Hae Yong Kim

Projeto de Operadores pela Aprendizagem, Difusão Anisotrópica e Marca d'Água de Autenticação

Tese apresentada à Escola Politécnica da Universidade de São Paulo para obtenção do título de Professor Livre Docente, junto ao Departamento de Engenharia de Sistemas Eletrônicos.

São Paulo
2004

Hae Yong Kim

Projeto de Operadores pela Aprendizagem, Difusão Anisotrópica e Marca d'Água de Autenticação

Tese apresentada à Escola Politécnica da Universidade de São Paulo para obtenção do título de Professor Livre Docente, junto ao Departamento de Engenharia de Sistemas Eletrônicos.

Especialidade: Processamento e Análise de Imagens

São Paulo
2004

FICHA CATALOGRÁFICA

Kim, Hae Yong
Projeto de operadores pela aprendizagem, difusão anisotrópica
e marca d'água de autenticação / Hae Yong Kim – São Paulo, 2004.
202 p.

Tese (Livre-Docência) – Escola Politécnica da Universidade
de São Paulo. Departamento de Engenharia de Sistemas Eletrônicos.

1. Processamento de imagens 2. Aprendizado computacional
3. Filtros elétricos digitais 4. Criptologia I. Universidade de São Paulo.
Escola Politécnica. Departamento de Engenharia de Sistemas Eletrônicos
II.t.

À minha querida esposa Claudia,

com amor e gratidão.

Agradecimentos

Gostaria de manifestar os meus agradecimentos a todos os amigos e amigas que tornaram possível a realização desta tese. Em especial, agradeço:

- A todos os meus orientandos e ex-orientandos, em especial àqueles que contribuíram nas pesquisas relatadas nesta tese: Paulo Barreto, Harold Bustos, Marco Antonio de Melo e Amir Afif.
- A todos os professores que colaboraram nas pesquisas relatadas nesta tese, especialmente ao prof. Zang Hee Cho, que me convidou como pesquisador visitante da University of California at Irvine, e ao prof. Ricardo de Queiroz que colaborou nas pesquisas de marca d'água para imagens binárias.
- Aos amigos(as) e colegas do Laboratório de Processamento de Sinais, professores(as) Denise Consoni, Flávio Cipparrone, Miguel Ramirez, Vitor Nascimento e Wagner Zucchi.
- Ao prof. Francisco Javier que me estimulou a fazer a livre docência.
- À FAPESP e ao CNPq, pelos auxílios financeiros que possibilitaram as nossas pesquisas.
- Aos meus pais e à minha irmã.

Resumo

Esta tese descreve as principais contribuições científicas do meu grupo de pesquisa após o meu doutoramento. Estas contribuições estão agrupadas em três capítulos:

1. Projeto de operadores pela aprendizagem: Tradicionalmente, um operador restrito à janela (W-operador), que desempenha uma determinada função no Processamento e Análise de Imagens, é projetada manualmente e esta tarefa pode ser tediosa. Um W-operador pode ser projetado automaticamente a partir das imagens amostras de entrada-saída por um processo de aprendizagem de máquina. Nesta tese, descrevemos o projeto automático de W-operadores "binária para binária" e "binária para níveis de cinza". Primeiro, descrevemos a aprendizagem provavelmente aproximadamente correta e a estimação estatística que constituem o embasamento teórico do projeto automático de W-operadores. Depois, analisamos os diferentes algoritmos de aprendizagem e propomos as adaptações neles para aumentar os seus desempenhos ao resolver os problemas tratados. Por fim, utilizamos as teorias e os algoritmos desenvolvidos para aumentar a resolução espacial das imagens binárias e meio-tom, e para efetuar o meio-tom inverso.

2. Difusão anisotrópica: A difusão anisotrópica é freqüentemente utilizada na segmentação de imagens, atenuação de ruídos e detecção de arestas. Esta tese descreve o uso da difusão anisotrópica em várias aplicações de Processamento e Análise de Imagens. Primeiro, descrevemos a teoria do espaço de escala linear (de onde se originou a difusão anisotrópica). Depois, descrevemos a difusão anisotrópica, incluindo uma versão baseada na estatística robusta. Mostramos, através de algumas aplica-

ções, que a difusão anisotrópica robusta é superior ao tradicional em termos da qualidade da imagem filtrada. Descrevemos o melhoramento do algoritmo de reconstrução tomográfica máxima entropia usando a difusão anisotrópica robusta. Por fim, descrevemos o aperfeiçoamento do modelo linear geral (um processo para detectar as áreas ativadas do cérebro em imagens de ressonância magnética funcional) usando a difusão anisotrópica robusta.

3. Marcas d'água de autenticação: Uma marca d'água é um sinal portador de informação embutido numa imagem digital que pode ser extraída mais tarde para fazer alguma asserção sobre a imagem hospedeira. As marcas d'água digitais são normalmente classificadas em robustas e frágeis. Esta tese trata somente das marcas d'água frágeis, também chamadas de autenticação. Primeiro, descrevemos a assinatura digital, um conceito amplamente utilizado nas marcas de autenticação de chave pública. Em segundo lugar, descrevemos as principais marcas de autenticação para as imagens estáticas de tonalidade contínua: Yeung-Mintzer e Wong. Descrevemos os principais ataques contra estas marcas e os meios para se defender contra eles. Em terceiro lugar, descrevemos as marcas d'água de autenticação para as imagens binárias e meio-tom.

Abstract

This thesis describes the main scientific contributions of my research group after my doctorate. These contributions are grouped in three chapters:

1. Operator design by machine learning: Traditionally, a windowed operator (W-operator) that plays a certain role in the Image Processing and Analysis is designed manually, and this task can be tedious. A W-operator can be designed automatically from sample in-out images by a machine learning process. In this thesis, we describe the automatic design of binary-to-binary and binary-to-grayscale W-operators. First, we describe the probably approximately correct learning and the statistical estimation that constitute the theoretic framework of the automatic W-operator design. Afterwards, we analyze the various learning algorithms and propose their adaptations to increase their performances in solving the applications addressed. We use the developed theories and algorithms to increase the spatial resolution of binary and halftone images, and to perform the inverse halftoning.

2. Anisotropic diffusion: Anisotropic diffusion is used frequently in image segmentation, noise attenuation and edge detection. This thesis describes the use of the anisotropic diffusion in various Image Processing and Analysis applications. First, we describe the linear scale space theory (from where the anisotropic diffusion has been derived). Afterwards, we expound the anisotropic diffusion theory, including a version based on the robust statistics. We show, through some applications, that the robust anisotropic diffusion is superior to the traditional in terms of the quality of the filtered image. We describe the improvement of the maximum entropy tomography

algorithm using the robust anisotropic diffusion. Finally, we describe the enhancement of the general linear model (a process to detect activated regions of brain in the functional magnetic resonance images) using the robust anisotropic diffusion.

3. Authentication watermarkings: A watermarking is an information-bearing signal embedded in a digital image that can be extracted later to make some assertion on the host image. Digital watermarks are usually classified in robust and fragile. This thesis is concerned only with the fragile watermarks, also called authentication watermarks. First, we explain the digital signature, a concept widely used for the public-key authentication watermarking. Second, we describe the main authentication watermarkings for static continuous-tone images: Yeung-Mintzer and Wong. We describe the principal attacks against these watermarks and the means to defend against them. Third, we describe the authentication watermarking techniques for binary and halftone images.

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Abreviaturas

Alice	Nome fictício que da pessoa que assina um documento digital usando a sua chave privada.
AWSF	Authentication watermarking by shuffling and flipping (marca d'água de autenticação pelo embaralhamento e reviramento).
AWST	Authentication watermarking by self toggling (marca d'água de autenticação pelo auto-reviramento).
A^{x}	Imagem amostra de entrada.
A^{y}	Imagem amostra de saída.
Bob	Nome fictício da pessoa que verifica a assinatura de um documento digital usando a chave pública.
BOLD	Blood oxygen level dependent (dependente do nível de oxigenação do sangue).
Contone	Continuous tone (tonalidade contínua).
DHPT	Data Hiding by Pair-Toggling (embutimento de dados por reviramento aos pares).
DHSPT	Data Hiding by Smart Pair Toggling (embutimento de dados por reviramento inteligente aos pares).
DHST	Data hiding by self toggling (embutimento de dados por auto-reviramento).
DS	Digital signature (assinatura digital).
DSA	Digital signature algorithm (algoritmo de assinatura digital).
DT	Decision tree (árvore de decisão).

EPM	Estimated parameters map (mapa dos parâmetros estimados).
e_error	Empirical error (erro empírico ou observado, escrito como “e-erro” no texto em português).
e-ótimo	Empiricamente ótimo, isto é, o melhor sobre os dados observados.
ek-NN	Aprendizagem *k*-NN empiricamente ótima.
fMRI	Functional magnetic resonance imaging (imageamento por ressonância magnética funcional).
Furo	Abreviação de “furo de espiar” (peephole).
HBC	Hash block chaining (encadeamento dos blocos de hash).
HSI	Hue, saturation, intensity (tonalidade, saturação, intensidade).
IH	Inverse halftoning (meio-tom inverso).
kd-árvore	Árvore *k* dimensional (kd-tree).
k-NN	*k* nearest neighbors (*k* vizinhos mais próximos).
LSB	Least significant bit (bit menos significativo).
LUT	Look-up-table (tabela de busca).
MAC	Message authentication code (código de autenticação de mensagem).
Mallory	Nome fictício de um “hacker” malicioso.
Marca	Abreviação de “marca d’água” (watermark).
MENT	Maximum entropy (máxima entropia).
mod	Módulo, resto de divisão inteira.
NN	Nearest neighbor (vizinho mais próximo).
OCR	Optical character recognition (reconhecimento de caracteres ópticos).
OD	Ordered dithering (excitação ordenada).
PAC	Probably approximately correct (provavelmente aproximadamente correto).
PET	Positron emission tomography (tomografia de emissão de pósitrons).
PSNR	Peak signal-to-noise ratio (razão entre o pico do sinal e o erro).
$Q^{\mathbf{x}}$	Imagem a-ser-processada.
$Q^{\mathbf{y}}$	Imagem de saída ideal.
$\hat{Q}^{\mathbf{y}}$	Imagem processada.
RAD	Robust anisotropic diffusion (difusão anisotrópica robusta).

RGB	Red, green, blue (vermelho, verde, azul).
RSA	Esquema de criptografia de chave pública de Rivest, Shamir e Adleman.
RMS	Root mean square (raiz da média quadrática).
SPM	Statistical parametric map (mapa estatístico paramétrico).
SPM{t}	SPM das estatísticas *t* de Student.
t_error	True error (erro real ou verdadeiro, escrito como “t-erro” no texto em português).
t-ótimo	Verdadeiramente ótimo (isto é, o melhor sobre a distribuição de probabilidade).
VIS	Visual impact score (nota de impacto visual).
W-operador	“Window operator” ou “windowed operator” (operador restrito à janela).
WZ-operador	“Windowed zoom operator” (operador restrito à janela para ampliação).
WZDT	Windowed zoom decision tree (árvore de decisão restrita à janela para ampliação).

Capítulo 1:
Introdução

Esta tese de livre docência apresenta sistematicamente os principais resultados das nossas pesquisas científicas em Processamento e Análise de Imagens após o meu doutoramento. Digo “nossas pesquisas” em vez de “minhas pesquisas”, pois o trabalho foi realizado em cooperação com os meus orientandos, contando com a colaboração de pesquisadores de diversas universidades e institutos de pesquisa. Ao longo do texto, deixo explícito quem foi o principal responsável por cada uma das pesquisas relatadas.

Essas pesquisas são classificadas em três áreas principais:

1. Projeto automático de operadores restritos à janela pela aprendizagem de máquina e as suas aplicações em diferentes problemas do Processamento e Análise de Imagens.
2. A difusão anisotrópica e o espaço de escala aplicados em diferentes problemas, especialmente para melhorar a reconstrução tomográfica e a detecção das áreas ativadas do cérebro em imagens de ressonância magnética funcional.
3. Marca d’água para autenticação de imagens em tonalidade contínua, binárias e meio-tom.

Assim, a presente tese está composta por três “subteses”, mais ou menos independentes, cada uma ocupando um capítulo. As nossas pesquisas fora destas três áreas não

estão aqui documentadas. As principais pesquisas não documentadas são os operadores baseados em lógica nebulosa [Ri02; Cn08] e o reconhecimento de formas [Ci10; Su05]. Mesmo dentro das três áreas, somente as principais pesquisas foram registradas neste documento.

Esta tese não possui um capítulo “conclusão”, pois as conclusões estão apresentadas no final de cada capítulo. Além disso, o presente capítulo “introdução” não faz uma introdução científica, pois mais uma vez cada capítulo possui uma introdução científica própria.

Esta tese está subdividida em capítulos (exemplo: capítulo 2), seções (exemplo: seção 2.1), subseções (exemplo: subseção 4.3.1) e subsubseções (sem numeração).

Escrevemos esta tese traduzindo, concatenando e adaptando trechos dos nossos artigos e textos didáticos. Assim, a maioria do material apresentada nesta tese já foi publicada em algum outro lugar.

Alguns dos programas utilizados nesta tese estão disponíveis em:

http://www.lps.usp.br/~hae/software.

Capítulo 2:
Projeto de Operadores pela Aprendizagem

Resumo e nossas contribuições

Um operador restrito à janela (W-operador) é uma transformação de imagem onde a cor de um pixel da imagem de saída é escolhida em função das cores da vizinhança desse pixel na imagem de entrada. Os W-operadores desempenham funções essenciais em diversas áreas do Processamento e Análise de Imagens. A maioria dos filtros utilizados no Processamento e Análise de Imagens são W-operadores (por exemplo, a convolução espacial, o filtro mediano, e os operadores morfológicos). A escolha de um W-operador adequado para uma dada aplicação normalmente é feita manualmente, o que é uma tarefa trabalhosa e tediosa. Temos pesquisado o uso da aprendizagem de máquina para automatizar esta tarefa, isto é, projetar um W-operador automaticamente a partir das imagens amostras entrada-saída. Este capítulo descreve as teorias que embasam o projeto automático de W-operadores (a aprendizagem provavelmente aproximadamente correta e a estimação estatística) e as nossas contribuições científicas nesta área. Nesta tese, abordamos somente o projeto de W-operadores de uma imagem binária para outra binária, e de uma imagem binária para outra em níveis de cinza, pois são as áreas onde encontramos as aplicações mais interessantes.

Formalizamos o problema de aprendizagem de W-operadores binários usando a teoria de aprendizagem PAC (provavelmente aproximadamente correta). Descrevemos como a estimação estatística pode ser utilizada para estimar os erros dos operadores projetado e ótimo. Também utilizamos a estimação estatística para comparar os dife-

rentes métodos de aprendizagem de máquina quanto a acurácia esperada do operador projetado e para escolher uma janela conveniente. Depois, aplicamos as teorias PAC e estimação estatística no problema de aumento da resolução espacial de imagens binárias e meio-tom. Por fim, aplicamos a aprendizagem no problema de meio-tom inverso.

Diferentemente das outras áreas descritas nesta tese, tenho realizado praticamente sozinho as pesquisas nesta área, com pouca ajuda dos meus orientandos e de outros pesquisadores. As nossas principais contribuições científicas na área de projeto automático de W-operadores pela aprendizagem de máquina são:

1) *Aumento de resolução de imagens binárias:* Esta contribuição científica foi publicada em [Ri03; Ci02; Cn10]. Nesta tese, ela está documentada na seção 2.3. Eu fui o principal responsável por esta contribuição, contando com alguma colaboração do meu ex-orientando de doutorado Paulo S. L. M. Barreto.
   Resumo: Num ambiente de escritório típico, equipamentos e softwares heterogêneos, cada um trabalhando numa resolução espacial diferente, devem interagir juntos. Assim, freqüentemente aparece o problema de conversão de resolução. Esta contribuição trata do problema de aumento de resolução espacial (ou ampliação) de documentos e imagens binárias (por exemplo, a conversão de uma imagem 300 dpi em 600 dpi). Uma solução nova, acurada e eficiente para este problema é proposta. Ela utiliza a aprendizagem *k*-NN (*k* vizinhos mais próximos) para projetar automaticamente os operadores de ampliação restritos à janela a partir dos pares de imagens entrada-saída de treinamento. O operador resultante é armazenado numa look-up-table, que é extremamente rápida computacionalmente. É útil conhecer, *a priori*, a complexidade de amostra (a quantidade de amostras de treinamento necessária para obter, com probabilidade 1-δ, um operador com a acurácia ε). Utilizamos a teoria de aprendizagem PAC (provavelmente aproximadamente correta) para calculá-la, nos casos sem ruído e ruidoso. Como a teoria PAC geralmente superestima a complexidade de amostra, a estimação estatística é utilizada para

estimar, *a posteriori*, um intervalo estreito para o erro. A estimação estatística também é usada para mostrar que a aprendizagem *k*-NN possui um bom viés indutivo que permite reduzir o tamanho necessário das imagens amostras.

2) *Aumento de resolução de imagens meio-tom:* Esta contribuição científica foi publicada em [Ri05; Ci05]. Nesta tese, ela está documentada na seção 2.4. Eu fui o principal responsável por esta contribuição.

   Resumo: Esta contribuição trata-se de uma técnica nova, acurada e eficiente para aumentar a resolução espacial de imagens meio-tom. Essa técnica faz uso de um processo de aprendizagem de máquina para projetar automaticamente um operador de ampliação a partir das imagens amostras de entrada-saída. Para ampliar com acurácia uma imagem meio-tom, uma ampla janela e grandes imagens amostras devem ser usadas. Infelizmente, neste caso, o tempo de execução da maioria das técnicas anteriores torna-se proibitivo. A nova solução supera esta dificuldade utilizando a aprendizagem pela árvore de decisão (decision tree, abreviado como DT). A aprendizagem DT original é alterada para obter uma técnica mais eficiente denominada aprendizagem WZDT. É útil conhecer, *a priori*, a complexidade de amostra (o número de amostras de treinamento necessário para obter, com probabilidade 1-δ, um operador com acurácia ε): usamos a aprendizagem provavelmente aproximadamente correta (PAC) para calculá-la. Como a teoria PAC normalmente superestima a complexidade de amostra, a estimação estatística é usada para avaliar, *a posteriori*, um intervalo estreito para o erro. A estimação estatística é também usada para escolher uma janela apropriada e para mostrar que a aprendizagem DT tem um bom viés indutivo. A nova técnica é mais acurada que a ampliação baseada em técnicas de meio-tom inverso simples. A qualidade da solução proposta está muito próxima da qualidade ótima possível de ser obtida, para um processo de ampliação baseada em vizinhança e usando a distância de Hamming para quantificar o erro.

3) *Meio-tom inverso pela aprendizagem.* Esta contribuição científica foi publicada em [Ci11]. Nesta tese, ela está documentada na seção 2.5. Eu fui o principal responsável por esta contribuição, com a colaboração do prof. Ricardo de Queiroz da UnB.

   Resumo: O meio-tom inverso (inverse halftoning, abreviado como IH) é o processo usado para obter uma imagem em níveis de cinza a partir da imagem meio-tom correspondente. Recentemente, as técnicas de IH baseadas na aprendizagem de máquina foram propostas. A aprendizagem por árvore de decisão tem sido aplicada com sucesso em várias tarefas de aprendizagem de máquina durante bastante tempo. Nesta pesquisa, propomos usar a árvore de decisão para resolver o problema de IH. Isto permite-nos reusar alguns algoritmos já desenvolvidos e testados. Especialmente, a maximização do ganho de entropia é uma idéia poderosa que faz com que o algoritmo de aprendizagem selecione automaticamente a janela ideal à medida que a árvore de decisão é construída. A nova técnica gerou imagens em níveis de cinza com PSNR vários dB acima daqueles previamente reportados na literatura. Além disso, ela possui uma implementação muito rápida, possibilitando usá-la em aplicações de tempo real.

4) Temos também utilizado o projeto automático de W-operadores por aprendizagem de máquina em outras aplicações, tais como emulação de operadores em níveis de cinza ou coloridos [Ri01; T02], reconhecimento de texturas [Ci01] e reconhecimento de caracteres (OCR) sem segmentação [Cn06], mas essas aplicações não estão documentadas nesta tese.

2.1 Introdução

Em Processamento e Análise de Imagens, os operadores restritos à janela (W-operadores) desempenham um papel fundamental. Um W-operador é uma transformação de imagem onde a cor de um pixel da imagem de saída é decidida em função

das cores do pixel na imagem de entrada correspondente e seus vizinhos (veja a figura 2.1).

Muitos operadores clássicos de diferentes ramos do Processamento e Análise de Imagens são W-operadores (convolução espacial, mediana, filtro de pilha, erosão, dilatação, abertura, fechamento, hit-miss, etc.). As transformações mais complexas de imagens (emagrecimento, esqueletonização, reconstrução, divisor d'água, etc.) costumam utilizar os W-operadores como seus blocos construtores.

Um W-operador, que desempenha um determinado papel numa aplicação de Processamento e Análise de Imagens, é tradicionalmente projetado manualmente, e esta tarefa é muitas vezes laboriosa e tediosa. Muitas técnicas diferentes têm sido propostas para facilitá-la. Temos trabalhado com a aprendizagem de máquina no projeto automático de W-operadores a partir das imagens exemplos.

Nesta abordagem, um W-operador Ψ é projetado automaticamente a partir da distribuição da probabilidade P responsável pela geração das imagens de entrada $Q^{\mathbf{x}}$ e de saída $Q^{\mathbf{y}}$. Por exemplo, suponha que $Q^{\mathbf{x}}$ seja uma imagem ruidosa e $Q^{\mathbf{y}}$ a imagem limpa correspondente. Supondo totalmente conhecido o processo estatístico P de corrupção da imagem $Q^{\mathbf{y}}$, é possível construir o operador Ψ de forma que a imagem processada $\hat{Q}^{\mathbf{y}} = \Psi(Q^{\mathbf{x}})$ seja "semelhante" à imagem ideal $Q^{\mathbf{y}}$. Isto é, Ψ é projetado para minimizar a esperança da diferença entre $\hat{Q}^{\mathbf{y}}$ e $Q^{\mathbf{y}}$. Por exemplo, os livros clássicos de processamento de imagens como [Gonzalez and Woods, 1992] trazem as técnicas lineares para a restauração de imagens, baseadas na transformada de Fourier bidimensional. Os trabalhos [Coyle and Lin, 1988] e [Lee et al., 1997] projetam o "filtro de pilha" que minimiza o erro médio absoluto e os trabalhos [Dougherty, 1992a] e [Dougherty, 1992b] projetam o operador morfológico que minimiza o erro médio quadrático.

Na prática, a distribuição P é normalmente desconhecida. Assim, uma abordagem mais pragmática emprega as imagens de treinamento $A^{\mathbf{x}}$ (entrada) e $A^{\mathbf{y}}$ (saída), que

são as realizações da distribuição P, ao invés da própria distribuição P, para projetar o operador Ψ automaticamente por um processo de aprendizagem de máquina.

Muitas abordagens diferentes de aprendizagem de máquina podem ser utilizadas para projetar W-operadores: algoritmos genéticos, redes neurais, aprendizagem bayesiana, etc. Mas, para o problema presente, o desempenho computacional é a pedra de toque que distingue os métodos úteis daqueles que são impraticáveis, pois as imagens e as janelas envolvidas são normalmente muito grandes, e assim uma técnica inadequada poderia levar meses ou anos para processar uma única imagem. Ousaríamos dizer que provavelmente a aprendizagem de máquina ainda não é mais amplamente utilizada para projetar W-operadores devido ao fraco desempenho computacional dos algoritmos de aprendizagem, quando estes são escolhidos sem uma preocupação criteriosa pelo seu desempenho.

O desempenho de um algoritmo de aprendizagem deve ser medida analisando três parâmetros: tempo para aprender um W-operador (tempo de treinamento), tempo para aplicar um W-operador previamente construído a uma imagem (tempo de aplicação) e a quantidade de memória de computador necessária (espaço necessário). Como uma propriedade essencial, o tempo de aplicação deve ser curto, pois de outro modo o método nunca poderá ser utilizado em qualquer aplicação prática, notavelmente nas aplicações de tempo real. Embora não seja tão essencial, é muito conveniente que o tempo de treinamento também seja curto, para não aborrecer o usuário. Finalmente, o requerimento do espaço usualmente não é muito preocupante, desde que o W-operador caiba dentro da memória de um computador comum.

Para atingir o desempenho computacional necessário, temos utilizado a aprendizagem k vizinhos mais próximos (k-NN) [Cover and Hart, 1967; Mitchell, 1997] e a aprendizagem por árvore de decisão (DT) [Mitchell, 1997]. Conforme descrevemos mais abaixo, estes dois métodos podem se tornar extremamente rápidos se as estruturas de dados convenientes forem utilizadas.

O viés indutivo (inductive bias) é um assunto bastante discutido na aprendizagem de máquina. O viés indutivo é o conjunto de suposições *a priori* pelas quais o aprendiz generaliza além dos dados observados para inferir a classificação de novas instâncias. Um algoritmo de aprendizagem que não fizesse suposições *a priori* no que diz respeito ao conceito alvo, não possuiria nenhuma base racional para classificar qualquer instância ainda não vista. Se um viés indutivo confiável for usado, a imagem processada será semelhante à imagem de saída ideal, mesmo usando somente uma pequena quantidade de amostras de treinamento. Ambas as técnicas k-NN e DT têm vieses indutivos sólidos. O viés indutivo da aprendizagem k-NN corresponde à suposição de que a classificação de uma instância será mais semelhante às classificações de outras instâncias que estão próximas em distância. Isto é especialmente verdadeiro para o problema que estamos tratando, pois é muito natural e intuitivo atribuir uma cor de saída semelhante aos padrões visualmente semelhantes. O viés indutivo da aprendizagem de árvore de decisão é conhecido como "a navalha de Occam" [Mitchell, 1997, cap. 3], que diz: "Prefira a hipótese mais simples que se ajusta aos dados". O algoritmo de construção de DT coloca os atributos de alto ganho de informação mais próximos da raiz. Esta prática corresponde a adotar o viés indutivo que prefere as árvores mais baixas às mais altas, isto é, a navalha de Occam.

Utilizando a teoria de aprendizagem computacional PAC [Mitchell, 1997; Anthony and Biggs, 1992; Haussler, 1992], é possível pré-calcular o tamanho necessário da amostra para que o W-operador aprendido atinja uma precisão ε com a probabilidade $1-\delta$, independentemente do método particular de aprendizagem de máquina adotado (basta que o método seja consistente, isto é, produza W-operador que concorde com os exemplos de treinamento). Porém, os resultados fornecidos por esta teoria costumam superestimar o tamanho da amostra, pois não considera o viés indutivo do particular método de aprendizagem. Este problema não pode ser contornado mesmo utilizando as teorias mais fortes, como a dimensão Vapnik-Chervonenkis [Vapnik, 1995; Mitchell, 1997]. Para superar esta dificuldade, além de utilizar a teoria PAC *a priori* (isto é, antes de realizar a aprendizagem), temos também utilizado os métodos

de estimação estatística *a posteriori*. A estimação estatística permite estimar precisamente a taxa de erro real do W-operador projetado.

O problema torna-se um pouco mais complexo quando há ruídos nas amostras de treinamento. Ou, equivalentemente, se o *professor* pode cometer alguns erros ao ensinar o *aprendiz*. Neste caso, o W-operador ótimo possui uma taxa de erro maior que zero. Esta taxa mínima de erro pode ser medida empiricamente e é possível construir um intervalo de confiança para essa medida. Além disso, utilizando a estimação estatística, dois métodos de aprendizagem diferentes podem ser comparados entre si quanto à acurácia, o que nos permite decidir, por exemplo, se o método k-NN é superior ou inferior à aprendizagem DT para uma determinada aplicação.

O algoritmo força-bruta para a aprendizagem k-NN é extremamente lento, pois para cada pixel da imagem a ser processada, deve-se fazer uma busca exaustiva na imagem de treinamento. Os trabalhos [Ci01] e [Ci02; Ri03] propõe duas soluções para o problema: o uso da kd-árvore (árvore binária multidimensional [Bentley, 1975; Friedman et al., 1977; Preparata and Shamos, 1985]) e look-up-table (LUT). A velocidade de treinamento da kd-árvore é bastante boa, porém a sua velocidade de aplicação somente é satisfatória para dimensões pequenas, piorando rapidamente com o aumento da dimensão. Por outro lado, a velocidade de aplicação da LUT é ótima em qualquer dimensão, porém a velocidade de treinamento e a memória necessária crescem exponencialmente com o aumento da dimensão.

A aprendizagem DT pode ser vista como uma kd-árvore sem o processo de backtracking. O uso da aprendizagem DT é especialmente recomendado para situações onde muitos atributos são irrelevantes para o conceito que está sendo aprendido. As experiências empíricas têm mostrado que o viés indutivo de DT é ligeiramente pior que k-NN para o problema de aprendizagem de W-operador. Porém, a árvore de decisão é rápida tanto no treinamento quanto na aplicação, propriedade que torna o seu uso na prática extremamente atraente.

O projeto de operadores pela aprendizagem computacional tem sido aplicado com sucesso em diferentes áreas, como na emulação de filtros desconhecidos [T02; Ri01; Ri03; Cn10], para atenuar ruídos [T02; Ri03], na segmentação da imagem de acordo com a textura [T02; Ci01], em OCR [Cn06], para aumentar a resolução de imagens binárias pela aprendizagem k-NN [Ci02; Ri03; Cn10], para aumentar a resolução de imagens meio-tom pela aprendizagem DT [Ci05; Ri05] e no problema de meio-tom inverso [Ci11]. Esta tese descreve detalhadamente somente as três últimas aplicações.

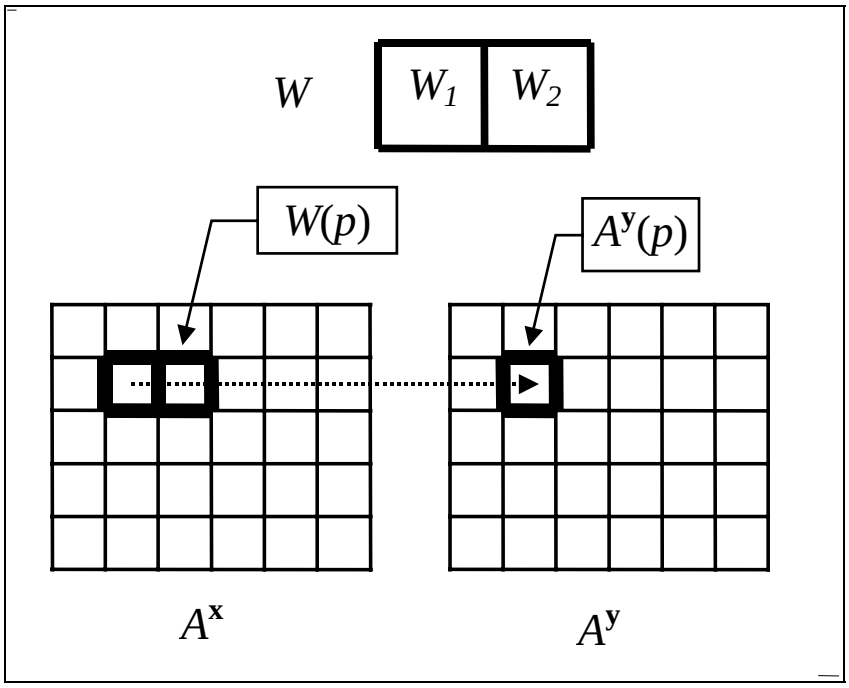


Fig. 2.1: Um W-operador decide a cor de um pixel p na imagem de saída $A^{\mathbf{y}}$ analisando uma vizinhança $W(p)$ do pixel p na imagem de entrada $A^{\mathbf{x}}$.

Organização deste capítulo

O restante deste capítulo está organizado como segue. A seção 2.2 apresenta as teorias que embasam o projeto automático de W-operadores por aprendizagem de máquina. O problema é formalizado como um processo de aprendizagem computacional PAC e analisamos os casos sem ruído e ruidoso. Depois, descrevemos a teoria da estimação estatística. Em seguida, expomos o algoritmo de aprendizagem *k*-NN e sugerimos uma pequena alteração nele para torná-lo empiricamente ótimo. Este algoritmo alterado será denotado como ek-NN. Explicamos também a aprendizagem por árvore de decisão (DT). Terminamos a seção explicando como a estimação estatística pode ser usada para comparar diferentes algoritmos de aprendizagem ou diferentes janelas. A seção 2.3 trata do problema de aumento de resolução de imagens binárias (documentos impressos ou manuscritos, ortográficos ou escaneados), usando a aprendizagem ek-NN. A seção 2.4 trata do problema de aumento de resolução de imagens meio-tom usando o algoritmo DT. Sugerimos uma alteração no algoritmo DT para torná-lo mais eficiente no problema de ampliação de imagens meio-tom. O algoritmo modificado é chamado de aprendizagem WZDT. A seção 2.5 trata do problema de meio-tom inverso usando a árvore de decisão. Finalmente, a seção 2.6 apresenta as nossas conclusões.

