

Resumo

A capacidade humana de extrair informações visuais de um objeto ou de uma cena é algo que, apesar dos inúmeros avanços na área de visão computacional, ainda não foi completamente transposto para os computadores. Para um ser humano, basta olhar para uma árvore, para perceber que seu tronco é áspero ou olhar um quadro branco para notar que sua textura é extremamente suave. A análise e segmentação de texturas é um dos passos na tentativa de aproximar a visão computacional do sistema visual humano através dos mais variados tipos de medida. O objetivo da segmentação é bem claro: definir as fronteiras entre texturas, as quais poderão, entre outras coisas, serem usadas para a identificação daquela amostra. No entanto, comparar duas amostras e definir sua semelhança é mais um dos obstáculos nesta tentativa de aproximação. Para ajudar a superar este obstáculo, faz-se necessário o uso de algoritmos capazes de analisar propriedades e atributos de imagens e através de um índice, definir o quão parecidas são as amostras. Este trabalho apresenta um estudo sobre o uso de índices de fidelidade de imagens, juntamente com três métodos estatísticos para a descrição de texturas com o objetivo de segmentar imagens.

Abstract

Despite the many advances in machine vision, the human ability to extract visual information from an object or a scene, has not yet been completely transposed to computers. One can tell the roughness of a tree trunk or how smooth a whiteboard is by just looking at them. Texture analysis and segmentation is an initial step in the attempt to bring machine vision and the human vision system closer, through many types of measures. The goal of texture segmentation is simple: to define the boundaries between textures, which may be used, among other things, to identify that sample. Comparing two samples to determine their similarity, however, is just one of the obstacles in this attempt to approach these two vision systems. To aid in overcoming this obstacle, it is essential to use algorithms that are able to compare and analyze properties and features of images and define how similar are the two samples. This work presents a study on the use of image fidelity indexes along with texture descriptors in order to perform image segmentation.

Sumário

Índice de Figuras	v
Índice de Tabelas	vii
Tabela de Símbolos e Siglas	viii
1 Introdução	10
1.1 Objetivos	11
1.2 Estrutura do trabalho	11
2 Texturas	12
2.1 Classificação de Texturas	13
2.2 Descritores de Texturas	14
2.2.1 Matriz de Co-Ocorrência de Tons de Cinza	14
2.2.2 Histogramas da Soma e da Diferença	14
2.2.3 Padrão Binário Local	15
2.3 Aplicações	16
2.3.1 Segmentação de Texturas	16
2.3.2 Recuperação de Imagem Baseada em Textura	17
3 Fidelidade de imagens	18
4 Experimentos	23
4.1 Imagens geradas	23
4.2 Avaliação de Similaridade usando Descritores de Texturas	24
4.2.1 GLCM	25
4.2.2 SDH	26
4.2.3 LBP	27
4.3 Recuperação de Imagem Baseada em Textura	28
4.3.1 TBIR com Q e janela de 8x8 <i>pixels</i>	29
4.3.2 TBIR com Q e janela de 32x32 <i>pixels</i>	30
4.3.3 TBIR com SSIM	30
Na Tabela 4-13 é possível ver os valores totais e normalizados para o <i>threshold</i> escolhido.	31
4.3.4 TBIR com LBP	31
4.3.5 TBIR com SDH	32
4.3.6 TBIR com GLCM	33
4.4 Segmentação de Imagens baseada em Texturas	34
4.4.1 Segmentação utilizando GLCM	35
4.4.2 Segmentação utilizando LBP	36
4.4.3 Segmentação utilizando SDH	37
4.4.4 Resultados da Segmentação do Banco de Imagens	38
4.5 Segmentação de Imagens utilizando Decomposição <i>QuadTree</i>	39
5 Conclusões	42

5.1	Conclusão	iv
5.2	Trabalhos Futuros	42
	Bibliografia	43
		44

Índice de Figuras

Figura 1-1. Exemplo de segmentação de texturas; regiões com diferentes tons refletem texturas diferentes que foram reconhecidas.	10
Figura 2-1. Exemplo do uso da informação da textura de uma imagem em duas dimensões para determinar formas em três dimensões. a) Imagem em duas dimensões e b) imagem em três dimensões, gerada a partir da informação da textura	13
Figura 2-2. Exemplo demonstrando os passos necessários para o cálculo do LBP. a) Imagem original, b) imagem binária gerada por comparação com o pixel central, c) máscara LBP e d) resultado da operação.	16
Figura 2-3. Objetivo da segmentação de texturas. a) Imagem composta por cinco texturas diferentes; em b) o resultado da segmentação ideal/perfeita da imagem [10].....	17
Figura 3-1. Comparativo entre imagens com compressão JPEG. Em a) Imagem original, em b) imagem com 15% de perda, c) com 30% de perda, d) com 80% de perda e e) com 90% sobre 80% de perda.	20
Figura 3-2. Exemplo de figuras com alterações visuais e estrutura preservada. a) Imagem Original; em b) Imagem com contraste e brilho alterados apresenta valor do SSIM = 0,4274.	21
Figura 3-3. Exemplo de figuras com alterações visuais e perda de informação estrutural. Em a) Imagem Original; em b) a imagem processada com um filtro passa-baixa apresenta valor do SSIM = 0,2850.....	22
Figura 4-1. Exemplos de imagens geradas para testes de segmentação.	24
Figura 4-2. Imagens atípicas geradas para testes de segmentação.....	24
Figura 4-3. Classes usadas para testar os descritores de texturas.	25
Figura 4-4. Classes da base de figuras usadas para CBIR.	29
Figura 4-5. Imagem usada para ajustes no valor de delta.	34
Figura 4-6. Resultado da segmentação com variação do delta: em a) delta = 0,52 (11 regiões); em b) delta = 0,55 (37 regiões) e em c) delta = 0,57 (47 regiões).....	35
Figura 4-7. Resultado da segmentação com variação do delta: em a) delta = 0,6 (1 região); em b) delta = 0,7 (4 regiões) e em c) delta = 0,8 (18 regiões).....	36
Figura 4-8. Resultado da segmentação com variação do delta: em a) delta = 0,63 (4 regiões); em b) delta = 0,66 (11 regiões) e em c) delta = 0,70 (30 regiões).....	36
Figura 4-9. Resultado da segmentação com variação do delta: em a) delta = 0,40 (4 regiões); em b) delta = 0,46 (11 regiões) e em c) delta = 0,50 (64 regiões).....	37
Figura 4-10. Resultado da segmentação com variação do delta: em a) delta = 0,44 (5 regiões); em b) delta = 0,48 (13 regiões) e em c) delta = 0,52 (27 regiões).....	37
Figura 4-11. Resultado da segmentação com variação do delta: em a) delta = 0,45 (4 regiões); em b) delta = 0,5 (18 regiões) e em c) delta = 0,52 (30 regiões).....	38
Figura 4-12. Exemplo do funcionamento do algoritmo de Decomposição <i>Quadtree</i> : a) Imagem a ser segmentada; b) o mapa gerado pelo <i>Quadtree</i> com limiar = 0,5 e c) sobreposição da imagem com o mapa para observarmos melhor o resultado da segmentação.....	40
Figura 4-13. Em a) Imagem original usada para o teste e em b) mapa gerado pelo <i>Quadtree Decomposition</i> original (Ponto de corte = 0,9; 580 regiões).....	40

Figura 4-14. Resultado da segmentação utilizando decomposição *Quadtree* e o índice Q para comparar as janelas usando descritores de textura: a) GLCM (255 regiões); b) LBP (142 regiões) e c) SDH (152 regiões).41

Índice de Tabelas

Tabela 2-1. Comparativo entre GCLM e SDH para o cálculo de propriedades de textura.....	15
Tabela 3-1. Resultados da comparação utilizando o índice Q sobre a imagem da Figura 3-1a....	20
Tabela 4-1. Resultados da comparação utilizando Q após descrição com GLCM para elementos da mesma classe.....	25
Tabela 4-2. Resultados da comparação utilizando Q após descrição com GLCM para elementos de outras classes.	26
Tabela 4-3. Resultados da comparação utilizando Q após descrição com SDH para elementos da mesma classe.	26
Tabela 4-4. Resultados da comparação utilizando Q após descrição com SDH para elementos de outras classes.	27
Tabela 4-5. Resultados da comparação utilizando Q após descrição com LBP para elementos da mesma classe.	27
Tabela 4-6. Resultados da comparação utilizando Q após descrição com LBP para elementos de outras classes.	28
Tabela 4-7. Média de valores para Q com janela de 8x8 <i>pixels</i>	29
Tabela 4-8. Valores do classificador usando Q com janela tamanho 8x8 <i>pixels</i>	30
Tabela 4-9. Média de valores de Q com janela de 32x32 <i>pixels</i>	30
Tabela 4-10. Valores do classificador para Q com janela de 32x32 <i>pixels</i>	30
Tabela 4-11. Média de valores de SSIM para os parâmetros originais.....	31
Tabela 4-12. Desvio Padrão e Porcentagem em relação à média por Classe de Textura.....	31
Tabela 4-13. Valores do classificador para SSIM.	31
Tabela 4-14. Média dos valores de Q para descritor LBP.....	32
Tabela 4-15. Valores do classificador para LBP.....	32
Tabela 4-16. Média dos valores de Q para SDH da soma.....	33
Tabela 4-17. Valores do classificador para SDH.	33
Tabela 4-18. Média dos valores de Q para o GLCM.	33
Tabela 4-19. Valores do classificador para GLCM.....	34
Tabela 4-20. Resultado da segmentação pelo SDH (janela = 32, delta = 0,44).....	38
Tabela 4-21. Resultado da segmentação pelo SDH (janela = 16, delta = 0,45).....	38
Tabela 4-22. Resultado da segmentação pelo LBP (janela = 32 delta = 0,65).....	39
Tabela 4-23. Resultado da segmentação pelo LBP (janela = 16, delta = 0,42).....	39

Tabela de Símbolos e Siglas

(Em ordem de aparição no texto)

Q – *Universal Quality Index* – Índice Universal de Qualidade

SSIM – *Structural Similarity* – Índice de Semelhança Estrutural

GLCM – *Gray Level Co-Occurrence Matrix* – Matriz de co-ocorrência de tons de cinza

SDH – *Sum and Difference Histograms* – Histogramas da Soma e da Diferença

LBP – *Local Binary Pattern* – Padrão Binário Local

CBIR – *Content-Based Image Retrieval* – Recuperação de Imagem Baseada em Conteúdo

TBIR – *Texture-Based Image Retrieval* – Recuperação de Imagem Baseada em Textura

EMQ – Erro Médio Quadrático

SVH – Sistema Visual Humano

PSNR – *Peak Signal-to-Noise Ratio* – Razão Sinal-Ruído de Pico

Agradecimentos

Gostaria de agradecer primeiramente aos meus pais, José Ronaldo de Melo Jucá e Magnólia de Melo Jucá pelos ensinamentos que recebi até hoje e pela oportunidade de estudar. Ao meu irmão, Victor de Melo Jucá, pela amizade e compreensão em vários momentos da minha caminhada até aqui.

Gostaria de agradecer ao Dr. Prof. Carlos Alexandre, meu orientador e amigo, por toda a atenção e comprometimento para o desenvolvimento deste trabalho e também por dedicação sobre-humana ao curso de computação da Escola Politécnica e ao DSC como um todo. A todos os professores do DSC, que colocam o departamento sempre acima de qualquer interesse e buscam incessantemente melhorar e aprimorar o ensino.

Aos amigos que fiz até hoje, sejam do colégio, através da música ou da faculdade: levo comigo um pouco de cada um e tenho certeza que não seria quem sou hoje sem vocês. Em especial o pessoal d'Os Caras por todos dias de estudo e noites de projeto e também por todos os encontros que ainda estão por vir.

Gostaria de agradecer a três amigos que fiz há muito tempo e tenho a sorte de poder contar com eles em qualquer momento: Fábio Correia, Felipe Lapa e Maria Beatriz Ribeiro. Palavras não são suficientes para dizer o quanto admiro e amo vocês. Sei que ainda temos longos anos pela frente!

Por último, mas não menos importante, a todas as pessoas que contribuíram direta ou indiretamente para a minha formação acadêmica e pessoal, ou torcem por mim de uma forma ou de outra.

Isso aqui é só o começo!

Capítulo 1

Introdução

A capacidade humana de reconhecer sutilezas entre diferentes texturas não é um processo simples de se reproduzir utilizando algoritmos automáticos, devido aos mais diversos fatores como: iluminação, direção, etc. O próprio conceito de textura é algo discutido até hoje. Entende-se por textura um padrão de tons que “envolve” os objetos. Dada essa falta de definição formal, torna-se também difícil modelar texturas. O modelo ainda aceito hoje foi proposto em 1973 por Haralick [1], que definiu 14 propriedades que individualizam uma textura (como, por exemplo, entropia, entropia da soma e variância).

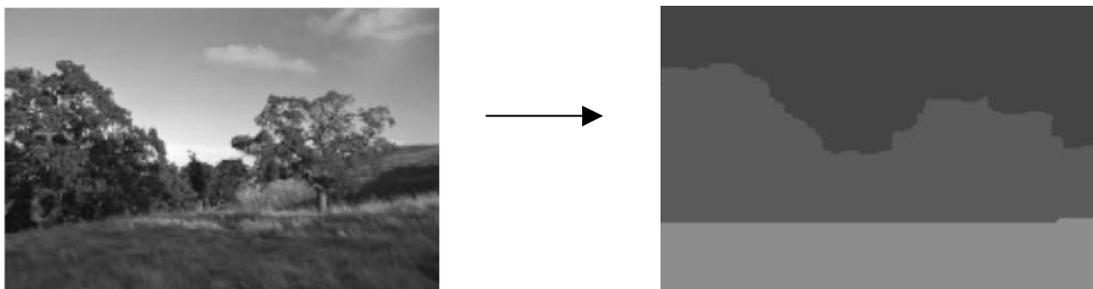


Figura 1-1. Exemplo de segmentação de texturas; regiões com diferentes tons refletem texturas diferentes que foram reconhecidas.

