano é isotrópico (circularmente simétrico), suavizando a imagem em todas as direções e fazendo com que as bordas tornem-se *borradas*.

3.5.1 Filtro gaussiano

O operador de suavização Gaussiano consiste numa operação de convolução utilizada para *borrar* a imagem digital com o objetivo de remover ruídos. A suavização utiliza um máscara diferente daquela usada pelo filtro que representa a forma da distribuição gaussiana (GONZALEZ; WOODS, 2002).

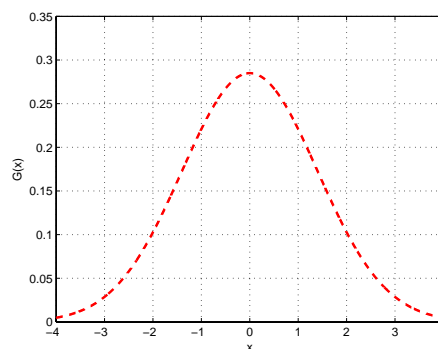


Figura 3.9: Forma da distribuição gaussiana com média zero e desvio padrão σ em 1D.

A distribuição gaussiana, com média zero e desvio padrão σ em uma dimensão, é ilustrada na Figura 3.9 e descrita pela Equação (24) definida por (PARKER, 1996).

$$G(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}}. \quad (24)$$

Em duas dimensões, a distribuição gaussiana isotrópica com média zero e desvio padrão

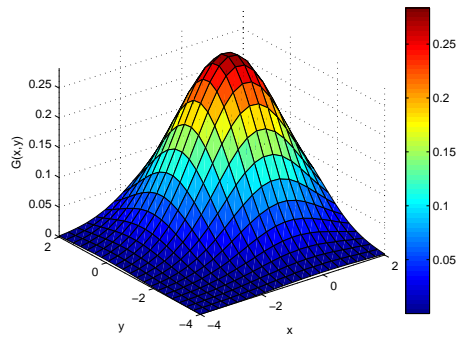


Figura 3.10: Forma da distribuição gaussiana com média zero e desvio padrão σ em 2D.

σ é ilustrada na Figura 3.10 e descrita, convenientemente, em termos das direções perpendiculares x e y , por

$$G(x, y) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x^2+y^2)}{2\sigma^2}}. \quad (25)$$

A idéia da suavização por filtragem gaussiana utiliza a distribuição 2D como uma função de ponto de espalhamento. Isso é obtido por meio da operação de convolução. A imagem é vista como uma coleção discreta de *pixels* e, nesse caso, é necessário produzir uma aproximação discreta da função de distribuição gaussiana antes de realizar a convolução (ACHARYAM; RAY, 2005).

Em teoria, a distribuição gaussiana é assintótica² e positiva em qualquer valor no conjunto dos \mathbb{R} . Assim, é necessária uma máscara de convolução infinitamente grande. Na prática, ela se torna zero para qualquer valor mais distante que três ou quatro desvios-padrões em relação à média, o que permite a truncagem da máscara a partir deste ponto.

O Quadro 1 ilustra uma máscara de convolução aplicada a um conjunto conveniente de valores inteiros que se aproxima de uma distribuição gaussiana com desvio padrão $\sigma = 1, 4$.

$$\frac{1}{115} \begin{array}{|c|c|c|c|c|} \hline 2 & 4 & 5 & 4 & 2 \\ \hline 4 & 9 & 12 & 9 & 4 \\ \hline 5 & 12 & 15 & 12 & 5 \\ \hline 4 & 9 & 12 & 9 & 4 \\ \hline 2 & 4 & 5 & 4 & 2 \\ \hline \end{array}$$

Quadro 1: Aproximação discreta de uma distribuição gaussiana com desvio padrão $\sigma = 1, 4$.

Uma vez que a máscara adequada tenha sido definida, a operação de suavização gaussiana pode ser desenvolvida utilizando métodos de convolução. Esta operação pode ser realizada de modo relativamente eficiente, desde que a Equação (25) seja separável nas suas componentes x

²A curva é assintótica, isto é, estende-se de $-\infty$ a $+\infty$, sem nunca tocar o eixo horizontal e, portanto, a função de x jamais se anula.

e y . Assim, a convolução em 2D pode ser executada primeiramente convoluindo na direção de x com a distribuição gaussiana em 1D e, depois, convoluindo na direção y , também em 1D.

O Quadro 2 ilustra a componente da máscara em 1D na direção x que pode ser utilizada para produzir a máscara completa da Quadro 1. A componente em y seria exatamente a mesma, orientada verticalmente.

$$\frac{1}{10,7} \begin{array}{|c|c|c|c|c|} \hline 1,3 & 3,2 & 3,8 & 3,2 & 1,3 \\ \hline \end{array}$$

Quadro 2: Máscara em 1D utilizada para calcular a convolução em 2D mais eficientemente.

O resultado obtido com a utilização da distribuição gaussiana representa uma média ponderada de cada *pixel* em relação a sua vizinhança. A técnica provê melhores resultados do que um filtro médio de igual tamanho.

O grau de suavização de uma imagem através da filtragem gaussiana é determinado pelo seu desvio-padrão. Logo, maior desvio-padrão requer máscara de tamanho maior para ser devidamente representado.

$$G(x, y) \approx \begin{bmatrix} 0,0113 & 0,0838 & 0,0113 \\ 0,0838 & 0,6193 & 0,0838 \\ 0,0113 & 0,0838 & 0,0113 \end{bmatrix} \quad (26)$$

Pode-se observar a Relação (26) que ilustra uma máscara de convolução aplicada a um conjunto de valores reais que se aproxima da distribuição gaussiana com desvio padrão $\sigma = 0.5$ que é utilizada neste trabalho.

3.6 Morfologia matemática

A morfologia matemática é introduzida na década de 1960 por Georges Matheron e Jean Serra (MATHERON, 1974; SERRA, 1982). A palavra morfologia tem origem grega e significa estudo das formas. Ela concentra seus esforços no estudo da estrutura geométrica das entidades presentes numa imagem. A morfologia matemática, em termos de imagens, representa um ramo não linear das técnicas de processamento imagens digitais, permitindo processar imagens com objetivos de realce, segmentação, detecção de bordas, entre outras. A morfologia matemática é uma ferramenta para a extração de componentes da imagem úteis na representação e descrição da forma de uma região (GONZALEZ; WOODS, 2002; PRATT, 2007; MCANDREW, 2004).

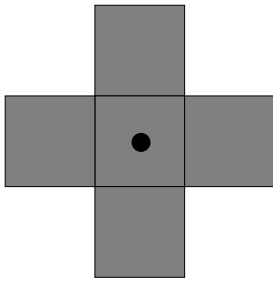


Figura 3.11: Disco.

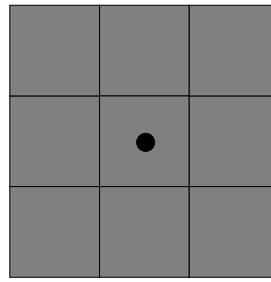


Figura 3.12: Quadrado.

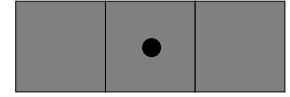


Figura 3.13: Linha.

A grande potencialidade da morfologia matemática reside na palavra elemento estruturante que é caracterizado como um conjunto completamente definido e conhecido (forma, tamanho). Comparado, a partir de uma transformação ao conjunto desconhecido da imagem. Alguns exemplos de elementos estruturantes (3×3) com origem no centro do elemento são mostrados nas Figuras 3.11, 3.12 e 3.13. A Figura 3.11 mostra o disco, a Figura 3.12 mostra o quadrado e a Figura 3.13 mostra a linha.

3.6.1 Definições básicas

Sejam A e B conjuntos de \mathbb{Z}^2 com componentes $a = (a_1, a_2)$ e $b = (b_1, b_2)$, respectivamente. A translação de A por $x = (x_1, x_2)$, denotada por $(A)_x$ é definida como

$$(A)_x = \{c | c = a + x, \text{ para } a \in A\}. \quad (27)$$

A reflexão de B , denotada por \hat{B} , é definido como

$$\hat{B} = \{x | x = -b, \text{ para } b \in B\}. \quad (28)$$

O complemento do conjunto A é definido como

$$A^c = \{x | x \notin A\}. \quad (29)$$

Finalmente, a diferença entre dois conjuntos A e B , denotada por $A - B$, é definida como

$$A - B = \{x | x \in A, x \notin B\} = A \cap B^c. \quad (30)$$

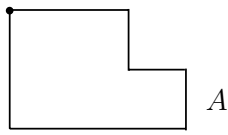


Figura 3.14: Conjunto A .

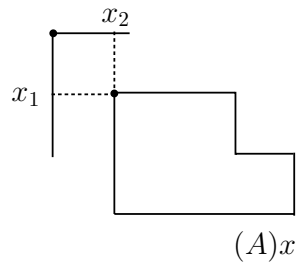


Figura 3.15: Conjunto A transladado.

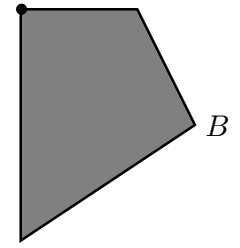


Figura 3.16: Conjunto B .

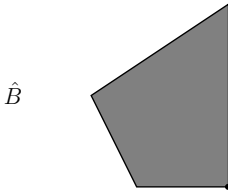


Figura 3.17: Reflexão de B .

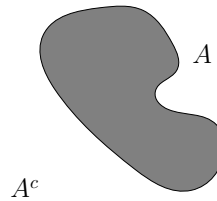


Figura 3.18: Conjunto A e seu complemento.

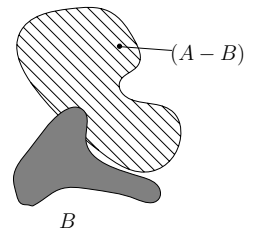


Figura 3.19: A diferença entre dois conjuntos.

As Figuras 3.14, 3.15, 3.16, 3.17, 3.18, 3.19 ilustram as definições básicas apresentadas, sendo que o ponto preto sobre a linha externa identifica a origem de cada conjunto. A Figura 3.14 mostra um conjunto A , enquanto a Figura 3.15 mostra a translação de A por $x = (x_1, x_2)$. A translação é realizada somando-se (x_1, x_2) a cada ponto de A . A Figura 3.16 mostra um conjunto B , enquanto a Figura 3.17 apresenta sua reflexão em torno da origem. Finalmente, a Figura 3.18 mostra um conjunto A e seu complemento, enquanto a Figura 3.19 apresenta a diferença entre o conjunto A , mostrado na Figura 3.18, e o conjunto B , apresentado na Figura 3.19.

3.6.2 Dilatação

Sejam A e B conjuntos de \mathbb{Z}^2 e \emptyset conjunto vazio, define-se a dilatação de A por B , denotada por $A \oplus B$, como

$$A \oplus B = \{x | (\hat{B})_x \cap A \neq \emptyset\}. \quad (31)$$

O processo de dilatação inicia-se na obtenção da reflexão de B em torno de sua origem, seguido da translação dessa reflexão por x . A dilatação de A por B é então o conjunto de todos os deslocamentos x tais que \hat{B} e A sobreponham-se em pelo menos um elemento não nulo. Essa interpretação permite que a Equação (31) seja reescrita como

$$A \oplus B = \{x | [(\hat{B})_x \cap A] \neq \emptyset\}. \quad (32)$$

O conjunto B é normalmente chamado de elemento estruturante da dilatação, bem como de outras operações morfológicas.

A dilatação de uma imagem em tons de cinza f por um elemento estruturante B é definida como (SOILLE, 2003; JAHNE, 2002; PRATT, 2007)

$$[\delta_B(f)](x) = \max_{b \in B} f(x + b) \quad (33)$$

onde $\delta_B(f)$ representa a dilatação de uma imagem pelo elemento estruturante B . O elemento b é o elemento estruturante B e a dilatação é representada pelo \max (máximo) de $f(x + b)$.

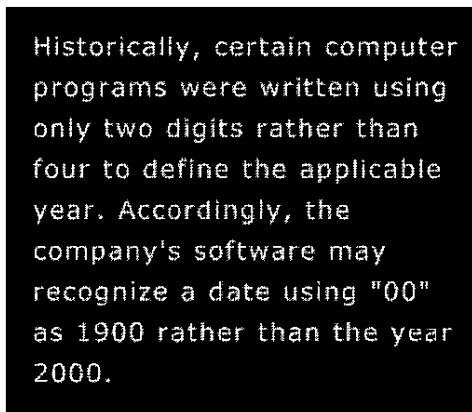


Figura 3.20: Imagem original.

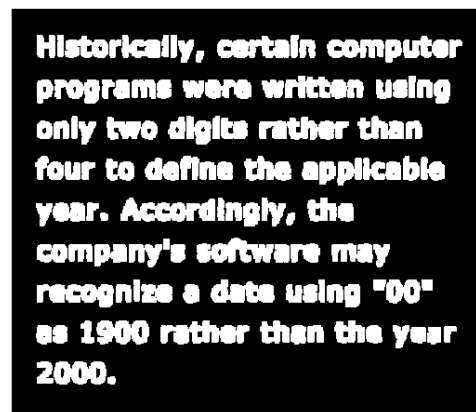


Figura 3.21: Resultado da dilatação.

A Figura 3.21 ilustra o resultado da dilatação na imagem em tons de cinza (Figura 3.20) pelo elemento estruturante disco (3×3) com origem no centro. A imagem resultante apresenta o efeito de aumento dos padrões claros da câmera sobre o fundo escuro da imagem, resultando em um aumento significativo do branco das letras.

3.6.3 Erosão

Para conjuntos A e B em \mathbb{Z}^2 , a erosão de A por B , denotada por $A \ominus B$, é definida por

$$A \ominus B = \{x | (B)_x \subseteq A\}, \quad (34)$$

o que significa que a erosão de A por B é o conjunto de todos os pontos x tais que B , quando transladado por x , fique contido em A .

A erosão de uma imagem em tons de cinza f por um elemento estruturante B é definida como (SOILLE, 2003; JAHNE, 2002; PRATT, 2007; MCANDREW, 2004)

$$[\varepsilon_B(f)]x = \min_{b \in B} \{f(x + b) - B(b)\} \quad (35)$$

onde ε_B representa a erosão pelo elemento estruturante B , b pertence ao elemento estruturante B e a erosão é representada pelo \min (mínimo) de $f(x + b) - B(b)$.



Figura 3.22: Imagem original.



Figura 3.23: Resultado da erosão.

A Figura 3.23 ilustra o resultado da erosão na imagem em tons de cinza mostrada na (Figura 3.22) pelo elemento estruturante disco (3×3) com origem no centro. Assim, a aplicação do operador de erosão apresenta os efeitos de aumento do fundo escuro sobre os padrões claros, resultando em uma diminuição significativa dos detalhes do cameraman, assim como do conjunto tripé e câmera.

3.6.4 Abertura

A abertura suaviza os contornos de uma imagem, quebra istmos³ estreitos e elimina protusões⁴ finas. A abertura de um conjunto A por um elemento estruturante B , denotado por $A \circ B$, é definida como

$$A \circ B = (A \ominus B) \oplus B, \quad (36)$$

significando que a abertura de A por B é simplesmente a erosão de A por B seguida da dilatação do resultado por B .