2.2 Aprendizagem de W-Operadores Binários

Introdução

Nesta seção, analisaremos o caso binário do problema do projeto automático de W-operadores pela aprendizagem de máquina. Faremos uso da teoria de aprendizagem PAC clássica [Anthony and Biggs, 1992] e generalizada [Haussler, 1992] para calcular a complexidade de amostra do problema de aprendizagem de operadores binários. Infelizmente, com freqüência, somente uma complexidade de amostra superestimada pode ser obtida utilizando esta teoria. Mesmo assim, ela será útil como um limite superior para a quantidade de amostras necessárias, e para mostrar a convergência do processo de aprendizagem. Além disso, a teoria de aprendizagem PAC irá nos permitir expressar rigorosamente o problema de aprendizagem do W-operador, e pode clarificar consideravelmente a compreensão do problema. Para superar o problema de superestimação da complexidade de amostra, além de utilizar a teoria PAC, temos também utilizado os métodos de estimação estatística. A estimação estatística permite estimar precisamente a taxa de erro real do W-operador projetado.

O problema

Vamos definir uma imagem binária como uma função $Q:\mathbb{Z}^2 \to \{0,1\}$. O suporte de uma imagem binária Q é um subconjunto finito de $\mathbb{Z}^2$ onde a imagem está de fato definida. O tamanho do suporte é o número de pixels da imagem e uma imagem é considerada estar preenchida com uma cor-de-fundo fora do seu suporte.

Um W-operador binário Ψ é uma função que mapeia uma imagem binária numa outra, definida através de um conjunto de w pontos chamado janela

$$W = \{W_1,\ldots,W_w\},\ W_i \in \mathbb{Z}^2$$

e um conceito ou uma função característica $\psi:\{0,1\}^w \to \{0,1\}$ como segue:

$$\Psi(Q)(p) = \psi(Q(W_1 + p),\ldots,Q(W_w + p)),$$

onde $p \in \mathbb{Z}^2$. Cada ponto W_i da janela é chamado *peephole* ou furo-de-espiar.

Sejam as imagens $A^{\mathbf{x}}$, $A^{\mathbf{y}}$, $Q^{\mathbf{x}}$ e $Q^{\mathbf{y}}$ respectivamente a imagem de entrada de treinamento, imagem de saída de treinamento, a imagem a ser processada e a imagem de saída ideal (supostamente desconhecida). Podemos supor que existe um único par de imagens de treinamento ($A^{\mathbf{x}}$ e $A^{\mathbf{y}}$), porque se existirem muitos pares, elas podem ser "coladas" para formarem um único par. A fim de projetar um W-operador $\hat{\Psi}$, o usuário deve escolher manualmente uma janela apropriada *W*.

Vamos denotar o conteúdo em $A^{\mathbf{x}}$, da janela *W* deslocada para $p \in \mathbb{Z}^2$, como $a_p^{\mathbf{x}}$ e denominá-lo uma instância de treinamento ou um padrão de entrada em torno do pixel *p*:

$$a_p^{\mathbf{x}} = [A^{\mathbf{x}}(W_1 + p), A^{\mathbf{x}}(W_2 + p),\ldots,A^{\mathbf{x}}(W_w + p)] \in \{0,1\}^w.$$

Cada padrão $a_p^{\mathbf{x}}$ está associado com uma cor de saída ou classificação $A^{\mathbf{y}}(p) \in \{0,1\}$. Vamos denotar os dados obtidos quando todos os pixels de $A^{\mathbf{x}}$ e $A^{\mathbf{y}}$ são varridos como uma seqüência

$$\vec{a} = \left((a_{p_1}^{\mathbf{x}}, A^{\mathbf{y}}(p_1)),\ldots,(a_{p_m}^{\mathbf{x}}, A^{\mathbf{y}}(p_m))\right)$$

e denominá-la seqüência de amostras (*m* é a quantidade dos pixels das imagens $A^{\mathbf{x}}$ e $A^{\mathbf{y}}$). Cada elemento $(a_{p_i}^{\mathbf{x}}, A^{\mathbf{y}}(p_i)) \in \vec{a}$ é chamado um exemplo ou uma amostra de treinamento. Vamos construir de forma semelhante a seqüência

$$\vec{q} = \left((q_{p_1}^{\mathbf{x}}, Q^{\mathbf{y}}(p_1)),\ldots,(q_{p_n}^{\mathbf{x}}, Q^{\mathbf{y}}(p_n))\right)$$

a partir de $Q^{\mathbf{x}}$ e $Q^{\mathbf{y}}$ (*n* é a quantidade de pixels de $Q^{\mathbf{x}}$ e $Q^{\mathbf{y}}$). Cada $q_{p_i}^{\mathbf{x}}$ é chamado um padrão de busca ou uma instância a-ser-processada, e a saída $Q^{\mathbf{y}}(p_i) \in \{0,1\}$ é chamada a cor de saída ideal ou a classificação ideal.

O aprendiz ou o algoritmo de aprendizagem **A** é requisitado para construir, baseado em $A^{\mathbf{x}}$ e $A^{\mathbf{y}}$, um W-operador $\hat{\Psi}$ tal que, quando $\hat{\Psi}$ é aplicado à $Q^{\mathbf{x}}$, espera-se que a

imagem resultante $\hat{Q}^{\mathbf{y}} = \hat{\Psi}(Q^{\mathbf{x}})$ seja semelhante à imagem de saída ideal $Q^{\mathbf{y}}$. Mais precisamente, o aprendiz **A** deve construir uma função característica ou hipótese $\hat{\psi}$ baseado em seqüência de amostras $\vec{a}$ de forma que, quando $\hat{\psi}$ é aplicado a um padrão a-ser-processado $q^{\mathbf{x}}_{p_i}$, espera-se que a sua classificação $\hat{Q}^{\mathbf{y}}(p_i) = \hat{\psi}(q^{\mathbf{x}}_{p_i})$ seja igual a $Q^{\mathbf{y}}(p_i)$ com alta probabilidade. A função $\hat{\psi}$ e a janela *W* juntas representam o W-operador $\hat{\Psi}$.

Caso sem ruído

Vamos estudar em primeiro lugar o caso sem ruído. Pois, embora a maioria dos problemas práticos seja ruidosa, o estudo do caso sem ruído irá nos ajudar a compreender melhor os casos ruidosos.

Num ambiente sem ruído, existe um conceito alvo claramente definido $\psi : \{0,1\}^W \to \{0,1\}$ que o aprendiz deve aprender. Em tal ambiente, podemos supor que as instâncias de treinamento $a^{\mathbf{x}}_{p_i}$ são geradas aleatória e independentemente no espaço $\{0,1\}^W$ por uma distribuição de probabilidade *P*. Além disso, as cores de saída $A^{\mathbf{y}}(p_i)$ são obtidas aplicando a função alvo ψ em cada $a^{\mathbf{x}}_{p_i}$, isto é, $A^{\mathbf{y}}(p_i) = \psi(a^{\mathbf{x}}_{p_i})$ para todos os pares $(a^{\mathbf{x}}_{p_i}, A^{\mathbf{y}}(p_i)) \in \vec{a}$.

O aprendiz **A** deve considerar algum conjunto $H \subset \left(\{0,1\}^W \to \{0,1\}\right)$ de possíveis hipóteses quando tenta aprender o conceito alvo ψ. Se nenhuma informação sobre ψ estiver disponível, o aprendiz deve assumir que $H = \left(\{0,1\}^W \to \{0,1\}\right)$. Porém, uma informação *a priori* pode simplificar bastante o processo de aprendizagem, pois ela pode reduzir substancialmente a cardinalidade do espaço das hipóteses *H*. Por exemplo, emular uma erosão Ψ com a informação de que Ψ é uma erosão é muito mais fácil do que emulá-la sem nenhuma informação *a priori* (exemplos 2.2 e 2.3). Uma erosão é um operador elementar de morfologia matemática e a sua definição encontra-se, por exemplo, em [Gonzalez and Woods, 1992]. No estágio de treinamento do

W-operador, o aprendiz **A** recebe uma seqüência de amostras $\vec{a}$ e procura uma hipótese $\hat{\psi} = \mathbf{A}(\vec{a})$ no espaço *H*.

Vamos definir o erro verdadeiro (t-erro) da hipótese $\hat{\psi}$ como a probabilidade de que $\hat{\psi}$ irá classificar incorretamente uma instância $q_{p_i}^{\mathbf{x}}$ escolhida aleatoriamente por *P*:

$$\text{t\_error}_P(\hat{\psi}) = P\left\{q_{p_i}^{\mathbf{x}} \in \{0,1\}^w \middle| \psi(q_{p_i}^{\mathbf{x}}) \neq \hat{\psi}(q_{p_i}^{\mathbf{x}})\right\}$$

De acordo com a teoria PAC [Mitchell, 1997; Anthony and Biggs, 1992], qualquer aprendiz consistente utilizando um espaço de hipótese finito *H* com uma função alvo $\psi \in H$ irá, com probabilidade maior que (1-δ), gerar uma hipótese $\hat{\psi}$ com erro menor que ε, depois de observar *m* exemplos escolhidos aleatoriamente pelo *P*, desde que

$$m \geq \frac{1}{\varepsilon}\left[\ln\left(\frac{1}{\delta}\right) + \ln\left(|H|\right)\right]. \tag{2.1}$$

Um aprendiz é consistente se, sempre que possível, gerar uma hipótese que se adapte perfeitamente aos dados de treinamento. O limite (2.1) freqüentemente está substancialmente superestimado, principalmente porque nenhuma suposição foi feita sobre o aprendiz exceto a consistência. Alguns exemplos de uso desta equação seguem.

Exemplo 2.1: Na figura 2.2, uma imagem de impressão digital $A^{\mathbf{x}}$ (2.2a) foi processada por W-operador Ψ, gerando a imagem $A^{\mathbf{y}}$ (2.2b). Este operador consistiu em união de 8 operadores hit-or-miss definidos dentro da janela 3×3. O operador hit-or-miss é um dos operadores elementares da morfologia matemática e a sua definição encontra-se, por exemplo, em [Gonzalez and Woods, 1992]. Vamos supor que de alguma forma conhecemos que Ψ está definida na janela 3×3. Utilizando esta informação e as imagens $A^{\mathbf{x}}$ e $A^{\mathbf{y}}$, um W-operador $\hat{\Psi}$ foi construído por um aprendiz consistente. De acordo com a equação (2.1), com probabilidade maior que 99%, o erro verdadeiro de $\hat{\Psi}$ será menor que 1%, desde que as imagens de treinamento tenham uma quantidade de pixels

$$m \geq \frac{1}{0{,}01}\left[\ln\left(\frac{1}{0{,}01}\right) + \ln\left(2^{2^9}\right)\right] \cong 35950\,.$$

Como as imagens $A^{\mathbf{x}}$ e $A^{\mathbf{y}}$ têm 200×200=40000 pixels, quase certamente $\hat{\Psi}$ irá apresentar uma taxa de erro menor que 1%. De fato, quando $\hat{\Psi}$ foi aplicado a uma outra imagem de impressão digital (figura 2.2c), uma imagem $\hat{Q}^{\mathbf{y}} = \hat{\Psi}(Q^{\mathbf{x}})$ (figura 2.2d) exatamente igual à saída ideal $Q^{\mathbf{y}} = \Psi(Q^{\mathbf{x}})$ foi produzida. Isto é, $\hat{\Psi}$ apresentou erro zero. Este teste foi repetido algumas vezes e as taxas de erro sempre foram zero. ■

Note que a análise acima somente é válida quando se pode supor que as imagens $A^{\mathbf{x}}$ e $Q^{\mathbf{x}}$ foram geradas por uma mesma distribuição de probabilidade. Isto é, $A^{\mathbf{x}}$ e $Q^{\mathbf{x}}$ devem ser do mesmo tipo: imagens de impressões digitais, documentos manuscritos, documentos impressos, etc.

Exemplo 2.2: Vamos resolver novamente o exemplo 2.1, desta vez supondo que o operador alvo é mais complexo e está definido dentro de uma janela 7×7. Neste caso:

$$m \geq \frac{1}{0{,}01}\left[\ln\left(\frac{1}{0{,}01}\right) + \ln\left(2^{2^{49}}\right)\right] \cong 3{,}9\times 10^{16}\,.$$

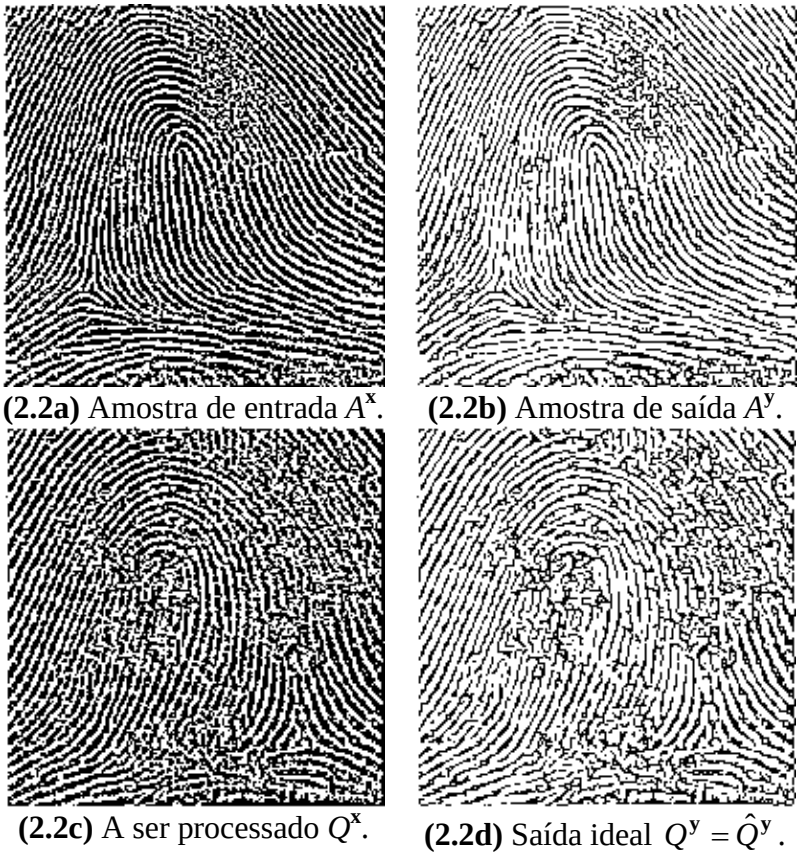
Isto é, as imagens amostras devem ser maiores que $(2\times10^8)\times(2\times10^8)$! Claramente, uma imagem tão grande não pode ser obtida na prática. ■

Exemplo 2.3: Vamos resolver novamente o exemplo 2.2, desta vez supondo que temos conhecimento de que o operador alvo é uma erosão cujo elemento estruturante cabe dentro de uma janela 7×7. Como cada um dos 49 furos pode pertencer ou não ao elemento estruturante, o operador alvo tem de ser uma das 2^{49} erosões. Assim, $|H| = 2^{49}$ e:

$$m \geq \frac{1}{0{,}01}\left[\ln\left(\frac{1}{0{,}01}\right) + \ln\left(2^{49}\right)\right] \cong 3857\,.$$

Isto é, qualquer par de imagens de treinamento maiores que 63×63 será suficiente. Compare com o tamanho das imagens $(2\times10^8)\times(2\times10^8)$ do exemplo 2.2. ■

A simplificação acima somente é válida quando se utiliza um algoritmo de aprendizagem projetado especialmente para as erosões. Resultados semelhantes podem ser obtidos para outros operadores elementares tais como dilatação, hit-or-miss, união de k erosões, e assim por diante.



(2.2a) Amostra de entrada $A^{\mathbf{x}}$. **(2.2b)** Amostra de saída $A^{\mathbf{y}}$.

(2.2c) A ser processado $Q^{\mathbf{x}}$. **(2.2d)** Saída ideal $Q^{\mathbf{y}} = \hat{Q}^{\mathbf{y}}$.

Fig. 2.2: Aprendizagem de W-operador num ambiente sem ruído.

Caso ruidoso

Para modelar o caso ruidoso, vamos supor que cada exemplo $(a_p^{\mathbf{x}}, A^{\mathbf{y}}(p)) \in \vec{a}$ tenha sido gerado independentemente por uma distribuição de probabilidade conjunta *P* desconhecida no espaço $\{0,1\}^w \times \{0,1\}$. Vamos também supor que cada elemento $(q_{p_i}^{\mathbf{x}}, Q^{\mathbf{y}}(p_i)) \in \vec{q}$ tenha sido gerado pela mesma distribuição *P*.

O erro verdadeiro da hipótese ψ agora deve ser definido como a probabilidade de que ψ classifique incorretamente um exemplo $(q_{p_i}^{\mathbf{x}}, Q^{\mathbf{y}}(p_i))$ escolhido aleatoriamente por *P*:

$$\text{t\_error}_P(\psi) = P\left\{(q_{p_i}^{\mathbf{x}}, Q^{\mathbf{y}}(p_i)) \in \{0,1\}^w \times \{0,1\} \middle| \psi(q_{p_i}^{\mathbf{x}}) \neq Q^{\mathbf{y}}(p_i)\right\}$$

Na situação ruidosa, não existe uma função alvo claramente definida. No seu lugar, existe uma função ψ^* com o menor erro verdadeiro. Vamos definir o erro empírico (e-erro) de uma hipótese ψ sobre uma seqüência $\vec{a}$ como a proporção de erros cometidos quando ψ classifica as instâncias de $\vec{a}$:

$$\text{e\_error}_{\vec{a}}(\psi) = \left(\frac{1}{m}\right) \left| \left\{(a_{p_i}^{\mathbf{x}}, A^{\mathbf{y}}(p_i)) \in \vec{a} \;\middle|\; \psi(a_{p_i}^{\mathbf{x}}) \neq A^{\mathbf{y}}(p_i)\right\} \right|,$$

onde *m* é o comprimento de $\vec{a}$.

Seja $\hat{\psi}$ a hipótese com o menor e-erro sobre $\vec{a}$ e seja ψ^* a hipótese com o menor erro verdadeiro. Então [Haussler, 1992]

$$\Pr\left[\text{t\_error}_P(\hat{\psi}) - \text{t\_error}_P(\psi^*) > \varepsilon\right] < \delta,$$

desde que *H* seja finito e o comprimento *m* de $\vec{a}$ satisfaça:

$$m \geq \frac{1}{2\varepsilon^2}\left[\ln\left(\frac{1}{\delta}\right) + \ln\left(2|H|\right)\right]. \qquad (2.2)$$

Infelizmente, a complexidade de amostra acima é uma superestimativa ainda maior que a da equação (2.1). Dada uma seqüência de amostras $\vec{a}$, a hipótese empiricamente ótima (e-ótima) $\hat{\psi}$ pode ser construída facilmente. Vamos definir que um aprendiz

A é e-ótimo se ele gerar sempre uma hipótese e-ótima sobre a seqüência de treinamento. Se **A** fosse e-ótimo, dado um padrão de busca $q_{p_i}^{\mathbf{x}}$, qual deveria ser a sua classificação $\hat{\psi}(q_{p_i}^{\mathbf{x}}) = \mathbf{A}(\vec{a})(q_{p_i}^{\mathbf{x}})$? Sejam $(a_{r_1}^{\mathbf{x}}, A^{\mathbf{y}}(r_1))$, ..., $(a_{r_N}^{\mathbf{x}}, A^{\mathbf{y}}(r_N))$ os N exemplos de treinamento de $q_{p_i}^{\mathbf{x}}$ em $\vec{a}$, isto é, $a_{r_j}^{\mathbf{x}} = q_{p_i}^{\mathbf{x}}$, $1 \le j \le N$ (não há outros exemplos de $q_{p_i}^{\mathbf{x}}$ em $\vec{a}$ além desses). Como há ruído, os N exemplos acima podem não concordar sobre a classificação de $q_{p_i}^{\mathbf{x}}$. Para minimizar e-erro, a classificação deve ser decidida pela maioria dos votos desses exemplos de treinamento:

$$\hat{\psi}(q_{p_i}^{\mathbf{x}}) \leftarrow \text{moda}(A^{\mathbf{y}}(r_1), \ldots, A^{\mathbf{y}}(r_N)).$$

Note que todo aprendiz e-ótimo é consistente num ambiente sem ruído. Apresentamos abaixo um exemplo.

Exemplo 2.4: As imagens de impressões digitais 2.2a e 2.2c foram corrompidas pelo ruído "sal e pimenta", resultando em imagens 2.3a e 2.3b. Em média, 1 em cada 40 pixels mudou de cor. Gostaríamos de projetar um W-operador 3×3 $\hat{\Psi}$ tal que uma imagem semelhante à saída ideal $A^{\mathbf{y}}$ (figura 2.2b) resulte, apesar do ruído, quando a imagem 2.3a é processada por $\hat{\Psi}$. Para atingir este objetivo, um W-operador $\hat{\Psi}$ foi projetado por um aprendiz e-ótimo usando as imagens 2.3a e 2.2b como amostras de treinamento. Como as imagens 2.3a e 2.2b têm 200×200 pixels, com probabilidade pelo menos 99%, a diferença entre os erros verdadeiros do operador ótimo Ψ^* e do operador $\hat{\Psi}$ será menor que 6,71%, i.e., $\text{t\_error}_P(\hat{\psi}) - \text{t\_error}_P(\psi^*) \le 0{,}0671$, pois:

$$\frac{1}{2 \times 0{,}0671^2}\left[\ln\left(\frac{1}{0{,}01}\right) + \ln\left(2 \times 2^{2^9}\right)\right] \cong 40000.$$

No exemplo 2.5, este problema será analisado novamente. ■

(2.3a) Amostra de entrada ruidosa.

(2.3b) Imagem ruidosa a-ser-processada.

(2.3c) Imagem processada.

Fig. 2.3: Aprendizagem de W-operador num ambiente ruidoso.

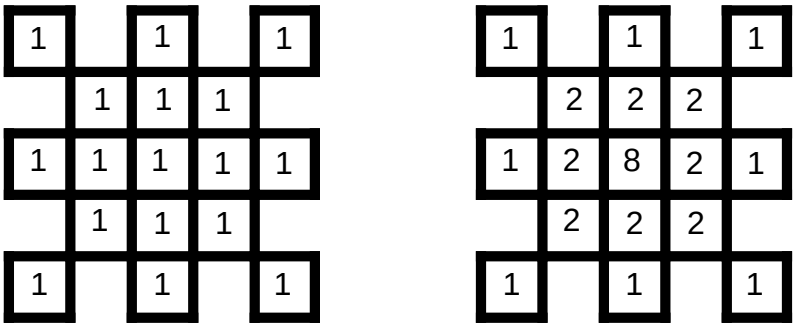


Fig. 2.4: Janelas sem e com pesos, com 17 furos-de-espiar.

Estimação estatística da taxa de erro

Esta subsubseção irá expor as técnicas para calcular um limite mais estreito para a taxa de erro. Estas técnicas serão muito úteis, pois as equações (2.1) e (2.2) normalmente superestimam a complexidade de amostra e a taxa de erro. Ao contrário das fórmulas anteriores, as técnicas desta subsubseção podem ser aplicadas somente após ter projetado W-operador, com a condição adicional de que a imagem de saída ideal $Q^{\mathbf{y}}$ esteja disponível. É lícito supor que a saída ideal estará disponível para se realizar testes, pois estamos supondo que um par de imagens entrada-saída de treinamento está disponível para projetar W-operador. E, se as imagens de treinamento estão disponíveis, elas podem ser quebradas em dois pedaços: imagens de treinamento ($A^{\mathbf{x}}$, $A^{\mathbf{y}}$) e imagens de teste ($Q^{\mathbf{x}}$, $Q^{\mathbf{y}}$).

Portanto, supondo que a saída ideal $Q^{\mathbf{y}}$ esteja disponível, uma simples contagem de pixels diferentes entre $Q^{\mathbf{y}}$ e $\hat{Q}^{\mathbf{y}}$ irá fornecer o e-erro. E, dada a acurácia observada de uma hipótese sobre uma amostra de dados limitada, é possível conhecer o quanto esta irá conseguir estimar a acurácia sobre exemplos adicionais. Para isso, vamos construir intervalos de confiança unilateral ou bilateral. Explicações adicionais sobre intervalos de confiança da média de variáveis aleatórias binomiais encontram-se em [Mitchell, 1997] ou em muitos livros elementares de Estatística. Com N% de confiança:

$$\text{t\_error}_P(\hat{\psi}) \in \text{e\_error}_{\vec{q}}(\hat{\psi}) \pm z_N \sqrt{\frac{\text{e\_error}_{\vec{q}}(\hat{\psi})(1-\text{e\_error}_{\vec{q}}(\hat{\psi}))}{n}}, \qquad (2.3)$$

$$\text{t\_error}_P(\hat{\psi}) \leq \text{e\_error}_{\vec{q}}(\hat{\psi}) + z'_N \sqrt{\frac{\text{e\_error}_{\vec{q}}(\hat{\psi})(1-\text{e\_error}_{\vec{q}}(\hat{\psi}))}{n}}, \qquad (2.4)$$

onde n é o comprimento de $\vec{q}$; z_N define a metade da largura do menor intervalo em torno da média que inclui N% da massa da probabilidade total sob distribuição normal com desvio-padrão 1; e $z'_N \equiv z_{2N-1}$. Por exemplo, $z'_{84\%} = z_{68\%} = 1{,}00$, $z'_{95\%} = z_{90\%} = 1{,}64$, $z'_{99\%} = z_{98\%} = 2{,}33$ e $z_{99\%} = 2{,}58$. As fórmulas (2.3) e (2.4) nor-

malmente produzem uma estimativa da taxa de erro muito mais acurada que as equações (2.1) e (2.2).

No caso sem ruído, basta conhecer um limite superior para a taxa de erro verdadeiro do operador projetado ($\text{t\_error}_P(\hat{\psi})$). Porém, para os casos ruidosos, o erro mínimo ($\text{t\_error}_P(\psi^*)$) também deve ser estimado pois, como o operador projetado nunca poderá atingir uma taxa de erro verdadeiro menor que o mínimo, um operador pode ser considerado uma boa solução se o seu erro verdadeiro estiver próximo do mínimo. Infelizmente, não há meios para se estimar $\text{t\_error}_P(\psi^*)$ diretamente, pois o operador ótimo é desconhecido. Descrevemos abaixo um artifício que tem conseguido estabelecer bons limites inferiores para $\text{t\_error}_P(\psi^*)$. Embora muito simples, nunca vimos esta técnica descrita na literatura.

Para isso, vamos construir a hipótese $\hat{\psi}^*$ e-ótima sobre $\vec{q}$. Se o aprendiz **A** for e-ótimo, $\hat{\psi}^* = \mathbf{A}(\vec{q})$. Note que estamos treinando o operador com as próprias imagens ($Q^{\mathbf{x}}$, $Q^{\mathbf{y}}$) que serão utilizadas no teste. Claramente, $\text{e\_error}_{\vec{q}}(\hat{\psi}^*) \leq \text{e\_error}_{\vec{q}}(\psi^*)$ e $\text{e\_error}_{\vec{q}}(\hat{\psi}^*)$ pode ser medido experimentalmente. Então, utilizamos a seguinte desigualdade para estabelecer um limite inferior para $\text{t\_error}_P(\psi^*)$:

$$\begin{aligned} \text{t\_error}_P(\psi^*) &\geq \text{e\_error}_{\vec{q}}(\psi^*) - z'_N \sqrt{\frac{\text{e\_error}_{\vec{q}}(\psi^*)(1-\text{e\_error}_{\vec{q}}(\psi^*))}{n}} \\ &\geq \text{e\_error}_{\vec{q}}(\hat{\psi}^*) - z'_N \sqrt{\frac{\text{e\_error}_{\vec{q}}(\hat{\psi}^*)(1-\text{e\_error}_{\vec{q}}(\hat{\psi}^*))}{n}} \end{aligned} \quad (2.5)$$

A desigualdade acima é verdadeira, com nível de confiança *N*%, toda vez que:

$$\frac{b+1-\sqrt{b(b+1)}}{2(b+1)} \leq \text{e\_error}_{\vec{q}}(\hat{\psi}^*) \leq \text{e\_error}_{\vec{q}}(\psi^*) \leq \frac{b+1+\sqrt{b(b+1)}}{2(b+1)} \quad (2.6)$$

onde $b = n/(z'_N)^2$. Note que a desigualdade (2.6) é verdadeira para praticamente todos os problemas práticos e conseqüentemente a desigualdade (2.5) também é sempre verdadeira na prática.

Exemplo 2.5: No exemplo 2.4, tínhamos concluído com 99% de confiança que o operador $\hat{\Psi}$ obtido comete no máximo 6,71% mais erros que o operador ótimo 3×3. A fim de estabelecer um limite de erro mais estreito, o e-erro de $\hat{\Psi}$ (a diferença entre as imagens 2.2d e 2.3c) foi medido e descobriu-se que valia 4,992%. Utilizando a equação (2.3), concluímos com 99% de confiança que o erro verdadeiro de $\hat{\Psi}$ pertence ao intervalo (4,992±0,281)%. O operador $\hat{\Psi}^*$ e-ótimo sobre as imagens de teste (figuras 2.3b e 2.2d) foi construído e cometeu e-erro 4,723% quando processou a imagem 2.3b. Utilizando a desigualdade (2.5), concluímos com 99% de confiança que o erro verdadeiro do operador ótimo 3×3 é maior que (4,723-0,247)%. Conseqüentemente, com confiança de pelo menos 99%, o erro verdadeiro de $\hat{\Psi}$ é no máximo 0,797% maior que o erro verdadeiro do operador ótimo, isto é:

$$\text{t\_error}_P(\hat{\psi}) - \text{t\_error}_P(\psi^*) \leq 0{,}00797 .$$

Este resultado confirma que a equação (2.2) superestima a taxa de erro, pois 0,797% é muito menor que 6,71%. ■

Viés indutivo ek-NN

Nas subsubseções anteriores, tínhamos suposto que o aprendiz era e-ótimo (ou consistente) para calcular a complexidade de amostra. Porém a e-otimalidade sozinha não especifica inteiramente um algoritmo de aprendizagem, pois existem muitos diferentes aprendizes e-ótimos. Para especificar completamente um aprendiz, um método de generalização (viés indutivo) também deve ser escolhido.

Para a aprendizagem de W-operador, sugerimos que se utilize a generalização *k*-NN [Mitchell, 1997], pois nos parece bastante natural que padrões semelhantes sejam classificados similarmente. Uma outra possibilidade seria utilizar a generalização dada pela árvore de decisão [Mitchell, 1997], pois se aproxima muito da generalização *k*-NN. Evidentemente, ao se escolher um viés indutivo, deve-se levar em conta a existência de algoritmos computacionalmente eficientes que consigam implementá-lo. Também se deve tomar cuidado para que a generalização mantenha a e-otimalidade pois, caso contrário, a teoria PAC se tornará inválida.

Para ilustrar o perigo da não e-otimalidade, considere a aprendizagem *k*-NN. O seu viés indutivo corresponde à suposição de que a classificação de uma instância será mais parecida à classificação de outras instâncias que estão próximas em distância. No algoritmo *k*-NN ingênuo, o treinamento consiste simplesmente em armazenar os dados de treinamento apresentados. De acordo com a regra *k*-NN, para cada padrão de busca $q_{p_i}^{\mathbf{x}}$, os *k* padrões exemplos de entrada "mais parecidos" devem ser procurados em $\vec{a}$. Como estamos lidando com imagens binárias, as distâncias entre $q_{p_i}^{\mathbf{x}}$ e os padrões de treinamento devem ser medidas utilizando a distância de Hamming (isto é, o número de bits discordantes) ou a distância de Hamming com pesos. No último caso, pode-se dar mais peso a alguns furos de espiar (por exemplo, os furos centrais) do que a outros (por exemplo, os furos periféricos). A figura 2.4 mostra duas janelas sem e com pesos. A saída é definida como a classificação mais comum entre os *k* exemplos de treinamento mais próximos. Claramente, esta regra *k*-NN original não é e-ótima. Porém, mudando-a ligeiramente como segue, ela torna-se e-ótima:

1) Se o padrão a-ser-processado $q_{p_i}^{\mathbf{x}}$ aparecer uma ou mais vezes em $\vec{a}$, a sua classificação será dada pela maioria dos votos somente dessas instâncias de treinamento. Isto é, sejam $a_{r_1}^{\mathbf{x}}$, ..., $a_{r_N}^{\mathbf{x}}$ as instâncias de treinamento tais que $a_{r_j}^{\mathbf{x}} = q_{p_i}^{\mathbf{x}}, 1 \le j \le N$. Então, faça $\hat{q}_{p_i}^{\mathbf{y}} \leftarrow \underset{1 \le j \le N}{\text{moda}}\left(A^{\mathbf{y}}(r_j)\right)$. Neste caso, *N* pode ser maior, igual ou menor que *k*.