Segmentação de texturas pode ser realizada por classificação a qual pode ser, basicamente, de duas formas: supervisionada ou não-supervisionada. Dizemos que a segmentação é supervisionada quando existem informações disponíveis sobre a imagem ou sobre a classe da textura, sendo assim, consiste basicamente de identificar regiões cujos parâmetros estão presentes em alguma textura usada durante o treinamento. Por outro lado, a segmentação não-supervisionada possui mais aplicações, mesmo possuindo em geral um custo computacional elevado, já que é mais genérica [2].

A análise e segmentação de texturas possuem diversas aplicações, variando desde aplicações médicas até a classificação de plantas [3]. Uma dessas aplicações médicas, por exemplo, visa detectar tumores em mamografias [4], melhorando a resposta do diagnóstico auxiliado por computador. Em outra aplicação, desta vez na área de automação industrial, a

análise de texturas é usada pra detectar automaticamente defeitos de fabricação e arranhões em peças de couro [5].

As métricas para avaliação objetiva de imagens podem ser classificadas de acordo com a disponibilidade da imagem original (sem distorção) junto com a imagem a ser avaliada. A maioria das abordagens utiliza referência completa, ou seja, um original sem distorção está disponível para comparações. Em alguns casos, não existe uma imagem original, sendo este caso conhecido como sem referência. Neste trabalho serão utilizados índices de fidelidade de imagens em avaliações com referência, tomando como base janelas de *pixels* de uma mesma imagem, para identificar e segmentar regiões com texturas semelhantes.

1.1 Objetivos

O objetivo central deste trabalho consiste em avaliar o uso de índices de fidelidade para comparação de texturas. Para tanto, foi necessário implementar algoritmos para descrever texturas, obter submatrizes de matrizes maiores, comparar, testar e segmentar imagens de acordo com o índice, entre outros. Além disso, algumas modificações foram realizadas em algoritmos já existentes, como por exemplo, o índice de fidelidade Q [20].

1.2 Estrutura do trabalho

Este trabalho está dividido em 5 Capítulos, incluindo este de Introdução. Os capítulos seguintes fazem a fundamentação teórica necessária para os experimentos que poderão ser encontrados no capítulo 4. Assim, o texto está dividido da seguinte maneira:

- Capítulo 2 – Define textura, classificação de texturas, alguns pontos a serem levados em consideração na escolha do algoritmo e descreve cada um dos métodos estatísticos que serão usados nos experimentos.
- Capítulo 3 – Introduz os algoritmos de fidelidade de imagens que foram usados: Q e SSIM, além de alguns resultados em comparações simples.
- Capítulo 4 – Neste capítulo, poderão ser encontrados os experimentos que foram realizados utilizando os métodos descritos no capítulo 2 juntamente com os índices introduzidos no capítulo 3.
- Capítulo 5 – Este capítulo conclui o trabalho e fala sobre resultados alcançados e esperados, além de trabalhos futuros.

Capítulo 2

Texturas

Apesar de texturas serem largamente utilizadas, não existe uma definição precisa devido a sua grande variabilidade. Textura pode ser amplamente definida como a aparência ou característica visual e tátil de uma superfície. De acordo com Slansky [6], “uma região de imagem tem uma textura constante se o conjunto de suas propriedades locais naquela região é constante, varia lentamente ou é aproximadamente periódico”. A região da imagem, medida estatística ou propriedade que se repete sobre uma região texturizada é chamada de elemento de textura ou *texel*. No entanto, é importante perceber que, “regiões texturizadas produzem interpretações diferentes de acordo com a distância com que são observadas e de acordo com o grau de atenção visual” [7].

A textura possui um papel central na percepção humana, provendo dados para reconhecimento e interpretação e por isso é uma importante característica/propriedade para a análise de imagens digitais. Em algumas tarefas de visão computacional como inspeção de superfícies, classificação de cenas e determinação de formas existe uma forte dependência da correta extração e compreensão das informações da textura.

Em imagens digitais, a caracterização de uma textura pode ser percebida através de variações das intensidades ou cores capturadas. Apesar de não haver nenhuma informação quanto à causa das variações, as diferenças existentes nos *pixels* da imagem fornecem meios práticos de analisar as propriedades das texturas de um objeto.

Existem basicamente quatro questões relativas à análise de texturas: (i) classificação de texturas, (ii) segmentação de texturas, (iii) síntese de texturas e (iv) determinação de formas. Na classificação, o problema é identificar a textura de uma região dentre um dado conjunto de texturas conhecidas. Na segmentação, o objetivo é definir as fronteiras entre texturas com características semelhantes presentes numa imagem e pode ser feita de maneira supervisionada ou não-supervisionada. Criar um modelo, que poderá ser usado no futuro para gerar a textura é o foco do estudo da síntese de texturas. E por fim, determinar formas a partir de texturas significa considerar uma imagem em duas dimensões como sendo uma projeção de uma imagem em três dimensões; distorções aparentes na sua textura são usadas para estimar a orientação das superfícies [8].



Figura 2-1. Exemplo do uso da informação da textura de uma imagem em duas dimensões para determinar formas em três dimensões. a) Imagem em duas dimensões e b) imagem em três dimensões, gerada a partir da informação da textura

2.1 Classificação de Texturas

Na classificação de texturas, o objetivo é atribuir uma amostra desconhecida a uma classe de um conjunto de texturas conhecido. O processo envolve duas etapas: a de aprendizado e a de reconhecimento. Na primeira, o objetivo é construir um modelo/descrição para o conteúdo de cada textura das classes conhecidas. O conteúdo referentes à texturas, da imagem de treinamento, é obtido através do método de análise escolhido e resulta num conjunto de propriedades para cada imagem.

Essas propriedades, que podem ser números escalares, histogramas ou distribuições empíricas, caracterizam a textura em relação à estrutura espacial, contraste, rugosidade, orientação, por exemplo. Na etapa de reconhecimento, o conteúdo da amostra desconhecida é descrito utilizando o mesmo método de análise e comparado aos resultados das classes conhecidas utilizando um algoritmo de classificação. A amostra pode então ser atribuída como pertencente à classe com o melhor resultado ou, caso o resultado não seja bom o suficiente segundo um critério estabelecido, rejeitada [9].

Alguns aspectos devem ser considerados ao escolher o método de análise de textura, de acordo com o problema:

1. Invariância da iluminação (tons de cinza): o quanto o algoritmo é tolerante a mudanças nos tons de cinza. Isto é importante em aplicações industriais onde as condições de iluminação podem ser instáveis.
2. Invariância da escala espacial: como algoritmo se comporta se a escala das amostras desconhecidas é diferente da escala das classes do treinamento.
3. Invariância à rotação: o algoritmo suporta se a rotação da amostra muda em relação ao ponto de vista.
4. Robustez em relação à ruído: o quanto o algoritmo tolera ruído nas amostras de entrada.
5. Complexidade computacional: alguns algoritmos requerem tanto poder computacional que não podem ser considerados para aplicações com *throughput* elevado, como por exemplo, inspeção visual em tempo real.
6. Tamanho da janela: o quão grande deve ser a janela para que o algoritmo produza uma boa descrição do conteúdo da textura.

Existem outros aspectos que podem ser levados em consideração como por exemplo generatividade (se o algoritmo facilita a síntese de texturas) e tamanho da janela ou amostra (quão grande deve ser a janela para que o algoritmo produza uma descrição útil da textura).

2.2 Descritores de Texturas

Várias técnicas para descrever texturas foram propostas e em 1998 foi sugerida uma divisão em quatro categorias de métodos de análise de texturas: (i) estatísticos, (ii) geométricos, (iii) baseados em modelo e (iv) de processamento de sinal [10]. Neste trabalho serão usados três métodos estatísticos. Mais informações sobre os outros métodos podem ser encontradas em [11][12][13].

Os métodos estatísticos analisam a distribuição espacial dos tons de cinza, calculando características locais em cada ponto da imagem e derivando um conjunto de estatísticas a partir destas distribuições. Podem ser divididos de acordo com o número de *pixels* que definem as características locais: primeira ordem (um *pixel*), segunda ordem (dois *pixels*) ou ordem superior (três *pixels* ou mais). A diferença básica entre eles é que, enquanto os métodos de primeira ordem estimam propriedades (média, variância) de *pixels* individuais, os de ordem maior levam em conta a interação espacial e a localização relativa entre mais de um *pixel*. Os métodos estatísticos mais usados são a co-ocorrência de tons de cinza [1] e diferenças de tons de cinza [14], que inspiraram uma variedade de modificações, como por exemplo o LBP (*Local Binary Pattern* – Padrão Binário Local) [15], que incorpora estatísticas de microestruturas locais, combinando desta forma abordagem estatística e estrutural para a análise de texturas.

2.2.1 Matriz de Co-Ocorrência de Tons de Cinza

Uma GLCM (*Gray Level Co-Occurrence Matrix* - Matriz de co-ocorrência de tons de cinza) [1] funciona como um contador e contém informação sobre a posição de *pixels* que possuem tons de cinza semelhantes. A idéia é visitar cada *pixel* da imagem analisando sua vizinhança; uma célula (i, j) da matriz é incrementada sempre que dois *pixels* i e j estão separados por uma distância *d*. É comum levar em conta a direção entre estes *pixels*, portanto temos normalmente quatro direções (horizontal, vertical e as duas diagonais), produzindo assim, para cada valor de *d*, quatro matrizes de 256x256 (para uma imagem com 256 tons de cinza).

Apesar da quantidade de dados produzida (excedendo até a imagem original normalmente), estes dados serão analisados e usados para calcular valores numéricos simples (descritores) que encapsulam a informação.

2.2.2 Histogramas da Soma e da Diferença

Dada a grande quantidade de informação gerada pelo GLCM, o SDH (*Sum and Difference Histograms* – Histogramas da Soma e da Diferença) [16] foi proposto como alternativa computacionalmente mais eficiente já que não envolvia o cálculo de matrizes; nesta nova proposta, o *pixel* i é somado com dois indicadores de deslocamento, dx e dy; por fim, o histograma resultante é incrementado.

Desta forma, Unser mostrou ser possível calcular nove das quatorze propriedades propostas por Haralick, além de aproximar as outras cinco. Utilizando esta abordagem (substituindo somatórios duplos por simples, como pode ser visto na Tabela 2-1), o ganho computacional e de memória pode ser estimado por um fator $N_g/4$. Numa imagem com 32 níveis de cinza, por exemplo, a execução é oito vezes mais rápida e oito vezes menos memória é necessária quando comparada à avaliação utilizando matrizes de co-ocorrência.

Tabela 2-1. Comparativo entre GLCM e SDH para o cálculo de propriedades de textura.

Propriedades	GLCM	SDH
Média	$f_1 = \sum_i \sum_j i \cdot \hat{P}(i, j)$	$= \frac{1}{2} \sum_i i \cdot \hat{P}_s(i) = \mu$
Variância	$f_2 = \sum_i \sum_j (i - \mu)^2 \cdot \hat{P}(i, j)$	$= \frac{1}{2} \left(\sum_i (i - \mu)^2 \cdot \hat{P}_s(i) + \sum_j j^2 \cdot \hat{P}_d(j) \right)$
Energia	$f_3 = \sum_i \sum_j \hat{P}(i, j)^2$	$= \sum_i \hat{P}_s(i)^2 \cdot \sum_j \hat{P}_d(j)^2$
Contraste	$f_4 = \sum_i \sum_j (i - j)^2 \cdot \hat{P}(i, j)$	$= \sum_j j^2 \cdot \hat{P}_d(j)$

2.2.3 Padrão Binário Local

Unidades de textura são os oito elementos periféricos, que podem ter valores 0, 1 ou 2, obtidos de uma vizinhança de tamanho 3×3 *pixels*. No total, existem $3^8 = 6561$ unidades de textura possíveis descrevendo padrões espaciais de três níveis numa vizinhança de 3×3 *pixels*.

Foi baseado nesta idéia que o LBP (*Local Binary Pattern* – Padrão Binário Local) [15] foi proposto; no caso binário, existem apenas $2^8 = 256$ possibilidades. A máscara 3×3 (a) é limiarizada de acordo com o valor do *pixel* central (b), e a matriz resultante é multiplicada, elemento a elemento, por uma máscara formada por potências de dois (c). O somatório dos elementos da matriz final produz o valor desta unidade de textura, no nosso exemplo, 169 (d). Esta sequência de operações pode ser vista com maior detalhe na Figura 2-2.

$$\begin{array}{cccc}
 \begin{bmatrix} 6 & 5 & 2 \\ 7 & 6 & 1 \\ 9 & 3 & 7 \end{bmatrix} &
 \begin{bmatrix} 1 & 0 & 0 \\ 1 & \times & 0 \\ 1 & 0 & 1 \end{bmatrix} &
 \begin{bmatrix} 1 & 2 & 4 \\ 8 & \times & 16 \\ 32 & 64 & 128 \end{bmatrix} &
 \begin{bmatrix} 1 & 0 & 0 \\ 8 & \times & 0 \\ 32 & 0 & 128 \end{bmatrix} \\
 \text{(a)} & \text{(b)} & \text{(c)} & \text{(d)}
 \end{array}$$

Figura 2-2. Exemplo demonstrando os passos necessários para o cálculo do LBP. a) Imagem original, b) imagem binária gerada por comparação com o pixel central, c) máscara LBP e d) resultado da operação.