A partir dos operadores básicos da morfologia matemática (erosão e dilatação), é definida a abertura como a diferença entre a imagem original e a imagem erodida pelo elemento estruturante B (SOILLE, 2003; FILHO; NETO, 1999; GONZALES; WOODS, 2009; PRATT, 2007)

$$\rho_B^- = f - \varepsilon_B(f) \quad (37)$$

onde ρ_B^- representa a abertura, f representa a imagem original, B é o elemento estruturante e ε_B a erosão pelo elemento estruturante B .

³Formação estreita que promove a união entre duas outras maiores.

⁴Que avança ou sai para fora do plano a que está unido: que sobressai ou ressalta.

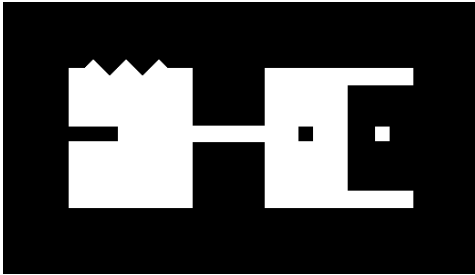


Figura 3.24: Imagem original.

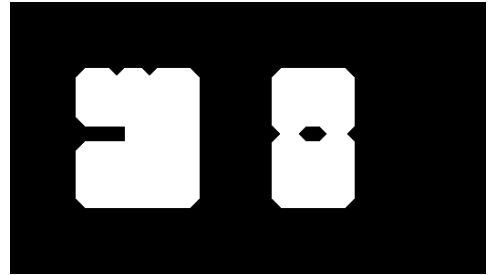


Figura 3.25: Resultado da abertura.

A Figura 3.25 ilustra o resultado da aplicação da operação de abertura na imagem em tons de cinza da Figura 3.24 pelo elemento estruturante disco (20×20) com origem no centro. Assim, a aplicação do operador de abertura apresenta os efeitos de quebra dos istmos estreitos da imagem e redução de quase todas as protuberâncias.

3.7 Histograma adaptativo

O histograma de uma imagem digital com níveis de cinza no intervalo $[0, L - 1]$ é uma função discreta $p(r_k) = \frac{n_k}{n}$, em que r_k é o k -ésimo nível de cinza, n_k é o número de *pixels* na imagem com esse nível de cinza, n é o número total de *pixels* na imagem e $k = 0, 1, 2, \dots, L - 1$ corresponde ao intervalo de escala de cinza, onde o 0 é considerado negro e $L - 1$ é considerado branco. Todos os valores intermediários são níveis de cinza.

A função discreta $p(r_k)$ dá uma estimativa da probabilidade de ocorrência do nível de cinza r_k . Nesse sentido, um gráfico dessa função para todos os valores de k fornece uma descrição global da aparência de uma imagem. Assim, o objetivo do histograma é representar a distribuição estatística de níveis de cinza de uma imagem (GONZALEZ; WOODS, 2002; MCANDREW, 2004; JAHNE, 2002).



Figura 3.26: Imagem original.



Figura 3.27: Resultado do histograma adaptativo.

A Figura 3.27 apresenta o resultado da aplicação do histograma adaptativo, na imagem em tons de cinza da Figura 3.26. Observamos os detalhes que passaram a ser perceptíveis na Figura 3.27, observa-se a nitidez dos contornos da roda e do aro.

O histograma adaptativo pode ser baseado em propriedades de média de intensidade e desvio padrão em uma vizinhança. A média de intensidade e desvio padrão são duas propriedades frequentemente usadas devido a relevância para a aparência de uma imagem, isto é, a média é uma medida do brilho médio e o desvio padrão é uma medida do contraste (FARID, 2007; PRATT, 2007).

Uma transformação local típica baseada nesses conceitos mapeia a intensidade de uma imagem de entrada $f(x, y)$ em uma nova imagem $g(x, y)$ através da transformação em cada *pixels* (x, y) :

$$g(x, y) = \psi(x, y) \times [f(x, y) - \bar{\mu}(x, y)] + \bar{\mu}(x, y) \quad (38)$$

$$\psi(x, y) = k \frac{\bar{\mu}_g}{\sigma(x, y)} \quad 0 < k < 1. \quad (39)$$

As Equações (38) e (39), $\bar{\mu}(x, y)$ e $\sigma(x, y)$ são o nível de cinza médio, o desvio padrão computado em uma vizinhança centrada em (x, y) , $\bar{\mu}_g$ é a média global de $f(x, y)$ e k é a constante pertencente ao intervalo indicado na Equação (39).

Os valores das quantidades variáveis ψ , $\bar{\mu}$, σ dependem da vizinhança pré-definida de (x, y) . A aplicação do fator de ganho local $\psi(x, y)$ é a diferença entre $f(x, y)$ e a média local amplifica as variações locais.

Como $\psi(x, y)$ é inversamente proporcional ao desvio padrão da intensidade, áreas com baixo contraste recebem ganho maior. A média é então adicionada para restaurar o nível médio de intensidade da imagem na região local.

3.8 Transformada de Hough

A Transformada de Hough (TH), proposta por Paul V. C. Hough, foi adequada por Richard O. Duda e Peter E. Hart, em 1972, para o tratamento de imagens digitais, utilizando coordenadas polares para definir uma reta através dos parâmetros ângulo-raio ao invés de inclinação-intersecção. Esse método, de Duda e Hart, envolve o mapeamento de retas do espaço imagem para conjuntos de pontos num espaço de parâmetros ângulo-raio (DUDA; HART, 1972; HOUGH, 1959, 1962).

A abrangência da técnica de Hough pode ser ilustrada através de alguns trabalhos recentes que incluem a eliminação de objetos espúrios, como rastros de aeronaves e satélites artificiais, em imagens astronômicas (STORKEY et al., 2004). O reconhecimento de tumores na região da aorta, em imagens de ecoendoscopia, e a detecção da região dos olhos para identificação da direção do olhar, em sistemas de interação homem-máquina, são outras possibilidades de abrangência dessa técnica (TOENNIES et al. apud PISTORI; PISTORI; COSTA, 2005; PISTORI; NETO; COSTA, 2003).

3.8.1 Transformada de Hough para retas

O método desenvolvido por Hough consiste em detectar pontos colineares⁵ ou quase colineares numa imagem. Para saber se os pontos são colineares, deve-se calcular o coeficiente das retas que passam pelo ponto. Em um dado ponto da imagem podem passar infinitas retas. Uma reta pode ser definida por dois parâmetros, utilizando-se coordenadas polares (ρ, θ) (DUDA; HART, 1972), em que ρ indica a distância mínima da reta à origem do plano cartesiano e, θ o coeficiente angular que o segmento de reta r faz com o eixo x das abscissa. Os pontos dessa reta podem ser representados por

$$\rho = x \cos(\theta) + y \sin(\theta). \quad (40)$$

Os novos parâmetros utilizados para representar o espaço são definidos agora por ρ e θ . Logo, o problema de detectar pontos colineares pode ser convertido no problema de se encontrar curvas concorrentes. De acordo com Duda e Hart (DUDA; HART, 1972), as propriedades para se transformar pontos em curvas são:

- a) Um ponto no plano imagem corresponde a uma curva senoidal no plano de parâmetros;
- b) Um ponto no plano de parâmetros corresponde a uma reta no plano da imagem;
- c) Pontos pertencentes a mesma linha reta no plano imagem correspondem a curvas através de um ponto no plano de parâmetros;

⁵Pontos colineares: são pontos que pertencem a uma mesma reta.

- d) Pontos pertencentes a mesma curva no plano de parâmetros correspondem a linhas através do mesmo ponto no plano imagens.

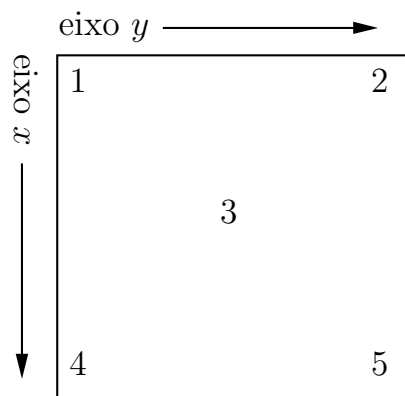


Figura 3.28: Pontos no espaço imagem.

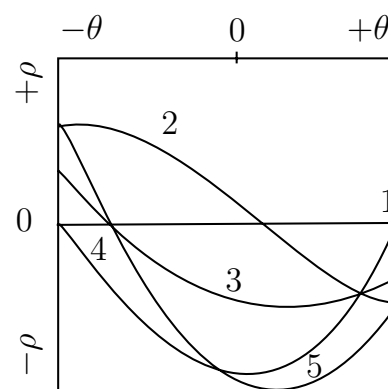


Figura 3.29: Mapeamento dos pontos no espaço de Hough.

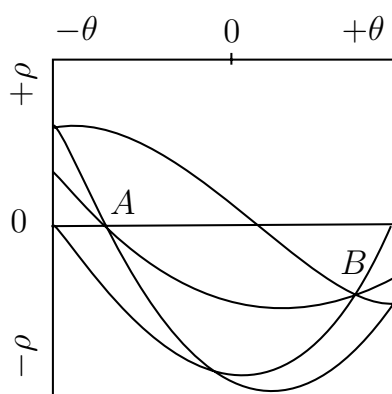


Figura 3.30: Detecção das propriedades de colinearidade.

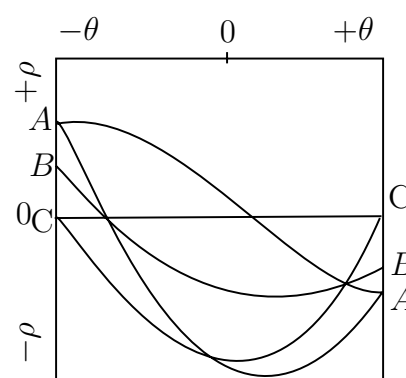


Figura 3.31: Propriedade de reflexão.

A Figura 3.28 mostra uma imagem com cinco pontos rotulados e a Figura 3.29, mostra o mapeamento dos pontos no espaço de Hough (ρ, θ) . Na Figura 3.30 ilustra-se as propriedades de colinearidade, em que o ponto A denota a intersecção entre as curvas correspondentes aos pontos 1, 3 e 5 no plano (x, y) e similarmente o ponto B corresponde aos pontos colineares 2, 3 e 4. A Figura 3.31 ilustra a propriedade de reflexão em que A , B e C se repetem enquanto a senóide inicia um novo ciclo, entretanto com sinal negativo.

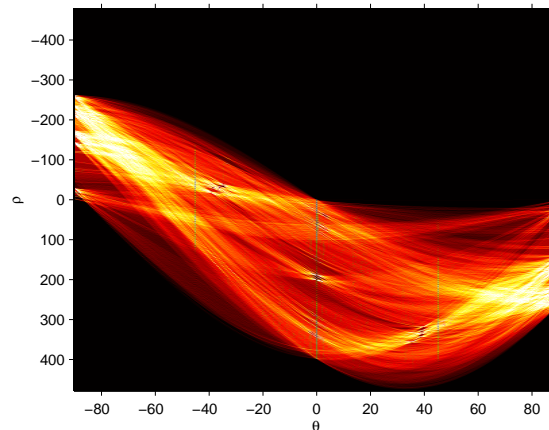


Figura 3.32: Exemplo de um arranjo acumulador (espaço de Hough).

A construção do arranjo acumulador bidimensional é definida dentro de um erro aceitável, em (ρ, θ) formando uma grade. Essa grade pode ser limitada no intervalo $(0 \leq \theta < \rho)$ e $(-R \leq \rho < R)$, onde R é o tamanho da maior diagonal da imagem. A Figura 3.32 ilustra um exemplo de um arranjo acumulador.

A detecção de retas consiste em localizar as células da matriz do espaço de Hough, calculada pela Equação (40) que apresentem os maiores valores, ou seja, determinar os pontos que possuem máximos locais.

3.8.2 Transformada de Hough para círculos

O problema da detecção de circunferências com raio fixo consiste em determinar quais pontos de uma imagem pertencem a uma mesma circunferência de raio r . Ou seja, tem-se um conjunto de coordenadas (x, y) e pretende-se encontrar valores possíveis para os parâmetros (x_c, y_c) correspondentes aos pontos centrais de circunferências. Para isso, constrói-se um espaço de Hough que, para o caso específico, pode ser visto como uma matriz com a mesma dimensão da imagem digital com as colunas e linhas representando, respectivamente, os possíveis valores de x_c e y_c . Cada célula da matriz recebe, inicialmente, o valor zero e, para cada ponto (x, y) da imagem, incrementa-se, no espaço de Hough, todas as células (x_c, y_c) representando centros de circunferências, de raio r , que passam por (x, y) .

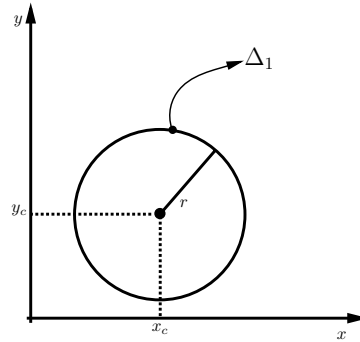


Figura 3.33: Exemplo de um círculo perfeito.

Considere o círculo Δ_1 perfeito a seguir, de raio r e centro de coordenadas (x_c, y_c) na Figura 3.33.

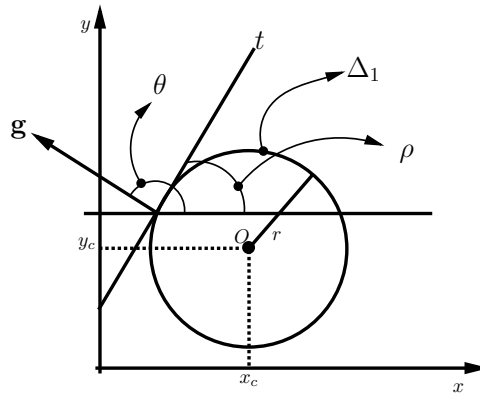


Figura 3.34: Exemplo de uma tangente t e um vetor gradiente \mathbf{g} .

Na Figura 3.34 é traçada uma tangente t ao círculo Δ_1 e um vetor gradiente \mathbf{g} perpendicular à t , em que θ é o ângulo formado entre o vetor gradiente \mathbf{g} e o eixo horizontal e ρ é o ângulo formado entre a tangente t e o eixo horizontal.