2) Por outro lado, se o padrão a-ser-processado $q_{p_i}^{\mathbf{x}}$ nunca foi visto antes, procure pelas suas *k* instâncias mais semelhantes em $\vec{a}$ e escolha o voto majoritário delas. Isto é, sejam $a_{r_1}^{\mathbf{x}}$, ..., $a_{r_N}^{\mathbf{x}}$ as *N* instâncias mais semelhantes à $q_{p_i}^{\mathbf{x}}$, de acordo com alguma medida de distância. Então, novamente faça $\hat{q}_{p_i}^{\mathbf{y}} \leftarrow \underset{1 \le j \le N}{\text{moda}}\left(A^{\mathbf{y}}(r_j)\right)$. Neste caso, *N* pode ser igual ou maior que *k* (se houver empate), mas nunca pode ser menor que *k*.

Chamamos esta regra modificada de aprendizagem *k* vizinhos mais próximos empiricamente ótima (abreviado como ek-NN). A aprendizagem ek-NN parece ser muito apropriada para ser usada na aprendizagem de W-operadores. Porém, para ser realmente útil, deveriam existir estruturas de dados e algoritmos que permitam uma implementação eficiente. As implementações possíveis são as mesmas da aprendizagem *k*-NN, já vistas na introdução: a força-bruta, a LUT e a kd-árvore. A força-bruta é muito lenta. A LUT é extremamente rápida na aplicação, porém a sua velocidade de treinamento e a memória gasta crescem exponencialmente com o aumento da janela. A kd-árvore pode ser treinada rapidamente e a memória gasta é razoável, porém o seu tempo de busca torna-se proibitivo nas dimensões altas.

Aprendizagem por árvore de decisão

A aprendizagem ek-NN vista na subsubseção anterior não pode ser usada para projetar W-operadores definidos em janelas amplas, pois não existem algoritmos e estruturas de dados eficientes. Assim, somos forçados a buscar alternativas. Vamos examinar a aprendizagem por árvore de decisão (DT) [Mitchell, 1997]. Ela é uma das técnicas mais amplamente utilizadas para aproximar funções alvos discretos. A função aprendida é representada como uma árvore (no nosso problema, uma árvore binária). Na realidade, a árvore de decisão é muito similar à kd-árvore usada na aprendizagem *k*-NN. A diferença principal está no estágio de busca: não existe um processo de backtracking. Isto torna a busca muito rápida, na prática milhões de vezes mais rápida que a kd-árvore, superando a deficiência que torna impossível o uso da kd-árvore em aprendizagem de W-operador com janela grande. A eliminação de backtracking também elimina a necessidade de armazenar padrões de entrada nas folhas, diminuindo o uso de memória.

A aprendizagem DT é e-ótima. Esta propriedade fixa os valores de saída para todos padrões de busca que aparecem pelo menos uma vez na seqüência de treinamento. Por outro lado, se o aprendiz nunca viu o padrão de busca, o valor de saída é escolhido de acordo com o viés indutivo de aprendizagem DT: prefira as árvores que colocam atributos com alto ganho de informação mais próximos à raiz sobre aqueles que

não fazem isso. Este costume torna o comportamento da aprendizagem DT bastante similar ao da aprendizagem ek-NN. Também aproxima o viés indutivo conhecido como a "navalha de Occam": prefira a hipótese mais simples que explica os dados observados.

Para explicar a construção de uma árvore de decisão, sejam dados *n* padrões amostras de entrada com as correspondentes cores de saída:

$$\vec{a} = \left((a^{\mathbf{x}}_{p_1}, A^{\mathbf{y}}(p_1)),\ldots,(a^{\mathbf{x}}_{p_m}, A^{\mathbf{y}}(p_m))\right),\ a^{\mathbf{x}}_{p_i} \in \{0,1\}^w \text{ e } A^{\mathbf{y}}(p_i) \in \{0,1\}.$$

No processo de geração da árvore DT, um atributo de corte $s \in [1...w]$ é escolhido e o espaço de padrões $\{0,1\}^w$ é cortado em duas metades. Todas as amostras com atributo *s* preto irão pertencer a um semi-espaço e aquelas com branco ao outro. Em cada corte, um nó interno é criado e o atributo de corte *s* armazenado nele.

Para obter uma árvore otimizada, em cada estágio de corte, o atributo *s* deve ser escolhido de forma que o ganho de informação seja maximizado. Assim, em cada corte, os ganhos de informação de todos os atributos são calculados e o atributo com o maior ganho é escolhido como o atributo de corte. O ganho de informação é a redução de entropia esperada causada ao particionar os exemplos de acordo com o atributo *s*:

$$\text{Gain}(\vec{a}, s) = \text{Entropy}(\vec{a}) - \left(\frac{b}{m}\text{Entropy}(\vec{a}_{v_s=0}) + \frac{m-b}{m}\text{Entropy}(\vec{a}_{v_s=1})\right)$$

onde $\vec{a}_{v_s=0}$ ($\vec{a}_{v_s=1}$) é a subseqüência de $\vec{a}$ com todas as amostras cujo valor no atributo *s* é preto (branco). Utilizamos a notação v_s para denotar o valor do atributo *s*. A entropia de uma seqüência de amostra $\vec{a}$ com *b* saídas pretas (e conseqüentemente *m*-*b* saídas brancas) é:

$$\text{Entropy}(\vec{a}) \quad = \quad -\left(\frac{b}{m}\right)\log_2\left(\frac{b}{m}\right) - \left(\frac{m-b}{m}\right)\log_2\left(\frac{m-b}{m}\right).$$

Para cada um dos dois semi-espaços obtidos, o processo de corte continua recursivamente, gerando subespaços cada vez menores. Este processo pára quando cada subespaço contiver ou somente amostra com a mesma cor de saída ou somente amostras

com o mesmo padrão de entrada (mas com duas diferentes cores de saída). No primeiro caso, um nó terminal é criado e a cor de saída é armazenada nele. No segundo caso, um nó terminal também é criado e, para assegurar a e-otimalidade, a moda das cores de saída é avaliada e armazenada.

A árvore de decisão construída representa a função característica $\hat{\psi}$. Dado um padrão de busca $q_{p_i}^{\mathbf{x}}$, a sua cor de saída $\hat{Q}^{\mathbf{y}}(p_i) = \hat{\psi}(q_{p_i}^{\mathbf{x}})$ é calculada executando uma busca na árvore. A busca começa no nó raiz. Em cada nó interno, a direção a seguir (esquerda ou direita) é escolhida de acordo com o valor do padrão de busca no atributo de corte *s*. O processo é repetido até chegar a um nó terminal. O valor da função característica $\hat{\psi}$ é a cor de saída armazenada no nó terminal.

Dadas *m* amostras e *n* pontos de busca no espaço de padrões de dimensão *w*, pode ser mostrado que a árvore de decisão pode ser construída em tempo médio $O(wm\log m)$. A aplicação leva $O(n\log m)$ e a complexidade de uso de memória é $O(m)$. Esta análise mostra que tanto a construção quanto a busca são extremamente rápidas, enquanto a memória é utilizada economicamente mesmo em dimensões altas.

Comparação dos diferentes vieses indutivos

Freqüentemente, estamos interessados em comparar o desempenho de dois algoritmos de aprendizagem $\mathbf{A}_1$ e $\mathbf{A}_2$ em vez de duas hipóteses específicas. Por exemplo, podemos querer determinar se o viés indutivo de ek-NN é mais efetivo que os outros. Em outras palavras, gostaríamos estimar a diferença esperada entre as taxas de erros verdadeiros:

$$E\left[\text{t\_error}_P(\mathbf{A}_1(\vec{a})) - \text{t\_error}_P(\mathbf{A}_2(\vec{a}))\right] = \sum_{\vec{a}\in(\{0,1\}^w\times\{0,1\})^m}\left[\text{t\_error}_P(\mathbf{A}_1(\vec{a})) - \text{t\_error}_P(\mathbf{A}_2(\vec{a}))\right]P^m(\vec{a})\cdot$$

Para estabelecer um intervalo de confiança para a quantidade acima, os dois aprendizes $\mathbf{A}_1$ e $\mathbf{A}_2$ devem ser treinados utilizando *K* seqüências de treinamento independen-

tes $\vec{a}_i$, $1 \le i \le K$, e as hipóteses resultantes aplicadas a *K* diferentes seqüências de teste $\vec{q}_i$, $1 \le i \le K$. Este processo irá gerar *K* diferenças entre os e-erros de $\mathbf{A}_1$ e $\mathbf{A}_2$:

$$\delta_i = \text{e\_error}_{\vec{q}_i}(\mathbf{A}_1(\vec{a}_i)) - \text{e\_error}_{\vec{q}_i}(\mathbf{A}_2(\vec{a}_i)),\ 1 \le i \le K.$$

Intervalos de confiança unilateral ou bilateral podem ser construídos a partir de δ_1, ..., δ_K utilizando a distribuição *t* de Student. Com confiança *N*%:

$$\underset{\vec{a} \in P^m}{E}\left[\text{t\_error}_P(\mathbf{A}_1(\vec{a})) - \text{t\_error}_P(\mathbf{A}_2(\vec{a}))\right] \in \overline{\delta} \pm t_{N,k-1} s_{\overline{\delta}} \tag{2.7}$$

$$\underset{\vec{a} \in P^m}{E}\left[\text{t\_error}_P(\mathbf{A}_1(\vec{a})) - \text{t\_error}_P(\mathbf{A}_2(\vec{a}))\right] > \overline{\delta} - t'_{N,k-1} s_{\overline{\delta}} \tag{2.8}$$

onde:

- $s_{\overline{\delta}} \equiv \sqrt{\dfrac{1}{K(K-1)} \sum_{i=1}^{K} (\delta_i - \overline{\delta})^2}$;
- $\overline{\delta} \equiv \left(\delta_1 + \cdots + \delta_K\right) / K$;
- $t_{N,K-1}$ define a meia largura do menor intervalo em torno da média que inclui *N*% da massa de probabilidade total sob a distribuição *t* normalizada com (*K*-1) graus de liberdade; e $t'_{N,K-1} \equiv t_{2N-1,K-1}$.

Por exemplo, $t'_{95\%,2} = t_{90\%,2} = 2{,}92$, $t'_{97.5\%,2} = t_{95\%,2} = 4{,}30$ e $t'_{99\%,2} = t_{98\%,2} = 6{,}96$.

2.3 Aumento de Resolução de Imagens Binárias

Introdução

Esta seção descreve uma contribuição científica original minha. Os resultados descritos nesta seção estão documentados em artigos [Ri03; Ci02; Cn10].

Nesta seção, usaremos a teoria desenvolvida na seção anterior para aumentar a resolução de imagens binárias de documentos impressos ou manuscritos. Num ambiente de escritório típico, as imagens digitais e os documentos são manipulados por um conjunto de equipamentos e softwares não-homogêneos que formam um sistema capaz de escanear, editar, mostrar, imprimir, transmitir, efetuar OCR, e executar várias outras tarefas de Processamento e Análise de Imagens. Como cada componente do sistema pode operar numa resolução espacial diferente, freqüentemente aparece a necessidade da conversão de resolução, para permitir que as imagens e os documentos digitais migrem de um componente do sistema a outro.

A diminuição da resolução espacial é uma tarefa relativamente fácil. Em contraste, o aumento da resolução espacial (ou ampliação ou zoom) é difícil, pois a imagem de entrada normalmente não contém toda a informação necessária para gerar uma imagem de saída perfeitamente ampliada. Além disso, a ampliação ideal depende do "contexto" da aplicação. Por exemplo, o operador ótimo, projetado para ampliar duas vezes os caracteres "Times, 12 pt., 300 dpi", pode não ser ótimo para uma outra fonte ou um documento manuscrito.

Muitos algoritmos de ampliação de imagens foram desenvolvidos para imagens em níveis de cinza e coloridas. Porém, parece que a ampliação de imagens binárias tem recebido muito menos atenção até agora. Loce et al. [Loce and Dougherty, 1997; Loce et al., 1997] apresentam algumas técnicas, entre o pequeno número publicadas na literatura, para a ampliação de imagens binárias. Isto causa certa surpresa, pois a ampliação de imagens binárias é muitas vezes necessária na prática. Por exemplo, con-

sidere o número de vezes em que uma imagem em 300 dpi teve que ser impressa numa impressora 600 dpi.

Muitas tarefas de processamento de imagens estão baseadas em operadores restritos à janela (W-operadores). Usaremos a aprendizagem *k*-NN (*k* vizinhos mais próximos) para projetar os operadores de ampliação restritos à janela (WZ-operadores, Z de zoom).

Loce et al. [Loce and Dougherty, 1997; Loce et al., 1997] em essência expõem duas técnicas para ampliar as imagens binárias. Eles usam filtros não-crescentes e crescentes (utilizamos as palavras “filtro” e “operador” como sinônimos). Nesta seção, propomos algumas melhorias sobre essas técnicas anteriores.

Primeiro, como a distribuição de probabilidade verdadeira que governa o processo de ampliação é normalmente desconhecida, na prática as estatísticas derivadas das imagens amostras de entrada-saída devem ser utilizadas no seu lugar. Conseqüentemente, o melhor operador que alguém pode obter na prática é o operador que é ótimo sobre as imagens de treinamento (não levando em conta o viés indutivo). Chamamos isto de operador empiricamente ótimo (e-ótimo). A técnica de filtro crescente [Loce and Dougherty, 1997, chap. 9; Loce et al., 1997] pode gerar uma solução sub-ótima, enquanto que a abordagem [Loce and Dougherty, 1997, chap. 6] e a nossa sempre projetam um operador e-ótimo. Além disso, os trabalhos anteriores não analisam a diferença entre os operadores empiricamente ótimo e verdadeiramente ótimo, assumindo implicitamente que as estatísticas derivadas das imagens amostras são uma aproximação próxima da verdadeira distribuição de probabilidade. Propomos usar as técnicas estatísticas para estimar a diferença entre as duas taxas de erro. Além disso, as técnicas prévias não adotam qualquer viés indutivo explícito. O viés indutivo é o conjunto de suposições *a priori* pelo qual o aprendiz generaliza além dos dados de treinamento observados, para inferir as classificações das novas instâncias. Um aprendiz que não assume nenhuma suposição *a priori* quanto à identidade do conceito alvo não possui nenhuma base racional para classificar qualquer instância ainda não vista. Adotamos a aprendizagem *k*-NN porque ela possui um viés indutivo sólido,

exaustivamente testado em muitas aplicações diferentes. Mostramos experimentalmente a sua eficácia para o problema em questão.

Em segundo lugar, os trabalhos anteriores parecem necessitar de certa intervenção humana no projeto de operador. A nossa técnica é, ao contrário, totalmente automática.

Em terceiro lugar, a técnica de filtro crescente é uma tentativa de melhorar a técnica não-crescente, visando a implementação em hardware: ela está focalizada em projetar um operador "logicamente eficiente", isto é, um operador representado utilizando um número pequeno de portas lógicas. A nossa abordagem está focalizada em implementação por software, onde a redução de lógica perde a sua atratividade porque uma lógica mais simples não necessariamente significa uma técnica mais rápida. Em seu lugar, o uso de algoritmos e estruturas de dados apropriados pode levar a métodos mais rápidos, reduzindo a complexidade computacional. Para acelerar a aplicação de W-operadores, Jones e Svalbe [Jones and Svalbe, 1994] usam look-up-table (LUT), Kim et al. [Ri01; Ci01; Cn06] usam uma estrutura de dados em forma de árvore, e Robert e Malandain [Robert and Malandain, 1998] usam diagrama de decisão binária. A LUT é extremamente rápida no estágio de aplicação e permite a implementação da aprendizagem *k*-NN exata, mas a sua demanda pela memória e tempo de treinamento cresce exponencialmente à medida que a janela cresce. A estrutura de árvore requer somente uma quantidade moderada de memória e tempo de treinamento, mas o seu tempo de aplicação é bem maior que LUT (quando ela implementa a aprendizagem *k*-NN exata usando uma estrutura de dado conhecida como kd-árvore, que requer um processo de back-tracking) ou ligeiramente maior que LUT (quando ela implementa uma árvore de decisão, uma estratégia de aprendizagem de máquina bastante semelhante à aprendizagem *k*-NN). O diagrama de decisão binária é tão rápido quanto a estrutura de árvore em aplicação e usa menos memória, mas o seu processo de treinamento é muito lento. Adotamos a solução LUT, pois os resultados experimentais mostraram que a ampliação de documentos impressos e manuscritos não necessita de janelas grandes.

Projeto de WZ-operador pela aprendizagem k-NN

Vamos definir o operador de aumento de resolução restrito à janela (WZ-operador, Z de zoom). Um WZ-operador Ψ é definido através da janela W e f^2 funções características ψ_0, ..., ψ_{f^2-1}, onde f é o fator de zoom. Trabalharemos somente com aumentos de resolução por fatores inteiros f. Além disso, para simplificar a notação, assumiremos que os fatores de aumento de linha e coluna são iguais. Por exemplo, $f=2$ aumenta a resolução espacial duas vezes em cada coordenada. Cada função característica é uma função booleana $\psi_i : \{0,1\}^w \to \{0,1\}$ e referiremos ao conjunto de f^2 funções ψ_i como ψ ($\psi : \{0,1\}^w \to \{0,1\}^{(f^2)}$). As funções ψ_i convertem um pixel de entrada p em f^2 pixels de saída y_i baseado no conteúdo da janela W deslocada para p, isto é, para $0 \le i < f^2$ (figura 2.5):

$$y_i = \Psi(Q)(f\ p + d_i) = \psi_i(Q(W_1 + p), \ldots, Q(W_w + p)),$$

onde $p \in \mathbb{Z}^2$ e d_i é o vetor de deslocamento associado à i-ésima função característica. Na figura 2.5, as funções características ψ_0, ..., ψ_3 convertem o pixel p em pixels y_0, ..., y_3 baseado no conteúdo da janela 3×3.

Para poder aplicar um WZ-operador a uma imagem $Q^{\mathbf{x}}$, a regra ek-NN deve ser aplicada a cada padrão a-ser-ampliado $q^{\mathbf{x}}_{p_i}$ de $Q^{\mathbf{x}}$. Infelizmente, este processo é excessivamente lento: para aumentar a resolução de cada pixel, a imagem amostra $A^{\mathbf{x}}$ inteira deve ser analisada. As nossas experiências mostram que este algoritmo ingênuo leva meses ou mesmo anos para aumentar a resolução de uma única imagem, utilizando um computador convencional.

Utilizamos look-up-table (LUT) para acelerar este processo. A LUT permite implementar a aprendizagem ek-NN e é extremamente rápida no tempo de avaliação, o que a torna adequada para aplicações de tempo real. Porém, a sua demanda pela memória e tempo de treinamento aumentam exponencialmente à medida que o tamanho da janela cresce. Isto torna impossível o seu uso para janelas grandes. Felizmente, as experiências mostraram que as janelas pequenas com 3×3, 4×4 ou 17

furos-de-espiar (figura 2.4) podem gerar bons WZ-operadores para aumentar a resolução de documentos impressos ou manuscritos.

Evidentemente, una função booleana $\psi:\{0,1\}^w \rightarrow \{0,1\}$ pode ser representada como uma LUT com 2^w linhas, numeradas de 0 a 2^w-1, onde cada célula é ou 0 ou 1. Portanto, uma tabela para representar f^2 funções deve ter 2^w linhas e f^2 colunas, ocupando $2^w f^2$ bits. Por exemplo, usando uma janela 4×4 e fator de zoom f=3, a tabela irá ocupar 589824 bits ou 73728 bytes. Cada coluna irá representar uma função característica ou hipótese $\hat{\psi}_i$.

O processo de aprendizagem *k*-NN deve preencher a LUT. O índice *l* de cada linha representa um padrão binário $q_l^{\mathbf{x}}$, de comprimento *w*. Para cada $q_l^{\mathbf{x}}$, a regra ek-NN deve ser aplicada. Este processo pode ser acelerado substancialmente criando um vetor onde cada padrão de entrada de $A^{\mathbf{x}}$ aparece uma única vez, junto com o número de votos para as saídas branca e preta. Depois, a busca é executada neste vetor, em vez de em $A^{\mathbf{x}}$. Note que o vetor de padrões não repetidos pode ser criado rapidamente utilizando qualquer algoritmo de ordenação $O(m \log m)$, como quicksort ou heapsort [Cormen et al., 1990], seguido por um algoritmo $O(m)$ para eliminar os padrões repetidos.

Depois que a LUT esteja completamente preenchida, dado um padrão a-ser-ampliado, os pixels de saída y_i podem ser calculados sem esforço simplesmente indexando a linha correspondente da LUT.

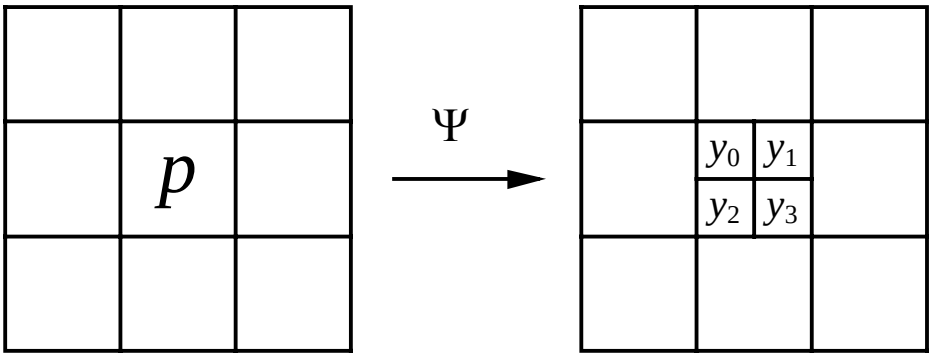


Fig. 2.5: Operador de aumento de resolução restrito à janela 3×3 (WZ-operador) com fator de zoom f =2.

	janela	ψ_1	ψ_2	ψ_3	ψ_4	média
$e\_error_{\vec{q}}(\hat{\psi}_i)$	3×3	1,10%	1,03%	1,05%	1,05%	1,058%
$e\_error_{\vec{q}}(\hat{\psi}_i^*)$, e-ótima sobre $\vec{q}$	3×3	1,09%	1,02%	1,04%	1,03%	1,045%
$e\_error_{\vec{q}}(\hat{\psi}_i)$	17	1,01%	0,95%	1,00%	1,02%	0,995%
$e\_error_{\vec{q}}(\hat{\psi}_i^*)$, e-ótima sobre $\vec{q}$	17	0,95%	0,89%	0,92%	0,92%	0,920%

Tab. 2.1: Erros empíricos obtidos usando a regra e1-NN ao aumentar a resolução de documento impresso.

Aumento de resolução de caracteres impressos

Para testar as idéias expostas acima, projetamos um WZ-operador 3×3 para aumentar a resolução de documentos contendo caracteres “Times 12 pt.” (tanto normal como itálico) de 300 dpi para 600 dpi. As imagens de treinamento foram obtidas imprimindo os documentos eletrônicos para arquivos “.PS” através de um *driver* de uma impressora *PostScript*, e então convertendo esses arquivos para as imagens binárias. Embora as imagens estejam sem ruído, o problema deve ser considerado ruidoso, pois um único padrão 3×3 em 300 dpi pode corresponder a dois ou mais padrões diferentes em 600 dpi.

Vamos utilizar a equação 2.2 para estimar o tamanho necessário das imagens de treinamento para, usando a janela 3×3, obter um WZ-operador $\hat{\Psi}$ com uma taxa de erro no máximo 2% maior que o operador ótimo. Usando nível de confiança 99%:

$$m \geq \frac{1}{2\varepsilon^2}\left[\ln\left(\frac{1}{\delta}\right)+\ln\left(2|H|\right)\right]=\frac{1}{2\times 0{,}02^2}\left[\ln\left(\frac{1}{0{,}01}\right)+\ln(2)+2^9\times\ln(2)\right]\cong 450238\,.$$

Temos dois pares de imagens de amostra independentes ($A^{\mathbf{x}}$, $A^{\mathbf{y}}$) e ($Q^{\mathbf{x}}$, $Q^{\mathbf{y}}$) com caracteres Times 12 pt. (figura 2.6) cujos tamanhos são (554×813, 1108×1626) e (558×740, 1116×1480), respectivamente. Note que a imagem $A^{\mathbf{x}}$ é grande o suficiente para obter a acurácia desejada, pois 554×813=450402. Um WZ-operador foi construído utilizando a aprendizagem 1-NN. O treinamento levou 5s e a aplicação menos que 1s num Pentium 300MHz.

A imagem processada $\hat{Q}^{\mathbf{y}}$ (figura 2.6c) e a imagem ideal $Q^{\mathbf{y}}$ (figura 2.6b) diferiam em 1,058% dos pixels e eles são visualmente bastante semelhantes. Note que na realidade 4 funções características independentes foram projetadas e os seus e-erros individuais estão descritos na primeira linha da tabela 2.1. Uma vez que o e-erro foi medido, pode surgir a seguinte pergunta: “É possível aumentar substancialmente a acurácia do operador projetado?” Utilizaremos as desigualdades (2.5) e (2.6) para

mostrar que é impossível obter qualquer melhora substancial na qualidade do WZ-operador, enquanto janela 3×3 estiver sendo utilizada. Mostraremos que:

1) O e-erro obtido é uma boa estimativa do erro verdadeiro de $\hat{\Psi}$.

2) O erro verdadeiro do operador 3×3 ótimo está muito próximo ao do $\hat{\Psi}$.

Usando equação 2.5, com confiança 99%:

$$\text{t\_error}_P(\hat{\psi}) \leq 0{,}01058 + 2{,}33\sqrt{\frac{0{,}01058(1-0{,}01058)}{1116\times1480}} = (1{,}058 + 0{,}019)\%,$$

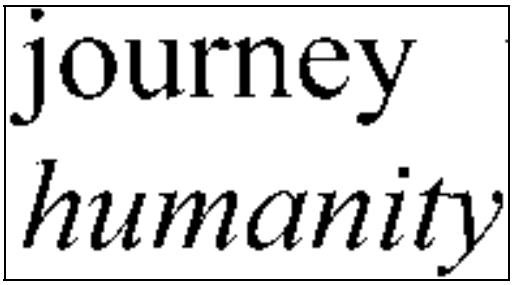
o que demonstra a primeira afirmação.

Para demonstrar a segunda afirmação, projetamos WZ-operador $\hat{\Psi}$* e-ótimo sobre ($Q^{\mathbf{x}}$, $Q^{\mathbf{y}}$) e o aplicamos na imagem $Q^{\mathbf{x}}$. Os e-erros obtidos estão mostrados na segunda linha da tabela 2.1. Usando os dados obtidos e a equação 2.6, concluímos com confiança 99% que:

$$\text{t\_error}_P(\psi^*) \geq 0{,}01045 - 2{,}33\sqrt{\frac{0{,}01045(1-0{,}01045)}{1116\times1480}} = (1{,}045 - 0{,}018)\%.$$

Isto mostra claramente que não pode existir qualquer WZ-operador 3×3 substancialmente melhor que $\hat{\Psi}$ pois, com probabilidade 99%, o erro verdadeiro de $\hat{\Psi}$ é no máximo 1,077% enquanto que com a mesma probabilidade o erro verdadeiro do WZ-operador 3×3 ótimo é pelo menos 1,027%.

Uma vez que demonstramos que o WZ-operador obtido é virtualmente o melhor WZ-operador 3×3, uma outra questão pode surgir: “Poderia melhorar a qualidade do operador escolhendo uma janela maior”? Repetimos os testes utilizando a janela com 17 furos sem peso (figura 2.4). A terceira linha da tabela 2.1 mostra os e-erros obtidos. A qualidade de WZ-operador melhorou somente ligeiramente. Além disso, a linha 4 mostra que, mesmo usando uma janela com 17 furos, o erro mínimo não pode ser substancialmente menor que 0,92%. Desta vez, o treino levou 148s mas a aplicação ainda levou menos de 1s.



(2.6a) Imagem original $Q^{\mathbf{x}}$ em 300 dpi.

(2.6b) Saída ideal $Q^{\mathbf{y}}$ em 600 dpi.

(2.6c) Imagem 600 dpi $\hat{Q}^{\mathbf{y}}$ gerada pela aprendizagem.

Fig. 2.6: Aumento de resolução de caracteres impressos (Times, 12 pt.) usando WZ-operadores projetados pela aprendizagem 1-NN.

	Janela	teste 1	teste 2	teste 3	média
1. Viés aleatório	17	1,638%	1,680%	1,622%	1,647%
2. e1-NN	17 sem peso	1,218%	1,238%	1,174%	1,210%
3. e5-NN	17 sem peso	1,208%	1,234%	1,166%	1,203%
4. e10-NN	17 sem peso	1,206%	1,236%	1,168%	1,203%
5. e20-NN	17 sem peso	1,212%	1,241%	1,166%	1,206%
6. e40-NN	17 sem peso	1,218%	1,244%	1,168%	1,210%
7. e1-NN	17 com peso	1,191%	1,206%	1,143%	1,180%
8. e5-NN	17 com peso	1,180%	1,202%	1,130%	1,171%
9. e10-NN	17 com peso	1,178%	1,199%	1,129%	1,169%
10. e20-NN	17 com peso	1,180%	1,200%	1,124%	1,168%
11. e40-NN	17 com peso	1,184%	1,201%	1,130%	1,172%
12. $\text{e\_error}_{\bar{q}}(\hat{\psi}_i^*)$	17	0,920%	1,012%	0,922%	0,952%
13. Replicação de pixels	-	1,540%	1,670%	1,580%	1,597%

Tab. 2.2: Erros empíricos usados para comparar os diferentes vieses indutivos.

Avaliação do viés indutivo ek-NN

Nesta subsubseção, iremos testar se o viés indutivo da aprendizagem ek-NN é efetivo no aumento da resolução. Para esta finalidade, as diferentes aprendizagens ek-NN foram comparadas com o aprendiz e-ótimo com viés indutivo aleatório. Um aprendiz com viés indutivo aleatório classifica aleatoriamente qualquer padrão não visto. Para tornar evidente as diferenças dos vieses indutivos, pequenas imagens de treinamento (116×516, 232×1032) foram usadas. Por outro lado, as imagens de teste ($Q^{\mathbf{x}}$, $Q^{\mathbf{y}}$) foram razoavelmente grandes (740×558, 1480×1116) para obter estimativas acuradas dos erros verdadeiros. Os testes foram repetidos 3 vezes, cada vez utilizando um conjunto de imagens completamente independente.

Os resultados estão listados na tabela 2.2. A primeira linha apresenta taxas de erro do aprendiz e-ótimo com o viés aleatório. As linhas 2-6 apresentam e-erros das aprendizagens ek-NN para diferentes valores de *k* usando uma janela com 17 furos sem peso, e linhas 7-11 usando janela com 17 furos com peso. A linha 12 é o e-erro do operador e-ótimo sobre as imagens de teste. Finalmente, como mera curiosidade, a linha 13 mostra os e-erros obtidos pela simples replicação de cada pixel quatro vezes.

Para mostrar a eficácia do viés indutivo ek-NN, vamos comparar o seu viés aleatório (linha 1) com o de e1-NN sem peso (linha 2). Note que e1-NN sem peso apresenta o maior taxa de e-erro entre os ek-NN's. A diferença média entre os dois aprendizes foi $\overline{\delta} = 0{,}437\%$. Esta diferença é significativa estatisticamente? Para responder a esta questão, vamos construir um intervalo de confiança. Usando a equação 2.8, com confiança 95%:

$$\underset{\vec{a}\in P^m}{E}\left[\text{t\_error}_P(\mathbf{A}_1(\vec{a})) - \text{t\_error}_P(\mathbf{A}_2(\vec{a}))\right] > (0{,}437 - 0{,}025)\%\,.$$

Isto mostra claramente que o viés indutivo ek-NN ajuda a diminuir a taxa de erro.

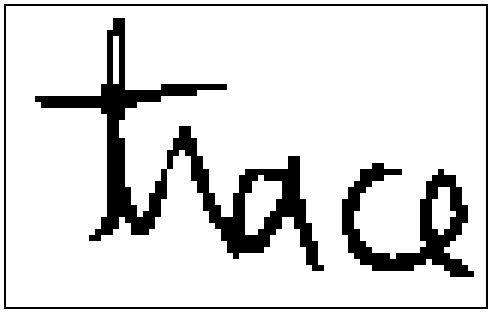
De acordo com a tabela 2.2, parece que as janelas com peso geram menos erros que as janelas sem peso e que o erro torna-se mínimo para $k \cong 10$. Porém, como essas

diferenças são muito pequenas, mais testes são necessários para validar essas suposições.