O contraste desta unidade pode ser facilmente calculado através da diferença da média simples entre os valores acima do limiar (no nosso exemplo, $6+7+9+7/4$) e abaixo ($5+2+1+3/4$).

2.3 Aplicações

Apresentamos aqui algumas aplicações relacionadas com texturas para motivar os estudos nessa área.

2.3.1 Segmentação de Texturas

Segmentação de texturas consiste em dividir uma imagem em regiões, de acordo com a textura presente em cada uma. No entanto, esta definição explica muito pouco sobre as dificuldades inerentes e limitações práticas envolvidas neste problema. Para começar, devemos considerar o termo ‘segmentação’ [17].

Uma imagem consiste de um *array* de *pixels* e o objetivo da segmentação é dar a um nome (*label*) a cada um deles de forma que possamos agrupa-los em regiões. Uma região é formada por um grupo de *pixels* que possui o mesmo nome, mas o que constitui uma região ‘correta’? Nem sempre é possível definir claramente a fronteira que separa duas amostras de texturas diferentes ou nem sempre queremos que texturas muito parecidas sejam separadas como se fossem 2 amostras.

É importante perceber que, para a segmentação propriamente dita, não é necessário identificar quais texturas foram usadas e sim, onde ficam as fronteiras entre as texturas diferentes. Na Figura 2-3, por exemplo, para resultado da segmentação, não é importante saber que “a textura do canto esquerdo inferior é de tijolos brancos”.

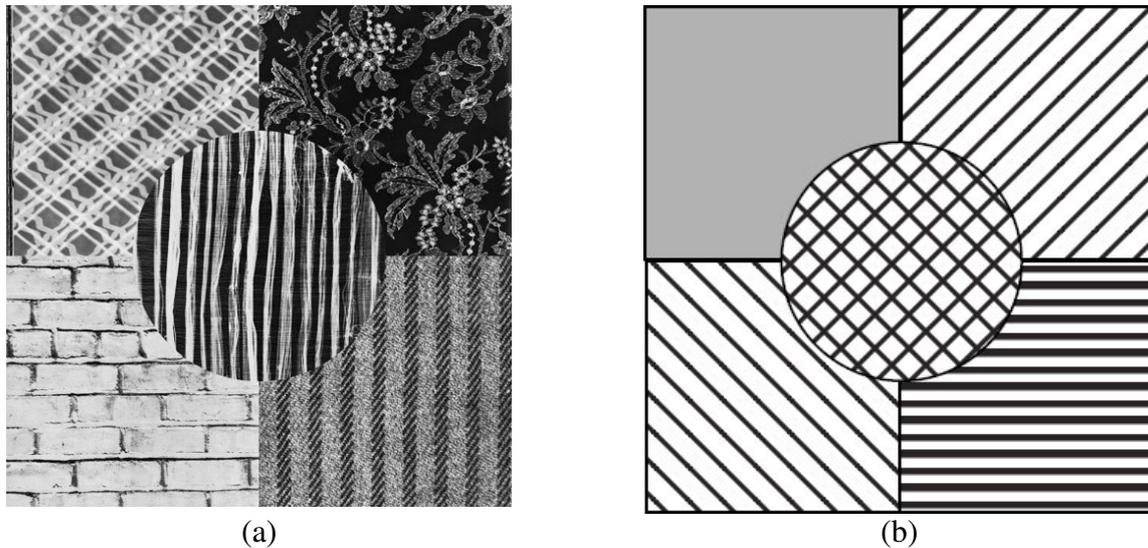


Figura 2-3. Objetivo da segmentação de texturas. a) Imagem composta por cinco texturas diferentes; em b) o resultado da segmentação ideal/perfeita da imagem [10].

Neste trabalho, será abordada a segmentação não-supervisionada, que apesar de mais computacionalmente intensiva, é mais genérica, permitindo assim uma maior aplicabilidade do estudo aqui desenvolvido.

2.3.2 Recuperação de Imagem Baseada em Textura

Recuperação de Imagem Baseada em Conteúdo (CBIR - *Content-Based Image Retrieval*) é uma área da visão computacional usada para pesquisar por imagens semelhantes aos parâmetros de entrada em grandes bancos de dados visuais, de acordo com o interesse do usuário [9]. Existem essencialmente duas maneiras de realizar esse tipo de busca: a primeira envolve fornecer um exemplar do tipo de imagem que se busca, por exemplo, uma paisagem; a segunda consiste em uma busca semântica, por exemplo, “procurar fotos de cachorros”. Este segundo tipo de pesquisa/operador é extremamente difícil para computadores já que existem cachorros de tamanho e cor diferentes e não necessariamente são facilmente identificáveis. Quando o objeto de busca é uma textura, a área é conhecida como Recuperação de Imagem Baseada em Textura (TBIR – *Texture-Based Image Retrieval*)

Os descritores de conteúdo de imagem podem considerar tanto o conteúdo visual quanto o semântico. Conteúdo visual pode ser dividido entre geral que inclui medidas baseadas na cor, textura e forma e de domínio específico, como o reconhecimento facial, que depende da aplicação e envolve conhecimento sobre um domínio específico. Já os descritores semânticos se apóiam fortemente em deduções e procedimentos sobre o conteúdo visual da imagem.

Capítulo 3

Fidelidade de imagens

As métricas para avaliação objetiva de imagens podem ser classificadas de acordo com a disponibilidade da imagem original (sem distorção). A maioria das abordagens utiliza *referência completa*, ou seja, um original sem distorção está disponível para comparações. Em alguns casos, não existe uma imagem original, sendo este caso conhecido como *sem referência* [18].

Existem basicamente duas abordagens quando o assunto é avaliação objetiva de imagens: a primeira consiste de métricas definidas matematicamente, como erro médio quadrático (EMQ), e a segunda, de métricas baseadas no sistema visual humano (SVH).

Métricas matematicamente definidas têm como principal vantagem a simplicidade computacional para o cálculo e a independência das condições visuais e de observadores individuais. Atualmente o EMQ e o PSNR (*Peak Signal-to-Noise Ratio*) são considerados o padrão em medida de qualidade de imagens e definidos por

$$EMQ = \frac{1}{m \cdot n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \|I(i, j) - K(i, j)\|^2 \quad (3-1)$$

$$PSNR = 10 \cdot \log_{10} \left(\frac{S^2}{EMQ} \right) \quad (3-2)$$

No entanto, com o uso do EMQ por exemplo, é possível que imagens com uma diferença perceptível de qualidade resultem no mesmo valor, já que este é sensível a variação de energia e não à perda real de informação. O PSNR apenas “adapta” (coloca em escala) o resultado do EMQ à variação máxima S da imagem (255 para uma imagem de 8 bits, por exemplo), já que um EMQ = 100 para uma imagem de 8 bits e outra de 10 bits têm visibilidade diferente [19].

Em [20] foi proposto um índice baseado em métricas matemáticas, chamado de *Universal Quality Index* (Índice Universal de Qualidade), no sentido de que esse índice não depende das condições das imagens que estão sendo testadas, das condições de visualização ou de cada observador individualmente. Esse índice pode ser calculado de acordo com a fórmula:

$$Q = \frac{4\sigma_{xy}\bar{x}\bar{y}}{(\sigma_x^2 + \sigma_y^2)[(\bar{x})^2 + (\bar{y})^2]}, \quad (3-3)$$

onde \bar{x} e \bar{y} são a média aritmética, calculadas por $\bar{x} = \frac{1}{N} \sum_{i=0}^N x_i$, σ_x^2 e σ_y^2 são a variância, calculadas por $\sigma_x^2 = \frac{1}{N-1} \sum_{i=0}^N (x_i - \bar{x})^2$ e σ_{xy} é o desvio-padrão, calculado por $\sigma_{xy} = \frac{1}{N-1} \sum_{i=0}^N (x_i - \bar{x})(y_i - \bar{y})$.

O índice leva em conta três fatores ao avaliar uma imagem: perda de correlação, distorção de luminância e distorção de contraste. Para melhor entender isso, podemos reescrever a fórmula como:

$$Q = \frac{\sigma_{xy}}{\sigma_x \sigma_y} \cdot \frac{2\bar{x}\bar{y}}{(\bar{x})^2 + (\bar{y})^2} \cdot \frac{2\sigma_x \sigma_y}{\sigma_x^2 \sigma_y^2}, \quad (3-4)$$

onde $\frac{\sigma_{xy}}{\sigma_x \sigma_y}$ representa o coeficiente de correlação entre x e y, que mede o grau de correlação linear entre x e y, variando entre 1 e -1. O valor máximo ocorre quando $y_i = ax_i + b$ para todo $i = 1, 2, \dots, N$, onde a e b são constantes e $a > 0$; $\frac{2\bar{x}\bar{y}}{(\bar{x})^2 + (\bar{y})^2}$ representa o quão próximas são as médias das luminâncias de x e y. Varia entre 0 e 1 e o valor máximo é atingido somente quando $\bar{x} = \bar{y}$ e $\frac{2\sigma_x \sigma_y}{\sigma_x^2 \sigma_y^2}$ representa o quão similar é o contraste das duas imagens, já que σ_x e σ_y podem ser vistos como uma estimativa dos contrastes de x e y. Este coeficiente também varia entre 0 e 1, e só é máximo se, e somente se, $\sigma_x = \sigma_y$.

Na Figura 3-1, temos cinco imagens que foram armazenadas no formato JPEG com diferentes taxas de perda, variando entre ‘nenhuma’ (para a imagem original da Figura 3-1a) e ‘90% sobre 80%’ de perda para a imagem da Figura 3-1e.

O resultado do uso deste índice Q com as imagens mostradas na Figura 3-1 demonstra uma correta classificação quanto a fidelidade das amostras em relação à original (Figura 3-1a). Isso pode ser visto na Tabela 3-1.



Figura 3-1. Comparativo entre imagens com compressão JPEG. Em a) Imagem original, em b) imagem com 15% de perda, c) com 30% de perda, d) com 80% de perda e e) com 90% sobre 80% de perda.

Tabela 3-1. Resultados da comparação utilizando o índice Q sobre a imagem da Figura 3-1a.

Imagem	Índice Q
Figura 3-1b	0,8861
Figura 3-1c	0,8379
Figura 3-1d	0,6958
Figura 3-1e	0,5667

Porém, já que nenhuma informação estrutural é levada em conta na operação, é possível que uma imagem com o contraste mais bem “espalhado” tenha um índice maior do que a imagem original (já que o espalhamento do contraste é considerada uma melhoria, por assim dizer).

Já as métricas baseadas no SVH procuram algumas características como, por exemplo, a alta capacidade em extrair informação estrutural para avaliar imagens. Em [18], é proposta uma métrica para avaliar semelhança estrutural comparando padrões locais de *pixels* que foram normalizados para luminância e contraste. Além disso, é sugerida uma nova filosofia onde, ao invés de adaptar modelos matemáticos e penalizar erros baseados na sua visibilidade (ou no quanto são perceptíveis), uma imagem recebe uma pontuação baixa de qualidade se informação estrutural da imagem original é permanentemente perdida. O novo índice é chamado de SSIM (*Structural Similarity*)

Essa nova abordagem pode ser melhor entendida através de uma comparação com a filosofia da “sensibilidade ao erro”. Enquanto esta estima ‘erros perceptíveis’ para quantificar a degradação de uma imagem, a nova abordagem considera as degradações de uma imagem como ‘mudanças perceptíveis na variação da informação estrutural’.

Para a elaboração do índice, foi levado em conta o fato de que enquanto a luminância da superfície de um objeto que está sendo observado é o produto da iluminação e das propriedades reflexivas, as estruturas dos objetos da cena não depende da iluminação. As características estruturais são, portanto, os atributos que representam as estruturas dos objetos da cena, independente da luminância e contraste médios. Já que existe variação nestes dois atributos, são usados a luminância e constraste locais na definição do índice.

O SSIM pode ser calculado através da fórmula:

$$SSIM(x, y) = \frac{(2\bar{x}\bar{y} + C_1)(2\sigma_{xy} + C_2)}{((\bar{x})^2 + (\bar{y})^2 + C_1) \cdot (\sigma_x^2 + \sigma_y^2 + C_2)}, \quad (3-5)$$

onde $C_1 = (K_1L)^2$ é uma constante, onde L é o valor máximo do *pixel* (255 para uma imagem de 8 bits) e $K_1 \ll 1$.

É possível perceber que o índice Q é um caso especial desta fórmula, quando $C_1 = C_2 = 0$ o que causa uma certa instabilidade quando $(\bar{x})^2 + (\bar{y})^2$ e $\sigma_x^2 + \sigma_y^2$ atingem valores muito próximos a zero.

Para exemplificar, vamos mostrar primeiro duas imagens onde, apesar da diferença no EMQ, a informação da estrutura permanece quase intacta já que a segunda recebeu apenas alterações no brilho e no contraste. Isso pode ser visto na Figura 3-2.