Da geometria elementar, tem-se a Equação (41) paramétrica do círculo que é

$$(x - x_c)^2 + (y - y_c)^2 = r^2. \quad (41)$$

Aplicando a derivada de primeira ordem, relativa ao gradiente, na Equação (41), tem-se a

$$2(x - x_c) + 2(y - y_c) \frac{\partial y}{\partial x} = 0, \quad (42)$$

onde tem-se que

$$\frac{\partial y}{\partial x} = tg\rho \quad \text{e} \quad (43)$$

$$\theta = \frac{\pi}{2} + \rho, \quad \text{ou seja} \quad \rho = \theta - \frac{\pi}{2}. \quad (44)$$

Substituindo, tem-se que

$$\frac{\partial y}{\partial x} = \operatorname{tg}\left(\theta - \frac{\pi}{2}\right), \quad (45)$$

$$\frac{\partial y}{\partial x} = -\left(\frac{1}{\operatorname{tg}\theta}\right). \quad (46)$$

Substituindo a Equação (46) na Equação (42), obtêm-se

$$2(x - x_c) + 2(y - y_c) \times \left(-\frac{1}{\operatorname{tg}\theta}\right) = 0. \quad (47)$$

Simplificando os termos

$$(x - x_c) - \frac{(y - y_c)}{\operatorname{tg}\theta} = 0. \quad (48)$$

Logo

$$(x - x_c) = \frac{(y - y_c)}{\operatorname{tg}\theta}, \quad (49)$$

$$(y - y_c) = (x - x_c)\operatorname{tg}\theta. \quad (50)$$

Substituindo a Equação (50) na Equação (41), têm-se

$$(x - x_c)^2 + (x - x_c)^2\operatorname{tg}^2\theta = r^2. \quad (51)$$

Isolando $(x - x_c)^2$, chega-se a

$$(x - x_c)^2 \times [1 + \operatorname{tg}^2\theta] = r^2. \quad (52)$$

Como

$$\operatorname{tg}^2\theta = \frac{\operatorname{sen}^2\theta}{\operatorname{cos}^2\theta}, \quad (53)$$

pode-se substituir a Equação (52), ficando com

$$(x - x_c)^2 \times \left[1 + \frac{\operatorname{sen}^2\theta}{\operatorname{cos}^2\theta}\right] = r^2, \quad (54)$$

e, em seguida,

$$(x - x_c)^2 \times \left[\frac{\operatorname{sen}^2\theta + \operatorname{cos}^2\theta}{\operatorname{cos}^2\theta}\right] = r^2. \quad (55)$$

Como $\operatorname{sen}^2\theta + \operatorname{cos}^2\theta = 1$, tem-se

$$(x - x_c)^2 \times \left[\frac{1}{\operatorname{cos}^2\theta}\right] = r^2, \quad (56)$$

que levará a

$$(x - x_c)^2 = r^2 \times \operatorname{cos}^2\theta. \quad (57)$$

Simplificando

$$(x - x_c) = r \times \cos \theta \quad (58)$$

chega a

$$x_c = x - r \times \cos \theta. \quad (59)$$

Substituindo a Equação (59) na Equação (50), encontra-se

$$(y - y_c) = r \times \cos \theta \times \operatorname{tg} \theta. \quad (60)$$

Como

$$\operatorname{tg} \theta = \frac{\operatorname{sen} \theta}{\cos \theta}, \quad (61)$$

concluí-se que

$$(y - y_c) = r \times \operatorname{sen} \theta, \quad (62)$$

leva a

$$y_c = y - r \times \operatorname{sen} \theta, \quad (63)$$

onde x e y definem as coordenadas do ponto no novo espaço de parametrização do círculo, x_c e y_c definem as coordenadas do centro, e o raio r .

Ao final, para a detecção de círculos, será utilizado um processo de eleição de círculos em que os votos são atribuídos aos pontos de passagem dos possíveis círculos existentes na imagem. Assim, os votos são acumulados em uma matriz de acumulação de votos, sendo que a detecção de um possível círculo é obtida quando um valor máximo (cume) é obtido no acumulador de votos (PISTORI; PISTORI; COSTA, 2005; KWATRA, 2000; HADAD, 2000).

Para calcular todos os valores de (x_c, y_c) para um determinado ponto (x, y) , geralmente não se utiliza a Equação (64),

$$r^2 = (x - x_c)^2 + (y - y_c)^2, \quad (64)$$

pois sua parametrização em função de x_c e y_c não produz, diretamente, implementações eficientes.

As equações mais utilizadas são baseadas na representação em coordenadas polares, como as Equações (65) e (66)

$$x_c = x - r \times \cos \theta \quad (65)$$

$$y_c = y - r \times \operatorname{sen} \theta. \quad (66)$$

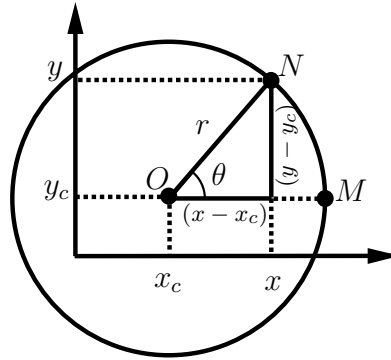


Figura 3.35: Circunferência de raio r e centro x_c e y_c .

É fácil verificar a validade das Equações (65) e (66) a partir da Figura 3.35 e dos conceitos de trigonometria. Tomando o ângulo θ da Figura 3.35, cujo seno é igual ao cateto oposto $(y - y_c)$ dividido pela hipotenusa r , e efetuando o cálculo da razão tem-se

$$\text{sen}\theta = \frac{y - y_c}{r}, \quad (67)$$

que resulta em

$$r \times \text{sen}\theta = y - y_c, \quad (68)$$

e, em seguida, em

$$y_c = y - r \times \text{sen}\theta. \quad (69)$$

Logo, o resultado expresso na Equação (69) também é uma coordenada polar descrita na Equação (65). Analogamente, efetuando o cálculo para o cosseno θ , sendo igual ao cateto adjacente $(x - x_c)$ dividido pela hipotenusa r , e efetuando o cálculo da razão tem-se

$$\text{cos}\theta = \frac{x - x_c}{r}, \quad (70)$$

que resulta em

$$r \times \text{cos}\theta = x - x_c, \quad (71)$$

e, em seguida, em

$$x_c = x - r \times \text{cos}\theta. \quad (72)$$

O resultado expresso na Equação (72) também é uma coordenada polar descrita na Equação (66).

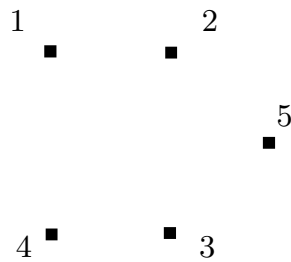


Figura 3.36: Plano imagem com 5 *pixels*.

O exemplo a seguir tem o objetivo de esclarecer o funcionamento da transformada de Hough circular. Dada uma imagem, Figura 3.36 com cinco *pixels*, representados por $[1, \dots, 5]$ deve-se identificar se há *pixels* que possam compor uma circunferência Δ_1 de raio $r = \frac{1}{\sqrt{2}}$.

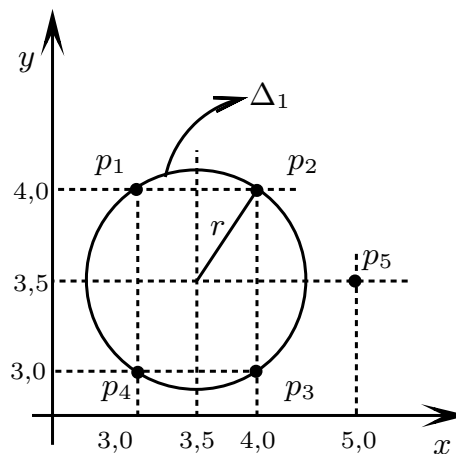


Figura 3.37: Circunferência mostrando os *pixels* no plano de parâmetros.

Os pontos da Figura 3.37 correspondem aos *pixels* da Figura 3.36. Através da transformada de Hough circular são identificados quais *pixels* compõem a circunferência Δ_1 . Pela Figura 3.37 pode-se observar, visualmente, que o ponto p_5 não faz parte da circunferência.

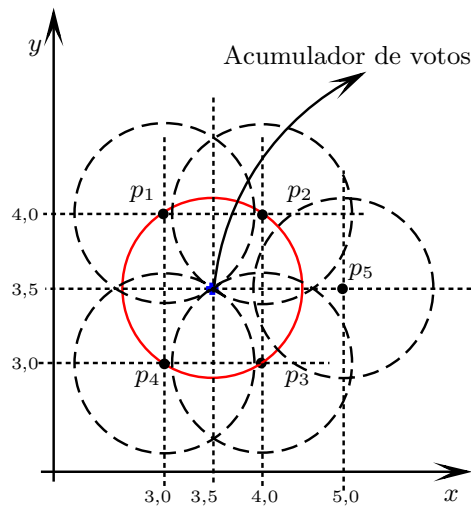


Figura 3.38: Plano de parâmetros de centro x_c, y_c para circunferências.

O acumulador de votos é então construído no plano de parâmetros para identificar o círculo desejado, apresentado na Figura 3.38 na cor vermelha. No círculo, cada *pixel* do plano imagem, Figura 3.36, torna-se uma circunferência. Pode-se observar que das cinco circunferências geradas na Figura 3.38, quatro se cruzam. Estas correspondem aos pontos p_1, p_2, p_3 e p_4 respectivamente.

Nota-se que o acumulador de votos registra o número total de curvas que passam por ele identificando o centro da circunferência no plano de parâmetros, coordenada $x_c = 3,5$ e $y_c = 3,5$. Desta forma conclui-se que o ponto p_5 realmente não compõe a circunferência no plano de parâmetros e que é possível localizar uma circunferência de raio $r = \frac{1}{\sqrt{2}}$ e obter sua equação que é dada pela Equação (64) e (73)

$$(x - 3,5)^2 + (y - 3,5)^2 = \left(\frac{1}{\sqrt{2}}\right)^2. \quad (73)$$

As dimensões do acumulador de votos devem ser definidas de forma a evitar perda de informações. Uma forma é considerar as dimensões iguais às da imagem no plano cartesiano, com tamanho (x_c, y_c) , onde o tamanho de x_c é igual ao tamanho de y_c . Todavia, neste caso, um *pixel* no plano cartesiano (imagem) que esteja ultrapassando os limites impostos pela borda (uma das quatro extremidades do plano imagem), provocaria a colocação de uma circunferência tendo como centro a borda, ultrapassando os limites do acumulador de votos, e assim perdendo informação. Para evitar essa perda, adotam-se as dimensões nas faixas, conforme Equação (74)

$$-raio \leq x_c \leq (X_{max} + raio) \quad \text{e} \quad -raio \leq y_c \leq (Y_{max} + raio) \quad (74)$$

onde X_{max} e Y_{max} são os maiores valores de x e y , respectivamente.

Logo, as dimensões do acumulador de votos serão dados por

$$acumulador[X_{max} + 2 * raio][Y_{max} + 2 * raio]. \quad (75)$$

A transformada de Hough circular apresenta no seu modo geral uma falha que é a geração de falsos picos. Esses picos são pontos (circunferência no espaço de imagem) que na realidade são decorrentes de *pixels* vizinhos do raio central da circunferência, os quais geram altos valores no acumulador de votos fazendo com que surjam circunferências que na realidade não existem.

Para contornar este problema foi proposta uma técnica conhecida por *backmapping* que proporciona meios para reduzir os falsos picos, que geralmente são encontrados na transformada de Hough circular (GERING; KLEIN, 1986; NIXON; AGUADO, 2008).

Algoritmo 1: Transformada de Hough circular.

Entrada: $f(x, y)$, *corfundo*, r_{max} , r_{min} .

Saída: *accum*.

```

1 início
2   para todo  $f(x, y) \neq \text{corfundo}$  faça
3     para todo  $r \leftarrow r_{min}$  até  $r_{max}$  faça
4       para todo  $\theta \leftarrow 0$  até 360 faça
5          $x_c \leftarrow x - \cos(\theta)r$ ;
6          $y_c \leftarrow y - \text{sen}(\theta)r$ ;
7          $\text{accum}(x_c, y_c, r) \leftarrow \text{accum}(x_c, y_c, r) + 1$ ;
8       fim
9     fim
10  fim
11 fim

```

O Algoritmo 1 mostra a descrição do procedimento para utilização da transformada de Hough circular, obtendo a matriz de acumulação de votos inicial.

No Algoritmo Hough circular, primeiramente são informados os parâmetros da imagem $f(x, y)$, a cor do fundo *corfundo* para distinção entre o objeto a ser encontrado e o fundo da imagem e o intervalo do raio, $[r_{min}, r_{max}]$ que são os raios dos círculos que faz o contorno de cada objeto encontrado.

O primeiro laço de repetição (linha 2 do Algoritmo 1) ocorre nos parâmetros da imagem $f(x, y)$ a procura do fundo, *corfundo* e do objeto. No segundo laço de repetição (linha 3 do Algoritmo 1) é utilizado o intervalo de raios $[r_{min}, r_{max}]$ e é traçado um círculo para cada ponto encontrado como borda do objeto.

No terceiro laço de repetição (linha 4 do Algoritmo 1) percorre-se toda a volta do objeto através do ângulo θ no intervalo $[0^\circ, 360^\circ]$ para cada tamanho de raio, para o cálculo de possíveis coordenadas de centro (linhas 5 e 6 do Algoritmo 1). Finalmente, o acumulador de votos obtém as coordenadas do centro (x_c, y_c) e seu respectivo raio r (linha 7 do Algoritmo 1). Incrementando de uma unidade a posição correspondente a cada conjunto de parâmetros encontrado.

3.8.3 Backmapping

Gering e Klein (GERING; KLEIN, 1986) introduziram um algoritmo denominado *backmapping* com o objetivo de enfatizar as partes mais relevantes no acumulador de votos, minimizando a geração de falsos picos decorrentes de *pixels* vizinhos do raio central da circunferência.

O Algoritmo 2 utiliza um novo acumulador *accumBack* para a transformada de Hough circular com as mesmas dimensões do acumulador de votos *accum* e executa o mesmo procedimento que a transformada de Hough circular do Algoritmo 1. Porém em vez de atribuir votos no arranjo acumulador *accumBack*, armazenam-se os parâmetros que identificam a posição no arranjo acumulador votos *accum* que possui o maior voto no Algoritmo 1.

Depois de localizada a posição de maior voto obtido pelo mapeamento para o espaço paramétrico de Hough, de um dado *pixel* na imagem, essa mesma posição é agora incrementada no acumulador *accumBack* de Hough. Isso faz com que se mantenham apenas os pontos *mais prováveis* de se encontrar segmentos realmente significativos. Esse processo tem como característica principal a redução significativa do tempo de análise do espaço de Hough e permite ainda uma diminuição do valor de limiar.

Algoritmo 2: Realização *backmapping*.

Entrada: *accum*, r_{min} , r_{max} , $valor_{max}$.

Saída: *accumBack*.

```

1 início
2   para todo  $accum(x_c, y_c, r)$  faça
3     para todo  $r \leftarrow r_{min}$  até  $r_{max}$  faça
4       se  $accum(x_c, y_c, r) > valor_{max}$  então
5          $valor_{max} \leftarrow accum(x_c, y_c, r)$ 
6          $x_{cmax} \leftarrow x_c$ ;
7          $y_{cmax} \leftarrow y_c$ ;
8       fim
9      $accumBack(x_{cmax}, y_{cmax}, r) \leftarrow$ 
10     $accumBack(x_{cmax}, y_{cmax}, r) + 1$ ;
11   fim
12 fim
```

O Algoritmo 2 mostra a descrição do procedimento para utilizar o *backmapping* empregando um novo acumulador da transformada de Hough circular.

O primeiro laço de repetição (linha 2 do Algoritmo 2), utiliza o vetor denominado *accum* da transformada de Hough circular descrita no Algoritmo 1 e os parâmetros de raios no intervalo $[r_{min}, r_{max}]$ e o parâmetro de valor máximo ($valor_{max}$) que serve como filtro eliminando os menores valores para o novo acumulador *accumBack*.

No segundo laço de repetição (linha 3 do Algoritmo 2) é utilizado o intervalo de raios $[raio_{min}, raio_{max}]$ em seguida é analisada a condição do acumulador (linha 4 do Algoritmo 2) de votos caso seja maior do que o parâmetro do valor máximo ($valor_{max}$), então o valor máximo recebe o valor do acumulador de votos (linha 5 do Algoritmo 2) e fornece o valor dos prováveis centros (x_c, y_c) e seu respectivo raio r (linha 6 e 7 do Algoritmo 2). Incrementando de uma unidade a posição correspondente cada conjunto de parâmetros encontrado (linha 9 do Algoritmo 2).