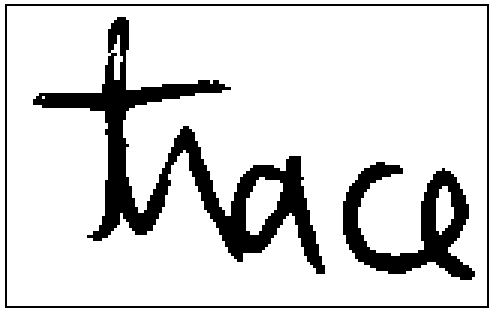
Documentos manuscritos

A técnica acima também foi aplicada para documentos manuscritos (figura 2.7). Os tamanhos das imagens treinamentos ($A^{\mathbf{x}}$, $A^{\mathbf{y}}$) e imagens de teste ($Q^{\mathbf{x}}$, $Q^{\mathbf{y}}$) foram (672×848, 1344×1696). O treino levou 9s usando a janela 3×3 sem peso, enquanto a aplicação levou menos de 1s. A imagem 2.7a é o documento original $Q^{\mathbf{x}}$, 2.7b é a saída ideal $Q^{\mathbf{y}}$ e 2.7c é a imagem processada $\hat{Q}^{\mathbf{y}}$. A diferença entre as imagens $Q^{\mathbf{y}}$ e $\hat{Q}^{\mathbf{y}}$ é 1,14%. O operador 3×3 $\hat{\Psi}^{*}$, e-ótimo sobre imagens de teste, apresentou e-erro de 1,13%. Isto mostra claramente que o WZ-operador projetado é virtualmente o melhor.

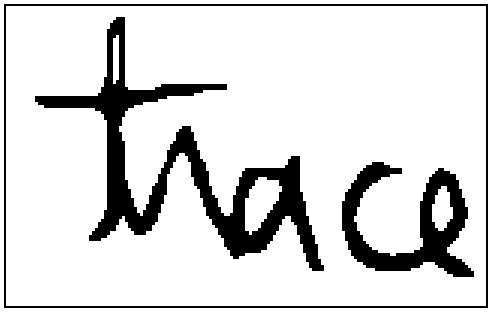
Como uma curiosidade, o WZ-operador 3×3 projetado para aumentar a resolução de documentos impressos foi aplicado em manuscrito 2.7a, gerando a figura 2.7d. O erro foi 1,56%. Isto mostra que o WZ-operador projetado para aumentar a resolução de caracteres impressos não é adequado para aumentar a resolução de documentos manuscritos, pois o erro de 1,56% é consideravelmente maior que 1,14%, obtido com o WZ-operador projetado para ampliar as imagens manuscritas. De um modo geral, a aptidão de um WZ-operador depende do contexto da aplicação.



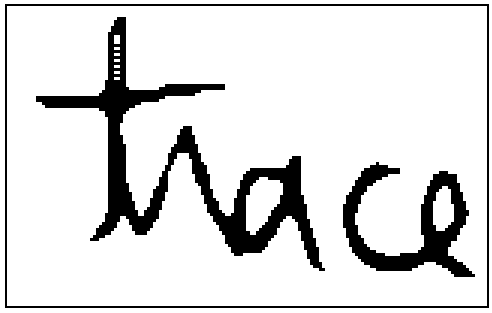
(2.7a) Imagem original $Q^{\mathbf{x}}$.



(2.7b) Imagem de saída ideal $Q^{\mathbf{y}}$ (supostamente desconhecida).



(2.7c) Imagem $\hat{Q}^{\mathbf{y}}$ com resolução aumentada.



(2.7d) Imagem obtida usando operador projetado para aumentar resolução de caracteres impressos.

Fig. 2.7: Aumento de resolução de um documento manuscrito, usando o WZ-operador projetado pela aprendizagem 1-NN.

2.4 Aumento de Resolução de Imagens Meio-Tom

Introdução

Esta seção descreve uma contribuição científica original minha. Os resultados descritos nesta seção estão documentados em artigos [Ri05; Ci05].

A maioria das impressoras jato-de-tinta ou laser atuais na verdade não consegue imprimir as tonalidades de cinza. Elas conseguem imprimir somente pontos minúsculos no papel (dispositivos coloridos não serão considerados aqui). Portanto, qualquer imagem em níveis de cinza deve primeiro ser convertida numa imagem binária por um processo de meio-tom digital antes que a impressão realmente seja efetuada. As técnicas de meio-tom simulam as tonalidades de cinza espalhando quantidades apropriadas de pontos pretos e brancos. Isto é, dada uma imagem em níveis de cinza $G: \mathbb{Z}^2 \to [0,1]$, o meio-tom gera uma imagem binária $B: \mathbb{Z}^2 \to \{0,1\}$ de tal forma que para qualquer pixel *p*:

$$\overline{B}(p) \cong G(p),$$

onde $\overline{B}(p)$ é o valor médio da imagem *B* numa vizinhança em torno do pixel *p*.

Existe uma variedade enorme de técnicas de meio-tom. Os dois mais amplamente conhecidos são a difusão de erro e a excitação ordenada (ordered dithering, abreviada como OD) [Knuth, 1987; Ulichney, 1987]. Existem muitas outras técnicas de meio-tom, por exemplo, a difusão de ponto e as máscaras de ruído azul [Knuth, 1987; Ulichney, 1987]. Algumas delas são projetadas para tecnologias de impressão específicas, para superar as limitações que certas impressoras têm em imprimir pequenos pontos isolados ou os pontos pretos e brancos finamente intercalados.

Muitas tarefas de Processamento e Análise de Imagens são realizadas com operadores restritos à janela (W-operadores). Alguns trabalhos utilizam a abordagem de aprendizagem de máquina para projetar automaticamente um W-operador a partir de imagens amostras de treinamento entrada-saída [Dougherty, 1992a; Dougherty,

1992b; Ri01; Ci01; Cn06; Ri03; Ci02]. Especificamente, propusemos armazenar um W-operador criado pelo processo de aprendizagem de máquina numa estrutura de dados em forma de árvore [Ri01; Cn06]. Aqui, usamos uma idéia similar para projetar o operador de ampliação de imagem (WZ-operador) para aumentar a resolução de imagens binárias meio-tom.

Na literatura, existem muitos artigos sobre o aumento de resolução de imagens em níveis de cinza. Surpreendentemente, somente uns poucos artigos foram escritos sobre a ampliação de imagens binárias [Ri03; Loce and Dougherty, 1997; Loce et al., 1997]. Todas essas técnicas estão baseadas em alguma forma de aprendizagem de máquina e podem ampliar de forma acurada os caracteres impressos ou manuscritos. Além disso, essas técnicas podem ser treinadas para executar algum processamento de imagem simples ao mesmo tempo em que aumenta a resolução. Por exemplo, elas podem atenuar o ruído enquanto aumenta a resolução. Infelizmente, essas técnicas não conseguem levar em conta uma vizinhança ampla para decidir as cores dos pixels com a resolução aumentada, pois os seus tempos de processamento explodem com o aumento dos tamanhos da janela e das imagens exemplos. Uma janela pequena (por exemplo, 3×3 ou 4×4) pode ser bom para ampliar os caracteres impressos ou manuscritos, porém ela não pode ampliar com acurácia as imagens meio-tom. As nossas experiências mostram que janelas do tamanho 8×8 ou 9×9 são necessárias para ampliar com acurácia uma imagem meio-tom.

Esta seção apresenta um algoritmo melhorado para ampliar as imagens binárias baseado em aprendizagem de máquina que permite ampliar de forma acurada até mesmo as imagens meio-tom. A nova técnica está baseada em aprendizagem por árvore de decisão (DT). No conhecimento do autor, esta é a primeira técnica que consegue ampliar direta e com acurácia as imagens meio-tom. A estrutura de dados em forma de árvore permite-nos escrever algoritmos eficientes. A complexidade do tempo de treinamento da nova técnica é somente $O(wm\log m)$, onde w é o tamanho da janela e m é o tamanho da imagem amostra de entrada. A complexidade de aplicação é somente $O(n\log m)$, onde n é o tamanho da imagem a-ser-ampliada. Isto significa que o de-

sempenho deteriora só muito lentamente à medida que os tamanhos da janela e amostras crescem. Esta propriedade torna possível usar as janelas e as imagens amostras grandes. A nova técnica também pode ser usada para ampliar os caracteres impressos ou manuscritos. A nova técnica é incapaz de ampliar com acurácia as imagens geradas pela difusão de erro [Knuth, 1987; Ulichney, 1987], ou por qualquer outro algoritmo de meio-tom onde as cores de saída não são escolhidas como uma função das cores numa vizinhança local. Note que a saída da difusão de erro num pixel particular na realidade depende de todos os pixels previamente processados. Porém, surpreendentemente, a aprendizagem DT pode efetuar o meio-tom inverso acurado das imagens obtidas por difusão de erro, conforme mostramos num artigo recente [Ci11].

O meio-tom inverso é a técnica usada para recuperar a imagem em níveis de cinza a partir de uma imagem binária meio-tom [Wong, 1995; Luo et al., 1998]. O meio-tom inverso simples consiste simplesmente num filtro passa-baixas, por exemplo, um filtro gaussiano. É possível ampliar as imagens meio-tom usando um algoritmo de meio-tom inverso. Porém, a nossa abordagem apresenta uma série de diferenças:

1. Na nossa abordagem, não é necessário ter acesso ao processo de meio-tom em si. É suficiente ter um conjunto de imagens de treinamento entrada-saída. O último é um requerimento mais suave que o primeiro, pois se alguém tiver acesso ao processo de meio-tom, qualquer quantidade de imagens amostras pode ser obtida. O contrário não é verdadeiro.
2. Apesar deste requerimento mais suave, as imagens obtidas pela nova técnica são mais acuradas que aquelas obtidas usando as técnicas de ampliação baseadas em processos de meio-tom inverso simples. Utilizamos o filtro passa-baixas gaussiano e a média local como os processos de meio-tom inverso.
3. Não comparamos o nosso método contra as outras técnicas mais sofisticadas de meio-tom inverso. Porém, demonstramos que a qualidade do nosso processo está bem próxima da melhor qualidade possível de se obter para um processo de ampliação baseada em vizinhança, utilizando a distância de Hamming para quantificar o erro.

Os programas e as imagens usados aqui estão disponíveis em:

http://www.lps.usp.br/~hae/software/halfzoom.

Projeto de WZ-operador pela aprendizagem por árvore de decisão

O operador de ampliação restrito à janela (WZ-operador) foi definido na seção 2.3. Conforme vimos, um WZ-operador pode ser imaginado como um conjunto de f^2 W-operadores (onde f é um fator de ampliação inteiro). O projeto de um WZ-operador é comparável ao projeto de f^2 W-operadores. Assim, um programa computacional que projeta W-operadores pode ser aplicado f^2 vezes para projetar um WZ-operador com fator de ampliação f. Porém, no projeto de WZ-operador, todas as f^2 seqüências de treinamento têm os mesmos padrões de entrada, embora elas normalmente têm diferentes cores de saída. Este fato pode ser explorado para escrever programas mais rápidos e que gastam menos memória, especialmente construídos para o projeto de WZ-operadores.

Para ampliar uma imagem meio-tom usando a aprendizagem DT original, f^2 árvores de decisões independentes devem ser construídas e aplicadas. Isto é uma perda de tempo e de memória computacional. Propomos usar, no projeto de WZ-operadores, uma aprendizagem DT ligeiramente alterada para economizar tempo e espaço, que denominamos de aprendizagem WZDT. A alteração consiste em escolher o atributo de corte $s \in [1...w]$ que torna os dois semi-espaços resultantes a conterem um número tão semelhante quanto possível de pontos de treinamentos (em vez de escolher o atributo que maximiza o ganho de entropia). O novo critério é computacionalmente mais simples que o original. Certamente, o novo teste não é tão bom quanto a maximização do ganho de entropia. Porém, à medida que o tamanho das amostras cresce, os comportamentos das aprendizagens WZDT e DT tornam-se cada vez mais semelhantes. Para amostras grandes, os dois métodos tornam-se inteiramente idênticos (veja os resultados experimentais adiante). Além disso, o novo critério não depende dos valores de saída, enquanto que o critério original depende. Conseqüentemente, usando o

novo critério, todas as f^2 árvores de decisão serão exatamente iguais, exceto pelos seus valores de saída. Assim, uma única árvore de decisão, onde f^2 valores de saída são armazenadas em cada folha, pode representar um WZ-operador. Isto diminui o uso de memória aproximadamente por um favor de f^2. A velocidade também seria melhorada por um fator de f^2. Porém, como o novo critério é computacionalmente mais simples que o original, a aceleração na prática é muito maior que f^2.

Complexidade de amostra e estimação estatística da taxa de erro

Nesta subsubseção, usaremos a teoria de aprendizagem PAC explicada na seção 2.2 para calcular a complexidade de amostra. A acurácia desta complexidade de amostra é então medida utilizando a estimação estatística também explicada na seção 2.2.

Vamos adotar o fator de ampliação $f = 2$ e a janela 4×4. Vamos usar a equação 2.2 para estimar o tamanho necessário *m* da seqüência de treinamento para obter, com nível de confiança 99%, um WZ-operador $\hat{\Psi}$ com o t-erro no máximo 14,5% mais alto que o t-erro do WZ-operador ótimo 4×4 :

$$m \geq \frac{1}{2\varepsilon^2}\left[\ln\left(\frac{1}{\delta}\right) + \ln\left(2|H|\right)\right] = \frac{1}{2\times 0{,}145^2}\left[\ln\left(\frac{1}{0{,}01}\right) + \ln(2) + 2^{16}\times\ln(2)\right] \cong 1{,}1\times 10^6 .$$

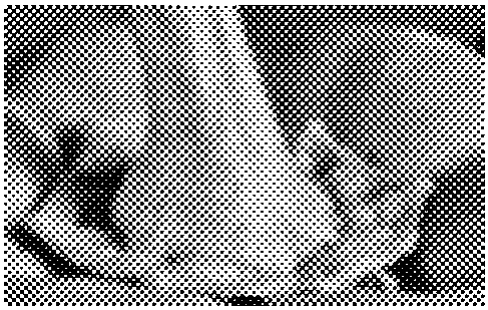
Usamos dois pares de imagens independentes entrada-saída (figuras 2.8a, 2.8b, 2.8c e 2.8d), respectivamente imagens *Peppers* ($A^{\mathbf{x}}$, $A^{\mathbf{y}}$) e *Lena* ($Q^{\mathbf{x}}$, $Q^{\mathbf{y}}$). Elas foram convertidas em imagens meio-tom em 150 e 300 dpi usando o HP LaserJet driver para Microsoft Windows, com a opção “pontos grandes”. $A^{\mathbf{x}}$ e $Q^{\mathbf{x}}$ são 1050×1050, e $A^{\mathbf{y}}$ e $Q^{\mathbf{y}}$ são 2100×2100. Portanto, a imagem $A^{\mathbf{x}}$ é suficientemente grande para produzir a acurácia requerida, pois 1050×1050 ≅ 1,1×10^6. Um WZ-operador $\hat{\Psi}$ foi construído pela aprendizagem WZDT. O treino levou 4s e a aplicação somente 1,2s num Pentium III 1GHz.

Por outro lado, para estabelecer um intervalo estreito para erro, o e-erro de $\hat{\Psi}$ (isto é, a proporção de pixels diferentes entre as imagens 2.8d e 2.8e) foi medido, resultando 7,540%. Usando a equação 2.3, concluímos com 99% de confiança que o t-erro de $\hat{\Psi}$ está contido no intervalo (7,540±0,032)%. Como explicamos anteriormente, o WZ-operador ótimo sobre as imagens de teste pode ser gerado por qualquer aprendiz e-ótimo usando as imagens de teste (Q^x, Q^y) como amostras de treinamento. Assim, usando as figuras 2.8c e 2.8d como amostras de treinamento, o WZ-operador e-ótimo $\hat{\Psi}^*$ foi projetado pela aprendizagem WZDT. Processando a imagem a-ser-ampliada (figura 2.8c) com $\hat{\Psi}^*$, obtivemos a imagem 2.8f. O e-erro desta imagem (a proporção de pixels diferentes entre 2.8d e 2.8f) foi 6,830%. O e-erro do operador e-ótimo $\hat{\Psi}^*$ é uma estimativa do t-erro do WZ-operador verdadeiramente ótimo Ψ^*. Usando a e-quação 2.5, concluímos com 99% de confiança que o t-erro de Ψ^* é pelo menos (6,830-0,028)%. Conseqüentemente, com confiança maior que 99%, o t-erro de $\hat{\Psi}$ é no máximo 0,77% maior que o t-erro do operador verdadeiramente ótimo Ψ^*, isto é:

$$\text{t\_error}_P(\hat{\psi}) - \text{t\_error}_P(\psi^*) \leq 0{,}0077\,.$$

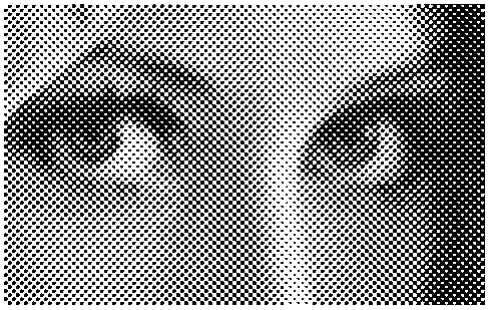
Este resultado confirma que a equação 2.2 produz uma taxa de erro superestimada, pois 0,77% é muito menor que 14,5%. Quanto maior for a janela, a estimativa da complexidade de amostra produzida pela equação 2.2 estará mais inflacionada.

(2.8a) Imagem amostra de entrada $A^{\mathbf{x}}$ em 150 dpi.

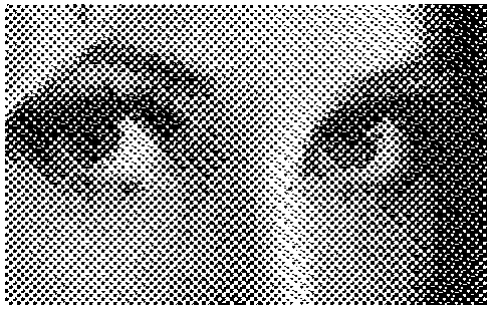


(2.8b) Imagem amostra de saída $A^{\mathbf{y}}$ em 300 dpi.

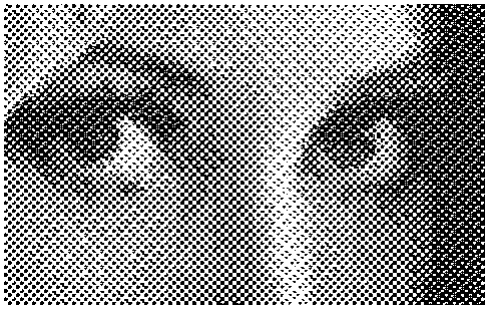
(2.8c) Imagem a-ser-ampliada $Q^{\mathbf{x}}$ em 150 dpi.



(2.8d) Imagem de saída ideal $Q^{\mathbf{y}}$ em 300 dpi.

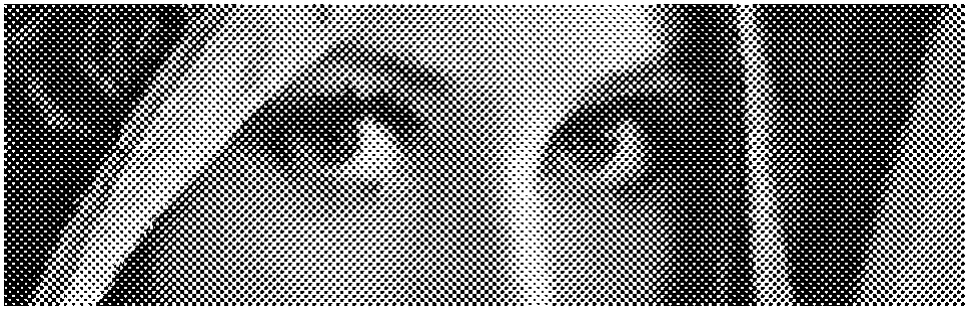


(2.8e) Imagem ampliada $\hat{Q}^{\mathbf{y}}$, usando a janela 4×4. Tamanho das imagens amostras: 1050×1050 e 2100×2100 pixels.



(2.8f) Imagem empiricamente ótima $\hat{\Psi}^{*}(Q^{\mathbf{x}})$, obtido usando a janela 4×4. As imagens de teste (figuras 2.8c e 2.8d) foram usadas como imagens amostras.

Fig. 2.8: Continua na próxima página.



(2.8g) Imagem ampliada $\hat{Q}^{\mathbf{y}}$, usando a janela 8×8. Tamanho das imagens amostras: 9610×1050 e 19220×2100 pixels. A diferença com a saída ideal foi em 1,466% dos pixels.

Fig. 2.8: Aumento de resolução das imagens meio-tom obtidas usando HP LaserJet driver (opção “pontos grandes”) e aprendizagem WZDT.

Escolha de uma janela adequada

Nesta subseção, selecionaremos uma janela apropriada para ampliar as imagens meio-tom na resolução 150 dpi geradas pelo HP LaserJet driver, opção “pontos grandes” (figura 2.8). A janela 4×4 que utilizamos na última subsubseção gerou uma taxa de e-erro excessivamente alta (7,540%). Testamos a aprendizagem WZDT com três conjuntos de imagens completamente independentes usando janelas quadradas de diferentes tamanhos. Os e-erros obtidos podem ser vistos na tabela 2.3. A janela 8×8 gerou o menor e-erro em todos os 3 testes. Isto não causa surpresa, pois o driver da HP provavelmente utiliza o algoritmo de difusão de ponto [Knuth, 1987] definido numa janela 8×8. Assim, parece que a janela 8×8 é a melhor escolha.

Entretanto, alguém poderia perguntar se temos evidências estatísticas para afirmar que a janela 8×8 é a melhor escolha. Usando a equação 2.8, podemos mostrar que, por exemplo, a janela 8×8 é melhor que 10×10. Podemos concluir, com 95% de confiança, que a diferença esperada entre os dois t-erros é pelo menos 0,096%, quando as amostras de tamanho m = 1050×1050 são usadas, isto é:

$$\mathop{\mathrm{E}}_{\vec{a}\in P^m}\left[\mathrm{t\_error}_P(\hat{\psi}_{8\times 8}) - \mathrm{t\_error}_P(\hat{\psi}_{10\times 10})\right] > 0{,}00096\,.$$

Porém, não podemos afirmar que a janela 8×8 seja melhor que 9×9 com 95% de confiança. Mais dados devem ser coletados para obter informação suficiente para formar uma evidência estatística.

Comparação de diferentes vieses indutivos

Nesta subsubseção, comparamos os diferentes vieses indutivos. Executamos 11 testes com as aprendizagens WZDT, DT original, e5-NN, e1-NN e o viés indutivo aleatório, sempre usando a janela 4×4. Não foi possível executar os testes usando uma janela maior (por exemplo, 8×8) pois a aprendizagem ek-NN é excessivamente lento: de acordo com as nossas estimativas, levaria 6 dias para executar um teste usando o algoritmo força bruta e 100 milhões de anos usando a implementação por look-up-

table. Para tornar as diferenças evidentes, pequenas imagens de treinamento ($A^{\mathbf{x}}$, $A^{\mathbf{y}}$) foram usadas (100×100, 200×200). Por outro lado, as imagens de teste ($Q^{\mathbf{x}}$, $Q^{\mathbf{y}}$) foram grandes (1050×1050, 2100×2100) para obter uma estimativa de t-erro acurada.

Os resultados estão ilustrados na tabela 2.4. O erro médio da aprendizagem WZDT é mais alto que os erros dos outros 3 algoritmos (DT original, e5-NN e e1-NN). Este resultado era esperado, pois escolhemos a aprendizagem WZDT devido ao seu desempenho computacional, com o sacrifício resultante da acurácia do WZ-operador obtido. Podemos executar testes para decidir se as diferenças observadas nas taxas de erro são estatisticamente significativas. Por exemplo, usando a equação 2.8, pode-se mostrar com 95% de confiança que a diferença esperada entre as taxas de t-erro dos algoritmos de WZDT e e1-NN é pelo menos 0,330%, usando m = 10000 exemplos de treinamento. Porém, um resultado similar não pode ser derivado para a diferença esperada entre os t-erros dos métodos de aprendizagem WZDT e DT. As diferenças entre os erros tende a desaparecer à medida que o tamanho das amostras cresce, conforme mostraremos na próxima subsubseção.

Por outro lado, os e-erros da aprendizagem WZDT é notavelmente menor que os e-erros do viés indutivo aleatório. Usando novamente a equação 2.8, pode-se mostrar com 95% de confiança que é esperado que o viés indutivo aleatório cometa pelo menos 1,442% mais erros que a aprendizagem WZDT, usando m = 10000 exemplos de treinamento. Isto mostra claramente que o viés indutivo da aprendizagem WZDT ajuda a diminuir a taxa de erro, mesmo que ele não seja tão efetivo quanto os outros vieses indutivos computacionalmente mais caros.

	4×4	5×5	6×6	7×7	8×8	9×9	10×10	11×11
teste 1 (%)	7,540	2,644	2,025	1,806	1,710	1,772	1,870	2,013
teste 2 (%)	6,105	2,431	2,330	2,374	2,249	2,305	2,486	2,616
teste 3 (%)	7,546	3,676	2,589	2,359	2,354	2,525	2,688	2,740
média (%)	7,064	2,917	2,315	2,180	2,104	2,201	2,348	2,456

Tab. 2.3: Erros empíricos obtidos usando a aprendizagem WZDT com janelas de diferentes tamanhos. Os tamanhos das imagens amostras foram 1050×1050 e 2100×2100 pixels.

	aprendizagem WZDT	aprendizagem original DT	aprendizagem e5-NN	aprendizagem e1-NN	viés indutivo aleatório
teste 1 (%)	8,699	8,664	8,592	8,602	9,262
teste 2 (%)	14,647	14,897	13,791	13,604	15,924
teste 3 (%)	11,541	11,796	11,563	11,471	13,150
teste 4 (%)	11,861	11,263	11,707	11,771	13,961
teste 5 (%)	14,700	13,434	15,922	13,925	18,349
teste 6 (%)	10,699	10,295	10,403	10,386	11,200
teste 7 (%)	14,439	14,436	13,384	13,898	17,369
teste 8 (%)	12,644	12,677	12,109	12,004	13,891
teste 9 (%)	14,483	13,965	13,675	13,995	17,660
teste 10 (%)	17,521	17,951	15,996	16,425	22,370
teste 11 (%)	22,523	20,982	20,267	20,362	24,607
média (%)	13,978	13,669	13,401	13,313	16,158

Tab. 2.4: Erros empíricos dos diferentes algoritmos de aprendizagem. Uma janela 4×4 e imagens amostras com 100×100 e 200×200 pixels foram usadas.

		1050×1050 (1 par)	3190×1050 (3 pares)	6400×1050 (6 pares)	9610×1050 (9 pares)	1050×1050 (e-ótimo)
aprendizagem WZTD	e-erro (%)	1,710	1,561	1,494	1,466	1,111
	treino (s)	11,04	37,35	84,36	146,11	10,49
	aplicação (s)	1,64	2,47	3,35	4,01	1,65
aprendizagem DT original	e-erro (%)	1,617	1,547	1,489	1,464	1,111
	treino (s)	538	2320	7×10^3	128×10^3	540
	aplicação (s)	2,9	3,9	4,0	4,8	3,1

Tab. 2.5: Os erros empíricos diminuem à medida que os tamanhos das amostras crescem. A última coluna mostra o erro do WZ-operador empiricamente ótimo, obtido usando as imagens de teste como as amostras de treinamento. A janela 8×8 foi usada.

Algoritmos de aprendizagem DT e WZDT

Nesta subsubseção, examinaremos cuidadosamente a variação do e-erro à medida que o número de exemplos de treinamento cresce, para obter o melhor WZ-operador possível. Executaremos todos os testes usando a janela 8×8, pois ela parece ser a melhor para a aplicação que estamos estudando. Testaremos somente as aprendizagens WZDT e DT, pois a aprendizagem ek-NN é excessivamente lenta para poder testar.

A tabela 2.5 mostra os resultados experimentais. Usamos, como as imagens amostras, 1, 3, 6 e 9 pares de imagens com (1050×1050, 2100×2100) pixels, grudadas horizontalmente mas separadas por algumas colunas brancas. O par de imagens de teste foi "Lena", com (1050×1050, 2100×2100) pixels (obviamente, o conjunto de imagens de treinamento não incluiu "Lena"). Como era esperado, os e-erros diminuíram à medida que o tamanho das amostras cresceu. Porém, os e-erros diminuíram muito pouco de 6 para 9 pares de imagens amostras, sugerindo que provavelmente já há uma quantidade suficiente de amostras de treinamento e o erro deve estar convergindo a algum limite inferior.

À medida que o tamanho das imagens amostras cresce, as diferenças entre as aprendizagens WZDT e DT diminuem. Para imagens amostras grandes (9 pares de imagens 1050×1050 e 2100×2100), as duas taxas de erro são praticamente idênticas: 1,466% e 1,464%. Porém, o treino da aprendizagem DT original leva 870 vezes mais tempo do que a aprendizagem WZDT. Portanto, na prática, a aprendizagem WZDT é o melhor algoritmo para ser usado para o projeto de WZ-operadores.

O melhor e-erro obtido pela aprendizagem WZDT é 1,466% (a penúltima coluna da tabela 2.5) e o menor e-erro possível é 1,111% (a última coluna da tabela 2.5). Usando as equações 2.4 e 2.5, concluímos com 95% de confiança que o t-erro do operador obtido é no máximo (1,466+0,009)% e o t-erro do operador verdadeiramente ótimo é pelo menos (1,111–0,008)%. Muito provavelmente, este limite inferior está subestimado. Para obter o menor e-erro, supusemos que a imagem de saída ideal estava disponível durante o estágio de treinamento. Isto não acontece numa situação real. As-

sim, o operador obtido pode ser considerado muito próximo do operador ótimo com respeito à distância de Hamming.

Ampliação baseada em meio-tom inverso

Nesta subsubseção, compararemos a aprendizagem WZDT com as ampliações baseadas em meio-tom inverso. As nossas experiências mostram que a aprendizagem WZDT é consideravelmente mais acurada que as ampliações baseadas em meio-tom inverso simples. Uma ampliação baseada em meio-tom inverso pode ser descrita como:

1. Dada uma imagem meio-tom *B*, use algum algoritmo de meio-tom inverso para obter a imagem em níveis de cinza *G* correspondente. Testamos dois filtros passa-baixas como os algoritmos de meio-tom inverso: o filtro gaussiano e a média móvel.
2. Aumente a resolução da imagem *G* usando alguma técnica de ampliação de imagem em níveis de cinza, obtendo a imagem ampliada *G*'. Utilizamos a interpolação linear como a técnica de ampliação em níveis de cinza.
3. Aplique o algoritmo de meio-tom à imagem *G*', para obter a imagem meio-tom ampliada *B*'.

Os e-erros obtidos estão listados na tabela 2.6. O menor e-erro foi 1,929% usando o filtro gaussiano e 1,947% usando o filtro média móvel. Ambas taxas de erro são consideravelmente mais altas que 1,466%, que é a menor taxa de erro obtida usando a aprendizagem WZDT. Os testes foram repetidos mais duas vezes usando imagens diferentes e resultados similares foram obtidos.

	Desvio-padrão da gaussiana (pixels) / tamanho da janela da média móvel (pixels)	erro empírico (%)
Meio-tom inverso usando a filtragem por gaussiana	2,0	3,192
	2,3	2,144
	2,5	1,962
	2,8	1,929
	3,0	2,012
	3,5	2,107
	4,0	2,286
Meio-tom inverso usando a filtragem média móvel	7×7	2,875
	8×8	1,947
	9×9	2,470

Tab. 2.6: Erros empíricos observados utilizando a ampliação baseada em meio-tom inverso.

Mais dados experimentais

Nesta subsubseção, aplicaremos a aprendizagem WZDT para ampliar as imagens meio-tom geradas por diferentes técnicas de meio-tom.

A figura 2.9 mostra a ampliação de imagens meio-tom geradas pela HP LaserJet driver, opção “pontos pequenos”. O melhor operador foi obtido usando a janela 8×8 e um par de imagens amostras com (9610×1050, 19220×2100) pixels. Aplicando-o à imagem “Lena”, a imagem processada apresentou uma taxa de e-erro 1,429%.