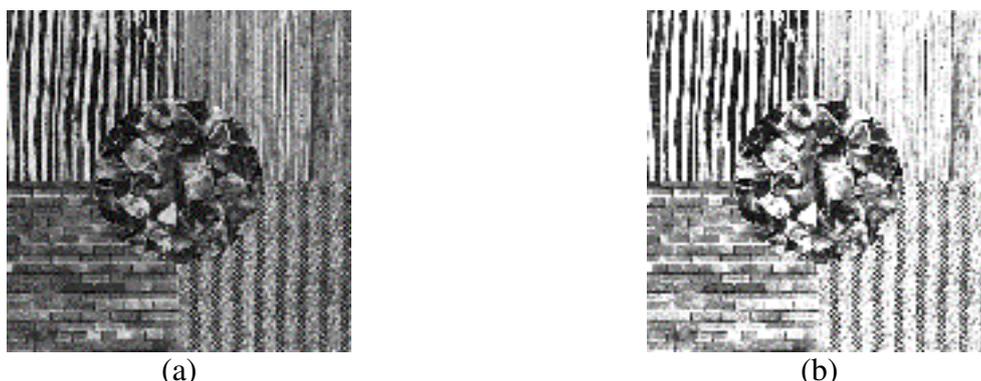
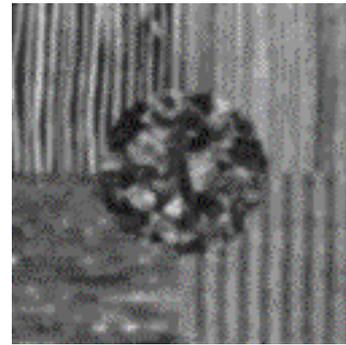


Figura 3-2. Exemplo de figuras com alterações visuais e estrutura preservada. a) Imagem Original; em b) Imagem com contraste e brilho alterados apresenta valor do SSIM = 0,4274.

Em seguida, a segunda imagem foi processada com um filtro passa-baixa (gaussiano) e o resultado da comparação demonstra perda de alguma informação estrutural (Figura 3-3).



(a)



(b)

Figura 3-3. Exemplo de figuras com alterações visuais e perda de informação estrutural. Em a) Imagem Original; em b) a imagem processada com um filtro passa-baixa apresenta valor do SSIM = 0,2850.

Capítulo 4

Experimentos

Para verificar o funcionamento dos algoritmos e também verificar o desempenho de índices de fidelidade na segmentação de texturas, uma série de experimentos foram realizados. Inicialmente, um total de vinte imagens foram geradas, utilizando texturas de *Brodatz* [21], para compor um banco de testes. Essa base de imagens de Brodatz é usada frequentemente em estudos relacionados a texturas. Assim, ela serve como um *benchmark* para a área. A base é formada por 112 imagens com 640x640 *pixels* armazenadas em tons de cinza.

Como primeiro experimento, cada um dos descritores vistos até agora foi testado sobre algumas amostras de textura escolhidas aleatoriamente visando determinar qual se comportava de maneira mais estável descrevendo padrões de texturas regulares. Em seguida, empregamos os índices de fidelidade estudados (Q e SSIM), além dos descritores comparados através do Q, para recuperação de imagens baseada em textura. No terceiro experimento, utilizamos os descritores para segmentar as imagens do banco de testes, variando alguns parâmetros objetivando a melhor separação das regiões de textura. E por fim, usamos o método de decomposição *Quadtree* [22] juntamente com descritores comparados através de Q para avaliar a segmentação nesse caso.

Como ambiente para a implementação e testes foi escolhido o MatLab [23] por sua facilidade de extensão e por se tratar de um padrão na área de visão computacional.

4.1 Imagens geradas

Para a realização dos experimentos, foram geradas vinte imagens de 512 *pixels* de largura por 512 *pixels* de altura formato bitmap-8bits utilizando texturas de *Brodatz*, divididas em três grupos (Figura 4-1):

- Seis imagens com quatro texturas diferentes.
- Seis imagens com cinco texturas diferentes, sendo uma delas posicionada sobre, no mínimo, duas outras.
- Seis imagens com cinco texturas diferentes, sendo uma delas circular, posicionada no centro da imagem.

Além destes grupos, foram geradas mais duas imagens atípicas, onde, nos grupos com cinco texturas, uma textura se repete, existindo assim cinco regiões onde foram usadas quatro texturas, como pode ser visto na Figura 4-2.

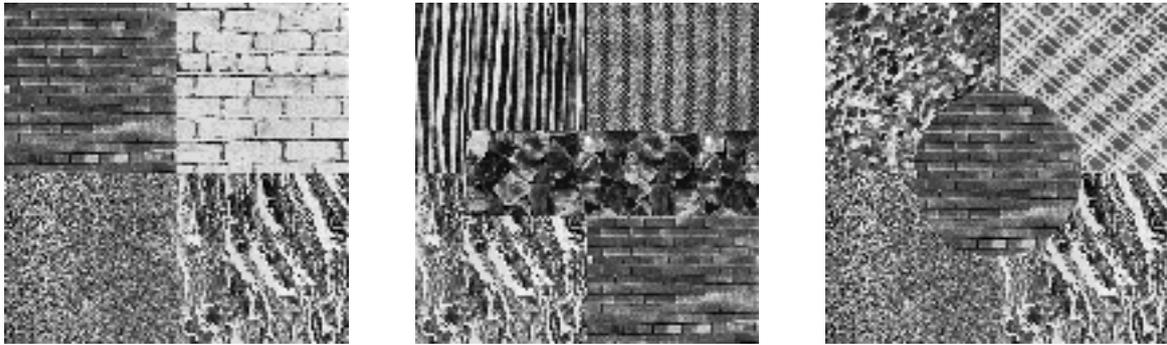


Figura 4-1. Exemplos de imagens geradas para testes de segmentação.



Figura 4-2. Imagens atípicas geradas para testes de segmentação.

4.2 Avaliação de Similaridade usando Descritores de Texturas

Os descritores foram testados inicialmente com quatro amostras de quatro classes de texturas distintas, escolhidas aleatoriamente. Para o LBP e SDH, foi necessário adaptar o índice de fidelidade de imagens Q para permitir a comparação em uma dimensão, já que é usado normalmente sobre matrizes de duas dimensões.

As quatro classes de textura abaixo foram divididas em quatro amostras de mesmo tamanho e foram calculados o GLCM, o SDH e o LBP de cada partição; em seguida os resultados foram usados como entrada do índice de fidelidade para calcular a semelhança entre eles.

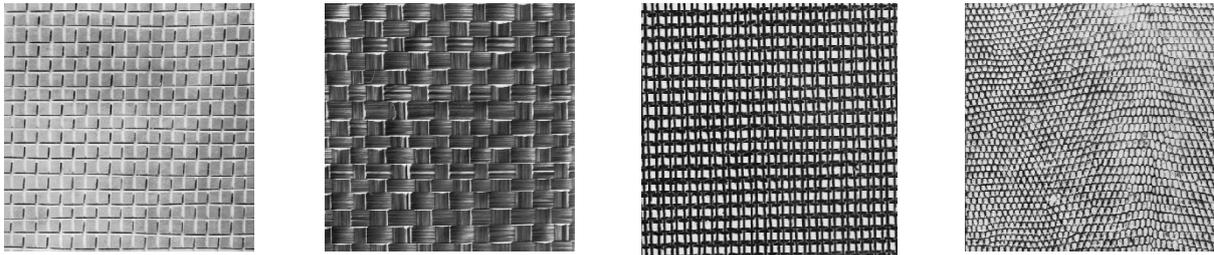


Figura 4-3. Classes usadas para testar os descritores de texturas.

Nas Tabelas 4-1, 4-3 e 4-5 é usada a seguinte convenção para indicar a comparação entre amostras: A1 -> A2, por exemplo, significa Amostra 1 comparada com a Amostra 2 da classe em questão. Já nas Tabelas 4-2, 4-4 e 4-6, a convenção é C2A1 por exemplo, indica a comparação com a “Classe 2 Amostra 1”.

4.2.1 GLCM

Apesar do GLCM não apresentar tanta certeza (ou um valor de Q próximo a 1) na comparação entre amostras da mesma classe (nota-se que o maior valor obtido não ultrapassa 0,66), o resultado da descrição gerada, para amostras de classes diferentes, mostra que para este experimento este algoritmo se saiu muito bem. A Tabela 4-1 contém os resultados das comparações amostra a amostra de cada classe.

Tabela 4-1. Resultados da comparação utilizando Q após descrição com GLCM para elementos da mesma classe.

	Amostras (A) da mesma Classe					
	A1 -> A2	A1 -> A3	A1 -> A4	A2 -> A3	A2 -> A4	A3 -> A4
Classe 1	0,6099	0,5115	0,6053	0,4925	0,6194	0,5073
Classe 2	0,6337	0,6238	0,6221	0,6220	0,6503	0,6190
Classe 3	0,5091	0,5545	0,4928	0,4846	0,5796	0,5074
Classe 4	0,5337	0,6194	0,6036	0,5363	0,5910	0,6154

Os valores na comparação com classes estrangeiras não ultrapassam 0,36 o que permite traçar um ponto de corte entre este valor e mínimo encontrado na tabela Tabela 4-1, obtendo uma separação perfeita. (Erro! A origem da referência não foi encontrada.)

Tabela 4-2. Resultados da comparação utilizando Q após descrição com GLCM para elementos de outras classes.

	C2A1	C2A2	C2A3	C2A4	C3A1	C3A2	C3A3	C3A4	C4A1	C4A2	C4A3	C4A4
C1A1	0,3304	0,3511	0,3326	0,3717	0,2392	0,2551	0,2638	0,2893	0,2592	0,3229	0,2809	0,3233
C1A2	0,3285	0,3413	0,3258	0,3594	0,2355	0,2498	0,2570	0,2785	0,2571	0,3201	0,2741	0,3157
C1A3	0,3348	0,3453	0,3289	0,3622	0,2497	0,2664	0,2691	0,2971	0,2578	0,3223	0,2751	0,3044
C1A4	0,3357	0,3588	0,3345	0,3709	0,2401	0,2580	0,2643	0,2911	0,2677	0,3331	0,2823	0,3298
C2A1	1,0000	0,6337	0,6238	0,6221	0,2557	0,2940	0,2758	0,3246	0,2794	0,3743	0,2793	0,3154
C2A2	0,6337	1,0000	0,6220	0,6503	0,2464	0,2829	0,2663	0,3117	0,2642	0,3575	0,2713	0,3069
C2A3	0,6238	0,6220	1,0000	0,6190	0,2574	0,2873	0,2794	0,3201	0,2695	0,3583	0,2737	0,3116
C2A4	0,6221	0,6503	0,6190	1,0000	0,2417	0,2748	0,2615	0,3063	0,2569	0,3503	0,2634	0,3027
C3A1	0,2557	0,2940	0,2758	0,3246	1,0000	0,5091	0,5545	0,4928	0,3295	0,2557	0,3267	0,2905
C3A2	0,2464	0,2829	0,2663	0,3117	0,5091	1,0000	0,4846	0,5796	0,3685	0,2993	0,3500	0,3283
C3A3	0,2574	0,2873	0,2794	0,3201	0,5545	0,4846	1,0000	0,5074	0,3212	0,2749	0,3528	0,3128
C3A4	0,2417	0,2748	0,2615	0,3063	0,4928	0,5796	0,5074	1,0000	0,3595	0,3190	0,3708	0,3503

4.2.2 SDH

O SDH (usando o histograma da soma) apresentou alguns problemas ao classificar algumas amostras da mesma classe. Como é possível notar na primeira entrada da tabela (Classe 1), o resultado do índice Q para a comparação entre as amostras 2 e 3 ficou com 0,42 e para as amostras 3 e 4 ficou em 0,40.

Este fato se repetiu para o primeiro resultado da Classe 4, onde a comparação entre as amostras 1 e 2 não superou 0,58. A Tabela 4-3 mostra estes resultados:

Tabela 4-3. Resultados da comparação utilizando Q após descrição com SDH para elementos da mesma classe.

	Amostras (A) da mesma Classe					
	A1 -> A2	A1 -> A3	A1 -> A4	A2 -> A3	A2 -> A4	A3 -> A4
Classe 1	0,8074	0,4803	0,7715	0,4224	0,8310	0,4060
Classe 2	0,7966	0,8320	0,7731	0,7939	0,8661	0,7944
Classe 3	0,8251	0,8754	0,7957	0,7786	0,8883	0,8510
Classe 4	0,5714	0,8051	0,7782	0,5849	0,6944	0,7531

Se para amostras da mesma classe o SDH apresentou valores baixos demais, quando usado para elementos de outras classes apresentou valores elevados demais, não permitindo portanto, traçar uma “linha” que separe com 100% de precisão uma classe de outra.

Por exemplo, se definirmos o ponto de corte em 0,40 (para atender à Tabela 4-3), estaríamos dizendo que a amostra 1 da classe 1 e a amostra 2 da classe 4 pertencem à mesma Classe. A Tabela 4-4 mostra todos os resultados das comparações realizadas.

Tabela 4-4. Resultados da comparação utilizando Q após descrição com SDH para elementos de outras classes.