4 RESULTADOS

Em geral, a aquisição das imagens envolve muitos problemas tecnológicos tais como o excesso de iluminação e, em geral, é feito por pessoal não treinado. A presença de ruído na imagem é sempre esperada e a não observação de alguns detalhes aumenta ainda mais a presença deles nas imagens obtidas. Uma das maiores contribuições deste trabalho é ter a pretensão de tratar imagens adquiridas da forma mais natural possível. Espera-se que o protótipo desenvolvido supere as dificuldades com a presença de ruído excessivo e contribua para minimizar os problemas tecnológicos no momento da aquisição das imagens.

O desenvolvimento do protótipo é feito no ambiente conhecido como *Matrix Laboratory* (MATLAB, 2007) baseado na plataforma *Windows XP*.



Figura 4.1: Imagem de entrada.

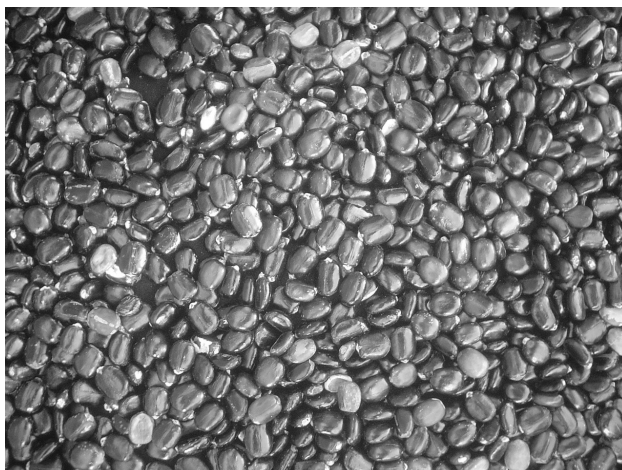


Figura 4.2: Suavização em níveis de cinza.

A imagem de entrada para o protótipo, mostrada na Figura 4.1 é lida e em seguida é feita a suavização em tons de cinza Figura 4.2. A função de suavização retorna uma matriz onde cada posição da matriz representa um *pixel* e seu valor representa um tom de cinza do referido *pixel*.

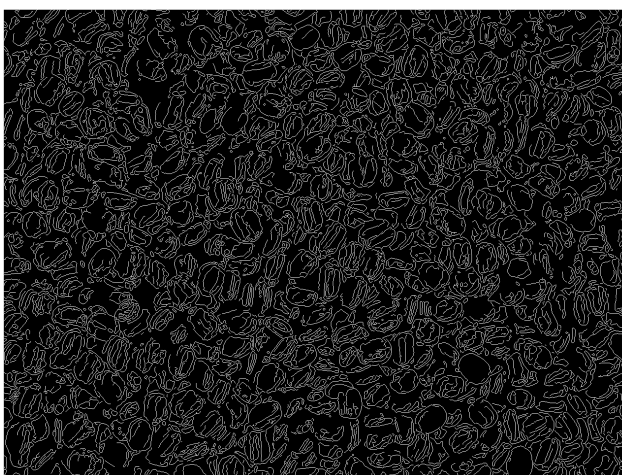


Figura 4.3: Bordas detectadas pelo método de *Canny*.

A Figura 4.3 mostra o resultado da aplicação do método de *Canny* para a detecção do contorno das sementes de mamona na imagem. Pode ser visto que os contornos são fracos e descontínuos. Os resultados obtidos com os outros métodos tradicionais, *Sobel*, *Prewitt* e *Roberts*, não são melhores que o resultado obtido com o método de *Canny*.

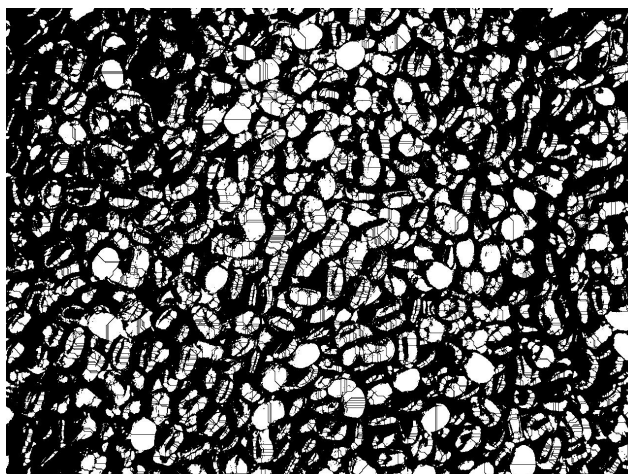


Figura 4.4: Segmentação usando *watershed* morfológicas.

O algoritmo baseado na transformada de *watershed* morfológica é utilizado para detectar as divisões entre as sementes. O método de segmentação *watershed* é aplicado na imagem que passa pelas técnicas de pré-processamento previamente (*Sobel*, *Prewitt*, *Roberts* e *Canny*), e o resultado também não é satisfatório. A Figura 4.4 mostra o resultado obtido com aplicação da transformada *watershed*.

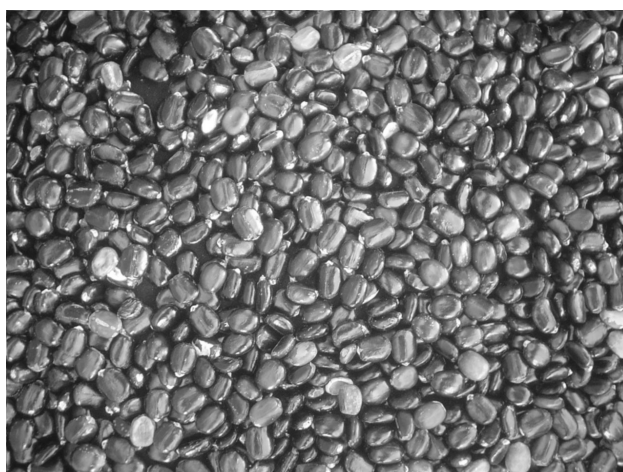


Figura 4.5: Filtro espacial com gaussiana, máscara (3×3) e convolução circular.

A solução dos problemas para a detecção de borda e a segmentação das sementes na imagem é feito pelo realce empregando a filtragem espacial gaussiana com uma máscara (3×3) e a convolução com uma máscara circular com objetivo de suavizar o ruído excessivo presente na imagem. A Figura 4.5 mostra o resultado da aplicação da filtragem espacial gaussiana com máscara (3×3) e a convolução com máscara circular.

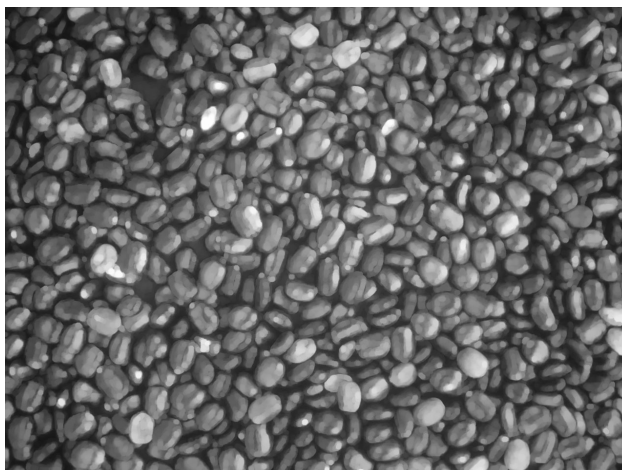


Figura 4.6: Morfologia de abertura.

Na etapa seguinte é feito o processamento morfológico da imagem com a morfologia dilatação e erosão, mas os resultados foram insatisfatórios para o isolamento do objeto e fundo. A morfologia de abertura com elemento estruturante em disco de (5×5) ofereceu melhor resultado para o isolamento das sementes. O resultado pode ser visto na Figura 4.6.

A morfologia de abertura apresenta um baixo contraste. O problema de baixo contraste é resolvido com a aplicação do histograma adaptativo que otimiza o contraste da imagem. O histograma adaptativo usado consiste em realçar o contraste da imagem operando em regiões pequenas de (8×8) *pixels* para evitar amplificação de ruído.



Figura 4.7: Histograma adaptativo.

O realce das sementes de mamona pode ser visto na Figura 4.7. A Figura 4.6 mostra o realce de contraste feito mapeando as variações dentro do intervalo de tons de cinza (V_{min}, V_{max}) para a variação máxima do dispositivo de visualização que, geralmente, é o intervalo $[0, 255]$. V_{min} e V_{max} são os valores de níveis de cinza mínimos e máximos, respectivamente mostrados na

Figura 4.6. Com esse procedimento obtêm-se a imagem clareada mostrada na Figura 4.7.

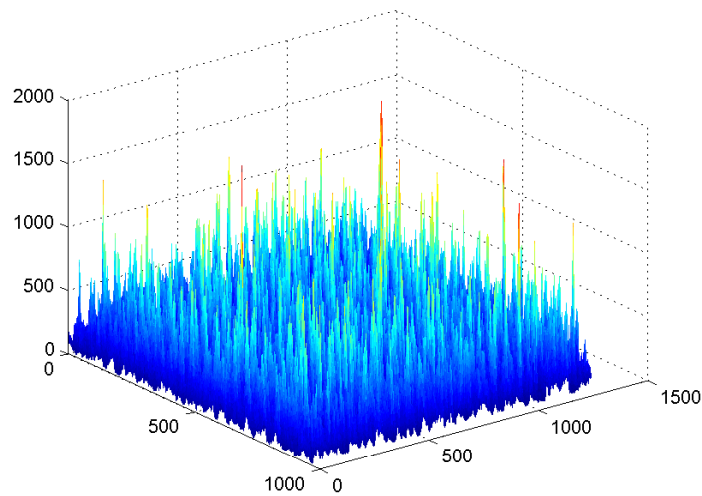


Figura 4.8: Matriz de acumulação de votos com dimensão de 1282×962 .

Para a solução do problema da segmentação das sementes aplica-se a transformada de Hough circular. Para converter cada *pixel* da borda da imagem em uma circunferência na matriz de acumulação de votos. Esse processo de conversão dos *pixels* da borda para a matriz de acumulação de votos é mostrada na Figura 4.8.

A Figura 4.8 mostra a matriz de acumulação de votos numa perspectiva tridimensional após a realização da transformada de Hough circular configurada para detectar círculos com raios de 25 *pixels*. A matriz de acumulação de votos é uma matriz com dimensão $(largura + (2 * raio)) \times (altura + (2 * raio))$ cujas posições são incrementadas de acordo com o traçado da circunferência.

Os picos observados na Figura 4.8 representam o número de intersecções que ocorrem nas posições da matriz. O maior pico indica o centro da circunferência na imagem. A matriz de acumulação de votos também pode ser referida como o arranjo acumulador da transformada de Hough circular.

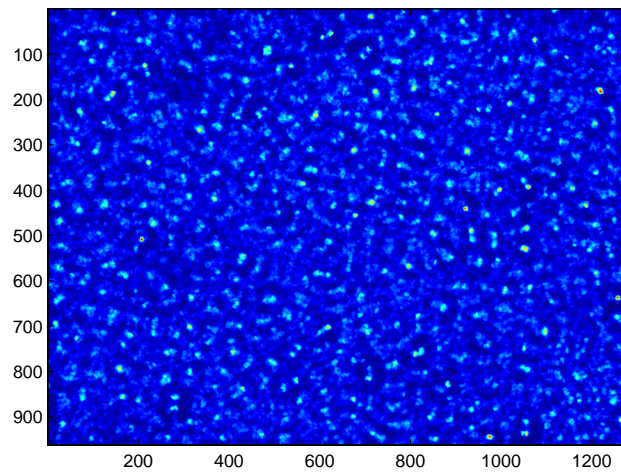


Figura 4.9: Representação 2D da matriz de acumulação de votos.

A abordagem empregada nesse trabalho é a geração de uma matriz de acumulação de votos tridimensional no qual é adicionada uma dimensão a matriz de acumulação de votos bidimensional, Figura 4.9 (*largura* \times *altura*) que se refere aos possíveis raios de curvatura (PEREIRA, 1995; MARTINEZ, 2002; ATIQUZZAMAN, 1999).

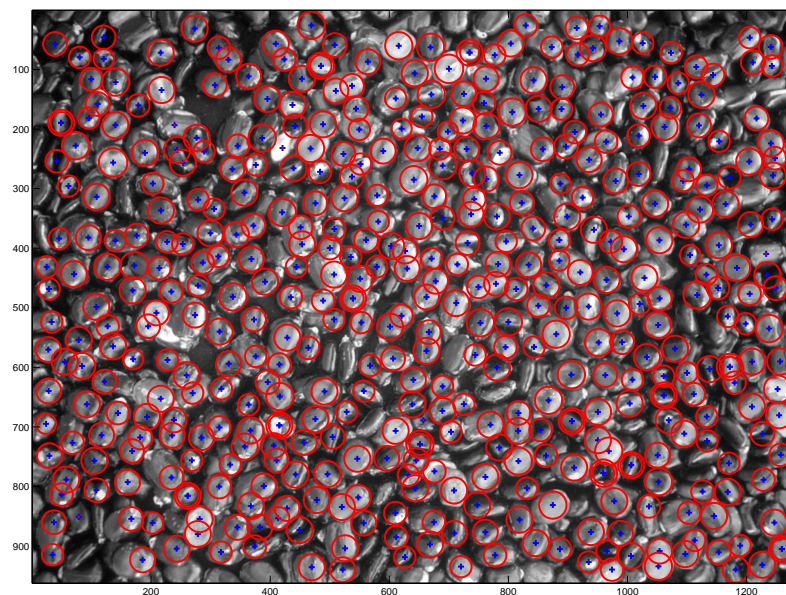


Figura 4.10: Imagem com círculos localizados (centros x_c, y_c).