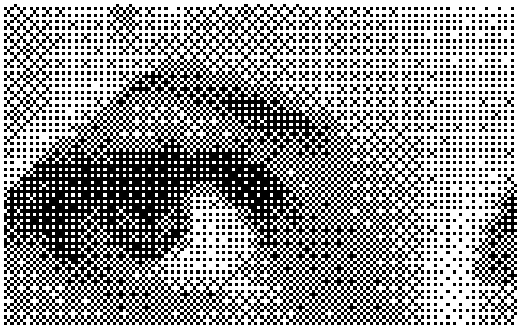
A figura 2.10 mostra a ampliação de imagens meio-tom geradas pelo algoritmo de excitação ordenada pontos aglutinados (clustered-dot ordered dithering) incluído no programa “Image Alchemy” de “Handmade Software, Inc”. A imagem processada tinha uma taxa de e-erro de 1,387%.

As imagens de entrada e saída não necessariamente devem usar a mesma técnica de meio-tom. Por exemplo, podemos usar imagens meio-tom 150 dpi geradas pela HP driver “pontos grandes” como entrada e imagens 300 dpi geradas pelo algoritmo de excitação ordenada pontos aglutinados como saída. Neste caso, a aprendizagem WZDT converte uma técnica de meio-tom numa outra ao mesmo tempo em que se aumenta a resolução. Testamos esta idéia e a imagem processada tinha uma taxa de e-erro de 1,494%. Também testamos o inverso: a conversão de uma imagem meio-tom 150 dpi gerada pela excitação ordenada pontos aglutinados na imagem meio-tom 300 dpi tipo HP pontos grandes. O e-erro resultante foi 1,687%.

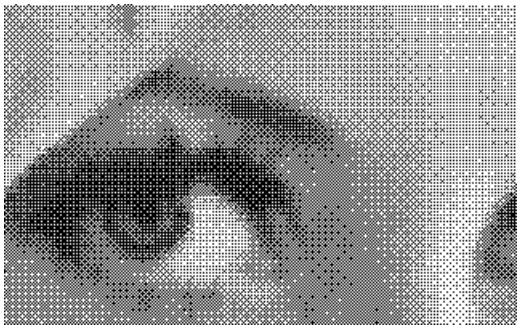
Finalmente, a aprendizagem WZDT foi aplicada para aumentar a resolução de imagens obtidas usando o algoritmo de difusão de erro. Infelizmente, resultados muito ruins foram obtidos. Usando driver HP opção “difusão de erro”, obtivemos uma taxa de e-erro de 12,90%. Usando o algoritmo de Floyd-Steinberg do programa Image Alchemy, obtivemos o e-erro de 42,77% (“algoritmo de difusão de erro” e “algoritmo de Floyd Steinberg” são sinônimos). Estes altos erros eram esperados, pois o algoritmo de difusão de erro não escolhe uma cor de saída em função das cores de uma vi-

zinhança local. Porém, surpreendentemente, a aprendizagem DT pode efetuar o meio-tom inverso com acurácia [Ci11].

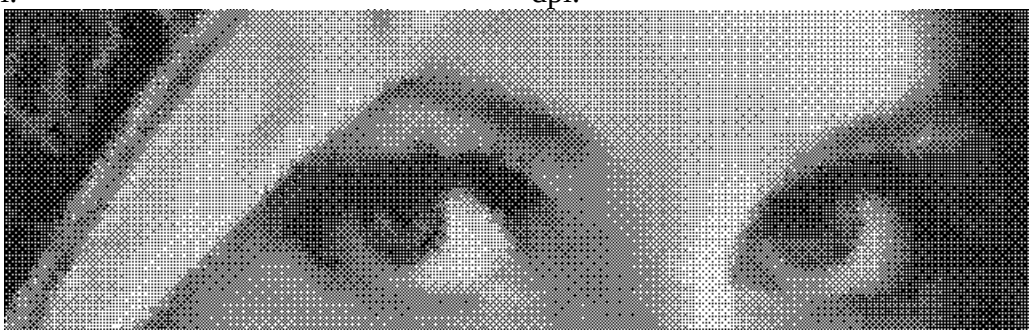
Uma imagem gerada pela difusão de erro pode ser ampliada por um processo de ampliação baseada em meio-tom inverso, resultando numa imagem com uma qualidade visual razoável. Porém, o e-erro resultante é muito alto. Uma imagem meio-tom gerada pela difusão de erro foi convertida numa imagem em níveis de cinza usando um filtro gaussiano com desvio padrão de 2,8 pixels. A imagem em níveis de cinza resultante foi ampliada e convertida novamente numa imagem meio-tom pela difusão de erro. O e-erro obtido foi 43,25%, embora a qualidade visual seja razoável (compare com 42,77% obtido com a aprendizagem WZDT). A distância de Hamming parece não ser uma medida apropriada para quantificar a qualidade das imagens produzidas por processos onde podem ocorrer “deslocamentos de fase”.



(2.9a) Imagem a-ser-ampliada $Q^{\mathbf{x}}$ em 150 dpi.



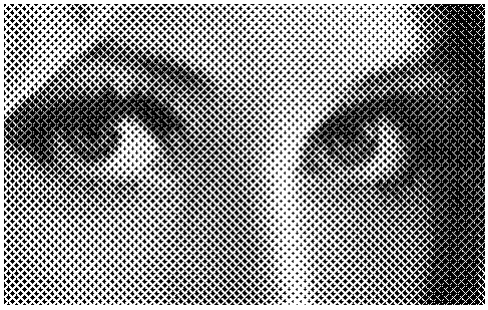
(2.9b) Imagem de saída ideal $Q^{\mathbf{y}}$ em 300 dpi.



(2.9c) Imagem ampliada $\hat{Q}^{\mathbf{y}}$, usando a janela 8×8. Os tamanhos das amostras de treinamento foram 9610×1050 e 19220×2100 pixels. A diferença para a saída ideal foi em 1,429% dos pixels.

Fig. 2.9: Aumento de resolução de imagens meio-tom geradas pela HP LaserJet driver, opção “pontos pequenos”, usando a aprendizagem WZDT.

(2.10a) Imagem a-ser-ampliada $Q^{\mathbf{x}}$ em 150 dpi.



(2.10b) Imagem de saída ideal $Q^{\mathbf{y}}$ em 300 dpi.

(2.10c) Imagem ampliada $\hat{Q}^{\mathbf{y}}$, usando a janela 8×8. Os tamanhos das imagens de treinamento foram 9610×1050 e 19220×2100 pixels. A diferença para a saída ideal foi em 1,387% dos pixels.

Fig. 2.10: Aumento de resolução das imagens meio-tom geradas pelo algoritmo de excitação ordenada, pontos aglutinados, usando a aprendizagem WZDT.

2.5 Meio-Tom Inverso pela Aprendizagem

Introdução

Esta seção descreve uma contribuição científica original minha. Os resultados descritos nesta seção estão documentados no artigo [Ci11].

A maioria de impressoras jato-de-tinta ou laser pode imprimir somente minúsculos pontos pretos sobre o papel. Assim, qualquer imagem em níveis de cinzas deve ser primeiro convertida numa imagem binária antes que a impressão seja efetuada. O processo de meio-tom simula os níveis de cinza distribuindo apropriadamente pixels pretos e brancos. As técnicas de meio-tom populares são a difusão de erro, a excitação ordenada (ordered dithering) e as máscaras de ruído azul [Roetling and Loce, 1994; Ulichney, 1998]. As imagens meio-tom podem ser ou ortográficas (dados digitais perfeitos antes da impressão) ou escaneadas.

O meio-tom inverso (em inglês, inverse halftoning ou descreening, abreviado aqui como IH) é o processo para recuperar a imagem em níveis de cinzas a partir da imagem meio-tom. Como meio-tom é um processo “muitos para um”, não existe uma única imagem em níveis de cinza para uma dada imagem meio-tom. Assim, outras propriedades das imagens devem ser utilizadas em IH. O método IH mais simples é um filtro passa-baixas. Embora este processo produza as imagens em níveis de cinza, borra as arestas e destrói os detalhes finos. Muitos métodos diferentes foram desenvolvidos para melhorar IH por filtragem passa baixas. Veja Luo et al. [Luo et al., 1998] para mais detalhes.

Mese e Vaidyanathan propuseram recentemente uma abordagem baseada na aprendizagem de máquina para fazer o meio-tom inverso das imagens ortográficas. Eles primeiro propuseram um algoritmo que usa look-up-table (LUT) [Mese and Vaidyanathan, 2001] e depois um outro que usa uma estrutura de árvore [Mese and Vaidya-

nathan, 2002]. Em ambos os métodos, existe uma fase de treinamento onde as imagens amostras são utilizadas para construir a estrutura de dados.

A aprendizagem de máquina tem sido usada em muitas aplicações de processamento de imagens. Em particular, pode-se usá-la para aumentar a resolução de imagens meio-tom usando uma LUT [Ri03; Ci02] ou uma árvore de decisão (DT) [Ci05; Ri05]. Estas técnicas podem ser adaptadas de forma direta para o problema IH.

Em [Mese and Vaidyanathan, 2002], um algoritmo de aprendizagem de máquina com uma estrutura de árvore "ad hoc" é empregado. Nós, ao contrário, propomos usar uma abordagem teórica e algorítmica baseada em aprendizagem DT para resolver IH. Mais especificamente, usamos uma versão da aprendizagem DT chamada ID3. Esta abordagem apresenta algumas vantagens sobre a proposta de [Mese and Vaidyanathan, 2002]. Há duas razões para isso. Primeiro, em [Mese and Vaidyanathan, 2002], o usuário deve selecionar cuidadosamente a máscara (isto é, a janela). Eles até apresentam um algoritmo para selecionar uma "boa" janela [Mese and Vaidyanathan, 2001]. A aprendizagem DT isenta o usuário de ter de escolher explicitamente uma janela, porque irá usar automaticamente somente aqueles atributos (furos de espiar, peepholes ou pixels) que são mais necessários para decidir a cor de saída em níveis de cinzas. Enquanto eles usaram janelas com no máximo 19 furos (que pode ser pequena demais para muitos problemas de IH), usando a aprendizagem DT, optamos por uma janela inicial muito maior (por exemplo, 8×8=64 pixels). O algoritmo de aprendizagem por si irá usar automaticamente somente os furos apropriados na ordem apropriada. A escolha dos furos está baseada na maximização de redução esperada da entropia, e esta política tem produzido árvores de alturas mais baixas e com boa capacidade de generalização [Mitchell, 1997; Quinlan, 1986]. Além disso, Mese e Vaidyanathan usam uma estrutura de dados não-ortodoxa (apesar de interessante) que combina a LUT e a árvore binária. Embora não há nada de errado com esta abordagem, a árvore de decisão original é claramente mais elegante, e apresenta um desempenho computacional equivalente.

Acreditamos que [Mese and Vaidyanathan, 2002] possua um desempenho estado de arte e que nosso algoritmo seja o próximo melhoramento natural, pois obtivemos imagens com PSNR que parecem estar 4 dB acima daquelas relatadas em [Mese and Vaidyanathan, 2002]. Os programas e as imagens usados aqui estão disponíveis em:

http://www.lps.usp.br/~hae/software/invhalf.

O problema

As imagens binária e em níveis de cinzas são definidas respectivamente como funções $Q^{\mathbf{x}}: \mathbb{Z}^2 \to \{0,1\}$ e $Q^{\mathbf{y}}: \mathbb{Z}^2 \to [0...255]$. O suporte de uma imagem é um subconjunto finito de $\mathbb{Z}^2$, onde a imagem está realmente definida.

Um operador restrito à janela Ψ "binário para níveis de cinza" é uma função que mapeia uma imagem binária $Q^{\mathbf{x}}$ numa imagem em níveis de cinza $Q^{\mathbf{y}}$. Ele é definido através de um conjunto de w pontos chamada janela $W = \{W_1, \ldots, W_w\}$, $W_i \in \mathbb{Z}^2$, e uma função característica $\psi : \{0,1\}^w \to [0...255]$ como segue:

$$Q^{\mathbf{y}}(p) = \Psi(Q^{\mathbf{x}})(p) = \psi(Q^{\mathbf{x}}(W_1 + p), \ldots, Q^{\mathbf{x}}(W_w + p)),$$

onde $p \in \mathbb{Z}^2$. Cada elemento W_i da janela é chamado furo de espiar (peephole) ou atributo (feature).

Sejam $A^{\mathbf{x}}$, $A^{\mathbf{y}}$, $Q^{\mathbf{x}}$ e $Q^{\mathbf{y}}$ respectivamente as imagens amostra de entrada, amostra de saída, a-ser-processada e ideal (esta supostamente desconhecida). Podemos supor que existe um único par de imagens de treinamento ($A^{\mathbf{x}}$ e $A^{\mathbf{y}}$). Se existirem mais pares, eles podem ser colados para formar um único par.

Vamos denotar o conteúdo em $A^{\mathbf{x}}$ da janela W deslocada para $p \in \mathbb{Z}^2$ como $a_p^{\mathbf{x}}$ e chamá-lo de instância de treinamento ou padrão de entrada de treinamento no pixel p:

$$a_p^{\mathbf{x}} = \left(A^{\mathbf{x}}(W_1 + p), A^{\mathbf{x}}(W_2 + p), \ldots, A^{\mathbf{x}}(W_w + p)\right) \in \{0,1\}^w.$$

A cada padrão $a_p^{\mathbf{x}}$, está associada uma cor de saída $A^{\mathbf{y}}(p) \in [0...255]$. Vamos denotar o conjunto obtido quando todos os pixels de $A^{\mathbf{x}}$ e $A^{\mathbf{y}}$ são varridos por

$$a = \left\{(a^{\mathbf{x}}_{p_1}, A^{\mathbf{y}}(p_1)), \ldots, (a^{\mathbf{x}}_{p_m}, A^{\mathbf{y}}(p_m))\right\}$$

e chamá-lo de conjunto amostra ou conjunto de treinamento (*m* é a quantidade de pixels das imagens $A^{\mathbf{x}}$ e $A^{\mathbf{y}}$). Vamos construir de forma similar o conjunto

$$q = \left\{(q^{\mathbf{x}}_{p_1}, Q^{\mathbf{y}}(p_1)), \ldots, (q^{\mathbf{x}}_{p_n}, Q^{\mathbf{y}}(p_n))\right\}$$

a partir das imagens $Q^{\mathbf{x}}$ e $Q^{\mathbf{y}}$ (*n* é a quantidade de pixels de $Q^{\mathbf{x}}$ e $Q^{\mathbf{y}}$). Cada padrão $q^{\mathbf{x}}_{p_i}$ é chamado de um padrão de busca ou uma instância a ser processada e a cor $Q^{\mathbf{y}}(p_i) \in [0...255]$ é chamada de cor de saída ideal.

No problema de IH, um algoritmo de aprendizagem **A** constrói um operador $\hat{\Psi}$ baseado em $A^{\mathbf{x}}$ e $A^{\mathbf{y}}$ tal que, quando $\hat{\Psi}$ é aplicado a $Q^{\mathbf{x}}$, espera-se que a imagem processada resultante $\hat{Q}^{\mathbf{y}} = \hat{\Psi}(Q^{\mathbf{x}})$ seja similar à imagem de saída ideal $Q^{\mathbf{y}}$. Para descrever este processo de forma mais precisa, vamos definir uma função de perda (ou erro) *l* que será usada para medir a diferença entre as saídas ideal e processada. Exemplos de possíveis funções de perda são:

Perda quadrática: $l(Q^{\mathbf{y}}(p), \hat{Q}^{\mathbf{y}}(p)) = (Q^{\mathbf{y}}(p) - \hat{Q}^{\mathbf{y}}(p))^2$

Perda absoluta: $l(Q^{\mathbf{y}}(p), \hat{Q}^{\mathbf{y}}(p)) = \left|Q^{\mathbf{y}}(p) - \hat{Q}^{\mathbf{y}}(p)\right|$.

Re-enunciando o problema IH, o aprendiz **A** deve construir uma função característica ou hipótese $\hat{\psi}$ baseada em conjunto amostra *a* tal que, quando $\hat{\psi}$ é aplicado a um padrão de busca $q^{\mathbf{x}}_{p_i}$ gerando a cor de saída $\hat{Q}^{\mathbf{y}}(p_i) = \hat{\psi}(q^{\mathbf{x}}_{p_i})$, a perda $l\left(Q^{\mathbf{y}}(p_i), \hat{Q}^{\mathbf{y}}(p_i)\right)$ deve ser baixa com alta probabilidade.

Algoritmos

Existem muitas técnicas de aprendizagem que poderiam ser usadas para conseguir o objetivo acima. Porém, para ser realmente útil, a técnica deve apresentar as seguintes características:

1. Ela deve generalizar para além do conjunto de treinamento. Isto é, a técnica deve gerar saídas com pequena perda não somente para os padrões do conjunto de treinamento mas também para outros padrões nunca vistos.
2. Ela deve ser rápida no estágio de aplicação do operador.
3. O espaço de memória necessário deve ser moderado.
4. A fase de treinamento não deve ser muito lenta, embora alguma lentidão possa ser tolerável.

Felizmente, a aprendizagem DT satisfaz todos esses requerimentos. Descrevemos brevemente abaixo o algoritmo de construção DT. Existem muitas versões de aprendizagem DT. Nesta seção, usamos o algoritmo ID3 [Mitchell, 1997; Quinlan, 1986]. No processo de geração de DT, o espaço de entrada $\{0,1\}^w$ é particionado em duas metades, e todos os padrões de treinamento com cor preta no atributo W_s irão pertencer a um semi-espaço e aqueles com cor branca a outro. A dimensão dos semi-espaços assim obtidos é um a menos que o espaço original, isto é, $\{0,1\}^{w-1}$. Para cada um dos dois semi-espaços obtidos, o processo que partição continua recursivamente, gerando espaços cada vez menores. Em cada partição, um nó interno é criado e o atributo *s* da partição é armazenado. Este processo pára quando cada espaço contiver ou somente amostras com a mesma cor de saída ou somente amostras com o mesmo padrão de entrada (mas com dois ou mais diferentes cores de saída). No primeiro caso, um nó terminal é criado e a cor de saída é armazenada nele. O segundo caso é chamado de conflito. Neste caso, um nó terminal é criado, mas a média dos valores de saída é calculada e armazenada (se a perda quadrática ou PSNR deve ser minimizada). A média deve ser substituída pela mediana se a perda absoluta deve ser minimizada.

Se não existem conflitos, o algoritmo acima deve classificar perfeitamente o conjunto de treinamento. Porém, os pesquisadores de aprendizagem de máquina observaram que esta estratégia pode levar a “superencaixamento” (overfitting). Dizemos que uma hipótese está superencaixada nos exemplos de treinamento se alguma outra hipótese que se encaixa mais pobremente nos exemplos de treinamento possui um desempe-

nho melhor sobre a distribuição global das instâncias. Na literatura, existem algumas estratégias sofisticadas para evitar o superencaixamento. Porém, usando-as, a fase de treinamento torna-se excessivamente demorada. Assim, usamos a seguinte estratégia simples: a média (ou mediana) é calculada toda vez que existirem *k* ou menos amostras num subespaço dos padrões. Denotaremos esta estratégia como *k*-ID3, por exemplo, 1-ID3 ou 10-ID3. Quando $k = 1$, temos o algoritmo ID3 original.

A questão central no algoritmo ID3 é como selecionar o atributo a ser usado para particionar o espaço dos padrões em cada nó interno da árvore. Diferentes escolhas gerarão diferentes árvores de decisão. Dadas duas ou mais DTs, é amplamente aceito que a mais simples (ou seja, a árvore de altura mais baixa) deve ser a preferida. Esta escolha é conhecida como "navalha de Occam" e muitos estudos apontam a sua superioridade, incluindo a nossa própria experiência com o problema IH. Uma maneira de implementar a "navalha de Occam" seria gerar todas as possíveis DTs e selecionar a mais baixa entre elas. Claramente, esta abordagem é impraticável pois levaria um tempo excessivamente longo.

O algoritmo ID3 utiliza o seguinte critério que segue de perto a "navalha de Occam". Consiste em colocar os atributos com alto ganho de informação mais perto da raiz. Vamos definir a entropia de um conjunto de amostras *a* onde cada amostra pode assumir um entre *c* diferentes valores de saída:

$$\text{Entropy}(a) \equiv \sum_{i=1}^{c} - p_i \log_2 p_i$$

onde p_i é a proporção de exemplos de *a* que pertence à classe *i*. Para o problema de IH, discretizamos (isto é, quantizamos) 256 possíveis valores em $c = 16$ categorias: [0...15], [16...31], etc. Esta discretização é usada somente para calcular a entropia. O valor preciso da saída é ainda armazenado nas folhas da DT. O ganho de informação, definido abaixo, é a redução esperada da entropia causada pela partição dos exemplos de acordo com o atributo *s*:

$$\text{Gain}(a, s) \equiv \text{Entropy}(a) - \sum_{v \in \{0,1\}} \frac{|a_v|}{|a|} \text{Entropy}(a_v)$$

onde a_v é o subconjunto de a na qual o atributo s tem valor v. Em cada nó interno, o algoritmo de aprendizagem escolhe o atributo que maximiza o ganho de informação.

Depois que a DT tiver sido construída, a sua aplicação é direta: dado um padrão de busca $q_p^{\mathbf{x}}$, a DT é percorrida de cima para baixo, até chegar numa folha. A informação contida nesta folha é então escolhida como o valor de saída $\hat{Q}^{\mathbf{y}}(p_i)$.

Denominaremos de "algoritmo seqüencial" o algoritmo de aprendizagem DT que escolhe os furos seqüencialmente numa ordem pré-determinada, sem usar o critério de maximização de ganho de entropia. Para verificar a eficácia do viés indutivo de maximização de ganho de entropia, compararemos os algoritmos seqüencial e ID3.

Resultados e Dados Experimentais

A técnica proposta foi implementada e testada. Usamos três pares de imagens com 1050×1050 pixels para o treinamento. Aplicamos o sistema IH resultante em três imagens 1050×1050 completamente independentes. Usamos a janela 8×8 em todos os casos. Os testes foram repetidos para 4 diferentes tipos de meio-tom:

1. Difusão de erro (algoritmo de Floyd-Steinberg).
2. Excitação ordenada, pontos dispersos (algoritmo de Bayer).
3. Excitação ordenada, pontos aglutinados.
4. O meio-tom do HP LaserJet driver para Windows, opção "pontos grandes".

A figura 2.11 mostra a aplicação da técnica proposta a imagens meio-tom obtidas pela difusão de erro. Parte de uma das imagens de treinamento é mostrada nas figuras 2.11a e 2.11b. Parte de uma das imagens a serem processadas é mostrada na figura 2.11c e a correspondente saída ideal (supostamente desconhecida) na figura 2.11d. As figuras 2.11e e 2.11f são as imagens obtidas usando algoritmos seqüencial e 10-ID3, respectivamente. As suas PSNRs são 26,75 e 34,75 dB para a imagem Lena. Isto demonstra que a maximização de ganho de informação desempenha um papel importante em melhorar a qualidade do sistema IH obtido. Aplicando o sistema IH a 3 imagens de teste diferentes (uma das quais é Lena), podemos ver que a PSNR varia

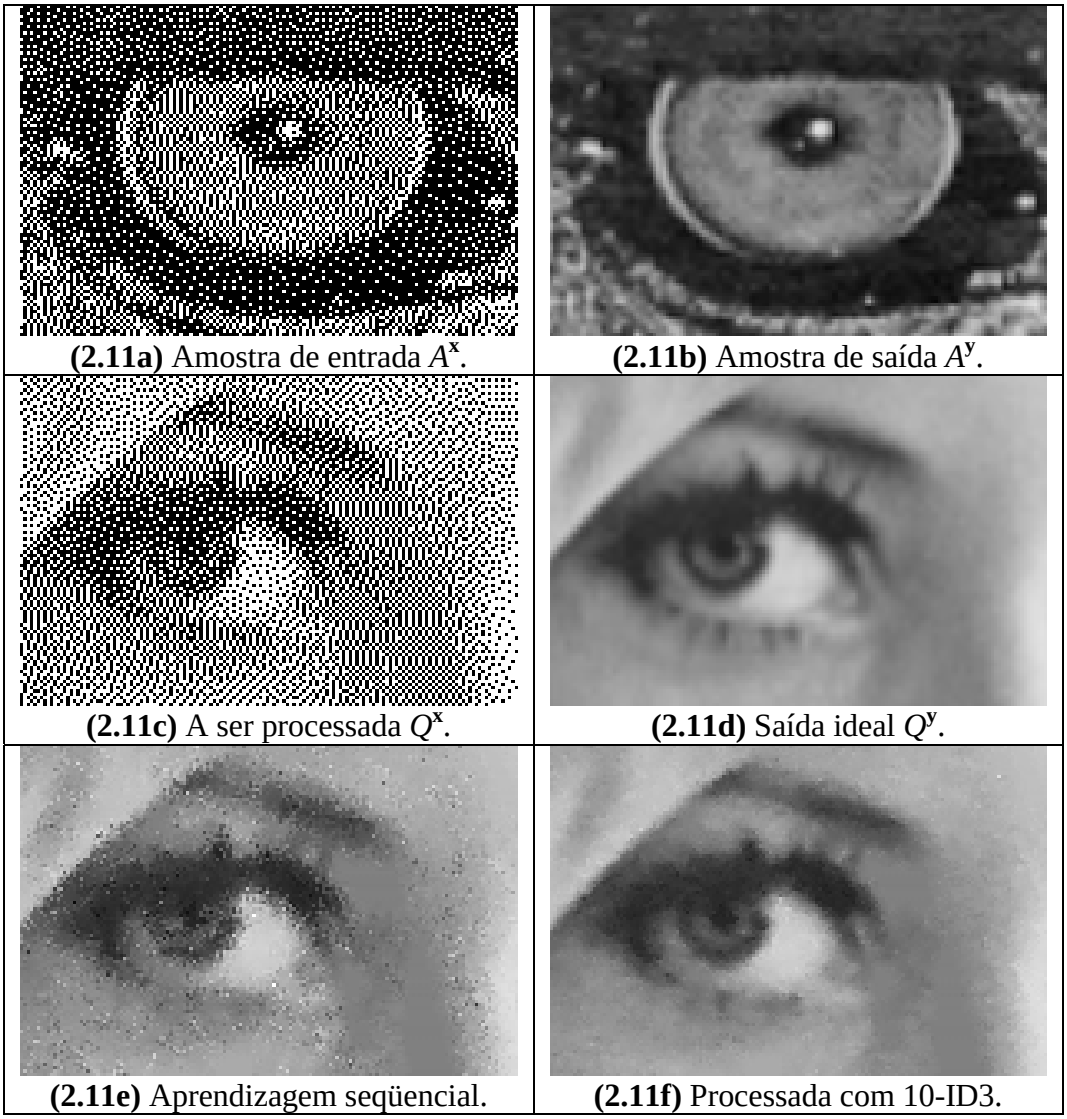
consideravelmente, conforme mostrada na tabela 2.7. A imagem “Lena” parece ser “boa” para fazer o meio-tom inverso, provavelmente por causa das suas amplas áreas suaves.

A figura 2.12 mostra partes das imagens meio-tom obtidas usando diferentes algoritmos e as suas respectivas imagens obtidas através do meio-tom inverso por 10-ID3. Suas PSNRs são mostradas na tabela 2.7. Em [Mese and Vaidyanathan, 2002], a PSNR relatada foi 27,08 dB para a difusão de erro. A nossa técnica produziu PSNR de 31,80 dB para um conjunto de imagens, que, embora diferente das usadas por [Mese and Vaidyanathan, 2002], contém imagens com características semelhantes. Testando somente a imagem Lena, [Mese and Vaidyanathan, 2002] relata 30,95 dB enquanto a nossa técnica produz 34,75 dB. Para a excitação ordenada, o nosso método produz 30,14 dB e 28,91 dB respectivamente para as opções “pontos dispersos” e “pontos aglutinados”. Os números correspondentes em [Mese and Vaidyanathan, 2002] são 25,82 dB e 24,26 dB. Isto parece apontar para um ganho de 4 dB em comparação com os resultados anteriores.

A fase de aplicação de IH leva somente 5 segundos num Pentium-1GHz, para uma imagem teste com 1050×1050 pixels. A fase de treinamento leva aproximadamente 20 minutos para ID3 e 1 minuto para a aprendizagem seqüencial (usando imagens com 1050×3150 pixels). A estrutura DT ocupa aproximadamente 5 MBytes.

		10-ID3 (dB)	1-ID3 (dB)	Seqüencial(dB)
Difusão de erro	3 imagens	31,80	31,02	25,39
	Somente Lena	34,75	33,20	26,75
OD disperso	3 imagens	30,14	29,73	28,11
	Somente Lena	33,69	33,17	32,97
OD aglutinado	3 imagens	28,91	28,56	27,78
	Somente Lena	32,59	32,01	31,83
HP laserjet, pontos grandes	3 imagens	27,66	27,32	27,35
	Somente Lena	31,72	31,16	31,07

Tab. 2.7: PSNRs obtidas usando diferentes algoritmos de aprendizagem DT e diferentes algoritmos de meio-tom. “3 imagens” refere às PSNRs obtidas testando o sistema IH em 3 imagens de teste diferentes, uma das quais era Lena. “Somente Lena” refere à PSNR obtida fazendo o meio-tom inverso da imagem Lena.



(2.11a) Amostra de entrada $A^{\mathbf{x}}$.

(2.11b) Amostra de saída $A^{\mathbf{y}}$.

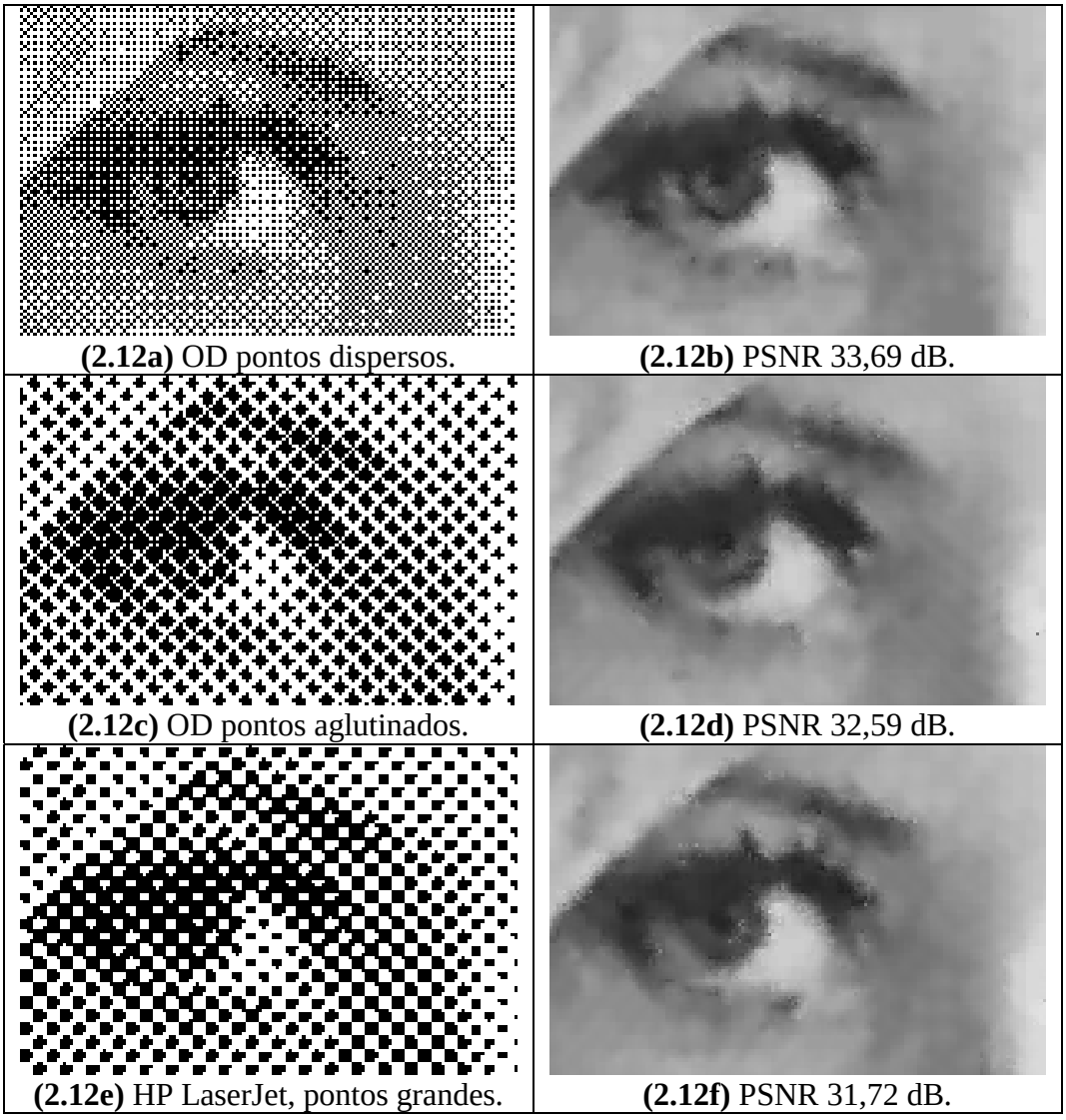
(2.11c) A ser processada $Q^{\mathbf{x}}$.