	C2A1	C2A2	C2A3	C2A4	C3A1	C3A2	C3A3	C3A4	C4A1	C4A2	C4A3	C4A4
C1A1	0,3026	0,3043	0,3091	0,2956	0,2988	0,3278	0,3100	0,3170	0,3482	0,4884	0,3492	0,4296
C1A2	0,3053	0,3047	0,3025	0,2937	0,3117	0,3673	0,3230	0,3498	0,3694	0,5506	0,3615	0,4396
C1A3	0,3026	0,2806	0,2844	0,2844	0,2622	0,2789	0,2782	0,2855	0,2918	0,3876	0,2847	0,3352
C1A4	0,2740	0,2726	0,2995	0,2946	0,3059	0,3583	0,3206	0,3418	0,3609	0,5129	0,3593	0,4568
C2A1	1,0000	0,7966	0,8320	0,7731	0,3060	0,3325	0,3531	0,3652	0,3522	0,3134	0,3659	0,3161
C2A2	0,7966	1,0000	0,7939	0,8661	0,2763	0,3037	0,3186	0,3287	0,3249	0,3237	0,3511	0,3119
C2A3	0,8320	0,7939	1,0000	0,7944	0,2799	0,3080	0,3276	0,3313	0,3266	0,2990	0,3615	0,3328
C2A4	0,7731	0,8661	0,7944	1,0000	0,2809	0,3109	0,3239	0,3346	0,3290	0,3165	0,3630	0,3342
C3A1	0,3060	0,3325	0,3531	0,3652	1,0000	0,8251	0,8754	0,7957	0,5066	0,3943	0,5155	0,4699
C3A2	0,2763	0,3037	0,3186	0,3287	0,8251	1,0000	0,7786	0,8883	0,5353	0,4698	0,5533	0,5189
C3A3	0,2799	0,3080	0,3276	0,3313	0,8754	0,7786	1,0000	0,8510	0,4706	0,3992	0,5365	0,4818
C3A4	0,2809	0,3109	0,3239	0,3346	0,7957	0,8883	0,8510	1,0000	0,5093	0,4413	0,5517	0,5081

4.2.3 LBP

O LBP apresentou excelente comportamento neste teste em particular: o resultado da comparação com imagens da mesma classe atingiu valores excelentes, enquanto as comparações com outras classes obtiveram valores reduzidos, possibilitando uma separação perfeita. Na Tabela 4-5 podemos notar que todas as comparações tiveram valores acima de 0,95.

Tabela 4-5. Resultados da comparação utilizando Q após descrição com LBP para elementos da mesma classe.

	Amostras (A) da mesma Classe					
	A1 -> A2	A1 -> A3	A1 -> A4	A2 -> A3	A2 -> A4	A3 -> A 4
Classe 1	0,9855	0,9807	0,9778	0,9818	0,9833	0,9874
Classe 2	0,9680	0,9747	0,9754	0,9704	0,9780	0,9797
Classe 3	0,9822	0,9852	0,9815	0,9860	0,9877	0,9833
Classe 4	0,9645	0,9826	0,9505	0,9683	0,9841	0,9589

A Tabela 4-6 mostra como o resultado da comparação com elementos de outras classes atingiu valores baixos, permitindo afirmar com precisão se a amostra analisada pertence ou não a amostra original.

Tabela 4-6. Resultados da comparação utilizando Q após descrição com LBP para elementos de outras classes.

	C2A1	C2A2	C2A3	C2A4	C3A1	C3A2	C3A3	C3A4	C4A1	C4A2	C4A3	C4A4
C1A1	0,7105	0,6784	0,7113	0,7008	0,8135	0,8094	0,8115	0,8028	0,8137	0,8278	0,8047	0,8026
C1A2	0,7076	0,6813	0,7141	0,7042	0,8091	0,8011	0,8025	0,7963	0,8174	0,8245	0,8112	0,8007
C1A3	0,7019	0,6765	0,7049	0,6959	0,7870	0,7807	0,7818	0,7720	0,8050	0,8163	0,7945	0,7907
C1A4	0,7271	0,7026	0,7320	0,7244	0,7990	0,7939	0,7959	0,7877	0,8119	0,8220	0,8022	0,7971
C2A1	1,0000	0,9680	0,9747	0,9754	0,6508	0,6659	0,6725	0,6726	0,6395	0,6390	0,6258	0,6260
C2A2	0,9680	1,0000	0,9704	0,9780	0,6138	0,6308	0,6369	0,6405	0,6156	0,6079	0,6000	0,5959
C2A3	0,9747	0,9704	1,0000	0,9797	0,6378	0,6506	0,6593	0,6590	0,6288	0,6270	0,6180	0,6155
C2A4	0,9754	0,9780	0,9797	1,0000	0,6413	0,6567	0,6619	0,6639	0,6344	0,6269	0,6205	0,6161
C3A1	0,6508	0,6659	0,6725	0,6726	1,0000	0,9822	0,9852	0,9815	0,8401	0,8772	0,8606	0,8801
C3A2	0,6138	0,6308	0,6369	0,6405	0,9822	1,0000	0,9860	0,9877	0,8453	0,8873	0,8637	0,8887
C3A3	0,6378	0,6506	0,6593	0,6590	0,9852	0,9860	1,0000	0,9883	0,8418	0,8795	0,8611	0,8810
C3A4	0,6413	0,6567	0,6619	0,6639	0,9815	0,9877	0,9883	1,0000	0,8340	0,8756	0,8549	0,8797

4.3 Recuperação de Imagem Baseada em Textura

Nessa aplicação, os algoritmos implementados foram testados utilizando uma base de figuras com 1.452 amostras de seis classes diferentes (242 amostras/classe). O foco dos algoritmos foi a recuperação de imagens baseada no conteúdo visual, mais especificamente, dados sobre a textura.

No primeiro teste, uma amostra de cada classe foi comparada com todas as demais usando o índice de fidelidade Q com uma janela de tamanho 8 (8×8 pixels). Em seguida, a janela foi aumentada para 32×32 pixels, buscando obter mais informações sobre cada amostra antes da comparação. No segundo teste, foi usado o SSIM para comparar as amostras e por último, uma amostra de cada classe foi descrita usando cada um dos algoritmos (GLCM, SDH e LBP) e então comparada com cada outra amostra da base de figuras usando o índice Q. As seis classes usadas nos testes podem ser vistas na Figura 4-4.

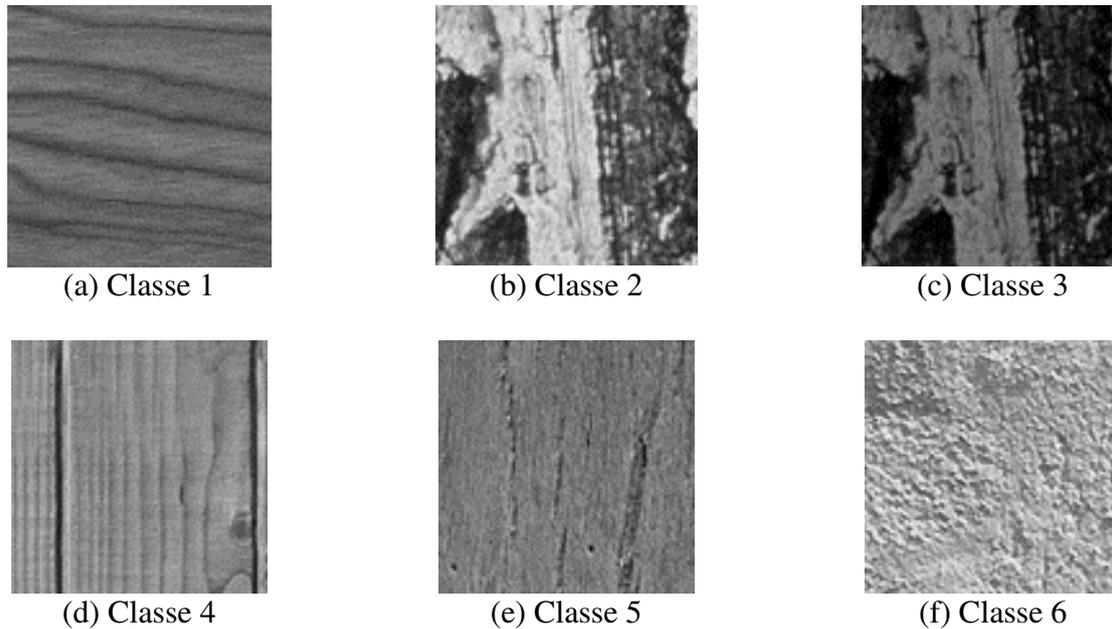


Figura 4-4. Classes da base de figuras usadas para CBIR.

A classe 4, como veremos nos testes a seguir, varia bastante dentre suas 242 amostras, alternando entre amostras mais claras e regulares e amostras escuras com falhas, o que dificulta a sua classificação.

4.3.1 TBIR com Q e janela de 8x8 pixels

No primeiro teste, o índice Q foi usado para recuperação de imagens baseada em textura. O desempenho não foi bom, indicando confusão ao classificar classes como a 2, 3, 4, se estabelecermos um ponte de corte (*threshold*) com valor elevado. (Tabela 4-7)

Tabela 4-7. Média de valores para Q com janela de 8x8 pixels.

	Média (Q8)					
	Classe 1	Classe 2	Classe 3	Classe 4	Classe 5	Classe 6
Classe 1	0,98350	0,83768	0,65746	0,92556	0,94394	0,78005
Classe 2	0,82102	0,77809	0,54417	0,82448	0,84728	0,81558
Classe 3	0,68000	0,57359	0,61977	0,60912	0,59962	0,45466
Classe 4	0,89221	0,74917	0,68074	0,81596	0,80808	0,62714
Classe 5	0,95679	0,85818	0,60057	0,94552	0,97918	0,87864
Classe 6	0,78107	0,81694	0,44433	0,86249	0,91131	0,98979

Para um *threshold* = 0,975, temos um total de apenas 668 amostras classificadas corretamente (verdadeiros positivos) de um máximo de 1.452 (levando em conta 242 acertos possíveis em cada uma das seis classes). O número de falsos positivos (amostras classificadas como pertencentes a uma classe errada) ficou em 184. (Tabela 4-8)

Tabela 4-8. Valores do classificador usando Q com janela tamanho 8x8 *pixels*.

Q8 <i>Threshold = 0,975</i>	Positivos		Negativos	
	Verdadeiros	Falsos	Verdadeiros	Falsos
Total	668	184	7076	784
Normalizado	0,460	0,127	0,975	0,108

4.3.2 TBIR com Q e janela de 32x32 *pixels*

O teste seguinte foi idêntico ao primeiro, exceto pela variação em um dos parâmetros do algoritmo de fidelidade Q: o tamanho da janela. O tamanho da janela influencia o algoritmo no sentido de que, uma janela maior permite obter mais informações sobre o trecho que está sendo analisado, o que influi diretamente em atributos matemáticos como média, variância, etc.

Apesar de o aumento no tamanho da janela, os resultados da classificação pouco mudaram. As classes 2, 3 e 4 continuaram apresentando confusão, no entanto, as classes 1, 5 e 6 apresentaram maior “certeza” na classificação, com valores próximos ao máximo, como por exemplo, 0,995 (classe 1), 0,987 (classe 5) e 0,995 (classe 6). A Tabela 4-9 apresenta a média dos resultados alcançados nesse experimento.

Tabela 4-9. Média de valores de Q com janela de 32x32 *pixels*.

	Média (Q32)					
	Classe 1	Classe 2	Classe 3	Classe 4	Classe 5	Classe 6
Classe 1	0,99542	0,92204	0,73170	0,94179	0,95586	0,78442
Classe 2	0,84931	0,90263	0,53195	0,88885	0,92594	0,92514
Classe 3	0,84020	0,70965	0,81997	0,74189	0,72463	0,52714
Classe 4	0,91394	0,77875	0,83592	0,81650	0,79823	0,58685
Classe 5	0,96423	0,95067	0,64308	0,95765	0,98700	0,88068
Classe 6	0,78588	0,89710	0,44118	0,87252	0,92006	0,99553

Para um *threshold* = 0,987, temos um resultado muito parecido com teste anterior: 682 verdadeiros positivos e 178 falsos positivos. Sendo assim, é possível concluir que para essa aplicação específica, usar apenas o índice Q não alcança bons resultados. (Tabela 4-10)

Tabela 4-10. Valores do classificador para Q com janela de 32x32 *pixels*.

Q32 <i>Threshold = 0,987</i>	Positivos		Negativos	
	Verdadeiros	Falsos	Verdadeiros	Falsos
Total	682	178	7082	770
Normalizado	0,470	0,123	0,975	0,106

4.3.3 TBIR com SSIM

No segundo teste, foi utilizado o SSIM sem alterações nos seus parâmetros originais. O resultado não foi bom, indicando problemas para classificar mesmo as texturas mais simples (sem muitas alterações visuais e sem muitas semelhanças com outras classes). Apenas a Classe 1 foi

classificada corretamente em relação às outras, se consideramos um ponto de corte próximo a 0,2864. Os resultados médios alcançados estão dispostos na Tabela 4-11.

Tabela 4-11. Média de valores de SSIM para os parâmetros originais.

	Média (SSIM)					
	Classe 1	Classe 2	Classe 3	Classe 4	Classe 5	Classe 6
Classe 1	0,2864	0,0764	0,0972	0,2500	0,1495	0,0482
Classe 2	0,0869	0,0470	0,0456	0,0813	0,0618	0,0305
Classe 3	0,1203	0,0481	0,0710	0,1050	0,0705	0,0266
Classe 4	0,2096	0,0647	0,0862	0,1848	0,1103	0,0373
Classe 5	0,2204	0,0674	0,0781	0,1991	0,1320	0,0479
Classe 6	0,0558	0,0329	0,0281	0,0534	0,0477	0,0355

Um fato que chamou a atenção durante os testes do SSIM, foi que o desvio padrão atingiu valores extremamente altos para metade das classes, indicando grande variabilidade no processo de classificação das texturas. Na Tabela 4-12, a relação da porcentagem do desvio padrão com a média para a Classe 2, Classe 3 e Classe 6 justifica os resultados obtidos: para um *threshold* = 0,275 por exemplo, temos apenas 162 verdadeiros positivos e 86 falsos positivos, valores baixos, se comparados com o uso de índice Q.