O resultado obtido com a transformada de Hough circular é produzido com a implementação do Algoritmo 1. Contudo, como visto na Seção terciária 3.8.2 da Seção secundária 3.8,

o algoritmo possui uma falha sendo solucionado pela implementação do Algoritmo 2 de *back-mapping* para reduzir os falsos picos, que geralmente são encontrados na transformada de Hough circular. O Algoritmo 1 Hough circular depende dos parâmetros imagem de entrada Figura 4.7 histograma adaptativo e raios $[r_{min}, r_{max}]$ gerando o acumulador de votos. O Algoritmo 2 depende dos parâmetros acumulador de votos gerados pela transformada de Hough circular e valor máximo. Como saída é criado o acumulador de votos com centros (x_c, y_c) representado pela Matriz (76) e raios (r) representado pela Matriz (77). E por fim, os valores de centros (x_c, y_c) e raios r são utilizados para produzir as circunferências e seus centros conforme Figura 4.10.

$$\begin{aligned}
\mathbf{XY} &= \begin{pmatrix} x_1 & y_1 \\ x_2 & y_2 \\ x_3 & y_3 \\ \vdots & \vdots \\ x_{153} & y_{153} \\ x_{154} & y_{154} \\ x_{155} & y_{155} \\ \vdots & \vdots \\ x_{253} & y_{253} \\ x_{254} & y_{254} \\ x_{255} & y_{255} \\ \vdots & \vdots \\ x_{353} & y_{353} \\ x_{354} & y_{354} \\ x_{355} & y_{355} \\ \vdots & \vdots \\ x_{453} & y_{453} \\ x_{454} & y_{454} \\ x_{455} & y_{455} \\ \vdots & \vdots \\ x_{514} & y_{514} \\ x_{515} & y_{515} \\ x_{516} & y_{516} \end{pmatrix} = \begin{pmatrix} 24, 4693 & 430, 2653 \\ 28, 3101 & 467, 9625 \\ 33, 1229 & 523, 1180 \\ \vdots & \vdots \\ 418, 7319 & 338, 5655 \\ 414, 9080 & 648, 2289 \\ 412, 1999 & 851, 9140 \\ \vdots & \vdots \\ 655, 6936 & 688, 9696 \\ 661, 0921 & 571, 4522 \\ 667, 7934 & 61, 9958 \\ \vdots & \vdots \\ 895, 6512 & 227, 8926 \\ 908, 6883 & 766, 9111 \\ 903, 9745 & 688, 1656 \\ \vdots & \vdots \\ 1152, 3790 & 393, 9675 \\ 1149, 6016 & 541, 9280 \\ 1150, 8500 & 465, 3886 \\ \vdots & \vdots \\ 1083, 8570 & 354, 1552 \\ 1136, 7803 & 598, 5617 \\ 1171, 0605 & 598, 1849 \end{pmatrix} \\
\mathbf{R} &= \begin{pmatrix} rad_1 & 18 \\ rad_2 & 18 \\ rad_3 & 18 \\ \vdots & \vdots \\ rad_{153} & 23 \\ rad_{154} & 22 \\ rad_{155} & 20 \\ \vdots & \vdots \\ rad_{253} & 20 \\ rad_{254} & 22 \\ rad_{255} & 23 \\ \vdots & \vdots \\ rad_{353} & 21 \\ rad_{354} & 22 \\ rad_{355} & 22 \\ \vdots & \vdots \\ rad_{453} & 23 \\ rad_{454} & 17 \\ rad_{455} & 20 \\ \vdots & \vdots \\ rad_{514} & 18 \\ rad_{515} & 18 \\ rad_{516} & 17 \end{pmatrix} = \begin{pmatrix} 18 \\ 18 \\ 18 \\ \vdots \\ 23 \\ 22 \\ 20 \\ \vdots \\ 20 \\ 22 \\ 23 \\ \vdots \\ 21 \\ 22 \\ 22 \\ \vdots \\ 23 \\ 17 \\ 20 \\ \vdots \\ 18 \\ 18 \\ 17 \end{pmatrix} \tag{76} \\
&= \begin{pmatrix} rad_1 & 18 \\ rad_2 & 18 \\ rad_3 & 18 \\ \vdots & \vdots \\ rad_{153} & 23 \\ rad_{154} & 22 \\ rad_{155} & 20 \\ \vdots & \vdots \\ rad_{253} & 20 \\ rad_{254} & 22 \\ rad_{255} & 23 \\ \vdots & \vdots \\ rad_{353} & 21 \\ rad_{354} & 22 \\ rad_{355} & 22 \\ \vdots & \vdots \\ rad_{453} & 23 \\ rad_{454} & 17 \\ rad_{455} & 20 \\ \vdots & \vdots \\ rad_{514} & 18 \\ rad_{515} & 18 \\ rad_{516} & 17 \end{pmatrix} \tag{77}
\end{aligned}$$

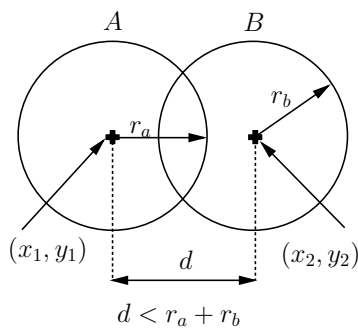


Figura 4.11: Círculos secantes.

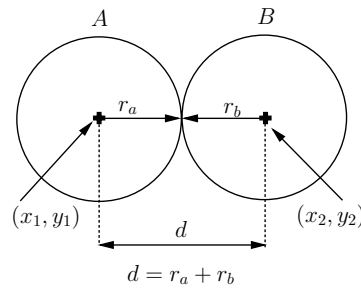


Figura 4.12: Círculos tangenciais externos.

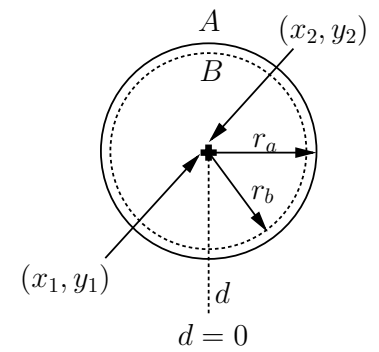


Figura 4.13: Círculos concêntricos.

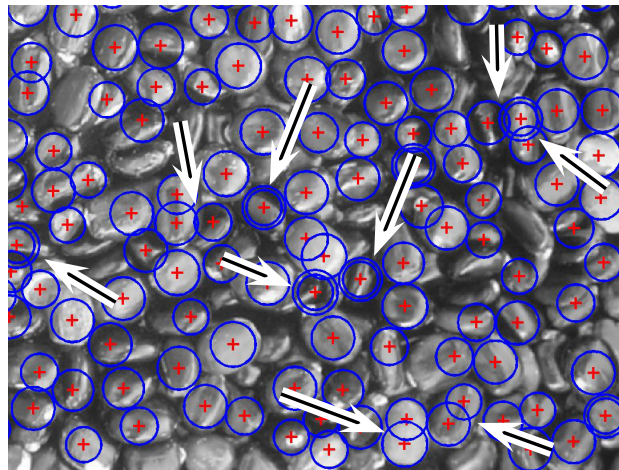


Figura 4.14: Recorte da Figura 4.10 dos pontos localizados pela transformada de Hough circular que apresentam anomalia.

A Figura 4.14 mostra a existência de três tipos de círculos formados pelas sementes na imagem, indicados pelas setas: círculos concêntricos, círculos tangenciais externos e círculos secantes. Os círculos concêntricos indicam o mesmo centro e raios diferentes para as sementes diferentes na imagem. Isso significa que existe a sobreposição das sementes. A Figura 4.13 ilustra essa situação. Os círculos secantes possuem dois pontos distintos em comum e significa que existe a sobreposição de parte das sementes. Essa situação é ilustrada pela Figura 4.11.

A situação ideal para a detecção automática das sementes da imagem é ilustrada pela Figura 4.12. Nessa situação é possível estabelecer a existência de duas sementes sem problema. Nos outros casos, não é possível fazer a detecção automática das sementes. Assim, é necessário extrair da imagem as sementes que correspondem aos círculos concêntricos e secantes.

Tabela 1: Acurácia do Experimento.

Descrição	Deteção automática	Porcentagem
Amostras não localizadas	49	8,67%
Amostras localizadas	516	91,33%
Círculos concêntricos	18	3,19%
Círculos secantes	256,5	45,40%
Círculos ideais	241,5	42,74%

O resultado obtido com a deteção automática é mostrada na Tabela 1. Sementes com círculos concêntricos são 18, com círculos secantes 256,5. Assim, são excluídas 274,5 sementes da imagem.

O procedimento para a retirada das sementes da imagem é baseado na distância entre dois pontos dado por

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}. \quad (78)$$

Vale lembrar que a imagem é uma função $f(x, y)$ discretizada no espaço e nesse caso é possível ordenar os valores da Matriz (76) em relação ao eixo x (abcissa). A distância entre todos os pontos localizados pela transformada de Hough circular são calculados a partir da Matriz (76). A distância encontrada é comparada com a distância dos raios de cada círculo armazenado na Matriz (77). Dessa maneira as sementes com raios concêntricos e secantes são excluídos da imagem.

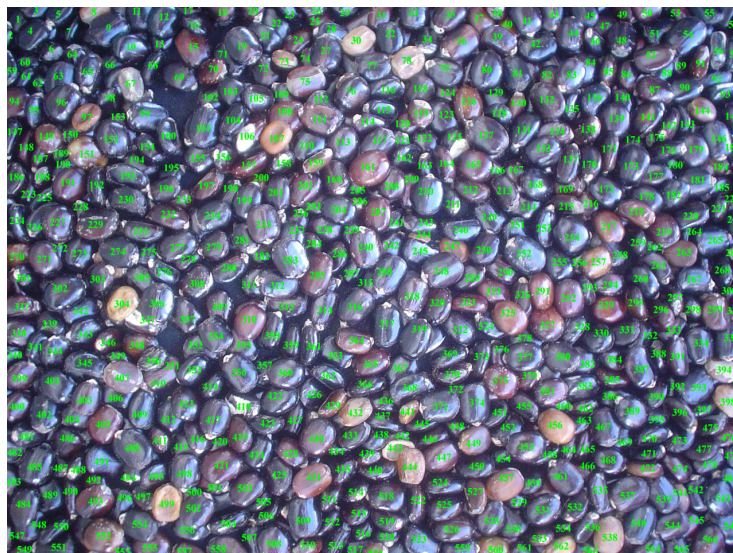


Figura 4.15: Deteção com marcação através de numeração nas sementes de mamona.

Na contagem manual encontra-se 565 sementes na imagem. Para garantir que todas as sementes são contadas um número é colocado sobre cada semente na imagem como mostra a

Figura 4.15. O resultado obtido com a detecção automática é mostrada na Tabela 1. Sementes com círculos concêntricos são 18, com círculos secantes 256,5. Assim, são excluídas 274,5 sementes da imagem.

5 CONCLUSÃO

São introduzidos conceitos clássicos tais como suavização em tons de cinza, suavização gaussiana e transformadas de Hough, dentre outros, visando a detecção das sementes de mamona em imagens digitais fotográficas.

Resultados obtidos com métodos tradicionais para a detecção de borda tais como *Sobel*, *Prewitt*, *Roberts* e *Canny* assim como a transformada de *watershed* são apresentados como parte do experimental de busca do método adequado. Finalmente, obteve-se resultado convincente com o emprego transformada de Hough circular.

A solução para o problema da detecção de borda é feito com a morfologia de abertura. A aplicação da morfologia melhora a relação do objeto com o fundo e deixa o fundo mais destacado o que provoca o escurecimento de toda a imagem. Maior clareamento da imagem é feito com o histograma adaptativo. Na sequência é aplicada a transformada de Hough circular. A redução dos picos falsos, geralmente encontrados na transformada de Hough circular, é feito com o algoritmo *backmapping* para determinar apenas os pontos mais prováveis para encontrar os segmentos significativos para a localização dos círculos relacionados com as sementes de mamona.

Uma das principais dificuldades superadas no trabalho é a presença de situações anômalas devido a má distribuição das sementes no momento em que as fotografias são feitas. É verificada a existência de círculos concêntricos e secantes o que interfere na eficácia da detecção automática. Esse problema é resolvido com extração das sementes com estas características para permitir a detecção automática.

O uso do protótipo desenvolvido mostrou que o excesso de iluminação presente na imagem é um complicador para a detecção automática. O ideal é que o houvesse um cuidado maior com ambiente para a diminuição de ruído. Recomenda-se ainda o espalhamento adequado das sementes para evitar a sobreposição ou a quase sobreposição e principalmente a remoção do fundo coloração preta.

A detecção automática identifica 42,74% das sementes presentes na imagem. Vale ressaltar, que nenhuma preparação do ambiente é feita no momento da aquisição das imagens. Isso significa baixo custo experimental quando comparado com outros meios de aquisição de dados como é o caso de um espectrômetro ou outros. O treinamento de pessoal técnico nesse caso é

quase inexistente porque o manuseio de uma máquina fotográfica é bastante popular.

Como trabalho futuro, pretende-se fazer a classificação dos objetos presentes na imagem de entrada (imagem da Figura 2.1) como identificar a classe de cada semente presente na imagem com o emprego de Redes Neurais Artificiais, por exemplo.

REFERÊNCIAS

ACHARYAM, T.; RAY, A. K. **Image Processing Principles and Applications**. Noboken, New Jersey, USA: John Wiley & Sons, 2005. 427p.

ATIQUZZAMAN, M. Coarse-to-Fine Search Technique to Detect Circles in Images. **The International of Advanced Manufacturing Techonolgy**, London, v.15, p.96–102, 1999.

BLANCHET, G.; CHARBIT, M. **Digital Signal and Image Processing using MATLAB**. Newport Beach, California, USA: ISTE, 2006. 747p.

BRASIL. Lei nº 11.097, de 13 de janeiro de 2005. **Casa Civil - Subchefia para Assuntos Jurídicos**, Brasília, DF, Jan. 2005. Disponível em: <http://www.planalto.gov.br/ccivil_03/_ato2004-2006/2005/Lei/L11097.htm>. Acesso em: 30 out. 2009.

BRASIL. Ministério da Agricultura, Pecuária e Abastecimento. **Plano Nacional de Agroenergia 2006-2011**, Brasília, DF, p.120, 2005.

BRASIL. **Balanco Energético Nacional 2008**: ano base 2007. Rio de Janeiro: EPE, 2008. 244p. Disponível em: <https://ben.epe.gov.br/downloads/Relatorio_Final_BEN_2008.pdf>. Acesso em: 30 out. 2009.

CASTLEMAN, K. R. **Digital Image Processing**. 2.ed. New Jersey,USA: Prentice Hall, 1995. 667p.

CONAB. **Acompanhamento de safra brasileira**: grãos, intenção de plantio, primeiro levantamento, 2009. Brasília: Companhia Nacional de Abastecimento, 2009. 39 p.

DUDA, R. O.; HART, P. E. Use of the Hough transformation to detect lines and curves in pictures. **Commun. ACM**, New York, NY, USA, v.15, n.1, p.11–15, 1972.

FARID, H. **Fundamentals of Image Processing**. Hanover, New Hampshire, USA: Dartmouth College, 2007. 69p.

FILHO, A. S. Melhoramento da mamona. In: OLIVEIRA, A. B. (Ed.). **Melhoramento de espécies cultivadas**. Viçosa, MG: UFV, 1999. p. 385–407.

FILHO, O. M.; NETO, H. V. **Processamento Digital de Imagens**. 2.ed. Rio: Brasport, 1999. 331p.

FILHO, P. P.; VILLANO, F. O emprego do éster da mamona nos motores dos veículos Fiat. In: CONGRESSO BRASILEIRO DE ENERGIA, 3, 1984, Rio de Janeiro,RJ. **Anais...** Rio de Janeiro: CBE, 1984. p. 903–912.

GERING, G.; KLEIN, F. Fast contour identifcation through efficient hough transform and simplified interpretation. **8th International Joint Conference on Pattern Recognition**, Paris, France, v.1, p.498–500, Oct 1986.

GOMES, J.; VELHO, L. **Computação Gráfica: imagem**. 2.ed. Rio de Janeiro, RJ: Instituto Nacional de Matemática Pura e Aplicada (IMPA), 2002. 373p.

GONZALES, R. C.; WOODS, R. E. **Processamento Digital de Imagens**. 3.ed. São Paulo: Pearson Prentice-Hall, 2009. 623p.

GONZALEZ, R. C.; WOODS, R. E. **Digital Image Processing**. 2.ed. New Jersey, USA: Pearson Prentice-Hall, 2002. 793p.

HADAD, R. M. **Identificação de Padrões em Imagens de Satélites –Formas Circulares Resumo**. 2000. 219p. Tese (Doutorado em Ciência da Computação) — Pós- Graduação em Ciência da Computação da Universidade Federal de Minas Gerais, Belo Horizonte, MG.

HOUGH, P. V. C. Machine analysis of bubble chamber pictures. In: INTERNATIONAL CONFERENCE ON HIGH ENERGY ACCELERATORS AND INSTRUMENTATION, 16, 1959, Switzerland. **Anais...** CERN, 1959.

HOUGH, P. V. C. Method and means for recognizing complex patterns. **United States Patent Office**, USA, n.3069654, p.6, dez 1962. Disponível em: <<http://www.freepatentsonline.com/3069654.html>>. Acesso em: 18 set. 2009.