(2.11d) Saída ideal $Q^{\mathbf{y}}$.

(2.11e) Aprendizagem seqüencial.

(2.11f) Processada com 10-ID3.

Fig. 2.11: Meio-tom inverso das imagens obtidas pela difusão de erro, pela aprendizagem DT. (a, b) Amostra de entrada $A^{\mathbf{x}}$ e amostra de saída $A^{\mathbf{y}}$ (Mandrill). (c, d) Imagem a ser processada $Q^{\mathbf{x}}$ e a saída ideal $Q^{\mathbf{y}}$, supostamente desconhecida (Lena). (e) Imagem obtida usando aprendizagem seqüencial (PSNR 26,75 dB). (f) Imagem obtida usando algoritmo 10-ID3 (PSNR 34,75 dB).



(2.12a) OD pontos dispersos. **(2.12b)** PSNR 33,69 dB.

(2.12c) OD pontos aglutinados. **(2.12d)** PSNR 32,59 dB.

(2.12e) HP LaserJet, pontos grandes. **(2.12f)** PSNR 31,72 dB.

Fig. 2.12: Meio-tom inverso pelo algoritmo 10-ID3 aplicado em diferentes tipos de imagens meio-tom. A coluna da esquerda é a imagem meio-tom e a coluna da direita é a correspondente imagem em níveis de cinza obtida pelo algoritmo de meio-tom inverso 10-ID3. (a, b) Excitação ordenada, pontos dispersos (algoritmo de Bayer). (c, d) Excitação ordenada pontos aglutinados. (e, f) Imagem meio-tom gerada pelo driver para HP LaserJet para MS-Windows, usando a opção “pontos grandes”.

2.6 Conclusões

O objetivo deste capítulo foi apresentar as nossas contribuições científicas no uso das técnicas de aprendizagem de máquina no projeto automático de operadores restritos à janela (W-operadores).

Para isso, formalizamos o problema de aprendizagem de W-operadores usando a teoria PAC. Descrevemos como a estimação estatística pode ser utilizada para verificar se um operador obtido pela aprendizagem está próximo (ou não) do operador ótimo. Também descrevemos como usar a estimação estatística para comparar a eficácia de dois diferentes algoritmos de aprendizagem quanto à acurácia esperada do operador obtido. Descrevemos diversos algoritmos de aprendizagem, juntamente com as suas complexidades computacionais do treinamento, da aplicação e da memória necessária. Adaptamos alguns algoritmos de aprendizagem para que sejam mais eficientes no problema que estamos tratando: a aprendizagem k-NN tornou-se ek-NN para que a teoria PAC pudesse ser aplicada, e a aprendizagem DT tornou-se WZDT para melhorar o seu desempenho computacional. Aplicamos as teorias e os algoritmos desenvolvidos em três problemas: a ampliação de imagens binárias, a ampliação de imagens meio-tom e o meio-tom inverso, obtendo em todos eles bons resultados quanto à acurácia da solução e ao desempenho computacional.

Capítulo 3:
Difusão Anisotrópica

Resumo e nossas contribuições

O espaço de escala é uma das teorias utilizadas para a análise multi-escala de imagens e sinais. A técnica do espaço de escala linear gera as imagens em resoluções grossas fazendo convolução da imagem original com um núcleo gaussiano ou, equivalentemente, usando a imagem original como a condição inicial de um processo de difusão. Esta abordagem possui um defeito sério: é difícil obter a localização acurada das arestas importantes nas escalas grossas. A difusão anisotrópica foi proposta para superar esta dificuldade. Nela, os coeficientes da difusão são escolhidos de forma a encorajar a suavização intra-região e evitar a suavização inter-região. Com isso, os ruídos são eliminados e a imagem é simplificada ao mesmo tempo em que mantém as arestas nítidas. Temos utilizado a difusão anisotrópica em várias aplicações do Processamento e Análise de Imagens. Este capítulo descreve as teorias do espaço de escala linear e da difusão anisotrópica, e as nossas contribuições científicas nesta área.

Primeiro, descrevemos a teoria do espaço de escala linear e mostramos o efeito de deslocamento das arestas nas escalas grossas. Em segundo lugar, descrevemos a teoria da difusão anisotrópica (a clássica e aquela baseada na estatística robusta) e mostramos duas aplicações: a filtragem de sinais do sensor de aceleração e a detecção de arestas em imagens. Em terceiro lugar, descrevemos o melhoramento do algoritmo de reconstrução tomográfica de máxima entropia através da difusão anisotrópica robusta. Por fim, descrevemos o melhoramento do modelo linear geral (um processo de

detecção das áreas ativadas do cérebro em imagens de ressonância magnética funcional) usando a difusão anisotrópica robusta.

As nossas contribuições na área da difusão anisotrópica e do espaço de escala são:

1) *Melhoramento do algoritmo de reconstrução tomográfica máxima entropia* (*MENT*) *através da difusão anisotrópica robusta* (*RAD*)*:* Esta contribuição científica foi publicada em [Cn13] e encontra-se submetida em [Su03]. Nesta tese, ela está documentada na seção 3.4. O principal responsável por esta contribuição foi meu ex-orientando de doutorado Harold I. A. Bustos. Temos outros trabalhos publicados na área de tomografia [Ci07; Cn11; Cn09] mas que não estão documentados nesta tese.

   Resumo: A máxima entropia (MENT) é uma técnica de reconstrução amplamente conhecida baseada na otimização da entropia. Se somente uma pequena quantidade de dados de aquisição estiver disponível, este algoritmo converge para uma imagem ruidosa e borrada. Propomos um melhoramento a este algoritmo que consiste em aplicar alternadamente a reconstrução MENT e a difusão anisotrópica robusta (RAD). Testamos esta idéia para a reconstrução de dados paralelos em ângulo completo, mas a idéia pode ser aplicada para qualquer cenário de aquisição de dados. A nova técnica tem gerado imagens surpreendentemente claras com arestas nítidas, mesmo utilizando uma quantidade de dados de projeção extremamente pequena.

2) *Melhoramento do modelo linear geral utilizado na detecção de áreas ativadas do cérebro a partir das imagens de ressonância magnética funcional (f-MRI) utilizando RAD:* Esta contribuição foi publicada em [Cn12] e encontra-se submetida em [Su01]. Nesta tese, ela está documentada na seção 3.5. Eu fui o principal responsável por esta contribuição, contando com a colaboração do Prof. H. Z. Cho da University of California, Irvine.

   Resumo: O imageamento por ressonância magnética funcional (fMRI) consegue mapear de forma não invasiva as áreas do cérebro com as atividades neuroniais aumentadas sem o uso de um agente de contraste exógeno. A baixa

taxa de relação sinal-ruído das imagens fMRI torna necessário o uso de sofisticadas técnicas de processamento de imagens, tais como o mapa estatístico paramétrico (SPM), para detectar as áreas de cérebro ativadas. As nossas pesquisas levaram a uma nova técnica para obter um SPM limpo a partir dos dados fMRI ruidosos. Ela está baseada na RAD. Uma aplicação direta da RAD à fMRI não funciona, em grande parte devido à falta de bordas nítidas entre as regiões ativadas e não-ativadas. Para superar esta dificuldade, propomos calcular o SPM a partir da fMRI ruidosa, obter os coeficientes de difusão no espaço SPM, e então efetuar a difusão nas imagens fMRI utilizando os coeficientes previamente calculados. Estes passos são iterados até a convergência. Resultados experimentais utilizando a nova técnica geraram SPMs surpreendentemente nítidos e sem ruídos, com alto grau de significância estatística.

3) *Uso da RAD para filtrar sinais de sensores.* Para ilustrar o processo de RAD, utilizei nesta tese uma adaptação de um artigo, elaborado principalmente pelo meu orientando de mestrado Marco A. A. de Melo, onde se usa a RAD para filtrar os sinais de sensor de aceleração [Su06] (seção 3.3).

4) Também temos algumas contribuições em filtro nebulosos no espaço de escala [Cn08] e no uso da RAD para segmentar imagens coloridas [Ci09], mas que não estão documentadas nesta tese.

3.1 Introdução

Percebemos os objetos no mundo como tendo estruturas em escalas grossas e finas. Uma floresta pode parecer simplesmente um amontoado verde quando vista de distância. À medida que nos aproximamos, começamos a distinguir as árvores individuais, os troncos, os galhos, as folhas, as nervuras das folhas, os orvalhos sobre as folhas, etc. Assim, a multi-escala constitui uma noção natural da percepção visual. A representação multi-escala de uma imagem em forma de pirâmide foi desenvolvida já

na década de 70. Nesta estrutura, quanto mais grossa for a escala, menos pixels conterá a imagem.

Em 1983, Witkin [Witkin, 1983] propôs que a escala poderia ser considerada como um parâmetro contínuo, generalizando a noção de pirâmide. A idéia essencial desta abordagem é muito simples: dada uma imagem digital *Q*, essa imagem na escala σ é a convolução da *Q* com a máscara gaussiana de desvio-padrão σ. Esta teoria é denominada de espaço de escala gaussiano ou linear. A imagem *Q* na escala σ=0 é a própria imagem original. À medida que se vai da escala fina para a escala grossa, a imagem se torna cada vez mais "borrada".

A convolução com a máscara gaussiana de desvio-padrão σ pode ser vista como a solução da equação de condução de calor, onde o valor da imagem original *Q* num ponto (*x*, *y*) é a temperatura inicial nesse ponto, o tempo decorrido é $t = \sigma^2 / 2$, e a imagem *Q* na escala σ representa as temperaturas no instante *t*. Assim, a convolução gaussiana é um processo de difusão isotrópica. Isotrópico significa "aquele que apresenta as mesmas propriedades físicas em todas as direções", segundo [Aurélio, 1999].

As pesquisas subseqüentes levaram a diferentes formas de simplificar a imagem original, utilizando filtros diferentes da convolução gaussiana. Por exemplo, Jackway e Deriche [Jackway and Deriche, 1996] propuseram o uso de operadores morfológicos, resultando no espaço de escala morfológico.

Uma outra forma de simplificar imagens foi proposta por Perona e Malik [Perona and Malik, 1987; Perona and Malik, 1990], e teve um grande impacto científico. Eles propuseram o uso da difusão anisotrópica, substituindo a difusão isotrópica. No espaço de escala linear (que utiliza a difusão isotrópica para simplificar uma imagem), uma imagem em escala grossa torna-se borrada e as arestas deslocam-se espacialmente de uma escala para outra. Utilizando a difusão anisotrópica, as arestas continuam nítidas mesmo em escalas grossas e permanecem na mesma posição mesmo mudando de escala.

Na formulação da difusão anisotrópica de Perona-Malik, existe uma função chamada parada-na-aresta (edge stopping function) *g*, que controla a intensidade da difusão de acordo com o gradiente do ponto que deve sofrer difusão. A função parada-na-aresta possui um parâmetro de escala σ que, em conjunto com o gradiente, indica se a difusão deve ser forte ou fraca. A correta escolha da função parada-na-aresta e da escala afetam de forma decisiva o resultado da filtragem da imagem. Perona e Malik sugeriram duas funções parada-na-aresta, sem apresentar uma justificativa fundamentada para a escolha.

A difusão anisotrópica robusta (RAD) [Black et al., 1998] foi proposta como um melhoramento da difusão anisotrópica de Perona-Malik. Esta técnica assume que a entrada é uma imagem constante por regiões corrompida pelo ruído gaussiano aditivo com média zero e pequena variância. O objetivo é estimar a imagem original a partir dos dados ruidosos. Black et al. usaram a estatística robusta para resolver este problema, e propuseram o uso da função "Tukey's biweight" como a função parada-na-aresta, de acordo com a teoria estatística adotada. Na prática, a RAD converge mais rapidamente e conserva ainda melhor as bordas do que a difusão de Perona-Malik.

A RAD mostra-se útil em diversas aplicações de Processamento e Análise de Imagens. Ela é um excelente detector de arestas. Também é um ótimo filtro de ruídos aditivos, que preserva as bordas ao mesmo tempo em que elimina os ruídos. Quando utilizado como um filtro, a RAD procura estimar a imagem original constante por regiões a partir da sua versão corrompida pelo ruído gaussiano aditivo. Esta característica torna-a extremamente eficiente em diversas aplicações.

Organização deste capítulo

O restante deste capítulo está organizado como segue. A seção 3.2 apresenta os conceitos básicos do espaço de escala linear (ou gaussiano), subdividido em casos unidimensional e bidimensional. A seção 3.3 apresenta os conceitos básicos do espaço de escala não-linear gerado pela difusão anisotrópica, incluindo a difusão anisotrópica robusta (RAD). A fim de ilustrar o processo da difusão anisotrópica, apresentamos

na seção 3.3 duas aplicações: a filtragem de sinais unidimensionais do sensor de aceleração e a detecção de arestas em imagens. A seção 3.4 apresenta o melhoramento do algoritmo MENT-estendido através da RAD, denominada MENT reconstrução-difusão. A seção 3.5 apresenta o melhoramento obtido no processamento da ressonância magnética funcional (fMRI) através do uso de RAD. Finalmente, na seção 3.6 apresentamos as nossas conclusões.

3.2 Espaço de Escala Linear

Caso unidimensional

Nesta subsubseção, vamos explicar o espaço de escala linear seguindo de perto o texto [Velho et al., 2000]. Historicamente, a teoria do espaço de escala foi elaborada primeiro para os sinais unidimensionais e depois estendida para as imagens. Para definir o espaço de escala, vamos definir antes dois conceitos básicos.

Definição (convolução): Dadas duas funções $f, g : \mathbb{R} \to \mathbb{R}$ a convolução é definida:

$$f(x) * g(x) = (f * g)(x) = \int_{-\infty}^{\infty} f(u) g(x-u) du$$

Definição (gaussiana): A distribuição normal $N(\mu, \sigma)$, onde μ é a média e σ é o desvio-padrão, é definida através da função gaussiana:

$$g(x, \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[\frac{-(x-\mu)^2}{2\sigma^2}\right]$$

A figura 3.1a mostra a função $g(x,0,1)$ e as suas derivadas. Costuma-se adotar $\sigma^2 = 2t$ e $\mu = 0$ para obter a notação:

$$G_t(x) = \frac{1}{\sqrt{4\pi t}} \exp\left[-\frac{x^2}{4t}\right]$$

Note que $\lim_{t \to 0} G_t(x)$ resulta no impulso de Dirac.

Com isso, já podemos definir o espaço de escala:

Definição (espaço de escala): Seja $f : \mathbb{R} \to \mathbb{R}$ um sinal unidimensional. O espaço de escala deste sinal é a função $F : \mathbb{R} \times \mathbb{R}^+ \to \mathbb{R}$ (representada por $F(x,t) = F_t(x)$) que é a solução da equação de calor:

$$\begin{cases} \dfrac{\partial F_t(x)}{\partial t} = \dfrac{\partial^2 F_t(x)}{\partial x^2} \\ F_0(x) = f(x) \end{cases}$$

Afirmação: O espaço de escala de $f:\mathbb{R}\to\mathbb{R}$ pode ser obtido através das convoluções com gaussianas:

$$F_t(x) = G_t(x) * f(x)$$

O espaço de escala gaussiano tem as seguintes propriedades básicas [Velho et al., 2000]:

- *Linearidade*: A transformação L_t que leva o sinal original $f(x)$ ao espaço de escala $F_t(x)$ é linear, isto é, $L_t\{f+\lambda g\}(x) = L_t\{f(x)\} + \lambda L_t\{g(x)\}$.
- *Invariância por translação*: Se T é uma translação qualquer, o espaço de escala de $T(f)$ é $T(F)$, isto é, $G_t(x) * Tf(x) = T\big(G_t(x) * f(x)\big)$.
- *Causalidade*: O sinal f é "simplificado" com o aumento da escala, isto é, os cruzamentos de zero não aumentam com o aumento de t.

A causalidade é a propriedade que permite afirmar que um sinal é simplificado pela convolução por gaussiana. Informalmente, uma "aresta" do sinal (isto é, um ponto do sinal onde há uma transição abrupta de nível) corresponde ao cruzamento de zero da segunda derivada. Em sinais unidimensionais, os cruzamentos de zero da segunda derivada desaparecem à medida que o sinal é filtrado por máscaras gaussianas com desvios-padrões cada vez maiores. Por outro lado, nunca pode aparecer um cruzamento de zero da segunda derivada numa escala grossa σ_g, se ela não estava presente em todas as escalas σ mais finas que σ_g. Uma aresta pouco acentuada não pode ser detectada numa escala grossa, enquanto que numa escala bem fina todas as arestas podem ser detectadas. Formalizando este conceito, temos:

Definição (cruzamento de zero): Um cruzamento de zero de uma função contínua $f(x)$ é um intervalo fechado $[a,b]$ (possivelmente com $a=b$) tal que:

$$\begin{cases} f([a,b]) = 0 \\ \lim\limits_{x \to a-} sinal(f(x)) = -\lim\limits_{x \to b+} sinal(f(x)) \neq 0 \end{cases}$$

Proposição (causalidade do cruzamento de zero): Dada uma função $f(x)$ contínua, considere o seu espaço de escala gaussiano $F_t(x)$. O número de cruzamentos de zero de $F_t(x)$ não aumenta à medida que *t* cresce.

Corolário: Se $f(x)$ é diferenciável, então o número de máximos e mínimos de $F_t(x)$ não aumenta à medida que *t* cresce.

Demonstração: Os máximos e mínimos (com relação a *x*) de $F_t(x) = G_t(x) * f(x)$ são os crizamentos de zero de

$$\frac{\partial F_t(x)}{\partial x} = G_t(x) * \frac{\partial f(x)}{\partial x}$$

que é o espaço de escala de $f'(x)$. ■

De forma semelhante, pode-se demonstrar que os cruzamentos da segunda derivada de *f* (as arestas) não aumentam à medida que *t* cresce.

Proposição: $F_t(x)$ é uma função suave (infinitamente diferenciável) para qualquer $t > 0$ fixo.

Demonstração: Note que

$$\frac{\partial^n F_t}{\partial x^n} = \frac{\partial^n}{\partial x^n}\left(G_t(x) * f(x)\right) = \frac{\partial^n G_t(x)}{\partial x^n} * f(x)$$

existe pois a função gaussiana $G_t(x)$ é suave para qualquer $t > 0$. ■

Caso bidimensional

Definição (normal): A distribuição normal bidimensional $N(x_0, y_0, \sigma)$, onde (x_0, y_0) é a média e σ é o desvio-padrão, é definida através da função gaussiana:

$$g(x, y, x_0, y_0, \sigma) = \frac{1}{2\pi\sigma^2} \exp\left[\frac{-(x-x_0)^2 - (y-y_0)^2}{2\sigma^2}\right]$$

A figura 3.1b mostra a função $g(x,y,0,0,1)$, e as figuras 3.1c-3.1f mostram o seu módulo do gradiente, as suas derivadas parciais, e o seu laplaciano. Costuma-se adotar $\sigma^2 = 2t$ e $\mu = 0$ para obter a notação:

$$G_t(x,y) = \frac{1}{4\pi t}\exp\left[-\frac{x^2+y^2}{4t}\right].$$

Definição (espaço de escala): Seja $f:\mathbb{R}^2 \to \mathbb{R}$ uma imagem 2-D. O espaço de escala desta imagem é a função $F:\mathbb{R}^2 \times \mathbb{R}^+ \to \mathbb{R}$ (denotada $F_t(x,y)$) que satisfaz a seguinte equação diferencial parcial ou equação de calor bidimensional:

$$\begin{cases} \dfrac{\partial F_t(x,y)}{\partial t} = \nabla^2 F_t(x,y) = \dfrac{\partial^2 F_t(x,y)}{\partial x^2} + \dfrac{\partial^2 F_t(x,y)}{\partial y^2} \\ F_0(x,y) = f(x,y) \end{cases}, \qquad (3.1)$$

Afirmação: A solução da equação diferencial parcial acima pode ser expressa como uma convolução com gaussianas bidimensionais:

$$F_t(x,y) = G_t(x,y) * f(x,y),$$

Proposição (separabilidade): A convolução acima pode ser calculada através de duas convoluções com gaussianas unidimensionais:

$$F_t(x,y) = G_t(y) \underset{(y)}{*} \left(G_t(x) \underset{(x)}{*} f(x,y) \right).$$

Esta propriedade permite acelerar a computação do espaço de escala gaussiano para as imagens.

Além da linearidade e da invariância por translações, o espaço de escala gaussiano bidimensional possui a invariância por rotações.

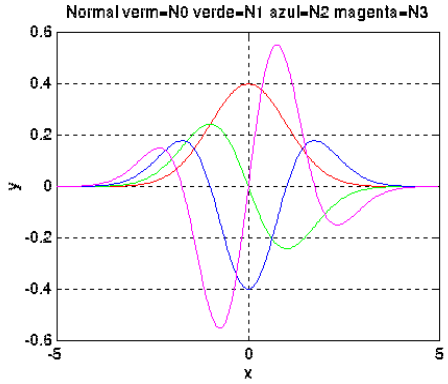
Proposição: Seja f uma imagem qualquer e $g = R_\theta f$ a rotação de f por ângulo θ. Então o espaço de escala G_t de *g* é a rotação por ângulo θ de F_t, isto é:

$$g = R_\theta f \quad \Rightarrow \quad G_t = R_\theta F_t .$$

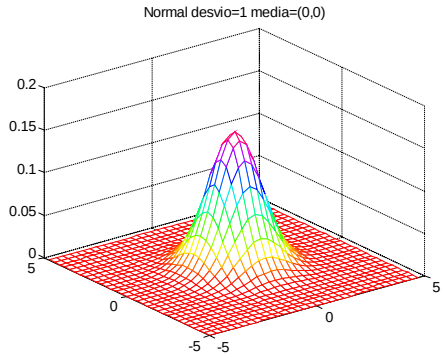
Infelizmente, o princípio de causalidade não vale para as imagens 2-D. Velho et al. [Velho et al., 2000] afirmam: “Tentemos agora entender o que será o princípio da causalidade em 2-D. Note que não faz sentido falar em número de cruzamentos de zero de uma imagem, já que em geral os cruzamentos de zero de uma imagem formam um conjunto de curvas, não um conjunto discreto de pontos. Por outro lado, pode-se falar do número de máximos e mínimos locais de uma imagem genérica (ou de um sinal *n*-dimensional). No entanto, *não é verdade* que o número de pontos críticos diminua com a escala no espaço de escala de uma imagem qualquer.”

Na prática, qualquer imagem digital está definida em um subconjunto finito de $\mathbb{Z}^2$, em vez de $\mathbb{R}^2$. Assim, é necessário discretizar de alguma forma o espaço de escala espacialmente. Muitas técnicas de discretização têm sido utilizadas para esta tarefa, por exemplo, a gaussiana amostrada, a gaussiana integrada e a gaussiana verdadeiramente discreta (obtida utilizando a função modificada de Bessel). Por outro lado, não é estritamente necessário discretizar o espaço de escala no tempo, pois é possível calcular “sob encomenda” qualquer pixel em qualquer escala real no espaço de escala discretizado espacialmente $\mathbb{Z}^2\times\mathbb{R}_+$. Porém, é computacionalmente vantajoso pré-calcular o espaço de escala para algumas escalas fixas, obtendo o espaço de escala discretizada espacial e temporalmente $\mathbb{Z}^2\times\mathbb{Z}_+$. Veja [Velho et al., 2000; Lindeberg, 1994] para maiores detalhes.

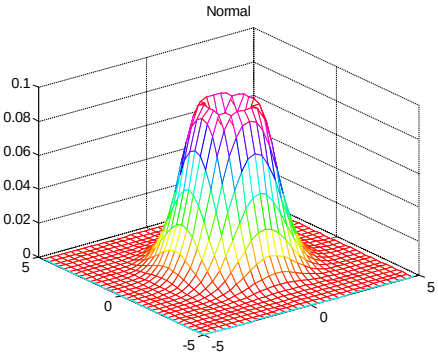
A figura 3.2 mostra a detecção de arestas de uma imagem no espaço de escala linear. A imagem original sofre convoluções com as gaussianas de diferentes desvios-padrões, gerando as imagens em diferentes escalas (primeira coluna). Calculando a convolução da imagem original com o laplaciano da gaussiana com diferentes desvios-padrões, obtém-se a segunda coluna (onde está ilustrado somente o sinal das imagens resultantes: preto indica positivo e branco indica negativo). Aplicamos os operadores morfológicos (dilatação seguida pela operação ou-exclusivo) sobre as imagens da segunda coluna para obter as arestas em diferentes escalas (terceira coluna).



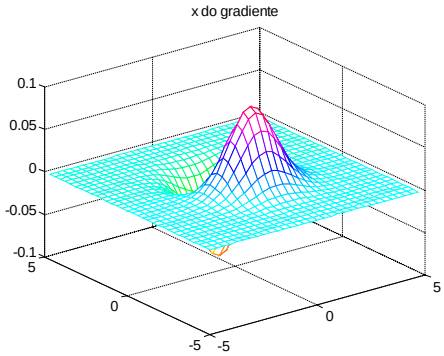
(3.1a) Função gaussiana com σ=1 (vermelho) e suas 1ª, 2ª e 3ª derivadas (respectivamente em verde, azul e magenta).



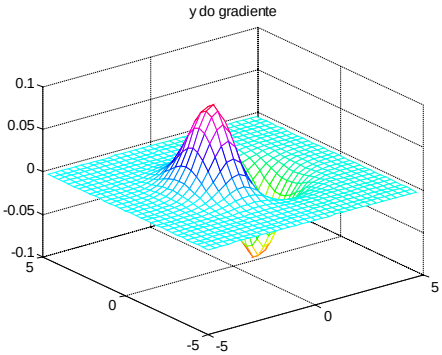
(3.1b) Função gaussiana *G* bidimensional com σ=1.



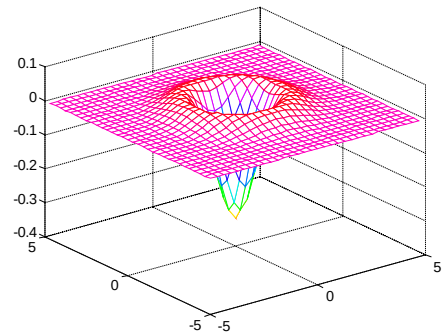
(3.1c) Módulo do gradiente da gaussiana $\|\nabla G(x,y)\|$.



(3.1d) Derivada parcial *x* da gaussiana $\partial G(x,y)/\partial x$.



(3.1e) Derivada parcial *y* da gaussiana $\partial G(x,y)/\partial y$.



(3.1f) Laplaciano da gaussiana $\nabla^2 G(x,y)$.

Fig. 3.1: Funções gaussianas unidimensional, bidimensional e suas derivadas.

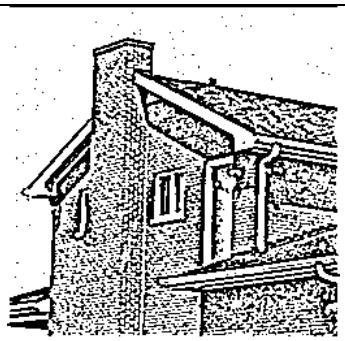
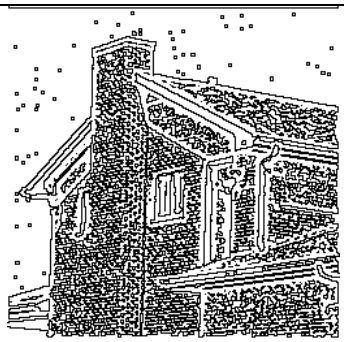
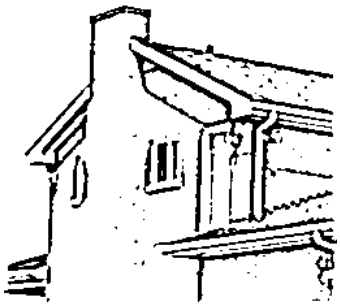
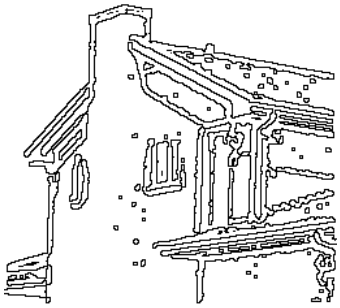
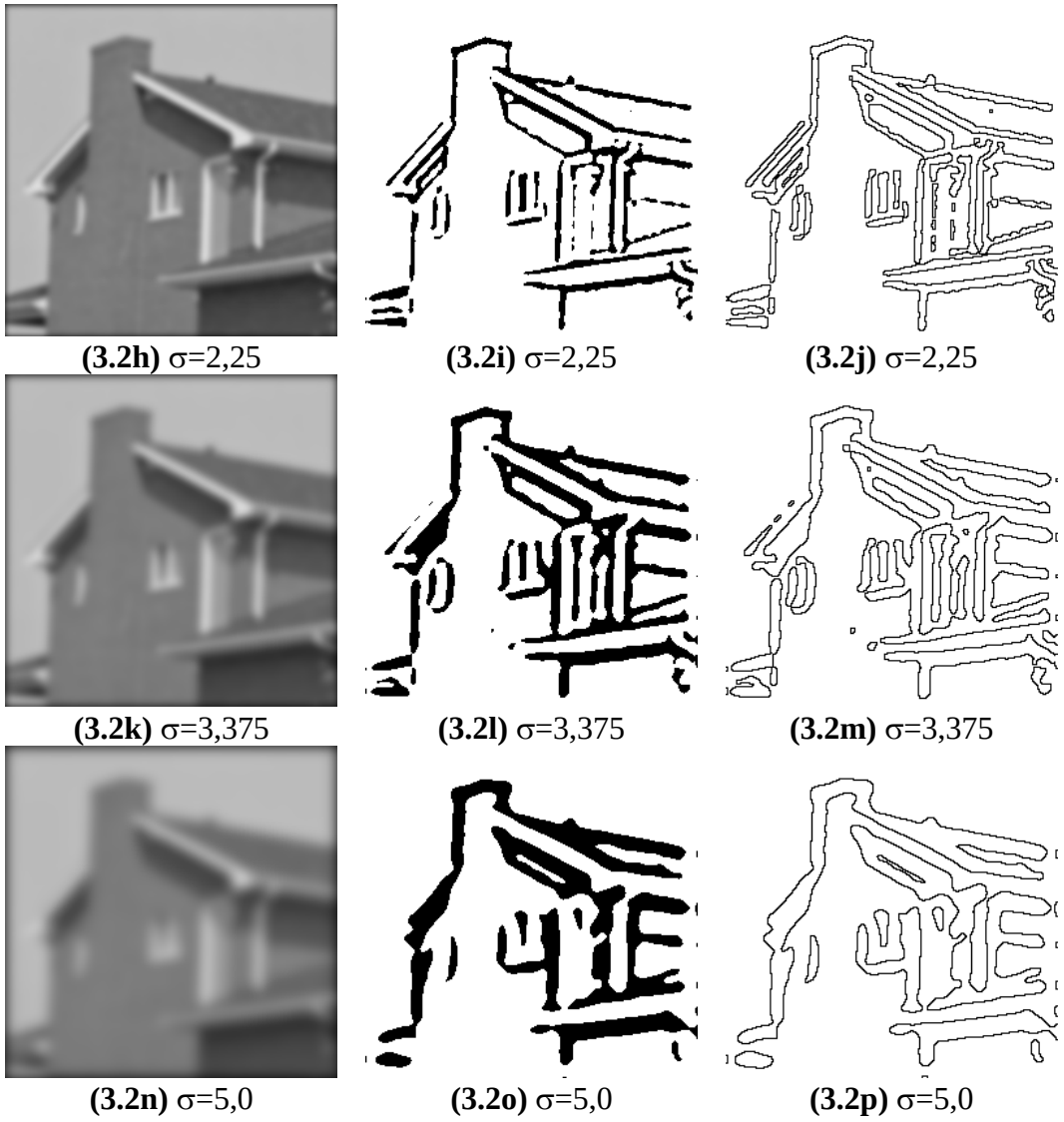
Imagem “casa.tga” no espaço de escala linear.	Sinal do laplaciano da imagem. Preto indica positivo e branco indica negativo.	Cruzamentos de zero do laplaciano (ou arestas).
 **(3.2a)** Imagem original ($\sigma \rightarrow 0$)		
 **(3.2b)** σ=1,0	 **(3.2c)** σ=1,0	 **(3.2d)** σ=1,0
 **(3.2e)** σ=1,5	 **(3.2f)** σ=1,5	 **(3.2g)** σ=1,5

Fig. 3.2: Continua na próxima página.