Tabela 4-12. Desvio Padrão e Porcentagem em relação à média por Classe de Textura.

	Desvio Padrão por Classe					
	Classe 1	Classe 2	Classe 3	Classe 4	Classe 5	Classe 6
Desvio Padrão	0,0666	0,0643	0,0628	0,0676	0,0590	0,0641
% em relação à Média	23%	137%	89%	37%	45%	181%

Na Tabela 4-13 é possível ver os valores totais e normalizados para o *threshold* escolhido.

Tabela 4-13. Valores do classificador para SSIM.

SSIM	Positivos		Negativos	
	Verdadeiros	Falsos	Verdadeiros	Falsos
Threshold = 0,275				
Total	162	86	7171	1284
Normalizado	0,112	0,059	0,988	0,177

4.3.4 TBIR com LBP

Usando o LBP obtivemos a segunda melhor separação entre os três algoritmos descritores. Para as texturas mais regulares (ou com menos alterações visuais), neste caso as classes 1, 5 e 6, a média dos resultados das comparações se manteve sempre acima de 0,93 com desvio padrão de até 0,02. Nos testes destas classes em específico, as outras classes ficaram com valores baixos, indicando uma boa identificação da textura alvo nestes casos.

Por outro lado, as classes 2, 3 e 4 apresentaram descrições confusas, o que era de se esperar já que as classes 2 e 3 são essencialmente a mesma textura com modificações de contraste

e brilho e portanto, semelhantes para o LBP (que leva em conta elementos estruturais além da distribuição espacial de tons de cinza). Já a classe 4 apresenta uma grande variabilidade entre suas amostras e por isso a dificuldade de classificá-la corretamente. Na Tabela 4-14 observamos os resultados médios alcançados nesse teste.

Tabela 4-14. Média dos valores de Q para descritor LBP.

	Média do valor Q para LBP					
	Classe 1	Classe 2	Classe 3	Classe 4	Classe 5	Classe 6
Classe 1	0,93883	0,37811	0,37672	0,35307	0,32721	0,55814
Classe 2	0,41178	0,95348	0,94768	0,80603	0,79188	0,62273
Classe 3	0,40389	0,95115	0,95042	0,81093	0,79163	0,61081
Classe 4	0,62903	0,78465	0,77818	0,74643	0,70813	0,72593
Classe 5	0,39116	0,77304	0,76083	0,82425	0,95107	0,60011
Classe 6	0,62603	0,63374	0,62360	0,55976	0,58560	0,96204

Utilizando um valor limiar de 0,945 tivemos 852 verdadeiros positivos e 302 falsos positivos além de 6.958 verdadeiros negativos (de um máximo possível de 7.260) como pode ser visto na Tabela 4-15.

Tabela 4-15. Valores do classificador para LBP.

LBP <i>Threshold = 0,945</i>	Positivos		Negativos	
	Verdadeiros	Falsos	Verdadeiros	Falsos
Total	852	6958	302	600
Normalizado	0,587	4,792	0,042	0,083

4.3.5 TBIR com SDH

Os resultados do SDH (utilizando o histograma da soma) foram melhores do que os obtidos pelo uso do LBP já que o SDH se mostrou mais robusto em relação às mudanças de brilho e contraste das classes 2 e 3. Assim sendo, as classes 1, 2, 3 e 6 apresentaram resultados bastante promissores (0,867; 0,772; 0,865 e 0,738) enquanto a média para as outras classes não alcançou mais do que 0,55.

Desta forma, fica claro que o SDH gerou uma descrição mais precisa de cada amostra, o que resultou numa boa identificação quando se tratava da mesma classe e uma excelente diferenciação quando comparada com classes diferentes. A Tabela 4-16 contém os valores médios gerados pela comparação através do índice Q.

Tabela 4-16. Média dos valores de Q para SDH da soma.

	Média do valor Q para SDH					
	Classe 1	Classe 2	Classe 3	Classe 4	Classe 5	Classe 6
Classe 1	0,867514	0,247479	0,532979	0,54649	0,553265	0,336159
Classe 2	0,273889	0,772468	0,305982	0,395155	0,375664	0,482604
Classe 3	0,551271	0,339904	0,865945	0,410542	0,407043	0,08368
Classe 4	0,543342	0,454817	0,606996	0,57599	0,563703	0,233215
Classe 5	0,589887	0,300118	0,378267	0,61141	0,644758	0,315814
Classe 6	0,308739	0,445462	0,093606	0,351605	0,335234	0,738131

Os resultados do classificador usando SDH, para o histograma da soma, foram bem eficientes:

- 1.131 verdadeiros positivos (de 1.452 possíveis);
- 114 falsos positivos;
- 321 falsos negativos;
- 7.146 verdadeiros negativos (de 7.260 possíveis).

Os valores obtidos pelo classificador podem ser observados na Tabela 4-17.

Tabela 4-17. Valores do classificador para SDH.

SDH <i>Threshold = 0,65</i>	Positivos		Negativos	
	Verdadeiros	Falsos	Verdadeiros	Falsos
Total	1131	114	7146	321
Normalizado	0,779	0,079	0,984	0,044

4.3.6 TBIR com GLCM

Os resultados do GLCM demonstraram bastante confusão ao classificar todas as classes (exceto a classe 1 com valor de 0,962 e máximo de 0,74 para outras classes, obtido na comparação com a classe 3). Por exemplo, as classes 3, 4 e 5 obtiveram valor médio menor comparado com a classe 1 do que com elas mesmas. (Tabela 4-18)

Tabela 4-18. Média dos valores de Q para o GLCM.

	Média do valor Q para GLCM					
	Classe 1	Classe 2	Classe 3	Classe 4	Classe 5	Classe 6
Classe 1	0,962333	0,455614	0,742337	0,71457	0,624961	0,535344
Classe 2	0,44855	0,468511	0,386792	0,428025	0,397095	0,392516
Classe 3	0,731516	0,397932	0,75477	0,611501	0,574247	0,374299
Classe 4	0,657177	0,408131	0,614628	0,625742	0,556601	0,387279
Classe 5	0,750146	0,441581	0,654616	0,653376	0,629252	0,462591
Classe 6	0,545901	0,400299	0,38976	0,461709	0,394088	0,547154

Essa confusão ao classificar se reflete na análise do resultado: para um limiar de 0,84, onde obtivemos 308 verdadeiros positivos e apenas 12 falsos positivos. Se reduzirmos o ponto de corte para 0,745 temos 375 verdadeiros positivos e 387 falsos positivos, ou seja, um resultado insatisfatório. (Tabela 4-19)

Tabela 4-19. Valores do classificador para GLCM.

GLCM <i>Threshold = 0,65</i>	Positivos		Negativos	
	Verdadeiros	Falsos	Verdadeiros	Falsos
Total	308	12	7248	1144
Normalizado	0,212	0,008	0,998	0,158

4.4 Segmentação de Imagens baseada em Texturas

Para os experimentos de segmentação de imagens, são usadas as imagens geradas com as texturas de Brodatz o processo se dá da seguinte forma:

1. A imagem inteira é analisada e um mapa é gerado contendo o valor de cada descritor (seja ele com GLCM, SDH ou LBP) para cada janela.
2. Em seguida, um ponto de corte (chamado de *delta*) é definido e será usado para caracterizar como a janela deverá ser classificada; cada janela é então comparada com todas as outras, utilizando o índice de fidelidade Q: se o resultado for maior que o ponto de corte, a janela é classificada como pertencendo à mesma classe das anteriores; caso contrário, considera-se que uma nova região foi encontrada.
3. Quando a última janela for classificada, é gerada uma imagem contendo as regiões que foram encontradas.

Os valores para o ponto de corte são determinados empiricamente, através de várias execuções sobre a mesma imagem já que variam bastante dependendo do descritor e do tamanho da janela. Com esses valores disponíveis, todas as imagens geradas são classificadas utilizando os três descritores vistos até agora. A imagem usada neste momento pode ser vista na Figura 4-5.

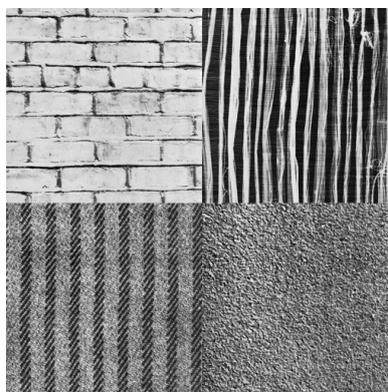


Figura 4-5. Imagem usada para ajustes no valor de delta.

4.4.1 Segmentação utilizando GLCM

A segmentação utilizando o GLCM foi executada com janelas quadradas de tamanho 32 e 16 *pixels*. Janelas de tamanho 8x8 *pixels* não foram testadas, devido ao tempo de processamento que leva para executar esse experimento. Por exemplo, o mapa gerado no primeiro passo do algoritmo, para janelas de tamanho 16, é uma matriz de 32x32 *pixels*, onde cada uma das suas 1.024 células é uma outra matriz de tamanho 256x256 *pixels* (que é o GLCM daquela janela) e o teste levou cerca de 15 horas para executar em computador Intel Core 2 Duo 2.0Ghz, com 2GB de RAM.

Se considerarmos a janela de tamanho 8x8 *pixels*, o mapa seria quatro vezes maior e um total de 4.096 janelas precisariam ser comparadas umas com as outras (como descrito no passo 3).

Variação nos valores de delta para janelas de tamanho 32x32 *pixels*

Para janelas de tamanho 32x32 *pixels*, o valor do delta variou entre 0,52 e 0,57. Os resultados da segmentação utilizando GLCM com esses valores pode ser visto na Figura 4-6.

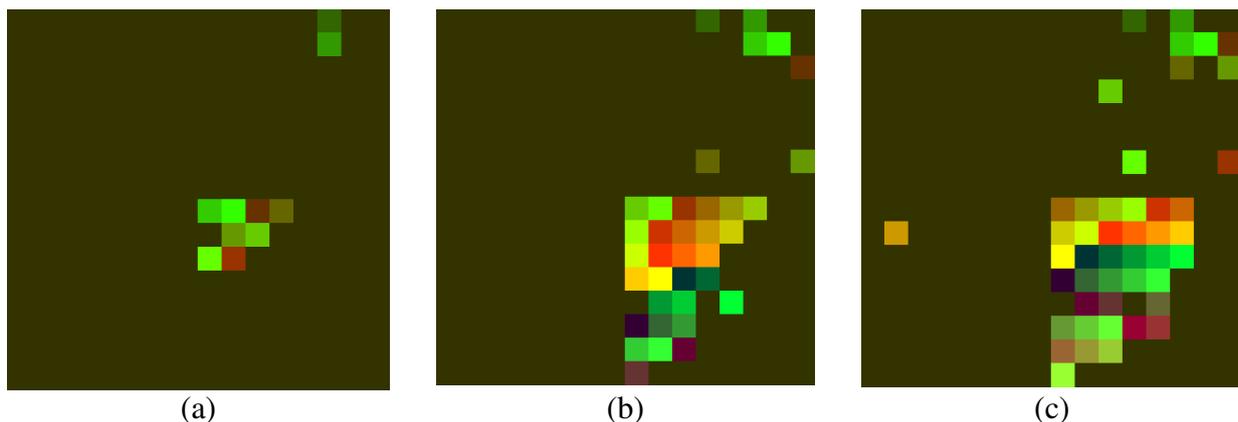


Figura 4-6. Resultado da segmentação com variação do delta: em a) delta = 0,52 (11 regiões); em b) delta = 0,55 (37 regiões) e em c) delta = 0,57 (47 regiões).

Variação nos valores de delta para janelas de tamanho 16x16 *pixels*

Para janelas de tamanho 16x16 *pixels*, o delta usado variou entre 0,6 e 0,8. O resultado da segmentação pode ser visto na Figura 4-7.

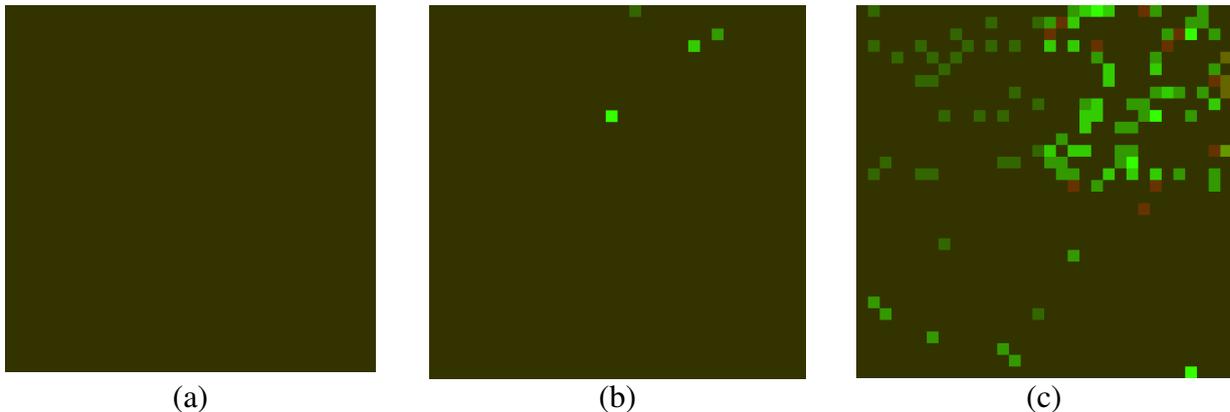


Figura 4-7. Resultado da segmentação com variação do delta: em a) delta = 0,6 (1 região); em b) delta = 0,7 (4 regiões) e em c) delta = 0,8 (18 regiões).