JAHNE, B. **Digital Image Processing**. 5.ed. Berlin, Germany: Springer-Verlag Berlin, Inc, 2002. 575p.

KWATRA, V. Detecting Coins using Hough Transform. **College of Computing - GeorgiaTech**, California, USA, Nov. 2000. Disponível em: <http://www.cc.gatech.edu/~kwatra/computer_vision/coins/coins.html>. Acesso em: 30 out. 2009.

LUCENA, A. M. A. de. **Qualidade das sementes de mamona, momento adequado para colheita de cachos e sua influência no valor da produção**. 2009. 148p. Tese (Doutorado em Processos Ambientais) — Programa de Pós-Graduação em Recursos Naturais, Universidade Federal de Campina Grande, Campina Grande, PB.

MACÊDO BELTRÃO, N. E. de. Ministério da Agricultura, Pecuária e Abastecimento. **Documentos 129: A Cadeia da Mamona no Brasil, com Ênfase para o Segmento P&D: Estado da Arte, Demandas de Pesquisa e Ações Necessárias para o Desenvolvimento**, Campina Grande, PB, p. 19, 2004.

MACÊDO BELTRÃO, N. E. de; CARDOSO, G. D.; SEVERINO, L. S.; PEREIRA, J. R.; GONDIM, T. M. D. S.; CARTAXO, W. V. Biodiesel do óleo da mamona e a produção de fitomassa: considerações gerais e singularidade. **Embrapa Brasileira de Pesquisa Agropecuária**, Campina Grande, PB, p.1–2, 2004.

MARTINEZ, A. C. **Um novo método para medidas de gotas de chuva com técnicas do processamento digital de imagens**. 2002. 186p. Dissertação (Mestrado em Engenharia Elétrica) — Programa de Pós-Graduação em Engenharia Elétrica, Universidade São Paulo, São Carlos, SP.

MATHERON, G. **Random Sets and Integral Geometry**. New York, USA: John Wiley & Sons, 1974. 261p.

MATLAB. **Image Processing Toolbox 6.4**. The MathWorks: Release Notes 7.4.0.287, 2007.

MCANDREW, A. **An Introduction to Digital Image Processing with Matlab**. Melbourne, Austrália: Victoria University of Technology, 2004. 226p.

NIXON, M.; AGUADO, A. **Feature Extraction & Processing**. 2.ed. London, UK: Elsevier Ltd, 2008.

PADILHA, F. R. R.; KHATCHATOURIAN, O. Redes neurais aplicadas na identificação de variedades de soja. In: CONGRESSO NACIONAL DE MATEMÁTICA APLICADA E COMPUTACIONAL, 30, 2007, Florianópolis,SC. **Anais...** Florianópolis: CNMAC, 2007. p.7.

PARKER, J. R. **Algorithms for Image Processing and Computer Vision**. New York,USA: John Wiley & Sons, Inc., 1996. 417p.

PEREIRA, A. S. **Processamento de imagens médicas utilizando a Transformada de Hough**. 1995. 263p. Tese (Doutorado em Física Computacional) — Programa de Pós-Graduação em Física Computacional, Universidade de São Paulo, São Paulo.

PIMENTEL, M. G.; NUNES, S. Avaliação da relação custo/volume/lucro no cultivo da mamona destinada à fabricação do biodiesel no agreste nordestino,. **Custos e agronegócio**, Recife, PE, v.4, n.2, p.23, maio-ago 2008.

PISTORI, H.; NETO, J. J.; COSTA, E. R. Utilização de Tecnologia Adaptativa na Detecção da Direção do Olhar. **SPC Magazine**, Lima,Perú, v.2, n.2, p.22–29, maio 2003.

PISTORI, H.; PISTORI, J.; COSTA, E. R. Hough-Circles: um módulo de detecção de circunferências para o imagej. In: WORKSHOP DE SOFTWARE LIVRE, 6, 2005, Porto Alegre,RS. **Anais...** Porto Alegre: WSL, 2005. p.2–5.

PRATT, W. K. **Digital Image Processing PIKS Scientific Inside**. 4.ed. Los Altos, California, USA: John Wiley & Sons, Inc., 2007. 769p.

RAMALHO, G.; MEDEIROS, F. Um sistema de inspeção visual automático aplicado à classificação e seleção de laranjas. In: CONGRESSO BRASILEIRO DA SOCIEDADE BRASILEIRA DE INFORMÁTICA APLICADA À AGROPECUÁRIA E AGROINDÚSTRIA , 4, 2003, Porto Seguro,BA. **Anais...** Porto Seguro: SBIAGRO, 2003. v.2, p.197–200.

RAMOS, P. Z. A. **Segmentação de imagens ultrassonográficas para detecção de nódulos**. 2010. 138 p. Dissertação (Mestrado em Engenharia Elétrica) — Programa de Pós-Graduação em Engenharia Elétrica, Universidade São Paulo, São Carlos,SP.

RODRIGUES, R. F. D.; OLIVEIRA, F. D.; FONSECA, A. M. As folhas de Palma Christi - Ricinus communis L. Euphorbiaceae jussieu. **Revista Lecta**, Bragança Paulista, v.20, n.2, p. 183–194, jul-dez 2002.

SERRA, J. **Image Analysis and Mathematical Morphology**. London: Academic Press, 1982. 610p.

SOILLE, P. **Morphological Image Analysis: principles and applications**. 2.ed. Secaucus, NY, USA: Springer-Verlag New York, Inc., 2003. 391p.

STAUGAARD, A. C. **Robotics and AI: an introduction to applied machine intelligence**. New Jersey, USA: Prentice-Hall, 1987.

STORKEY, A. J.; HAMBLY, N. C.; WILLIAMS, C. K. I.; MANNAND, R. G. Cleaning Sky Survey Databases using Hough Transform and Renewal String Approaches. **Monthly Notices of the Royal Astronomical Society**, London, n.347, p.37, fev 2004.

WEISS, E. A. **Oilseed Crops**. 2.ed. London: Blackwell Science Ltd, 2000. 364p. (Word Agriculture series).

YOUNG, I. T.; GERBRANDS, J. J.; VLIET, L. J. V. **Fundamentals of Image Processing**. 2.ed. Delft,Netherlands: Delft University of Technology, 1998. 111p.

GLOSSÁRIO

Assintótica	A curva é assintótica; isto é, estende-se de $-\infty$ a $+\infty$, sem nunca tocar o eixo horizontal, e portanto a função de x jamais se anula.
Biodiesel	É um éster de ácido graxo, renovável e biodegradável, obtido comumente a partir da reação química de óleos ou gorduras, de origem animal ou vegetal, com um álcool na presença de um catalisador (reação conhecida como transesterificação). Pode ser obtido também pelos processos de craqueamento e esterificação.
Deiscência	Abertura espontânea, por maturidade, de estruturas de uma planta, como fruto ou esporângio.
Estádio	Fase, período, época, estação.
Isotrópica	Circularmente simétrica.
Istmo	Formação estreita que promove a união entre duas outras maiores.
Mamoneira	Planta medicinal da família das euforbiáceas (<i>Ricinus communis L.</i>), de fruto capsular ovóide, achatado, de tamanho variável, com superfície lisa, brilhante e acinzentada, e da qual se extrai o óleo de rícino; mamona, mamoneira, carrapateira, carrapateira-branca, carrapato, caturra.
Maturação	Ato ou efeito de maturar; amadurecimento.
Protuberâncias	Coisa saliente.
Protusões	Que avança ou sai para fora do plano a que está unido: que sobressai ou resalta.


```

44 img_con = imfilter(img_gra,w,'conv','circular','same');
45 imwrite(img_con,'resultado_img_con.jpg');%Gravando resultado
46 %Filtro espacial linear
47 %-----
48 %           Morfologia Abertura
49 %-----
50 se = strel('disk',5); %Elemento estruturante disco de tam. cinco
51 img_ope = imopen(img_con,se);
52 imwrite(img_ope,'resultado_img_ope.jpg');%Gravando resultado
53 %
54 %-----
55 %           Histograma Adaptativo
56 %-----
57 img_his = adapthisteq(img_ope);
58 rawimg = img_his;
59 imwrite(img_his,'resultado_img_his.jpg');%Gravando resultado
60 tic; % inicio de tempo
61 %
62 %-----
63 %           Transformada de Hough Circular
64 %-----
65 %rawing, [15,25], 5, 20
66 [accum, circen, cirrad] = CircularHough_Grd(rawimg, [15,25],5,20);
67 toc; %Fim do tempo
68 figure, imagesc(accum); axis image;
69 %title('Projeção da Transformada Hough Circular');
70 figure, imagesc(img_con); colormap('gray'); axis image;%gray
71 hold on;
72
73 plot(circen(:,1), circen(:,2), 'b+', 'LineWidth', 2.0); %'r+'
74 for k= 1: size(circen, 1),
75     DrawCircle(circen(k,1), circen(k,2), cirrad(k), 32, 'r-');%'b-'
76 end
77 hold off;
78 %title(['Imagem com círculos encontrados', ...
79 %      '(Posições X,Y e Radianos)']);
80 figure, surf(accum, 'EdgeColor', 'none');axis ij;
81 %title('Domínio 3D da Transformada Hough Circular');
82 grid on
83 %
84 %-----
85 %           Apresentando Quatro imagens
86 %-----
87 figure
88 subplot(3,4,1), image(img), title('Imagem de Entrada')
89 subplot(3,4,2), image(img_gra), title('Suavização escala de cinza')
90 subplot(3,4,3), imshow(img_con, []), title('Convolução Circular')
91 subplot(3,4,4), image(img_ope), title('Morfologia Abertura')
92 subplot(3,4,5), image(rawimg), title('Histograma Adaptativo')
93 subplot(3,4,6), imagesc(accum); axis image; title('Acumulador de votos 1D')
94 subplot(3,4,7), imagesc(img_con); colormap('gray'); axis image;%gray
95 hold on;
96
97 plot(circen(:,1), circen(:,2), 'w+', 'LineWidth', 2.0); %'r+'
98 for k= 1: size(circen, 1),
99     DrawCircle(circen(k,1), circen(k,2), cirrad(k), 32, 'w-');%'b-'
100 end

```



```

101 hold off;
102 title(['Imagem com círculos encontrados', ...
103       '(Posições X,Y e Radianos)']);
104 subplot(3,4,8), surf(accum, 'EdgeColor','none');axis ij;
105 title('Acumulador de votos 3D');
106 grid on
107 %subplot(3,4,8),surf(accum, 'EdgeColor','none');axis ij;
108
109 imtool(img_ope)
110 left_image = img;
111 right_image = img_con;
112 my_image_compare_tool(left_image,right_image);
113 %
114 %-----
115 %                               Calculando precisão entre a população
116 %-----
117 W2 =sortrows([circen,cirrad]);
118 circen=W2(:, [1,2]); %selecionando 1 e 2 colunas
119 cirrad=W2(:,3);     %selecionando 3 coluna
120
121 [I,J]=size(circen);
122
123 pt_concentricos=0;
124 pt_secantes=0;
125
126 for I=1:I-1
127     for J=1:1
128         d(I,J)=sqrt((circen(I+1,1) - circen(I,1))^2 + (circen(I+1,2) - circen(I,2))^2 );
129         if (d(I,J) < cirrad(I,J) + cirrad(I+1,J))&& d(I,J)~=0
130             pt_secantes = pt_secantes + 1;
131         end
132         if d(I,J)==0
133             pt_concentricos = pt_concentricos + 1;
134         end
135     end
136 end
137 end
138
139
140 %
141 %-----
142 %                               Contagem através da imagem dsc00894.jpg
143 %-----
144
145 pre = 488;                               % Preta
146 bro = 44;                                % Bronzeada
147 ave = 9;                                  % Avermelhada
148 ama = 8;                                  % Amarelada
149 opa = 16;                                 % Opaca
150 populacao = pre + bro + ave + ama + opa; % População(Total das amostras)
151
152 [m,n]=size(circen);
153
154 populacao
155 n_localizados = populacao - m
156 pt_localizados = m
157 pt_concentricos

```

```

158 pt_secantes
159 result = pt_concentricos + pt_secantes;
160 pt_ideais = pt_localizados - result
161
162
163
164 %
165 %-----
166 %           Percentual através da imagem dsc00894.jpg
167 %-----
168 per_n_localizados = (100*n_localizados)/populacao
169 per_pt_localizados = (100*pt_localizados)/populacao
170 per_pt_concentricos = (100*pt_concentricos)/populacao
171 per_pt_secantes = (100*pt_secantes)/populacao
172 per_pt_ideais = (( pt_localizados - result)*100)/populacao
173 per_pt_localizados - per_pt_concentricos - per_pt_secantes
174 %
175 %-----
176 %           Copiando Arquivos *.m
177 %-----
178 copiando
179 %
180 %-----
181 %           Imprimindo em .eps
182 %-----
183 %print -depsc2 thc
184 %fn = 'thc'; print( gcf, '-depsc2', fn )
185 %
186 %-----
187 %           Término
188 %-----

```

ANEXO A FUNÇÕES UTILIZADAS NO CÓDIGO FONTE

A.1 Função CircularHough_Grd

```

1  %-----
2  %                               Função CircularHough_Grd
3  %-----
4  function [accum, varargout] = CircularHough_Grd(img, radrange, varargin)
5  %Detect circular shapes in a grayscale image. Resolve their center
6  %positions and radii.
7  %
8  % [accum, circen, cirrad, dbg_LMmask] = CircularHough_Grd(
9  %     img, radrange, grdthres, fltr4LM_R, multirad, fltr4accum)
10 % Circular Hough transform based on the gradient field of an image.
11 % NOTE: Operates on grayscale images, NOT B/W bitmaps.
12 %       NO loops in the implementation of Circular Hough
13 %       transform, which means faster operation but at the same
14 %       time larger memory consumption.
15 %
16 %%% INPUT: (img, radrange, grdthres, fltr4LM_R, multirad, fltr4accum)
17 %
18 % img:      A 2-D grayscale image (NO B/W bitmap)
19 %
20 % radrange: The possible minimum and maximum radii of the circles
21 %           to be searched, in the format of
22 %           [minimum_radius , maximum_radius] (unit: pixels)
23 %           **NOTE**:: A smaller range saves computational time
24 %           and memory.
25 % grdthres: (Optional, default is 10, must be non-negative)
26 %           The algorithm is based on the gradient field of the
27 %           input image. A thresholding on the gradient magnitude
28 %           is performed before the voting process of the Circular
29 %           Hough transform to remove the 'uniform intensity'
30 %           (sort-of) image background from the voting process.
31 %           In other words, pixels with gradient magnitudes smaller
32 %           than 'grdthres' are NOT considered in the computation.
33 %           **NOTE**:: The default parameter value is chosen for
34 %           images with a maximum intensity close to 255. For cases
35 %           with dramatically different maximum intensities, e.g.
36 %           10-bit bitmaps in stead of the assumed 8-bit, the
37 %           default value can NOT be used. A value of 4% to 10%
38 %           of the maximum intensity may work for general cases.
39 %
40 % fltr4LM_R: (Optional, default is 8, minimum is 3)
41 %           The radius of the filter used in the search of local