(3.2h) σ=2,25 **(3.2i)** σ=2,25 **(3.2j)** σ=2,25

(3.2k) σ=3,375 **(3.2l)** σ=3,375 **(3.2m)** σ=3,375

(3.2n) σ=5,0 **(3.2o)** σ=5,0 **(3.2p)** σ=5,0

Fig. 3.2: Detecção de arestas no espaço de escala linear. A imagem "casa.tga" no espaço de escala (primeira coluna), o sinal do laplaciano da imagem (segunda coluna) e as arestas ou os cruzamentos de zero do laplaciano da imagem (terceira coluna). Quanto σ cresce, as arestas menos importantes deixam de ser detectadas. Note que as arestas deslocam-se espacialmente à medida que σ cresce.

3.3 Difusão Anisotrópica

O espaço de escala linear possui muitas propriedades matemáticas atraentes. Porém, nas escalas grossas, a imagem torna-se borrada e as arestas deslocam-se espacialmente. Para manter as arestas nítidas, ao mesmo tempo em que se filtram os ruídos e os detalhes pouco importantes, Perona e Malik definiram o espaço de escala não-linear anisotrópica [Perona and Malik, 1990] modificando a equação diferencial parcial (3.1):

$$\begin{cases} \dfrac{\partial F_t(x,y)}{\partial t} = \nabla \cdot \left[g\left(\left\| \nabla F_t(x,y) \right\| \right) \nabla F_t(x,y) \right], \\ F_0(x,y) = f(x,y) \end{cases}$$

onde $\left\| \nabla F_t(x,y) \right\|$ é o magnitude do gradiente da F_t, e g é uma função "parada-na-aresta" (edge stopping function).

Perona e Malik discretizaram (espaço-temporalmente) a sua equação de difusão anisotrópica acima como:

$$I(s,t+1) = I(s,t) + \frac{\lambda}{|\eta_s|} \sum_{p \in \eta_s} g\left(\left| \nabla I_{s,p}(t) \right| \right) \nabla I_{s,p}(t), \qquad (3.2)$$

onde:

- $I(s,t)$ é a imagem discretizado espacial e temporalmente;
- s denota a posição de pixel numa grade discreta 2-D;
- t agora denota o passo de tempo discreto (número de iterações, $t \geq 0$);
- a constante $\lambda \in \mathbb{R}^+$ determina a velocidade de difusão (normalmente $\lambda = 1$);
- η_s representa o conjunto de vizinhos espaciais do voxel s. Para imagens 2-D, normalmente quatro pixels vizinhos são considerados: norte, sul, leste e oeste. Para imagens 3-D, seis voxels são normalmente considerados (os quatro voxels já mencionados mais os voxels "em cima" e "embaixo");
- $\nabla I_{s,p}(t)$ é a magnitude do gradiente da imagem I no ponto s na direção (s, p) na iteração t: $\nabla I_{s,p}(t) = I(p,t) - I(s,t), \quad p \in \eta_s$.

Perona e Malik sugeriram usar uma das duas funções parada-na-aresta abaixo (que vamos denotar por g_1 e g_2):

$$g_1(x) = \frac{1}{1 + \frac{x^2}{2\sigma^2}}$$

$$g_2(x) = \exp\left[\frac{-x^2}{2\sigma^2}\right]$$

A correta escolha da função g e da escala σ afeta substancialmente o quanto as descontinuidades serão preservadas.

Black et al. [Black et al., 1998] propuseram recentemente a difusão anisotrópica robusta (RAD). Esta técnica assume que a entrada é uma imagem constante por regiões corrompida pelo ruído gaussiano aditivo com média zero e pequeno desvio-padrão. O objetivo é estimar a imagem original a partir do dado ruidoso. Black et al. usaram a estatística robusta para resolver este problema. Eles calcularam uma imagem I que satisfaz o seguinte critério de otimização:

$$\min_I \sum_{s \in I} \sum_{p \in \eta_s} \rho_\sigma(I(p) - I(s))$$

onde $I(s)$ é o valor da imagem I no pixel s, η_s é a vizinhança espacial do pixel s, ρ é uma norma de erro robusta e σ é um parâmetro de escala. A equação acima pode ser resolvida pelo sistema (3.2), fazendo $g(x) = \rho'(x)/x$. Black et al. escolheram a função "Tukey's biweight" como a norma de erro ρ, de acordo com a teoria da estatística robusta. A correspondente função parada-na-aresta, que denotaremos como g_3, é:

$$g_3(x) = \begin{cases} \left[1 - \frac{x^2}{\sigma^2}\right]^2, & |x| \le \sigma \\ 0, & \text{caso contrário} \end{cases}$$

Para ter uma noção intuitiva da RAD, considere uma imagem constante por regiões, corrompida pelo ruído. A RAD executa a média da vizinhança intra-região, e evita

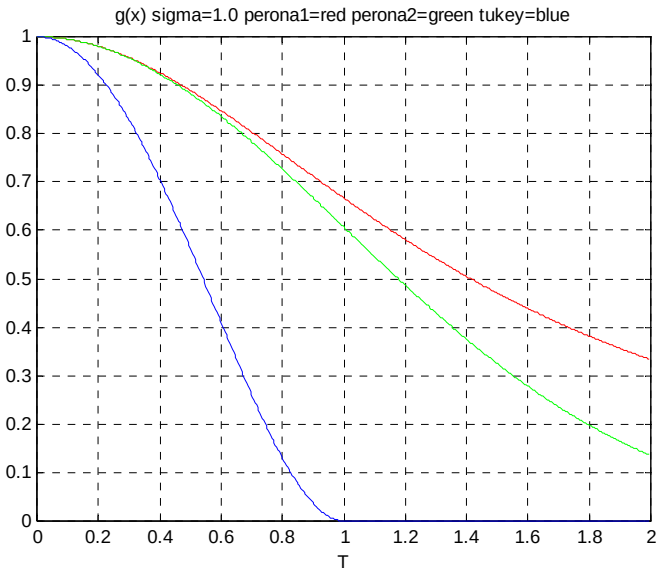
calcular a média inter-região. Assim, este processo atenua os ruídos ao mesmo tempo em que preserva as arestas entre as diferentes regiões nítidas.

A figura 3.3a mostra as três funções parada-na-aresta. Repare que as três estão em escalas diferentes, de forma que é necessário normalizá-las para poder compará-las. Para isso, considere a função $\psi(x) = xg(x) = \rho'(x)$. Esta função é denominada função de influência na estatística robusta e indica o quanto o erro cometido por uma medida particular (e quantificado pela norma de erro ρ) influencia na solução. A figura 3.3b mostra as 3 funções de influência correspondentes às 3 funções parada-na-aresta. Para normalizar as 3 funções parada-na-aresta, os pontos de máximo das 3 funções de influência foram calculados, e as funções ψ_1 e ψ_2 foram ajustadas de forma que os seus pontos de máximo coincidam com o ponto de máximo da ψ_3 $\left(x = \sqrt{0,2}\right)$. Fazendo isso, obtivemos as funções g_1 e g_2 normalizadas abaixo. A função g_3 não foi alterada.

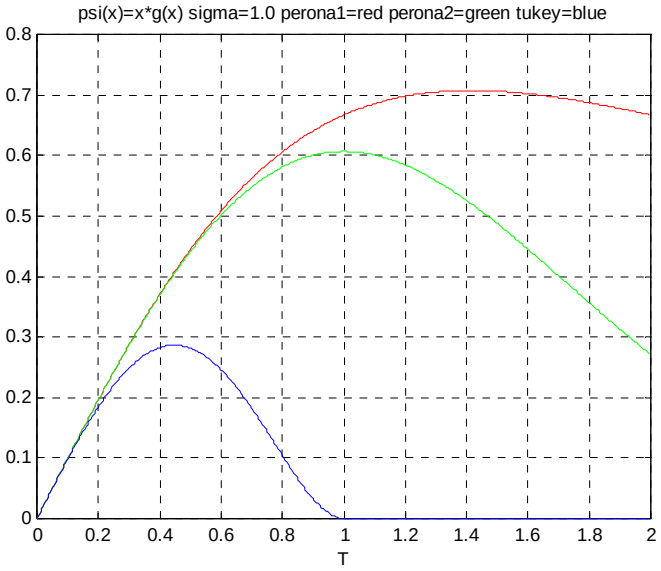
$$g_1(x) = \frac{1}{1 + \frac{5x^2}{\sigma^2}}$$

$$g_2(x) = \exp\left[\frac{-5x^2}{2\sigma^2}\right]$$

As figuras 3.3c e 3.3d mostram as funções parada-na-aresta e de influência normalizadas.

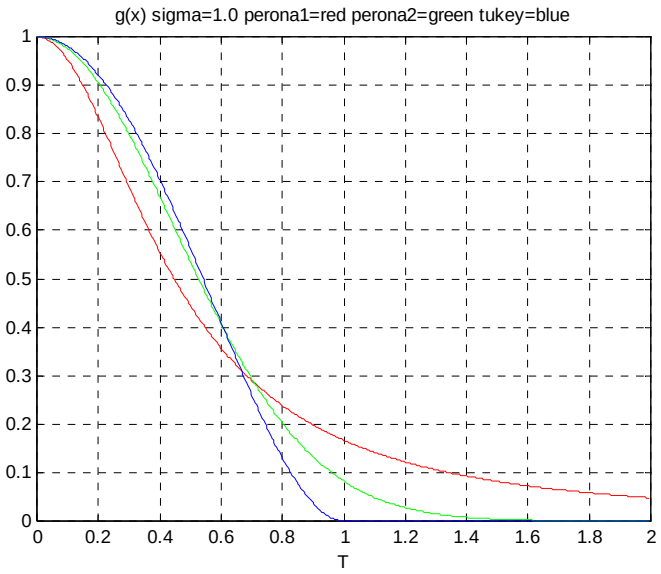


(3.3a) Funções parada-na-aresta não-normalizadas com σ=1: g_1 (Perona-Malik 1, em vermelho), g_2 (Perona-Malik 2, em verde) e g_3 (Tukey's biweight, em azul).

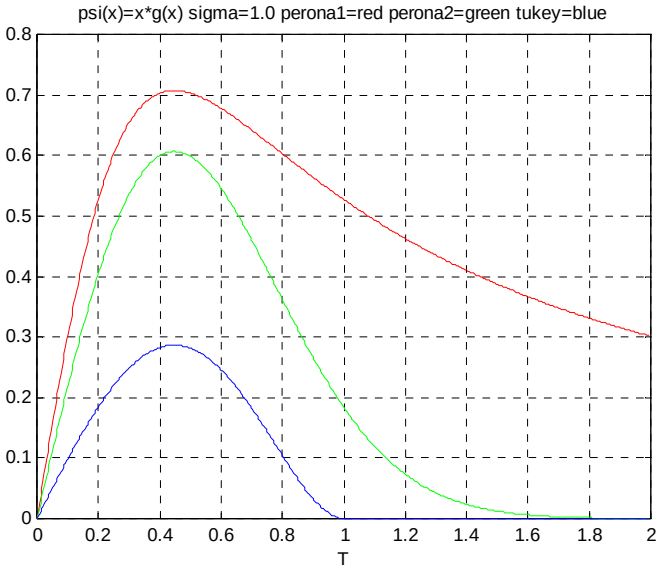


(3.3b) Funções de influência não-normalizadas: ψ_1 (Perona-Malik 1, em vermelho), ψ_2 (Perona-Malik 2, em verde) e ψ_3 (Tukey's biweight, em azul).

Fig. 3.3: Continua na próxima página.



(3.3c) Funções parada-na-aresta normalizadas com σ=1: g_1 (Perona-Malik 1, em vermelho), g_2 (Perona-Malik 2, em verde) e g_3 (Tukey's biweight, em azul).



(3.3d) Funções de influência normalizadas: ψ_1 (Perona-Malik 1, em vermelho), ψ_2 (Perona-Malik 2, em verde) e ψ_3 (Tukey's biweight, em azul).

Fig. 3.3: Funções parada-na-aresta e de influência, antes e depois da normalização.

Filtragem de sinais unidimensionais pela difusão anisotrópica

Esta subsubseção é uma adaptação resumida do artigo [Su06] elaborado principalmente pelo orientando de mestrado Marco A. A. de Melo.

Vamos ilustrar o uso da difusão anisotrópica, utilizando-a para filtrar sinais unidimensionais. O objetivo é filtrar os sinais do sensor de aceleração ADLX202E da Analog Devices [Analog, 2000], utilizado em sistemas automotivos para determinar a velocidade do veículo [Shih and Weinberg, 2001]. Inicialmente, iremos analisar o efeito das escolhas da função parada-na-aresta e escala na filtragem de um sinal artificial semelhante aos sinais do acelerômetro e contaminado com um ruído gaussiano. O uso de um sinal artificial permite calcular o erro cometido pelo filtro, pois neste caso dispomos do sinal sem ruído com o qual o sinal filtrado pode ser comparado. Usando um sinal real do acelerômetro, não há como calcular o erro, pois não temos acesso ao sinal sem ruído.

Como vimos na seção anterior, as funções parada-na-aresta dependem de um parâmetro de escala σ. Testamos duas maneiras de calcular automaticamente esta escala a partir do sinal ruidoso I. A primeira, denominada escala robusta, está baseada em estatística robusta e foi proposta por [Black et al., 1998]:

$$\begin{aligned}\sigma_1 &= 1{,}4826\,\mathrm{MAD}(\nabla I) \\ &= 1{,}4826\,\mathrm{median}_I\left[\left\|\nabla I\right\| - \mathrm{median}_I\left(\left\|\nabla I\right\|\right)\right]\end{aligned}$$

onde “MAD” denota o desvio absoluto mediano e a constante deriva do fato de que MAD de uma distribuição normal com média zero e variância unitária é 1/1,4826 = 0,6745.

A segunda forma de calcular a escala provém da análise utilizada em sistemas de transmissão de pulsos contaminados por ruído gaussiano, onde a informação está contida na amplitude e na posição dos pulsos [Carlson, 1986]. Esta análise pode ser aplicada aos sinais do acelerômetro, pois também neste caso a informação está presente na amplitude e no instante dos pulsos. Carlson sugere usar o limiar de detecção

de pulso maior ou igual a duas vezes o desvio-padrão σ_n do ruído gaussiano. Neste trabalho, testamos a seguinte escala:

$$\sigma_2 = 2\sigma_n .$$

O erro entre o sinal filtrado e o sinal sem ruído foi calculado usando a Raiz da Média Quadrática (RMS):

$$\mathrm{RMS}(S,R) = \sqrt{\frac{1}{N}\sum (S_n - R_n)^2}$$

onde:

- S_n é a amostra n do sinal filtrado.
- R_n é a amostra n do sinal original sem ruído.
- N é o número total de amostras dos sinais.

A figura 3.4a mostra o sinal sem ruído gerado artificialmente e a figura 3.4b mostra esse sinal contaminado com ruído gaussiano com desvio-padrão 0,2315. O sinal original e o sinal com ruído foram amostrados em 50 Hz. O erro RMS entre os dois sinais é 0,2314. Portanto, se o erro de algum sinal filtrado for maior que esse valor, o filtro estará introduzindo mais erro, em vez de diminuí-lo. As escalas σ_1 e σ_2 foram calculadas e são respectivamente 0,198 e 0,463.

Os erros obtidos com a filtragem do sinal por difusão anisotrópica usando as funções parada-na-aresta g_1, g_2 e g_3 estão mostrados nas tabelas 3.1 e 3.2. Na tabela 3.1 foram feitas 50 iterações da equação (3.2) e na tabela 2 foram feitas 100 iterações. Para cada função, são mostrados os 3 erros observados usando as escalas σ_1, σ_2 e a escala ótima σ^*. As escalas ótimas foram escolhidas manualmente para se obter o menor erro RMS.

Conforme esperado, em ambos os casos o menor erro foi obtido usando a função de Tukey g_3. A função Perona-Malik g_2 também gerou erros baixos, bastante próximos aos de Tukey. A figura 3.4c mostra o sinal filtrado pela difusão anisotrópica com a melhor qualidade. Pode-se notar que a informação útil foi preservada, ao mesmo tempo em que os ruídos indesejáveis foram eliminados.

Utilizando a função g_1, os erros aumentaram de 50 para 100 iterações (tabelas 3.1 e 3.2, para as escalas fixas σ_1 e σ_2). Isto é causado pela não-convergência do processo de difusão anisotrópica, quando se utiliza a função g_1. Neste caso, se o número de iterações for muito grande, o sinal filtrado converge para um sinal com amplitude constante. A figura 3.4d mostra esse fenômeno, onde os pulsos tiveram os picos decepados após 100 iterações.

Por outro lado, utilizando a função g_3 os erros diminuíram ligeiramente quando o número de iterações aumentou (para as escalas fixas σ_1, σ_2 e σ^*). O processo de difusão anisotrópica usando g_3 converge para um sinal otimamente filtrado, de forma que a saída não se altera após um número suficiente de iterações. A função g_2 é muito parecida com g_3, mas mesmo assim observa-se um pequeno aumento de erro ao aumentar o número de iterações (para σ_2).

A partir da tabela 2, podemos extrair as seguintes regras empíricas para calcular a escala ótima σ^* da difusão anisotrópica robusta:

$$\sigma^* = \frac{0{,}593}{0{,}198}\sigma_1 = 3{,}00 \times \sigma_1$$

$$\sigma^* = \frac{0{,}593}{0{,}463}\sigma_2 = 1{,}28 \times \sigma_2$$

A figura 3.4e mostra o sinal filtrado pelo filtro linear Butterworth passa-baixas de quinta ordem, com freqüência de corte de 5 Hz. Variamos a freqüência de corte até encontrar aquela onde o erro era mínimo. O sinal filtrado mostra uma forte distorção, alterando as amplitudes e os instantes de ocorrência dos picos do sinal. Esta distorção está refletida na alta taxa de erro RMS obtido (0,66135), mais de 8 vezes maior do que o menor erro obtido com a difusão anisotrópica (0,07957) e maior até que o erro do sinal ruidoso não filtrado (0,2314).

Também testamos o desempenho do filtro linear média móvel que consiste em calcular a média aritmética dos valores dos pontos vizinhos:

$$S_n = \left[Q_{n-1} + Q_n + Q_{n+1}\right]/3$$

onde S_n é o sinal filtrado e Q_n é o sinal original com ruído. O erro RMS obtido foi 0,15449.

Aplicamos os resultados obtidos até agora na filtragem de um sinal real do acelerômetro. A qualidade da filtragem deverá ser avaliada apenas visualmente, pois não é possível calcular o erro RMS, uma vez que não dispomos do sinal original sem ruído. O sinal original está mostrado na figura 3.5a e foi fornecido pela Analog Devices. As escalas σ_1 e σ_2 calculadas a partir desse sinal foram respectivamente 0,1927 e 0,506. Usando as duas regras empíricas propostas acima, obtemos os valores candidatos para a escala ótima de 0,578 e 0,648. A figura 3.5b mostra o sinal filtrado pela RAD com escala σ=0,648. A filtragem limpou fortemente o sinal, permitindo inclusive distinguir duas colisões na região final do sinal, difíceis de serem visualizadas no sinal ruidoso original.

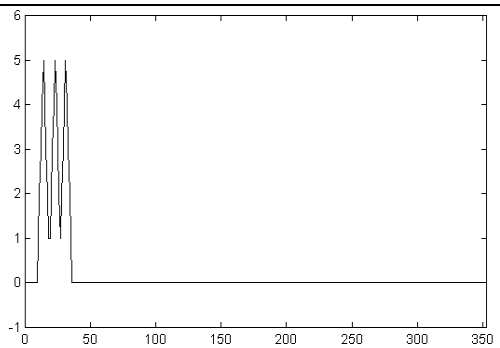
A figura 3.5c mostra o sinal filtrado pelo filtro Butterworth de quinta ordem, com freqüência de corte de 5 Hz. Esta filtragem elimina as informações contidas nas amplitudes e nos instantes de ocorrência dos pulsos.

	Perona-Malik g_1	Perona-Malik g_2	Tukey g_3
$\sigma_1 = 0{,}198$	0,12103	0,19114	0,22150
$\sigma_2 = 0{,}463$	0,19084	0,11196	0,10764
Escala ótima σ^*	0,10565 ($\sigma^* = 0{,}142$)	0,09269 ($\sigma^* = 0{,}374$)	0,08968 ($\sigma^* = 0{,}594$)

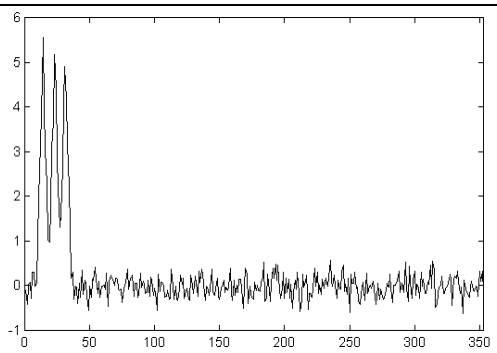
Tab. 3.1: Erros RMS obtidos filtrando o sinal da figura 3.4b pela difusão anisotrópica com 50 iterações, com diferentes funções parada-na-aresta e diferentes escalas.

	Perona-Malik g_1	Perona-Malik g_2	Tukey g_3
$\sigma_1 = 0{,}198$	0,13658	0,18285	0,22145
$\sigma_2 = 0{,}463$	0,26024	0,12009	0,10351
Escala ótima σ^*	0,10303 ($\sigma^* = 0{,}099$)	0,09393 ($\sigma^* = 0{,}382$)	0,07957 ($\sigma^* = 0{,}593$)

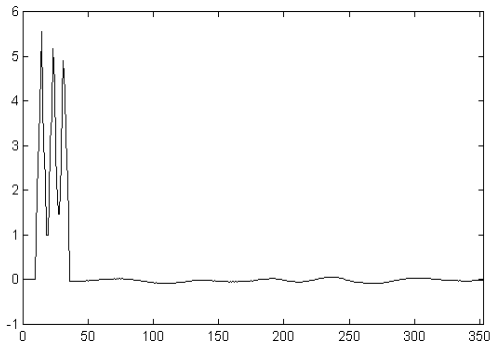
Tab. 3.2: Erros RMS obtidos com 100 iterações.



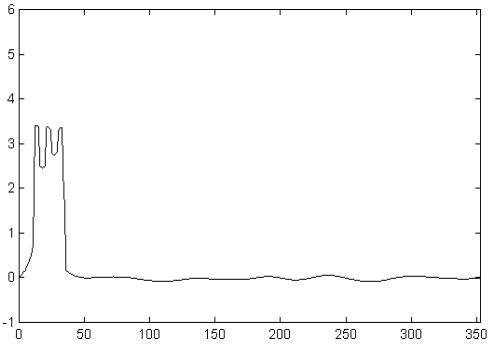
(3.4a) Sinal gerado artificialmente.



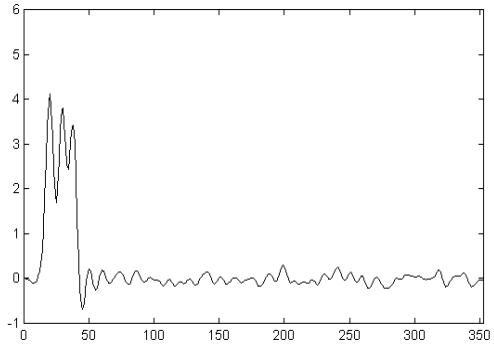
(3.4b) Sinal artificial com ruído gaussiano.



(3.4c) O sinal filtrado pela difusão anisotrópica com o menor erro foi obtido usando a função de Tukey g_3 (escala σ^*=0,593, 100 iterações).

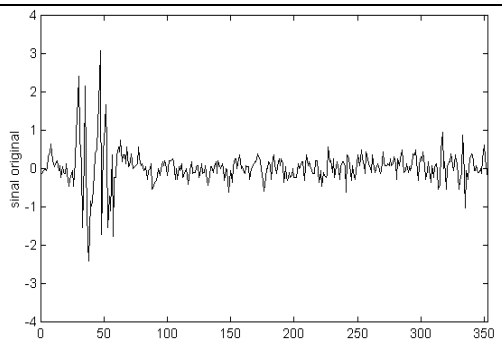


(3.4d) O sinal filtrado pela difusão anisotrópica usando a função Perona-Malik g_1 apresenta alta distorção quando o número de iterações é grande (100 iterações e escala σ_2 = 0,463).

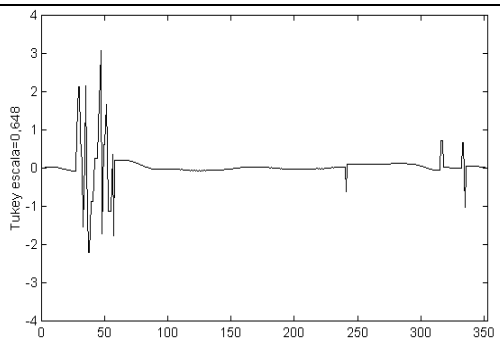


(3.4e) O sinal filtrado pelo Butterworth com freqüência de corte (ótima) 5 Hz apresenta alto erro RMS.

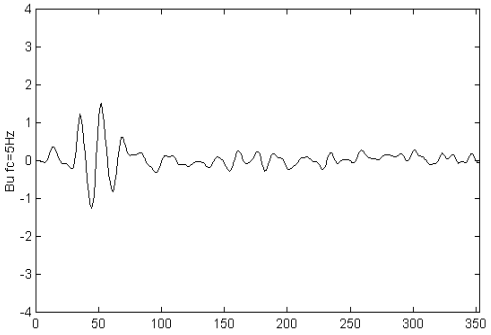
Fig. 3.4: Filtragem de um sinal sintetizado pela difusão anisotrópica.



(3.5a) Sinal original do sensor ADLX202E.



(3.5b) Sinal filtrado pela RAD (100 iterações, $\sigma = 0{,}648$).



(3.5c) Sinal obtido pelo filtro Butterworth com freqüência de corte 5 Hz.

Fig. 3.5: Filtragem do sinal do sensor de aceleração ADLX202E pela difusão anisotrópica e pelo filtro Butterworth.

Detecção de arestas pela difusão anisotrópica

A figura 3.6 mostra a detecção de arestas usando várias funções parada-na-aresta e diferentes escalas σ. O número de iterações foi mantido fixo em t_{max} = 50. Uma comparação visual entre as figuras 3.2 e 3.6 permite constatar que a difusão anisotrópica preserva muito melhor a nitidez e a localização das bordas do que a difusão isotrópica.

A figura 3.7 permite constatar a superioridade da função parada-na-aresta de Tukey sobre aquelas de Perona-Malik. Compare as imagens da figura 3.7 (500 iterações) com as imagens da última linha da figura 3.6 (50 iterações). Todas essas imagens foram obtidas usando a escala σ=0,08. Quando o número de iterações é grande a função g_1, e em menor grau a função g_2, borra as arestas. Enquanto isso a RAD (g_3) mantém as arestas perfeitamente nítidas.

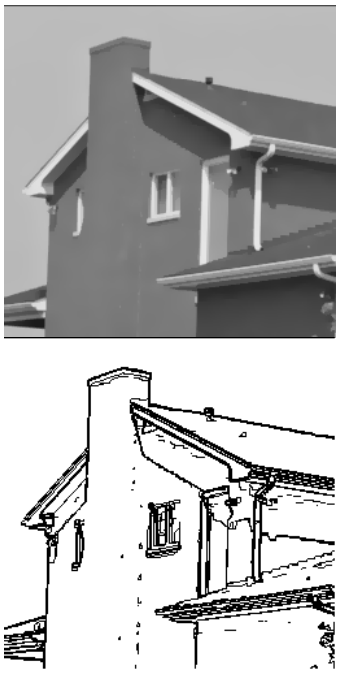
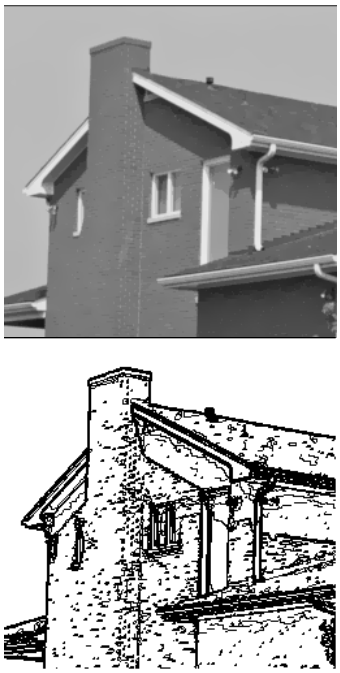
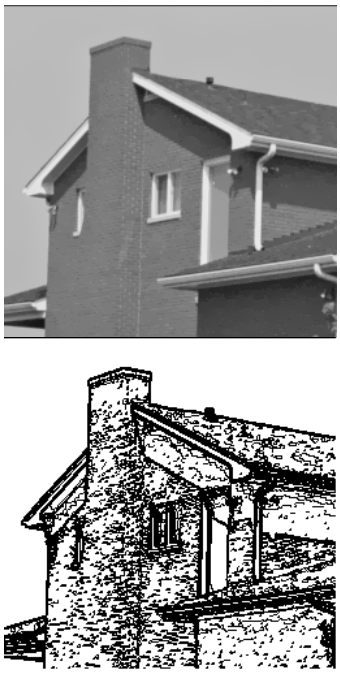
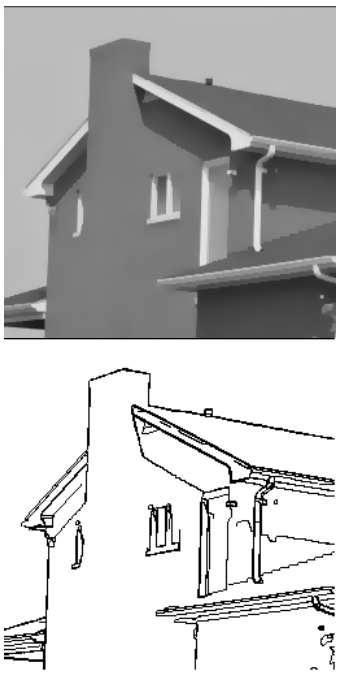
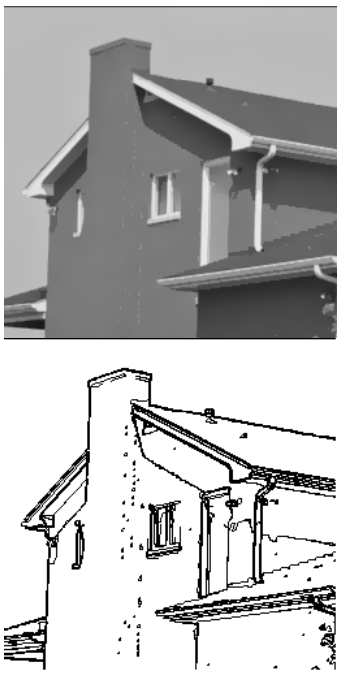
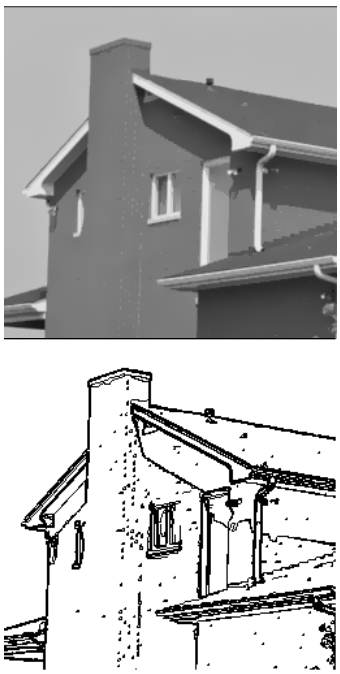
Perona-Malik 1 (g_1)	Perona-Malik 2 (g_2)	RAD (g_3)
		
(3.6a) $\sigma = 0{,}02$	**(3.6b)** $\sigma = 0{,}02$	**(3.6c)** $\sigma = 0{,}02$
		
(3.6d) $\sigma = 0{,}04$	**(3.6e)** $\sigma = 0{,}04$	**(3.6f)** $\sigma = 0{,}04$

Fig. 3.6: Continua na próxima página.