4.4.2 Segmentação utilizando LBP

Para a segmentação utilizando LBP, foram realizados testes com janelas quadradas de 16 e 32 pixels.

Variação nos valores de delta para janelas de tamanho 32x32 pixels

Após realizar o teste inicial com valores bem espalhados (variando entre 0,3 e 0,7) e não obter os resultados esperados, o valor do delta foi alterado para variar entre 0,63 e 0,7. O resultado da segmentação pode ser visto na Figura 4-8:

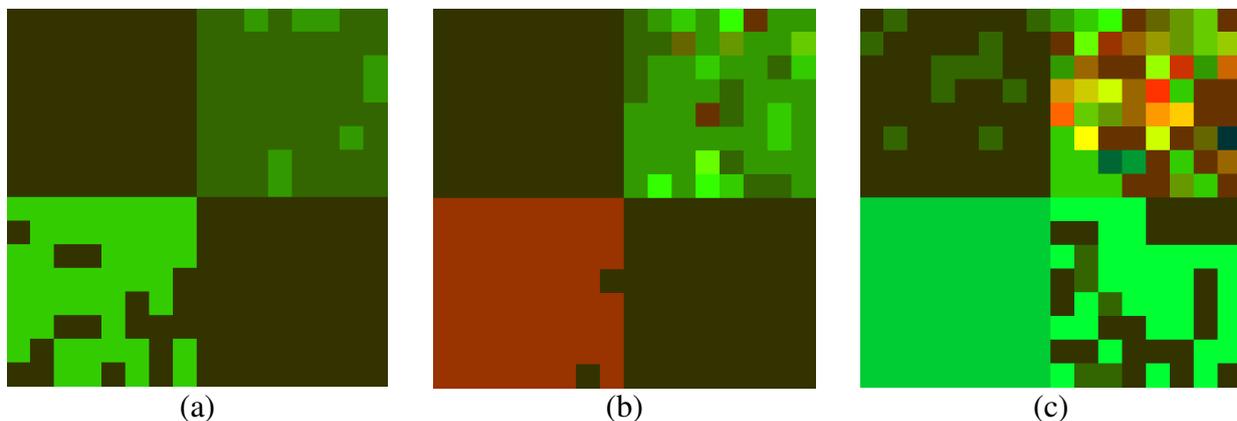


Figura 4-8. Resultado da segmentação com variação do delta: em a) delta = 0,63 (4 regiões); em b) delta = 0,66 (11 regiões) e em c) delta = 0,70 (30 regiões).

Variação nos valores de delta para janelas de tamanho 16x16 pixels

Para janelas de 16x16 pixels os valores de delta usados foram: 0,40; 0,46 e 0,5. Mesmo para essa pequena variação entre os valores, houve muita diferença entre o número de regiões encontradas. O resultado pode ser visto na Figura 4-9.

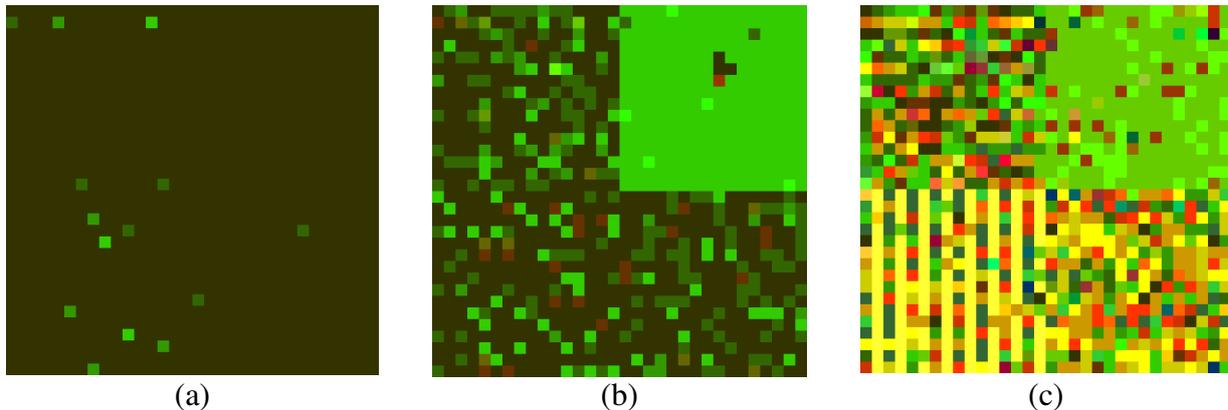


Figura 4-9. Resultado da segmentação com variação do delta: em a) delta = 0,40 (4 regiões); em b) delta = 0,46 (11 regiões) e em c) delta = 0,50 (64 regiões).

4.4.3 Segmentação utilizando SDH

Para a segmentação utilizando SDH, com o histograma da soma, foram utilizadas janelas quadradas de 8, 16 e 32 *pixels*.

Variação nos valores de delta para janelas de tamanho 32x32 *pixels*

Para janelas quadradas de tamanho 32 *pixels* testamos delta com valores iniciais de 0,44; 0,48 e 0,52. Como é possível observar na Figura 4-10, com 0,44 a segmentação resultou em poucas regiões. Quando o valor de delta foi igual a 0,48 tivemos 13 regiões encontradas. Já quando o valor de delta = 0,52 o número de regiões saltou para 27. (Figura 4-10)

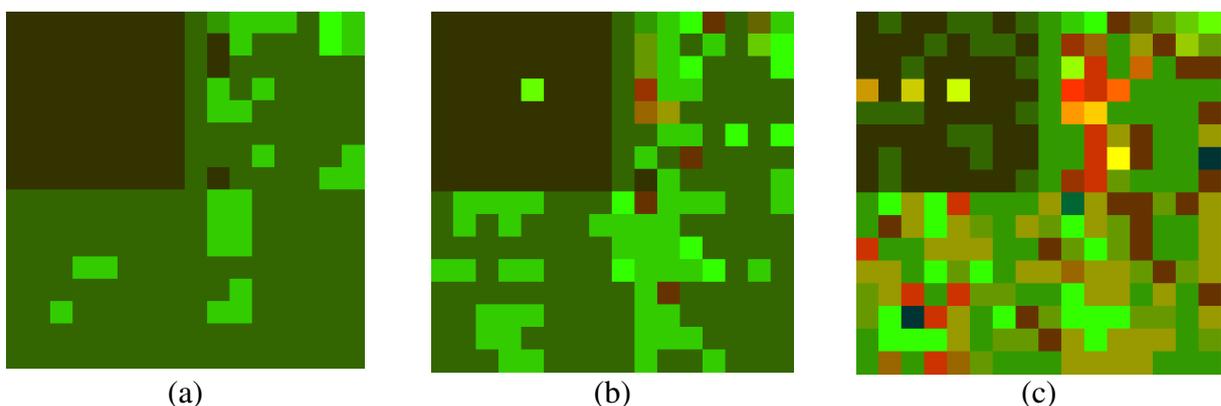


Figura 4-10. Resultado da segmentação com variação do delta: em a) delta = 0,44 (5 regiões); em b) delta = 0,48 (13 regiões) e em c) delta = 0,52 (27 regiões).

Variação nos valores de delta para janelas de tamanho 16x16 *pixels*

Inicialmente, foram usados valores entre 0,3 e 0,7 visando determinar as faixas onde acontecem sub-segmentação e sobre-segmentação, para só então escolher uma faixa mais restrita de valores e testar novamente. Neste teste em particular, foi utilizado delta variando entre 0,45 e 0,52 já que os valores usados inicialmente não deram uma boa resposta. O resultado pode ser observado Figura 4-11.

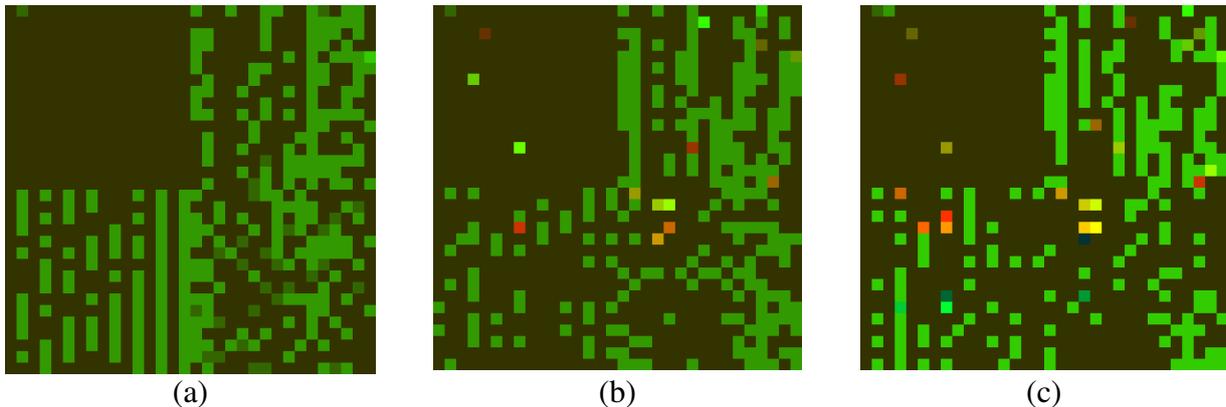


Figura 4-11. Resultado da segmentação com variação do delta: em a) delta = 0,45 (4 regiões); em b) delta = 0,5 (18 regiões) e em c) delta = 0,52 (30 regiões).

4.4.4 Resultados da Segmentação do Banco de Imagens

Após a definição de maneira empírica dos melhores valores para delta, variando de acordo com o tamanho da janela e com o algoritmo descritor, esses valores foram utilizados para segmentar o restante das imagens que foram geradas. Nas Tabelas 4-20, 4-21, 4-22 e 4-23, “4 Text.,” “5 Text.” e “5 Text. (Circ.)” referem-se às três classes de imagens que foram geradas: com 4 texturas, com 5 texturas e com 5 texturas sendo 1 delas circular. Na Tabela 4-20 pode-se observar o resultado da segmentação destas pelo SDH.

Tabela 4-20. Resultado da segmentação pelo SDH (janela = 32, delta = 0,44)

SDH - Janela = 32 e delta = 0,44			
	4 Text.	5 Text.	5 Text. (Circ.)
Imagem 1	5	10	1
Imagem 2	5	7	5
Imagem 3	5	8	5
Imagem 4	4	5	5
Imagem 5	6	4	11
Imagem 6	5	8	10

Para janelas de tamanho 16x16 *pixels* o SDH demonstrou uma certa instabilidade ao classificar algumas janelas, resultando em alguns casos onde foi encontrado um grande número de regiões. (Tabela 4-21)

Tabela 4-21. Resultado da segmentação pelo SDH (janela = 16, delta = 0,45)

SDH - Janela 16 e delta = 0,45			
	4 Text.	5 Text.	5 Text. (Circ.)
Imagem 1	4	188	1
Imagem 2	4	6	4
Imagem 3	2	113	5
Imagem 4	131	2	2
Imagem 5	3	109	113
Imagem 6	2	97	112

Já o LBP não apresentou instabilidade como o SDH e resultou em uma classificação mais próxima da real, mesmo apresentando problemas em alguns casos, como por exemplo: Imagem 5 e 6 da classe de 4 texturas, e a Imagem 6 da classe com uma textura circular. Os resultados para o LBP podem ser vistos na Tabela 4-22

Tabela 4-22. Resultado da segmentação pelo LBP (janela = 32 delta = 0,65)

LBP - Janela 32 e delta = 0,65			
	4 Text.	5 Text.	5 Text. (Circ.)
Imagem 1	5	6	6
Imagem 2	5	5	8
Imagem 3	3	2	10
Imagem 4	3	9	3
Imagem 5	1	8	3
Imagem 6	9	8	1

A Tabela 4-23 comprova que essa estabilidade do LBP se repetiu para janelas menores (16x16 *pixels* neste caso).

Tabela 4-23. Resultado da segmentação pelo LBP (janela = 16, delta = 0,42)

LBP - Janela 16 e delta = 0,42			
	4 Text.	5 Text.	5 Text. (Circ.)
Imagem 1	5	1	1
Imagem 2	8	8	4
Imagem 3	3	1	8
Imagem 4	4	3	3
Imagem 5	4	8	7
Imagem 6	1	6	4

4.5 Segmentação de Imagens utilizando Decomposição *QuadTree*

Foram realizados experimentos com Decomposição *Quadtree* [22]. A idéia do método é a seguinte:

1. A imagem de entrada é subdividida em quatro blocos de tamanho igual. Os blocos devem ser potências de dois, por isso, a imagem deve ser quadrada e de dimensões proporcionais ao tamanho das janelas (essa é uma limitação do método).
2. Em seguida, os *pixels* de cada bloco são testados em relação à algum critério de homogeneidade, por exemplo, se seus contrastes estão dentro de um intervalo estabelecido.
3. Se o critério não for atendido, o bloco é subdividido em quatro e testado novamente.
4. Este processo se repete até que todos os blocos atendam ao critério ou blocos de tamanho 1x1 *pixel* sejam encontrados. A imagem resultante pode conter blocos de tamanhos diferentes.

No exemplo do funcionamento da Decomposição *Quadtree* que está na Figura 4-12, onde foram utilizado os parâmetros padrões, o critério é: se a diferença entre a cor máxima de um bloco e a cor mínima for maior do que o limiar (0,5 neste exemplo), então ele deve ser dividido.