```

```

42 %             maxima in the accumulation array. To detect circles
43 %             whose shapes are less perfect, the radius of the
44 %             filter needs to be set larger.
45 %
46 % multirad:    (Optional, default is 0.5)
47 %             In case of concentric circles, multiple radii may be
48 %             detected corresponding to a single center position. This
49 %             argument sets the tolerance of picking up the likely
50 %             radii values. It ranges from 0.1 to 1, where 0.1
51 %             corresponds to the largest tolerance, meaning more radii
52 %             values will be detected, and 1 corresponds to the smallest
53 %             tolerance, in which case only the "principal" radius will
54 %             be picked up.
55 %
56 % fltr4accum:  (Optional. A default filter will be used if not given)
57 %             Filter used to smooth the accumulation array. Depending
58 %             on the image and the parameter settings, the accumulation
59 %             array built has different noise level and noise pattern
60 %             (e.g. noise frequencies). The filter should be set to an
61 %             appropriately size such that it's able to suppress the
62 %             dominant noise frequency.
63 %
64 %%%%%%%%%% OUTPUT: [accum, circen, cirrad, dbg_LMmask]
65 %
66 % accum:       The result accumulation array from the Circular Hough
67 %             transform. The accumulation array has the same dimension
68 %             as the input image.
69 %
70 % circen:      (Optional)
71 %             Center positions of the circles detected. Is a N-by-2
72 %             matrix with each row contains the (x, y) positions
73 %             of a circle. For concentric circles (with the same center
74 %             position), say k of them, the same center position will
75 %             appear k times in the matrix.
76 %
77 % cirrad:      (Optional)
78 %             Estimated radii of the circles detected. Is a N-by-1
79 %             column vector with a one-to-one correspondance to the
80 %             output 'circen'. A value 0 for the radius indicates a
81 %             failed detection of the circle's radius.
82 %
83 % dbg_LMmask:  (Optional, for debugging purpose)
84 %             Mask from the search of local maxima in the accumulation
85 %             array.
86 %
87 %%%%%%%%%% EXAMPLE #0:
88 % rawimg = imread('TestImg_CHT_a2.bmp');
89 % rawimg = img_gray([1:500],[1:600]);
90 % tic;
91 % [accum, circen, cirrad] = CircularHough_Grd(rawimg, [15 60]);
92 % toc;
93 % figure(1); imagesc(accum); axis image;
94 % title('Accumulation Array from Circular Hough Transform');
95 % figure(2); imagesc(rawimg); colormap('gray'); axis image;
96 % hold on;
97 % plot(circen(:,1), circen(:,2), 'r+');
98 % for k = 1 : size(circen, 1),

```

```

99 %     DrawCircle(circen(k,1), circen(k,2), cirrad(k), 32, 'b-');
100 % end
101 % hold off;
102 % title(['Raw Image with Circles Detected ', ...
103 %       '(center positions and radii marked)']);
104 % figure(3); surf(accum, 'EdgeColor', 'none'); axis ij;
105 % title('3-D View of the Accumulation Array');
106 %
107 % COMMENTS ON EXAMPLE #0:
108 % Kind of an easy case to handle. To detect circles in the image whose
109 % radii range from 15 to 60. Default values for arguments 'grdthres',
110 % 'fltr4LM_R', 'multirad' and 'fltr4accum' are used.
111 %
112 %%%%%%%%%% EXAMPLE #1:
113 % rawimg = imread('TestImg_CHT_a3.bmp');
114 % tic;
115 % [accum, circen, cirrad] = CircularHough_Grd(rawimg, [15 60], 10, 20);
116 % toc;
117 % figure(1); imagesc(accum); axis image;
118 % title('Accumulation Array from Circular Hough Transform');
119 % figure(2); imagesc(rawimg); colormap('gray'); axis image;
120 % hold on;
121 % plot(circen(:,1), circen(:,2), 'r+');
122 % for k = 1 : size(circen, 1),
123 %     DrawCircle(circen(k,1), circen(k,2), cirrad(k), 32, 'b-');
124 % end
125 % hold off;
126 % title(['Raw Image with Circles Detected ', ...
127 %       '(center positions and radii marked)']);
128 % figure(3); surf(accum, 'EdgeColor', 'none'); axis ij;
129 % title('3-D View of the Accumulation Array');
130 %
131 % COMMENTS ON EXAMPLE #1:
132 % The shapes in the raw image are not very good circles. As a result,
133 % the profile of the peaks in the accumulation array are kind of
134 % 'stumpy', which can be seen clearly from the 3-D view of the
135 % accumulation array. (As a comparison, please see the sharp peaks in
136 % the accumulation array in example #0) To extract the peak positions
137 % nicely, a value of 20 (default is 8) is used for argument 'fltr4LM_R',
138 % which is the radius of the filter used in the search of peaks.
139 %
140 %%%%%%%%%% EXAMPLE #2:
141 % rawimg = imread('TestImg_CHT_b3.bmp');
142 % fltr4img = [1 1 1 1 1; 1 2 2 2 1; 1 2 4 2 1; 1 2 2 2 1; 1 1 1 1 1];
143 % fltr4img = fltr4img / sum(fltr4img(:));
144 % imgfltrd = filter2( fltr4img , rawimg );
145 % tic;
146 % [accum, circen, cirrad] = CircularHough_Grd(imgfltrd, [15 80], 8, 10);
147 % toc;
148 % figure(1); imagesc(accum); axis image;
149 % title('Accumulation Array from Circular Hough Transform');
150 % figure(2); imagesc(rawimg); colormap('gray'); axis image;
151 % hold on;
152 % plot(circen(:,1), circen(:,2), 'r+');
153 % for k = 1 : size(circen, 1),
154 %     DrawCircle(circen(k,1), circen(k,2), cirrad(k), 32, 'b-');
155 % end

```

```

156 % hold off;
157 % title(['Raw Image with Circles Detected ', ...
158 %       '(center positions and radii marked)']);
159 %
160 % COMMENTS ON EXAMPLE #2:
161 % The circles in the raw image have small scale irregularities along
162 % the edges, which could lead to an accumulation array that is bad for
163 % local maxima detection. A 5-by-5 filter is used to smooth out the
164 % small scale irregularities. A blurred image is actually good for the
165 % algorithm implemented here which is based on the image's gradient
166 % field.
167 %
168 %%%%%%%%%%% EXAMPLE #3:
169 % rawimg = imread('TestImg_CHT_c3.bmp');
170 % fltr4img = [1 1 1 1 1; 1 2 2 2 1; 1 2 4 2 1; 1 2 2 2 1; 1 1 1 1 1];
171 % fltr4img = fltr4img / sum(fltr4img(:));
172 % imgfltrd = filter2( fltr4img , rawimg );
173 % tic;
174 % [accum, circen, cirrad] = ...
175 %     CircularHough_Grd(imgfltrd, [15 105], 8, 10, 0.7);
176 % toc;
177 % figure(1); imagesc(accum); axis image;
178 % figure(2); imagesc(rawimg); colormap('gray'); axis image;
179 % hold on;
180 % plot(circen(:,1), circen(:,2), 'r+');
181 % for k = 1 : size(circen, 1),
182 %     DrawCircle(circen(k,1), circen(k,2), cirrad(k), 32, 'b-');
183 % end
184 % hold off;
185 % title(['Raw Image with Circles Detected ', ...
186 %       '(center positions and radii marked)']);
187 %
188 % COMMENTS ON EXAMPLE #3:
189 % Similar to example #2, a filtering before circle detection works for
190 % noisy image too. 'multirad' is set to 0.7 to eliminate the false
191 % detections of the circles' radii.
192 %
193 %%%%%%%%%%% BUG REPORT:
194 % This is a beta version. Please send your bug reports, comments and
195 % suggestions to pengtao@glue.umd.edu . Thanks.
196 %
197 %
198 %%%%%%%%%%% INTERNAL PARAMETERS:
199 % The INPUT arguments are just part of the parameters that are used by
200 % the circle detection algorithm implemented here. Variables in the code
201 % with a prefix 'prm_' in the name are the parameters that control the
202 % judging criteria and the behavior of the algorithm. Default values for
203 % these parameters can hardly work for all circumstances. Therefore, at
204 % occasions, the values of these INTERNAL PARAMETERS (parameters that
205 % are NOT exposed as input arguments) need to be fine-tuned to make
206 % the circle detection work as expected.
207 % The following example shows how changing an internal parameter could
208 % influence the detection result.
209 % 1. Change the value of the internal parameter 'prm_LM_LoBndRa' to 0.4
210 %    (default is 0.2)
211 % 2. Run the following matlab code:
212 %     fltr4accum = [1 2 1; 2 6 2; 1 2 1];

```

```

213 %     fltr4accum = fltr4accum / sum(fltr4accum(:));
214 %     rawimg = imread('Frame_0_0022_portion.jpg');
215 %     tic;
216 %     [accum, circen] = CircularHough_Grd(rawimg, ...
217 %         [4 14], 10, 4, 0.5, fltr4accum);
218 %     toc;
219 %     figure(1); imagesc(accum); axis image;
220 %     title('Accumulation Array from Circular Hough Transform');
221 %     figure(2); imagesc(rawimg); colormap('gray'); axis image;
222 %     hold on; plot(circen(:,1), circen(:,2), 'r+'); hold off;
223 %     title('Raw Image with Circles Detected (center positions marked)');
224 % 3. See how different values of the parameter 'prm_LM_LoBndRa' could
225 %     influence the result.
226
227 % Author: Tao Peng
228 %     Department of Mechanical Engineering
229 %     University of Maryland, College Park, Maryland 20742, USA
230 %     pengtao@glue.umd.edu
231 % Version: Beta          Revision: Mar. 07, 2007
232
233
234 %%%%%%%%% Arguments and parameters %%%%%%%%%%%%%%%
235
236 % Validation of arguments
237 if ndims(img) ~= 2 || ~isnumeric(img),
238     error('CircularHough_Grd: ''img'' has to be 2 dimensional');
239 end
240 if ~all(size(img) >= 32),
241     error('CircularHough_Grd: ''img'' has to be larger than 32-by-32');
242 end
243
244 if numel(radrange) ~= 2 || ~isnumeric(radrange),
245     error(['CircularHough_Grd: ''radrange'' has to be ', ...
246         'a two-element vector']);
247 end
248 prm_r_range = sort(max([0,0;radrange(1),radrange(2)]));
249
250 % Parameters (default values)
251 prm_grdthres = 10;
252 prm_fltrLM_R = 8;
253 prm_multirad = 0.5;
254 func_compu_cen = true;
255 func_compu_radii = true;
256
257 % Validation of arguments
258 vap_grdthres = 1;
259 if nargin > (1 + vap_grdthres),
260     if isnumeric(varargin{vap_grdthres}) && ...
261         varargin{vap_grdthres}(1) >= 0,
262         prm_grdthres = varargin{vap_grdthres}(1);
263     else
264         error(['CircularHough_Grd: ''grdthres'' has to be ', ...
265             'a non-negative number']);
266     end
267 end
268
269 vap_fltr4LM = 2; % filter for the search of local maxima

```

```

270 if nargin > (1 + vap_fltr4LM),
271     if isnumeric(varargin{vap_fltr4LM}) && varargin{vap_fltr4LM}(1) >= 3,
272         prm_fltrLM_R = varargin{vap_fltr4LM}(1);
273     else
274         error(['CircularHough_Grd: ''fltr4LM_R'' has to be ', ...
275             'larger than or equal to 3']);
276     end
277 end
278
279 vap_multirad = 3;
280 if nargin > (1 + vap_multirad),
281     if isnumeric(varargin{vap_multirad}) && ...
282         varargin{vap_multirad}(1) >= 0.1 && ...
283         varargin{vap_multirad}(1) <= 1,
284         prm_multirad = varargin{vap_multirad}(1);
285     else
286         error(['CircularHough_Grd: ''multirad'' has to be ', ...
287             'within the range [0.1, 1]']);
288     end
289 end
290
291 vap_fltr4accum = 4; % filter for smoothing the accumulation array
292 if nargin > (1 + vap_fltr4accum),
293     if isnumeric(varargin{vap_fltr4accum}) && ...
294         ndims(varargin{vap_fltr4accum}) == 2 && ...
295         all(size(varargin{vap_fltr4accum}) >= 3),
296         fltr4accum = varargin{vap_fltr4accum};
297     else
298         error(['CircularHough_Grd: ''fltr4accum'' has to be ', ...
299             'a 2-D matrix with a minimum size of 3-by-3']);
300     end
301 else
302     % Default filter (5-by-5)
303     fltr4accum = ones(5,5);
304     fltr4accum(2:4,2:4) = 2;
305     fltr4accum(3,3) = 6;
306 end
307
308 func_compu_cen = ( nargout > 1 );
309 func_compu_radii = ( nargout > 2 );
310
311 % Reserved parameters
312 dbg_on = false; % debug information
313 dbg_bfigno = 4;
314 if nargin > 3, dbg_on = true; end
315
316
317 %%%%%%% Building accumulation array %%%%%%%%%%%%%%%
318
319 % Convert the image to single if it is not of
320 % class float (single or double)
321 img_is_double = isa(img, 'double');
322 if ~(img_is_double || isa(img, 'single')),
323     imgf = single(img);
324 end
325
326 % Compute the gradient and the magnitude of gradient

```



```

327 if img_is_double,
328     [grdx, grdy] = gradient(img);
329 else
330     [grdx, grdy] = gradient(imgf);
331 end
332 grdmag = sqrt(grdx.^2 + grdy.^2);
333
334 % Get the linear indices, as well as the subscripts, of the pixels
335 % whose gradient magnitudes are larger than the given threshold
336 grdmasklin = find(grdmag > prm_grdthres);
337 [grdmask_IdxI, grdmask_IdxJ] = ind2sub(size(grdmag), grdmasklin);
338
339 % Compute the linear indices (as well as the subscripts) of
340 % all the votings to the accumulation array.
341 % The Matlab function 'accumarray' accepts only double variable,
342 % so all indices are forced into double at this point.
343 % A row in matrix 'lin2accum_aJ' contains the J indices (into the
344 % accumulation array) of all the votings that are introduced by a
345 % same pixel in the image. Similarly with matrix 'lin2accum_aI'.
346 rr_4linaccum = double( prm_r_range );
347 linaccum_dr = [ (-rr_4linaccum(2) + 0.5) : -rr_4linaccum(1) , ...
348     (rr_4linaccum(1) + 0.5) : rr_4linaccum(2) ];
349
350 lin2accum_aJ = floor( ...
351     double(grdx(grdmasklin)./grdmag(grdmasklin)) * linaccum_dr + ...
352     repmat( double(grdmask_IdxJ)+0.5 , [1,length(linaccum_dr)] ) ...
353 );
354 lin2accum_aI = floor( ...
355     double(grdy(grdmasklin)./grdmag(grdmasklin)) * linaccum_dr + ...
356     repmat( double(grdmask_IdxI)+0.5 , [1,length(linaccum_dr)] ) ...
357 );
358
359 % Clip the votings that are out of the accumulation array
360 mask_valid_aJaI = ...
361     lin2accum_aJ > 0 & lin2accum_aJ < (size(grdmag,2) + 1) & ...
362     lin2accum_aI > 0 & lin2accum_aI < (size(grdmag,1) + 1);
363
364 mask_valid_aJaI_reverse = ~ mask_valid_aJaI;
365 lin2accum_aJ = lin2accum_aJ .* mask_valid_aJaI + mask_valid_aJaI_reverse;
366 lin2accum_aI = lin2accum_aI .* mask_valid_aJaI + mask_valid_aJaI_reverse;
367 clear mask_valid_aJaI_reverse;
368
369 % Linear indices (of the votings) into the accumulation array
370 lin2accum = sub2ind( size(grdmag), lin2accum_aI, lin2accum_aJ );
371
372 lin2accum_size = size( lin2accum );
373 lin2accum = reshape( lin2accum, [numel(lin2accum),1] );
374 clear lin2accum_aI lin2accum_aJ;
375
376 % Weights of the votings, currently using the gradient maginitudes
377 % but in fact any scheme can be used (application dependent)
378 weight4accum = ...
379     repmat( double(grdmag(grdmasklin)) , [lin2accum_size(2),1] ) .* ...
380     mask_valid_aJaI(:);
381 clear mask_valid_aJaI;
382
383 % Build the accumulation array using Matlab function 'accumarray'