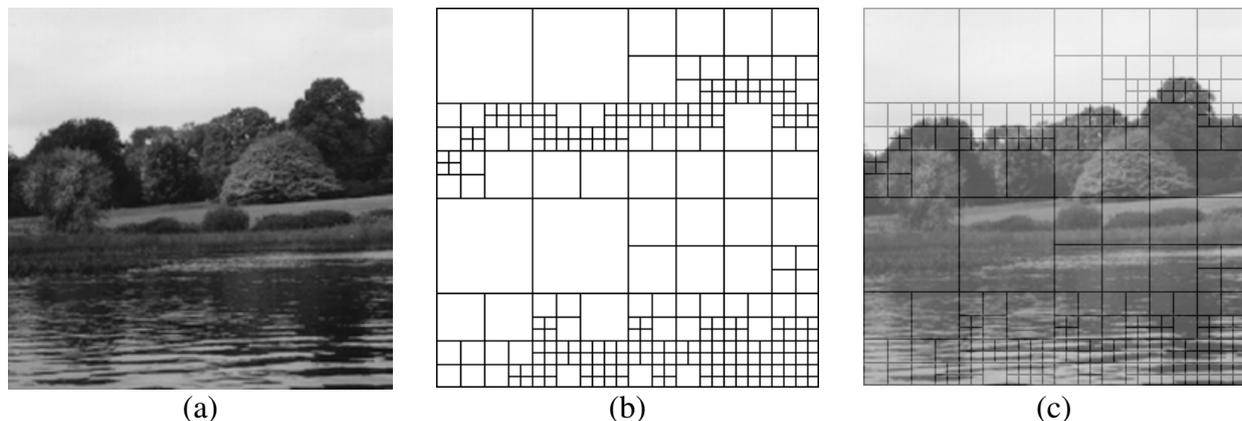


Figura 4-12. Exemplo do funcionamento do algoritmo de Decomposição *Quadtree*: a) Imagem a ser segmentada; b) o mapa gerado pelo *Quadtree* com limiar = 0,5 e c) sobreposição da imagem com o mapa para observarmos melhor o resultado da segmentação.

O algoritmo original do MatLab foi modificado para reduzir os blocos até o máximo de 8x8 *pixels* já que tamanhos menores que esse não fornecem informação suficiente para análise de textura. Além disso, os experimentos foram executados utilizando os parâmetros iniciais da decomposição *Quadtree* e os algoritmos descritores para comparar os resultados. Na Figura 4-13 podem ser vistas a imagem original usada pra os testes e o mapa de regiões gerados pelo *Quadtree* com ponto de corte definido como 0,9.

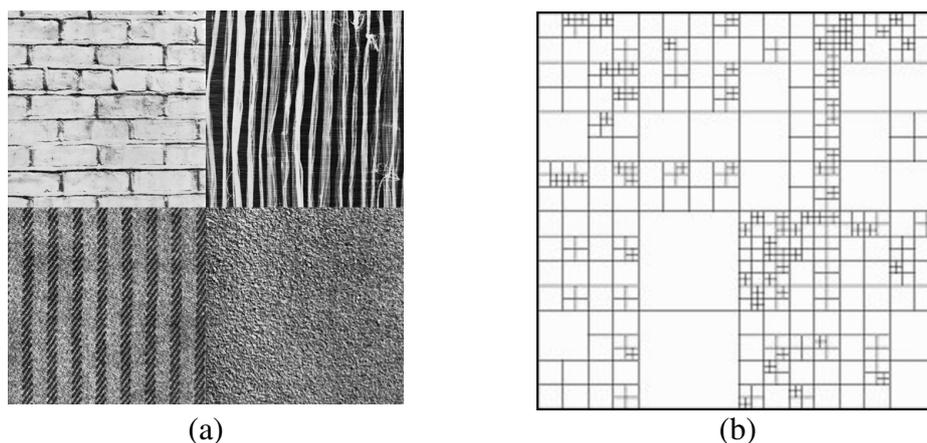


Figura 4-13. Em a) Imagem original usada para o teste e em b) mapa gerado pelo *Quadtree Decomposition* original (Ponto de corte = 0,9; 580 regiões).

A Figura 4-14 mostra os resultados através da comparação de cada região gerada pela decomposição usando o índice Q aplicado sobre os descritores de textura (GLCM, LBP e SDH). Na mesma figura apresentamos também o número de regiões encontradas por cada um.

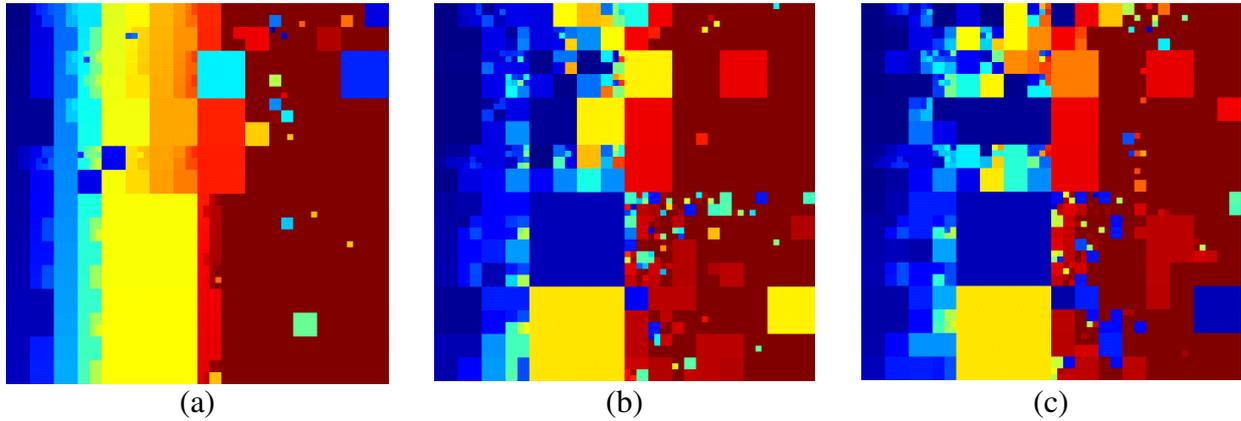


Figura 4-14. Resultado da segmentação utilizando decomposição *Quadtree* e o índice Q para comparar as janelas usando descritores de textura: a) GLCM (255 regiões); b) LBP (142 regiões) e c) SDH (152 regiões).

Capítulo 5

Conclusões

Neste capítulo é feita a conclusão do trabalho, levando em conta os resultados obtidos nos experimentos e os resultados esperados. Além disso, serão propostas algumas possibilidades de trabalhos futuros.

5.1 Conclusão

Apesar de inúmeros avanços na área de visão computacional, ainda não é possível simular todos os estágios da visão humana. Neste âmbito, seja num *hardware* para identificação de falhas em produtos numa indústria ou num software criado para a recuperação de imagens baseada em texturas, a análise e a segmentação de texturas é uma peça essencial na busca por uma visão computacional que consiga separar regiões com a eficiência e precisão do sistema visual humano.

Neste sentido, este trabalho propôs um conceito novo e pouco encontrado na literatura: o uso de índices de fidelidade de imagens como apoio à comparação de janelas de textura numa mesma imagem. Nos nossos experimentos, obtivemos bons resultados utilizando o LBP para descrever janelas da imagem e compara-las através do índice Q.

Utilizando a decomposição *Quadtree* esse ganho foi ainda maior: todos os experimentos realizados utilizando o índice de fidelidade Q geraram menos regiões que o *Quadtree* original e aproximando a quantidade real de regiões da quantidade produzida pela decomposição.

Além disso, toda a implementação foi realizada em MatLab, o que proporciona um ambiente fácil para diferentes testes de índices de comparação entre imagens (não necessariamente texturas), facilitando o desenvolvimento e experimentação dos algoritmos usados e permitindo a inclusão de novos índices.

5.2 Trabalhos Futuros

Como trabalhos futuros, algumas modificações podem ser feitas para a realização de mais testes, visando avaliar de forma mais completa o comportamento dos índices de fidelidade de imagem. Algumas sugestões:

- Uso de outro métodos não-estatísticos (por exemplo, métodos geométricos, baseados em modelo e de processamento de sinal) para descrever texturas. Dependendo das restrições da aplicação (algumas foram descritas na no Capítulo 2), pode ser possível obter melhores resultados utilizando outros métodos. Se por exemplo, a complexidade computacional não for uma restrição, talvez métodos geométricos gerem melhores resultados.
- Avaliar outros algoritmos de segmentação de imagens, como o *Watershed*. Além disso, é possível realizar permutações para combinar descritores, algoritmos/técnicas para comparar janelas e algoritmos de segmentação. Por exemplo, poderíamos usar um método geométrico para descrever texturas, compará-las através de redes neurais e segmentar usando *Watershed*.
- Implementar uma função para comparação de janelas e passá-la como parâmetro para a Decomposição *Quadtree*, não sendo necessário re-implementar a Decomposição, como foi feito neste trabalho, para usar o índice Q. O fato do MatLab ter sido usado como ambiente para implementação e testes, facilita a criação de tais funções.

Bibliografia

- [1] **Haralick, R., Shanmugam, K., Dinstein, I.**, Textural Features for Image Classification. *IEEE Transactions on Systems, Man and Cybernetics*. 1973.
- [2] **Zhou, Dongxiao.** Texture Analysis and Synthesis using a Generic Markov-Gibbs Image Model. <http://www.cs.auckland.ac.nz/~georgy/research/texture/thesis-html/node7.html> (Visitado em: 10 de Agosto de 2007).
- [3] **Samal, A., Brandle, J., Zhang, D.** Texture as the basis for individual tree identification. *Information Sciences*. 2006, Vol. 176, 5, pp. 565-576.
- [4] **Mello, C., Mascaro, A.** Image Fidelity Index Applied to Digital Mamography Texture Segmentation. Conferência Latino Americana de Informática, Chile, 2006.
- [5] **Lovergine, F.P., et al.** Leather inspection by oriented texture analysis with a morphological approach. *1997 International Conference on Image Processing*. 1997, Vol. 2, p. 669.
- [6] **Sklansky, J.** Image segmentation and feature extraction. *IEEE Transactions on Systems, Man, and Cybernetics*. 1978, Vol. 8, pp. 237-247.
- [7] **Chaudhuri, B., Sarkar, N. e Kundu, P.** Improved fractal geometry based texture segmentation technique. The Institution of Electrical Engineers. *IEE Proceedings Part E Computers and Digital Techniques*. 1993, Vol. 140, 5, pp. 233-241.
- [8] **Loh, Angeline M.** The Recovery of 3-D Structure Using Visual Texture Patterns. 2006.
- [9] **Long, Fuhui, Zhang, Hongjiang e Feng, David D.** Microsoft Research. Microsoft, 2003. http://research.microsoft.com/asia/dload_files/group/mcomputing/2003P/ch01_Long_v40-proof.pdf?0sr=a. (Visitado em: 3 de Novembro de 2007)
- [10] **Tuceryan, M. e Jain, A. K.** Texture analysis. In Chen, C. H., Pau, L. F. e Wang, P. S. P., editors, *Handbook of Pattern Recognition and Computer Vision*, pages 235-276. World Scientific Publishing, 1993.
- [11] **Keller, J. M.; Chen, S.; Crownover, R. M.;** Texture description and segmentation through fractal geometry. *Computer vision, graphic, and image processing*, vol. 45, p. 150, Fev. 1989.
- [12] **Pietikainen, M.; Rosenfeld, A.;** Gray level pyramid linking as an aid in texture analysis. *IEEE Transactions on Systems, Man, and Cybernetics*. Vol. SMC-12, no. 3, pp. 422-428. 1982
- [13] **Reed, T.; Wechsler, H.** Spatial/spatial-frequency representations for image segmentation and grouping. *Image and Vision Computing*. 1991, Vol. 9, no. 3, pp. 175-193.
- [14] **Weszka, J. S.; Dyer, C. R.; Rosenfeld, A.;** A comparative study of texture measures for terrain classification (photointerpretation based on Fourier power spectra and gray level statistics). *IEEE Transactions on Systems, Man, and Cybernetics*. Vol. SMC-6, pp. 269-285. Abr. 1976
- [15] **Ojala, T., Pietikäinen, M. e Harwood, D.** A comparative study of texture measures with classification based on feature distribution. *Pattern Recognition* 1996, Vol 29, no. 1, pp. 51-59.
- [16] **Unser, M.** Sum and difference histograms for texture classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 1986, Vols. PAMI-8, 1, pp. 118-125.

- [17] **Ruzon, M.** Texture Segmentation: An Introductory Primer. http://robotics.stanford.edu/~ruzon/tex_seg/node1.html. (Visitado em: 3 de Novembro de 2007)
- [18] **Wang, Zhou, et al.** Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Transactions on Image Processing*. Abril de 2004, Vol. 13, 4, pp. 600-612.
- [19] **Veldhuizen, T.** Grid Filters for Local Nonlinear Image Restoration <http://ubiety.uwaterloo.ca/~tveldhui/papers/MAScThesis/node18.html>. (Visitado em: 6 de Novembro de 2007)
- [20] **Wang, Zhou e C. Bovik, Alan. A.** Universal Image Quality Index. *IEEE Signal Processing Letters*. Março de 2002, Vol. 9, 3, pp. 81-84.
- [21] **Brodatz, P.** Textures: A photographic album for artists and designers. New York : Dover Publications, 1966.
- [22] **Samet, H.;** Connected component labeling using quadrees. *Journal of the Association for Computing Machinery*, 1981, no. 28, vol.3, pp. 487-501.
- [23] **Math Works:** www.mathworks.com (Visitado em: 27 de Novembro de 2007).