```

```

384 accum = accumarray( lin2accum , weight4accum );
385 accum = [ accum ; zeros( numel( grdmag ) - numel( accum ) , 1 ) ];
386 accum = reshape( accum, size( grdmag ) );
387
388
389 %%%%%%%%% Locating local maxima in the accumulation array %%%%%%%%%
390
391 % Stop if no need to locate the center positions of circles
392 if ~func_compu_cen,
393     return;
394 end
395 clear lin2accum weight4accum;
396
397 % Parameters to locate the local maxima in the accumulation array
398 % -- Segmentation of 'accum' before locating LM
399 prm_useaoi = true;
400 prm_aoithres_s = 2;
401 prm_aoiminsize = floor( min( [ min( size( accum ) ) * 0.25, ...
402     prm_r_range( 2 ) * 1.5 ] ) );
403
404 % -- Filter for searching for local maxima
405 prm_fltrLM_s = 1.35;
406 prm_fltrLM_r = ceil( prm_fltrLM_R * 0.6 );
407 prm_fltrLM_npix = max( [ 6, ceil( ( prm_fltrLM_R / 2 ) ^ 1.8 ) ] );
408
409 % -- Lower bound of the intensity of local maxima
410 prm_LM_LoBndRa = 0.2; % minimum ratio of LM to the max of 'accum'
411
412 % Smooth the accumulation array
413 fltr4accum = fltr4accum / sum( fltr4accum( : ) );
414 accum = filter2( fltr4accum, accum );
415
416 % Select a number of Areas-Of-Interest from the accumulation array
417 if prm_useaoi,
418     % Threshold value for 'accum'
419     prm_llm_thres1 = prm_grdthres * prm_aoithres_s;
420
421     % Thresholding over the accumulation array
422     accummask = ( accum > prm_llm_thres1 );
423
424     % Segmentation over the mask
425     [ accumlabel, accum_nRgn ] = bwlabel( accummask, 8 );
426
427     % Select AOIs from segmented regions
428     accumAOI = ones( 0, 4 );
429     for k = 1 : accum_nRgn,
430         accumrgn_lin = find( accumlabel == k );
431         [ accumrgn_IdxI, accumrgn_IdxJ ] = ...
432             ind2sub( size( accumlabel ), accumrgn_lin );
433         rgn_top = min( accumrgn_IdxI );
434         rgn_bottom = max( accumrgn_IdxI );
435         rgn_left = min( accumrgn_IdxJ );
436         rgn_right = max( accumrgn_IdxJ );
437         % The AOIs selected must satisfy a minimum size
438         if ( ( rgn_right - rgn_left + 1 ) >= prm_aoiminsize && ...
439             ( rgn_bottom - rgn_top + 1 ) >= prm_aoiminsize ),
440             accumAOI = [ accumAOI; ...

```

```

441         rgn_top, rgn_bottom, rgn_left, rgn_right ];
442     end
443 end
444 else
445     % Whole accumulation array as the one AOI
446     accumAOI = [1, size(accum,1), 1, size(accum,2)];
447 end
448
449 % Thresholding of 'accum' by a lower bound
450 prm_LM_LoBnd = max(accum(:)) * prm_LM_LoBndRa;
451
452 % Build the filter for searching for local maxima
453 fltr4LM = zeros(2 * prm_fltrLM_R + 1);
454
455 [mesh4fLM_x, mesh4fLM_y] = meshgrid(-prm_fltrLM_R : prm_fltrLM_R);
456 mesh4fLM_r = sqrt( mesh4fLM_x.^2 + mesh4fLM_y.^2 );
457 fltr4LM_mask = ...
458     ( mesh4fLM_r > prm_fltrLM_r & mesh4fLM_r <= prm_fltrLM_R );
459 fltr4LM = fltr4LM - ...
460     fltr4LM_mask * (prm_fltrLM_s / sum(fltr4LM_mask(:)));
461
462 if prm_fltrLM_R >= 4,
463     fltr4LM_mask = ( mesh4fLM_r < (prm_fltrLM_r - 1) );
464 else
465     fltr4LM_mask = ( mesh4fLM_r < prm_fltrLM_r );
466 end
467 fltr4LM = fltr4LM + fltr4LM_mask / sum(fltr4LM_mask(:));
468
469 % **** Debug code (begin)
470 if dbg_on,
471     dbg_LMmask = zeros(size(accum));
472 end
473 % **** Debug code (end)
474
475 % For each of the AOIs selected, locate the local maxima
476 circen = zeros(0,2);
477 for k = 1 : size(accumAOI, 1),
478     aoi = accumAOI(k,:);    % just for referencing convenience
479
480     % Thresholding of 'accum' by a lower bound
481     accumaoi_LBMask = ...
482         ( accum(aoi(1):aoi(2), aoi(3):aoi(4)) > prm_LM_LoBnd );
483
484     % Apply the local maxima filter
485     candLM = conv2( accum(aoi(1):aoi(2), aoi(3):aoi(4)) , ...
486         fltr4LM , 'same' );
487     candLM_mask = ( candLM > 0 );
488
489     % Clear the margins of 'candLM_mask'
490     candLM_mask([1:prm_fltrLM_R, (end-prm_fltrLM_R+1):end], :) = 0;
491     candLM_mask(:, [1:prm_fltrLM_R, (end-prm_fltrLM_R+1):end]) = 0;
492
493     % **** Debug code (begin)
494     if dbg_on,
495         dbg_LMmask(aoi(1):aoi(2), aoi(3):aoi(4)) = ...
496             dbg_LMmask(aoi(1):aoi(2), aoi(3):aoi(4)) + ...
497             accumaoi_LBMask + 2 * candLM_mask;

```

```

498     end
499     % **** Debug code (end)
500
501     % Group the local maxima candidates by adjacency, compute the
502     % centroid position for each group and take that as the center
503     % of one circle detected
504     [candLM_label, candLM_nRgn] = bwlabel( candLM_mask, 8 );
505
506     for ilabel = 1 : candLM_nRgn,
507         % Indices (to current AOI) of the pixels in the group
508         candgrp_masklin = find( candLM_label == ilabel );
509         [candgrp_IdxI, candgrp_IdxJ] = ...
510             ind2sub( size(candLM_label) , candgrp_masklin );
511
512         % Indices (to 'accum') of the pixels in the group
513         candgrp_IdxI = candgrp_IdxI + ( aoi(1) - 1 );
514         candgrp_IdxJ = candgrp_IdxJ + ( aoi(3) - 1 );
515         candgrp_idx2acm = ...
516             sub2ind( size(accum) , candgrp_IdxI , candgrp_IdxJ );
517
518         % Minimum number of qualified pixels in the group
519         if sum(accumaoi_LBMask(candgrp_masklin)) < prm_fltrLM_npix,
520             continue;
521         end
522
523         % Compute the centroid position
524         candgrp_acmsum = sum( accum(candgrp_idx2acm) );
525         cc_x = sum( candgrp_IdxJ .* accum(candgrp_idx2acm) ) / ...
526             candgrp_acmsum;
527         cc_y = sum( candgrp_IdxI .* accum(candgrp_idx2acm) ) / ...
528             candgrp_acmsum;
529         circen = [circen; cc_x, cc_y];
530     end
531 end
532
533 % **** Debug code (begin)
534 if dbg_on,
535     figure(dbg_bfigno); imagesc(dbg_LMmask); axis image;
536     title('Generated map of local maxima');
537     if size(accumAOI, 1) == 1,
538         figure(dbg_bfigno+1);
539         surf(candLM, 'EdgeColor', 'none'); axis ij;
540         title('Accumulation array after local maximum filtering');
541     end
542 end
543 % **** Debug code (end)
544
545
546 %%%%%%%%% Estimation of the Radii of Circles %%%%%%%%%
547
548 % Stop if no need to estimate the radii of circles
549 if ~func_compu_radii,
550     varargout{1} = circen;
551     return;
552 end
553
554 % Parameters for the estimation of the radii of circles

```

```

555 fltr4SgnCv = [2 1 1];
556 fltr4SgnCv = fltr4SgnCv / sum(fltr4SgnCv);
557
558 % Find circle's radius using its signature curve
559 cirrad = zeros( size(circen,1), 1 );
560
561 for k = 1 : size(circen,1),
562     % Neighborhood region of the circle for building the sgn. curve
563     circen_round = round( circen(k,:) );
564     SCvR_I0 = circen_round(2) - prm_r_range(2) - 1;
565     if SCvR_I0 < 1,
566         SCvR_I0 = 1;
567     end
568     SCvR_I1 = circen_round(2) + prm_r_range(2) + 1;
569     if SCvR_I1 > size(grdx,1),
570         SCvR_I1 = size(grdx,1);
571     end
572     SCvR_J0 = circen_round(1) - prm_r_range(2) - 1;
573     if SCvR_J0 < 1,
574         SCvR_J0 = 1;
575     end
576     SCvR_J1 = circen_round(1) + prm_r_range(2) + 1;
577     if SCvR_J1 > size(grdx,2),
578         SCvR_J1 = size(grdx,2);
579     end
580
581     % Build the sgn. curve
582     SgnCvMat_dx = repmat( (SCvR_J0:SCvR_J1) - circen(k,1) , ...
583         [SCvR_I1 - SCvR_I0 + 1 , 1] );
584     SgnCvMat_dy = repmat( (SCvR_I0:SCvR_I1)' - circen(k,2) , ...
585         [1 , SCvR_J1 - SCvR_J0 + 1] );
586     SgnCvMat_r = sqrt( SgnCvMat_dx .^2 + SgnCvMat_dy .^2 );
587     SgnCvMat_rp1 = round(SgnCvMat_r) + 1;
588
589     f4SgnCv = abs( ...
590         double(grdx(SCvR_I0:SCvR_I1, SCvR_J0:SCvR_J1)) .* SgnCvMat_dx + ...
591         double(grdy(SCvR_I0:SCvR_I1, SCvR_J0:SCvR_J1)) .* SgnCvMat_dy ...
592         ) ./ SgnCvMat_r;
593     SgnCv = accumarray( SgnCvMat_rp1(:) , f4SgnCv(:) );
594
595     SgnCv_Cnt = accumarray( SgnCvMat_rp1(:) , ones(numel(f4SgnCv),1) );
596     SgnCv_Cnt = SgnCv_Cnt + (SgnCv_Cnt == 0);
597     SgnCv = SgnCv ./ SgnCv_Cnt;
598
599     % Suppress the undesired entries in the sgn. curve
600     % -- Radii that correspond to short arcs
601     SgnCv = SgnCv .* ( SgnCv_Cnt >= (pi/4 * [0:(numel(SgnCv_Cnt)-1)]') );
602     % -- Radii that are out of the given range
603     SgnCv( 1 : (round(prm_r_range(1))+1) ) = 0;
604     SgnCv( (round(prm_r_range(2))+1) : end ) = 0;
605
606     % Get rid of the zero radius entry in the array
607     SgnCv = SgnCv(2:end);
608     % Smooth the sgn. curve
609     SgnCv = filtfilt( fltr4SgnCv , [1] , SgnCv );
610
611     % Get the maximum value in the sgn. curve

```

```

612     SgnCv_max = max(SgnCv);
613     if SgnCv_max <= 0,
614         cirrad(k) = 0;
615         continue;
616     end
617
618     % Find the local maxima in sgn. curve by 1st order derivatives
619     % -- Mark the ascending edges in the sgn. curve as 1s and
620     % -- descending edges as 0s
621     SgnCv_AscEdg = ( SgnCv(2:end) - SgnCv(1:(end-1)) ) > 0;
622     % -- Mark the transition (ascending to descending) regions
623     SgnCv_LMmask = [ 0; 0; SgnCv_AscEdg(1:(end-2)) ] & (~SgnCv_AscEdg);
624     SgnCv_LMmask = SgnCv_LMmask & [ SgnCv_LMmask(2:end) ; 0 ];
625
626     % Incorporate the minimum value requirement
627     SgnCv_LMmask = SgnCv_LMmask & ...
628         ( SgnCv(1:(end-1)) >= (prm_multirad * SgnCv_max) );
629     % Get the positions of the peaks
630     SgnCv_LMPos = sort( find(SgnCv_LMmask) );
631
632     % Save the detected radii
633     if isempty(SgnCv_LMPos),
634         cirrad(k) = 0;
635     else
636         cirrad(k) = SgnCv_LMPos(end);
637         for i_radii = (length(SgnCv_LMPos) - 1) : -1 : 1,
638             circen = [ circen; circen(k,:) ];
639             cirrad = [ cirrad; SgnCv_LMPos(i_radii) ];
640         end
641     end
642 end
643
644 % Output
645 varargout{1} = circen;
646 varargout{2} = cirrad;
647 if nargout > 3,
648     varargout{3} = dbg_LMmask;
649 end
650 %-----
651 %                               Término
652 %-----

```

A.2 Função DrawCircle

```

1 %-----
2 %                               Função DrawCircle
3 %-----
4 function DrawCircle(x, y, r, nseg, S)
5 % Draw a circle on the current figure using ploylines
6 %
7 % DrawCircle(x, y, r, nseg, S)
8 % A simple function for drawing a circle on graph.
9 %
10 % INPUT: (x, y, r, nseg, S)
11 % x, y:   Center of the circle
12 % r:      Radius of the circle

```

```
13 % nseg:    Number of segments for the circle
14 % S:      Colors, plot symbols and line types
15 %
16 % OUTPUT: None
17 %
18 % BUG REPORT:
19 % Please send your bug reports, comments and suggestions to
20 % pengtao@glue.umd.edu . Thanks.
21
22 % Author:  Tao Peng
23 %          Department of Mechanical Engineering
24 %          University of Maryland, College Park, Maryland 20742, USA
25 %          pengtao@glue.umd.edu
26 % Version: alpha      Revision: Jan. 10, 2006
27
28
29 theta = 0 : (2 * pi / nseg) : (2 * pi);
30 pline_x = r * cos(theta) + x;
31 pline_y = r * sin(theta) + y;
32
33 plot(pline_x, pline_y, S,'LineWidth',2.0);
34 %
35 %-----
36 %                Término
37 %-----
```

ÍNDICE

Anexo - A	Transformada de Hough para retas, 41
Funções utilizadas no código fonte, 75	Resultados, 53
Função CircularHough_Grd, 75	
Função DrawCircle, 86	
Apêndice - A	
Implementação do código fonte, 70	
Conclusão, 64	
Glossário, 68	
Introdução, 17	
Materiais e métodos, 21	
Processamento digital de imagem, 24	
Amostragem e quantização, 25	
Histograma adaptativo, 39	
Morfologia	
Abertura, 38	
Definições básicas, 35	
Dilatação, 36	
Erosão, 37	
Morfologia matemática, 34	
O processamento da vizinhança, 30	
Operações de convolução com máscaras, 27	
Convolução bidimensional, 28	
Convolução unidimensional, 27	
Processamento no domínio espacial, 32	
Filtro gaussiano, 32	
Representação matemática de imagem digital,	
25	
Transformada de Hough, 41	
Transformada de Hough para círculos, 43	