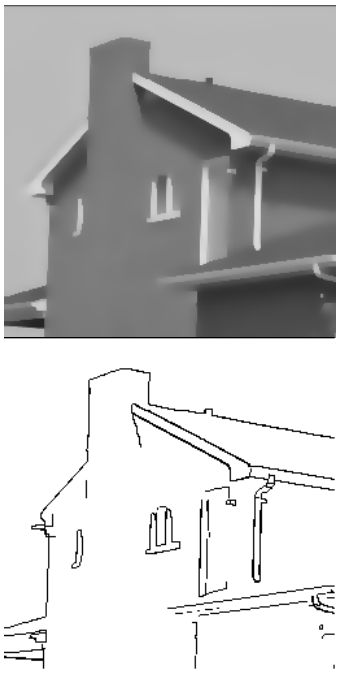
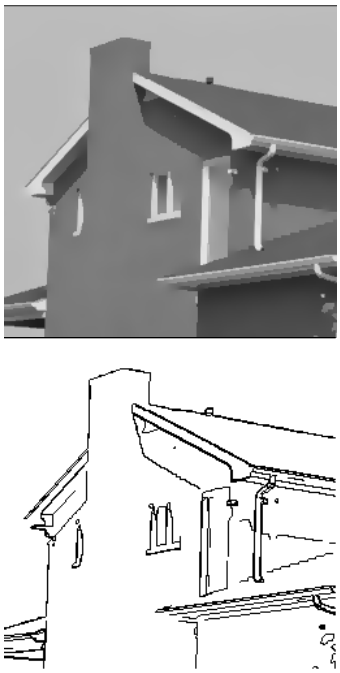
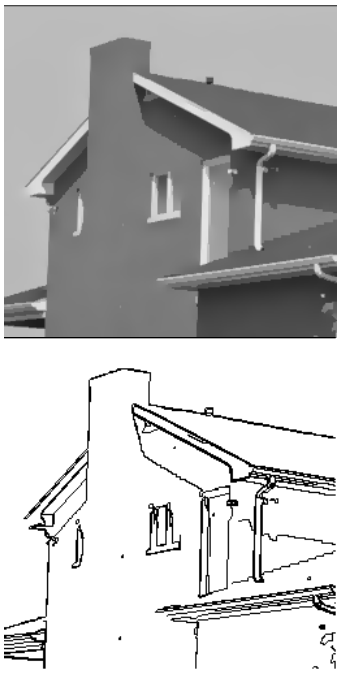
Perona-Malik 1 (g_1)	Perona-Malik 2 (g_2)	RAD (g_3)
		
(3.6g) σ = 0,08	**(3.6h)** σ = 0,08	**(3.6i)** σ = 0,08

Fig. 3.6: Detecção de arestas usando a difusão anisotrópica com diferentes funções parada-na-aresta e várias escalas σ. O número de iterações foi mantido fixo em t_{max} = 50.

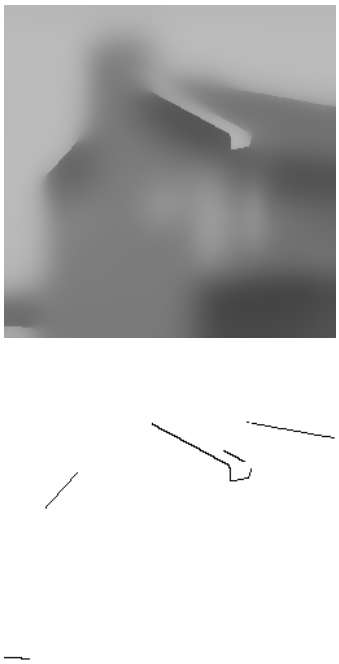
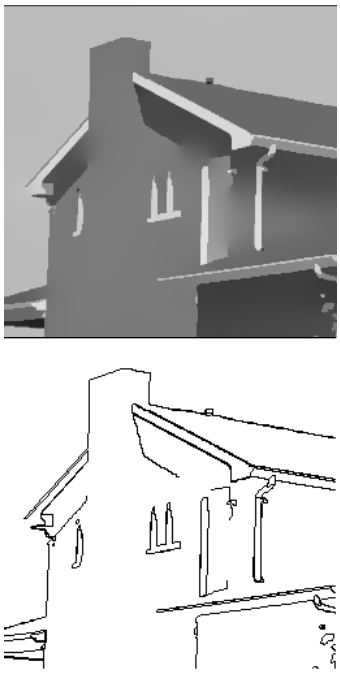
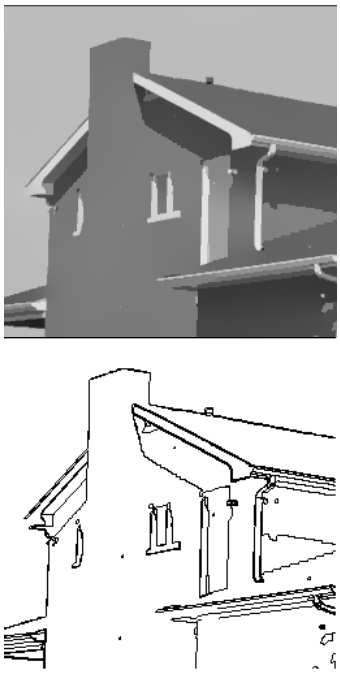
Perona-Malik 1 (g_1)	Perona-Malik 2 (g_2)	RAD (g_3)
		
(3.7a) t_{max} = 500, σ = 0,08	**(3.7b)** t_{max} = 500, σ = 0,08	**(3.7c)** t_{max} = 500, σ = 0,08

Fig. 3.7: Comportamento da difusão anisotrópica com grande número de iterações (t_{max} = 500). A função parada-na-aresta Perona-Malik 1 acaba borrando as arestas. A função de Tukey é a que consegue manter as arestas mais nítidas, pois está baseada na estatística robusta. Compare com a última linha da figura 3.6, onde tínhamos t_{max} = 50.

3.4 Melhoramento da Tomografia pela RAD

Introdução

Esta seção descreve uma contribuição científica original nossa. O principal responsável por esta contribuição foi o meu ex-orientando de doutorado Harold I. A. Bustos.

A reconstrução de imagem (ou tomografia) é a técnica usada para obter a distribuição de um meio não observável diretamente através das projeções. Existem muitas técnicas de reconstrução tomográfica, por exemplo, a retro-projeção filtrada, a transformada de Fourier, a reconstrução aritmética [Jain, 1989], a entropia máxima [Reis and Roberty, 1992; Dusassoy and Abdou, 1991], etc. Aqui, estamos interessados nas técnicas tomográficas que geram boas imagens reconstruídas mesmo usando uma pequena quantidade de dados de projeção. Em muitas situações práticas, é vantajoso minimizar a irradiação, para não danificar a amostra sendo irradiada ou para apressar a aquisição dos dados. O algoritmo de máxima entropia (MENT) é uma das melhores técnicas quando somente poucos dados estão disponíveis. Porém, mesmo este método produz imagens borradas com ruídos e artefatos numa situação com dados escassos. Delaney e Bresler [Delaney and Bresler, 1998] propuseram uma técnica tomográfica especialmente projetada para gerar imagens clarar a partir de poucos dados. Porém, na prática este algoritmo requer uma quantidade bastante grande de dados para gerar uma imagem nítida, conforme discutiremos mais adiante.

Conforme vimos nas seções anteriores, a difusão anisotrópica é uma técnica bem conhecida usada para filtragem, detecção de arestas e análise multi-escala de imagens. Recentemente, Black et al. descreveram a relação entre a difusão anisotrópica e a estatística robusta, resultando numa técnica teoricamente bem fundamentada denominada difusão anisotrópica robusta (RAD) [Black et al., 1998].

Nesta pesquisa, melhoramos o algoritmo MENT utilizando a RAD. A idéia principal do novo algoritmo é intercalar, em cada passo iterativo do MENT, uma filtragem

RAD. Esta idéia é completamente diferente de simplesmente pós-filtrar com a RAD uma imagem gerada pelo MENT. Como o algoritmo MENT gera imagens muito ruidosas e borradas numa situação de poucos dados, nenhuma pós-filtragem consegue melhorar substancialmente a qualidade da imagem gerada. Porém, incorporando a filtragem RAD no algoritmo MENT, imagens nitidamente reconstruídas podem ser obtidas mesmo usando poucos dados de projeção.

Descrevemos as nossas idéias para um cenário de aquisição de dados em feixe paralelo e ângulo completo (180^o) sem nenhum dado em falta. Implementamos e testamos a técnica proposta somente para esta situação. Porém, as idéias desenvolvidas aqui podem ser aplicadas de forma direta para qualquer circunstância de aquisição de dados: feixe paralelo ou em leque, ângulo completo ou limitado, com ou sem dados em falta. Testamos o nosso algoritmo usando uma quantidade extremamente pequena de dados. Sob esta condição, o algoritmo MENT original gera imagens ruidosas e borradas, onde mesmo as arestas importantes não são claramente reconstruídas. O nosso algoritmo gerou imagens nítidas.

Algoritmo de Máxima Entropia Estendida

Minerbo propôs o algoritmo MENT em [Minerbo, 1979] e depois muitos autores propuseram diferentes melhoramentos a este algoritmo. Por exemplo, Dusassoy e Abdou [Dusassoy and Abdou, 1991] introduziram o algoritmo MENT-estendido que pode levar em conta uma informação a priori sobre a imagem a ser reconstruída. Isto é, se uma aproximação f^* da imagem a ser reconstruída f for conhecida, este conhecimento pode ser usado para melhorar a reconstrução.

O funcional de Lagrange abaixo representa o custo funcional do MENT-estendido, sujeito às restrições dos dados de projeção:

$$L(f,\Lambda) = -\iint f(x,y)\log\left[\frac{f(x,y)}{ef^*(x,y)}\right]dxdy$$
$$-\sum_{j=1}^{J}\sum_{n=1}^{N}\Lambda_{j,n}\left[h_{j,n} - \iint f(x,y)\chi_{j,n}(x,y)\,dxdy\right]$$

onde:

- *e* é a base neperiana (2,71828...).
- $h_{j,n}$ é a intensidade do *n*-ésimo raio da projeção *j*.
- $\Lambda_{j,n}$ é o parâmetro de Lagrange associado à faixa (*j*, *n*). Se os dados fossem completos, este coeficiente seria sempre um.
- $\chi_{j,n}$ é a função indicadora da faixa (*j*, *n*). Esta função é 1 dentro da faixa (*j*, *n*) e 0 fora.
- $f^*(x,y)$ é o modelo a priori do objeto $f(x,y)$. Se nenhuma informação a priori estiver disponível, $f^*(x,y)$ pode receber o valor e^{-1}. Neste caso, MENT-estendido torna-se o algoritmo original MENT de Minerbo.

A otimização da equação acima permite-nos achar a solução do problema de reconstrução:

$$f^{(i)}(x,y) = f^*(x,y)\prod_{j=1}^{J}\sum_{n=1}^{N} F_{j,n}^{(i)}\chi_{j,n}(x,y), \quad 1 \le i \le \zeta \tag{3.3}$$

onde $F_{j,n}$ é a matriz dos parâmetros duais de Lagrange associada à faixa (*j*, *n*). Esses parâmetros são obtidos pelo seguinte sistema iterativo:

$$F_{j,n}^{(i)} = \begin{cases} z\,h_{j,n}, \; i = 0 \\ \dfrac{h_{j,n}}{\iint f^*(x,y)\prod\limits_{\substack{k=1 \\ k\neq j}}^{J}\sum\limits_{n=1}^{N}[F_{k,n}^{(i-1)}\chi_{k,n}(x,y)]\chi_{j,n}(x,y)\,dx\,dy}, 1 \le i \le \zeta \end{cases} \tag{3.4}$$

onde *z* é a largura da faixa (*j*, *n*). Após calcular $F_{j,n}^{(i)}$, eles devem ser inseridos em (3.3) para achar a imagem reconstruída na *i*-ésima iteração $f^{(i)}(x,y)$.

MENT Reconstrução-Difusão

Vamos supor que a imagem *f* a ser reconstruída seja constante por regiões. Conforme notamos antes, numa situação com poucos dados, o algoritmo MENT-estendido irá

reconstruir uma versão ruidosa e borrada da imagem *f*. Esta imagem pode estar tão fortemente corrompida que nenhuma pós-filtragem pode melhorá-la substancialmente. A RAD é um excelente estimador da imagem original *f* a partir da sua versão corrompida. Porém, utilizando-a como um processo de pós-filtragem, somente um ligeiro melhoramento pode ser obtido. A nossa idéia é usar esta imagem ligeiramente melhorada como o conhecimento a priori f^* do algoritmo MENT-estendido. Isto irá gerar uma imagem reconstruída de melhor qualidade. Esta imagem melhorada pode ser melhorada ainda mais pela RAD e usada como um novo conhecimento a priori pelo MENT-estendido, e assim por diante.

Inicialmente, aplicamos o algoritmo MENT-estendido, iterando a equação (3.4) ζ vezes. Usando a equação (3.3), a imagem reconstruída $f^{(\zeta)}$ é obtida. Esta imagem é filtrada pela RAD, iterando a equação (3.2) uma ou mais vezes. A imagem filtrada é usada como conhecimento a priori f^* pelo MENT-estendido para obter uma nova imagem reconstruída. Esta imagem é novamente filtrada pela RAD, e assim por diante.

Resultados Experimentais

Para mostrar a eficácia da técnica proposta, executamos alguns experimentos. O objeto teste simulado é um cilindro com diâmetro 100 e densidade 5, imerso no meio com densidade 0 (figura 3.8a). Este cilindro contém 5 cilindros menores com diâmetros e densidades variadas.

Experiências consistiram em reconstruir imagens 100×100 a partir de apenas 6 projeções paralelas distribuídas em ângulo completo (180^o), cada projeção com 100 raios irradiados. A imagem 3.8b foi obtida pelo algoritmo original MENT-estendido, iterando 10 vezes a equação 3.4. Esta imagem foi filtrada pela RAD (σ=32, 100 iterações), gerando a imagem 3.8c. Esta imagem foi utilizada como a estimativa inicial do MENT reconstrução-difusão. A reconstrução-difusão (isto é, uma execução de (3.4) seguida por uma execução de (3.2)) foi iterada 9 vezes (com σ=32), gerando a imagem 3.8d. Sem dúvida, o novo algoritmo gerou uma imagem melhor. As médias das

diferenças absolutas entre a imagem ideal 3.8a e as imagens 3.8b, 3.8c, 3.8d foram respectivamente 12,7%, 12,8% e 7,5%.

Para mostrar a superioridade da nossa proposta, vamos definir uma projeção reconstruída $h_{j,n}^{(i)}$, calculada a partir da imagem reconstruída na *i*-ésima iteração $f^{(i)}$ como:

$$h_{j,n}^{(i)} = \int\int f^{(i)}(x,y)\chi_{j,n}(x,y)\,dx\,dy\,.$$

Vamos definir a norma euclidiana $k^{(i)}$ entre as projeções originais $h_{j,n}$ e as projeções calculadas a partir da imagem reconstruída $h_{j,n}^{(i)}$ como:

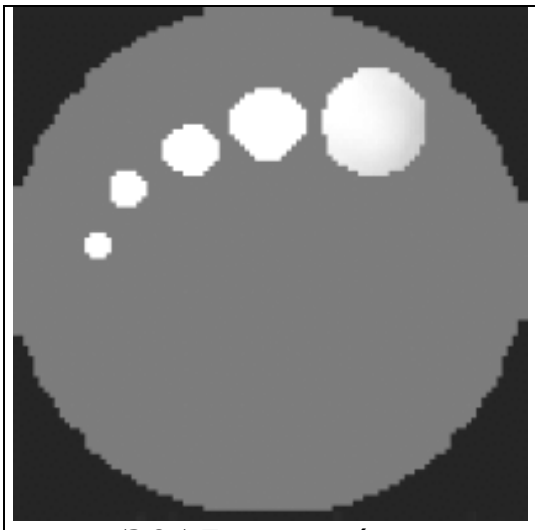
$$k^{(i)} = \sqrt{\sum_{j=1}^{J}\sum_{n=1}^{N}\left(h_{j,n}^{(i)} - h_{j,n}\right)^2}\,.$$

A figura 3.9 mostra as normas euclidianas $k^{(i)}$ em diferentes iterações da reconstrução. As primeiras 10 iterações correspondem ao MENT-estendido original e as últimas 9 iterações ao MENT reconstrução-difusão proposto. Sem dúvida, a norma euclidiana converge mais rapidamente utilizando o algoritmo proposto. O parâmetro de escala σ=32 foi escolhido para maximizar a convergência da norma euclidiana.

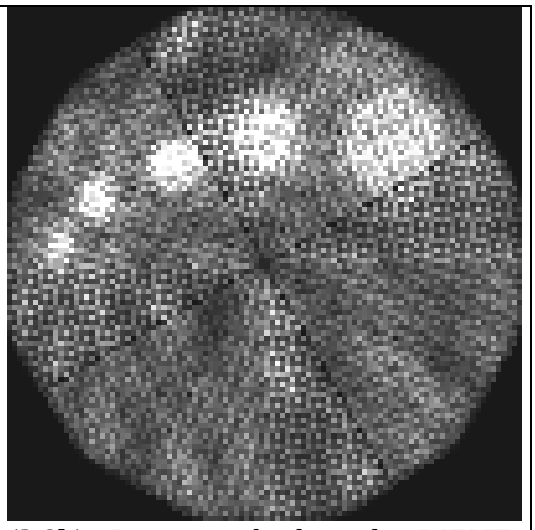
Um outro fantom gerado artificialmente está ilustrado na figura 3.10a. Foram irradiadas seis projeções paralelas distribuídas sobre 180^o com 100 raios por projeção (600 raios ao todo). O algoritmo MENT-estendido original foi executado sobre estes dados, gerando a imagem 3.10b. Esta imagem foi filtrada pela RAD (σ=50, 70 iterações), gerando a imagem 3.10c. Olhando esta figura, fica claro que uma pós-filtragem não consegue gerar uma imagem reconstruída nítida. A imagem 3.10c foi usada como a estimativa inicial da imagem a ser reconstruída pela MENT reconstrução-difusão. Este algoritmo foi iterado 70 vezes usando parâmetro de escala σ=50, gerando a imagem 3.10d. Sem dúvida, o algoritmo proposto gerou a melhor imagem. Todas as imagens têm resolução de 100×100 pixels. As médias das diferenças absolutas entre a imagem ideal 3.10a e as imagens 3.10b, 3.10c e 3.10d foram respectivamente 14,89%, 15,22% e 8,15%. Usando o bem-conhecido algoritmo de retro-

projeção filtrada sobre os mesmos dados, a imagem de baixa qualidade 3.10e foi obtida.

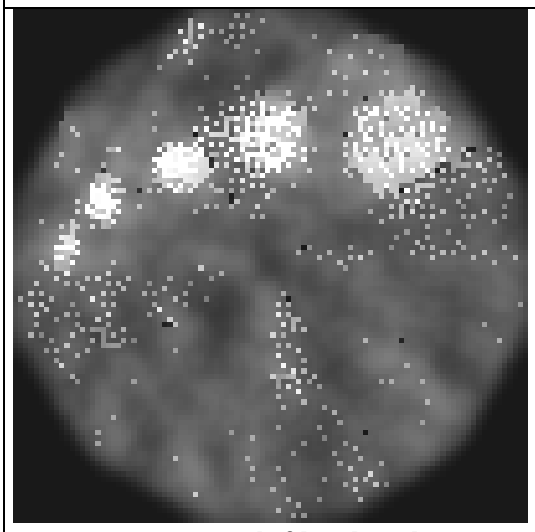
Devido à relevância do trabalho de Delaney e Bresler [Delaney and Bresler, 1998], comparamos rapidamente os nossos resultados com os deles. As experiências descritas em [Delaney and Bresler, 1998] coletam uma projeção a cada grau, em ângulo completo (-90°, 90°) ou limitado (-75°, 75°), com somente um ou dois ângulos em falta. No nosso caso, uma projeção paralela é coletada a cada 30°, em ângulo completo. Portanto, o nosso problema é muito mais severamente subdeterminado que o problema considerado por Delaney e Bresler. O nosso algoritmo pode reconstruir imagens nítidas a partir dos dados de projeção altamente subdeterminados.



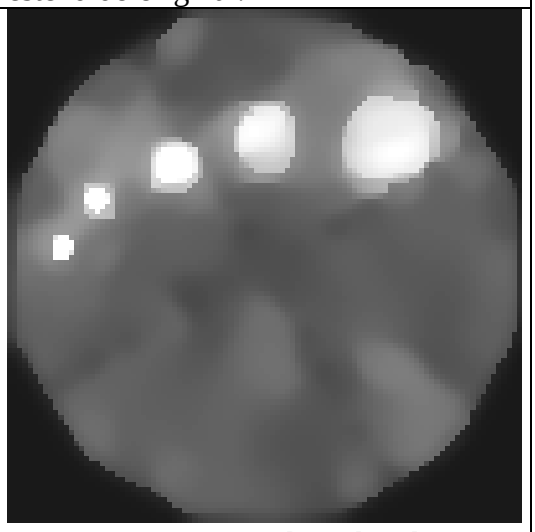
(3.8a) Fantom sintético.



(3.8b) Imagem obtida pelo MENT-estendido original.



(3.8c) Imagem 3.8b filtrada pela RAD.



(3.8d) Imagem gerada pelo algoritmo MENT reconstrução-difusão proposto.

Fig. 3.8: Comparação do algoritmo MENT-estendido com o MENT reconstrução-difusão.

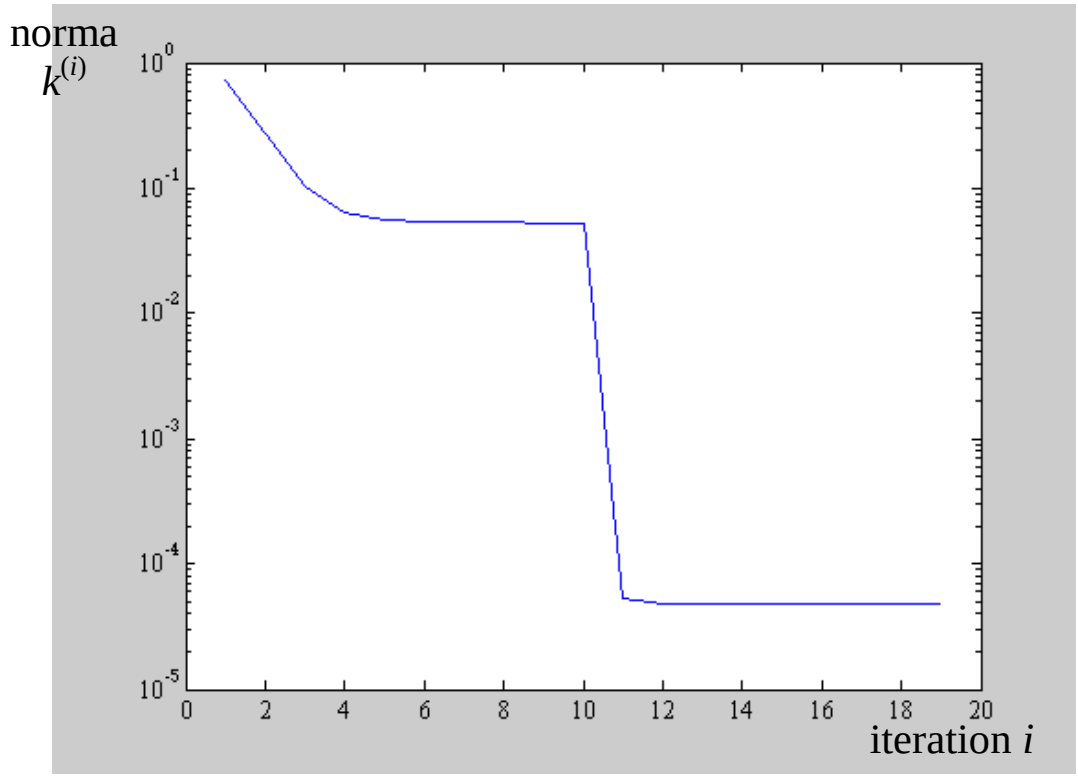
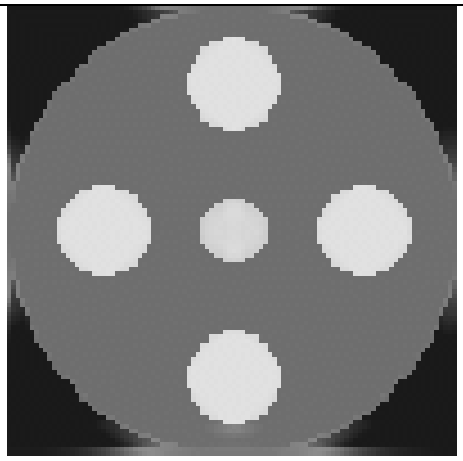
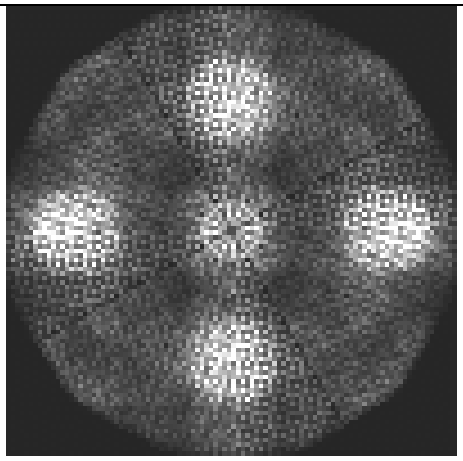


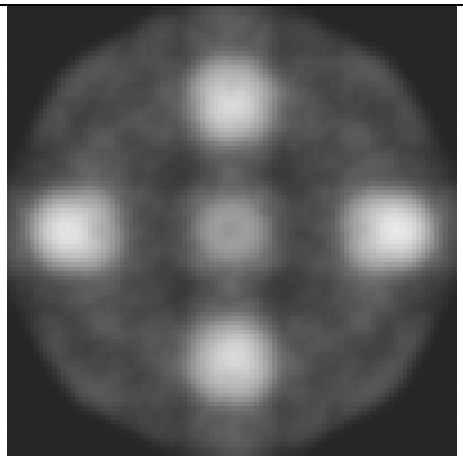
Fig. 3.9: Norma euclidiana $k^{(i)}$ da diferença entre as projeções originais e as projeções calculadas a partir das imagens reconstruídas na i-ésima iteração. As primeiras 10 iterações correspondem ao algoritmo MENT estendido original e as últimas 9 iterações correspondem ao MENT reconstrução-difusão proposto.



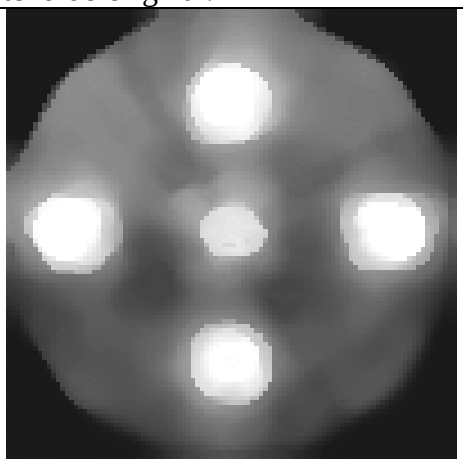
(3.10a) Fantom sintético.



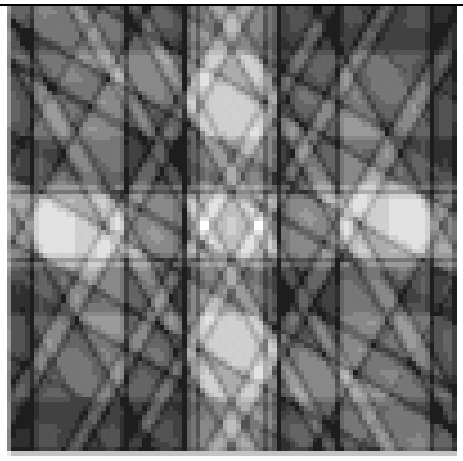
(3.10b) Imagem obtida pelo MENT-estendido original.



(3.10c) Imagem 3.10b filtrada pela RAD.



(3.10d) Imagem gerada pelo algoritmo MENT reconstrução-difusão proposto.



(3.10e) Imagem reconstruída pela retroprojeção filtrada.

Fig. 3.10: Comparação entre os algoritmos MENT-estendido, MENT reconstrução-difusão e retroprojeção filtrada, utilizando os mesmos dados.

3.5 Melhoramento da fMRI pela RAD

Introdução

Esta seção descreve uma contribuição científica original minha. Eu fui o principal responsável por esta contribuição, e o trabalho foi realizado em colaboração com Prof. H. Z. Cho da University of California, Irvine.

O objetivo do imageamento funcional é mapear as atividades de um cérebro vivo no espaço e no tempo. O "padrão ouro" para medir a atividade celular cerebral envolve a gravação, direta e invasiva, do potencial elétrico da membrana dos neurônios individuais. Porém, tais medidas são limitadas a certas condições experimentais. Para estudos em sujeitos humanos, os métodos não-invasivos tais como PET (positron emission tomography) ou fMRI (functional magnetic resonance imaging) devem ser aplicados.

Desde o início dos anos 80, a PET dominou o campo de neuroanatomia funcional, mas nos últimos 10 anos a fMRI desenvolveu-se numa técnica alternativa e poderosa. Aumentos locais na atividade neuronial provocam a relativa desoxigenação do sangue e o aumento da perfusão, que rapidamente reverte a desoxigenação inicial, levando ao aumento da oxigenação que dura durante vários segundos. A fMRI é sensível a oxigenação do sangue (efeito normalmente abreviado como BOLD - blood oxygen level dependent) e possui a escala espaço-temporal de aproximadamente 1-3 mm e 1 ou mais segundos. Os limites inferiores da resolução efetiva da fMRI são fisiológicos e impostas pela organização espaço-temporal da resposta hemodinâmica evocada (2-5 mm e 5-8 segundos).

Em virtude da meia-vida relativamente longa dos traçadores radioativos utilizados, a PET pode medir somente respostas somadas durante um período de tempo razoavelmente longo. Em contraste, a fMRI pode ser usada de segundo paradigmas periódico (estímulos repetidos) ou relacionado a eventos. No paradigma periódico, o sujeito se

alterna entre os períodos de estímulo e descanso. Neste caso, cada voxel da fMRI consiste numa série temporal que pode ser dividido em “épocas”, sendo cada época o período de tempo que corresponde à atividade seguida pelo descanso. No paradigma relacionado a eventos, o sujeito executa a atividade durante somente um período curto de tempo.

A baixa razão sinal-ruído nas imagens fMRI obriga o uso de sofisticadas técnicas de Processamento e Análise de Imagens para detectar as áreas ativadas do cérebro. Em primeiro lugar, os dados devem passar através de transformações espaciais para corrigir o movimento da cabeça do sujeito durante a aquisição de fMRI. Se a experiência envolver sujeitos diferentes, os dados devem além disso ser normalizados, isto é, as imagens devem ser arqueadas de forma que todas elas se conformem a algum cérebro padrão.

Depois das transformações espaciais, as análises estatísticas são efetuadas. Muitos procedimentos estatísticos diferentes foram propostos para analisar os dados fMRI dependentes do nível de oxigenação [Lange et al., 1999; Gold et al., 1998]. Um dos procedimentos estatísticos mais populares é o modelo linear geral [Friston et al., 1995]. Neste modelo, o usuário especifica manualmente uma “matriz de projeto” (design matrix) e faz uso da regressão linear múltipla para estimar os parâmetros, isto é, determinar quão bem a série temporal de cada voxel se encaixa dentro da matriz de projeto especificada. Estes parâmetros são então utilizados para computar a significância estatística de um efeito. Estas estatísticas, dispostas espacialmente, formam o mapa estatístico paramétrico (SPM - statistical parametric map). As notas de curso [Friston, 1997] são uma boa referência sobre o processamento de fMRI através do modelo linear geral.

Mesmo com todos esses aparatos de processamento de imagens, uma fMRI ruidosa sempre dá origem a um SPM ruidoso. Os filtros passa-baixas simples não podem ser usados indiscriminadamente pois eles borram as arestas das áreas ativadas. As técnicas tradicionais de filtragem que preservam as arestas também não podem ser usadas pois não existe uma fronteira clara entre as áreas ativadas e não-ativadas.

Na literatura existem muitos trabalhos para atenuar o ruído e aglutinar as regiões ativadas nos dados fMRI [Goutte et al., 1999; Ardekani and Kanno, 1998; Kershaw et al., 1999; Chuang et al., 1999; Friston et al., 1994]. Em particular, Solé et al. [Solé et al., 2001] propuseram recentemente uma técnica denominada "média anisotrópica" (anisotropic averaging). Esta técnica foi inspirada na difusão anisotrópica, introduzida por Perona e Malik [Perona and Malik, 1990]. A média anisotrópica calcula um conjunto inicial de voxels claramente ativados utilizando os coeficientes de correlação. Este conjunto é então utilizado para construir uma complexa "medida de similaridade" para calcular os pesos da média ponderada. Apesar de Solé et al. tentar explicar a definição da sua medida de similaridade com argumentos intuitivos, somos impelidos a perguntar se não existiria uma forma mais natural e simples de definir essa medida. Além disso, a sua técnica pode ser usada somente para processar fMRI periódica, pois a sua medida de similaridade está baseada na transformada de Fourier da série temporal de cada voxel. Num fMRI relacionado a eventos, a transformada de Fourier não faz o mínimo sentido.

Nos trabalhos [Cn12; Su01], propusemos uma outra técnica para obter um SPM nítido a partir de fMRI ruidosa utilizando o modelo linear geral. Em vez de definir uma medida de similaridade altamente complexa baseada no conjunto de voxels claramente ativados, usamos a magnitude do gradiente dos parâmetros estimados como argumentos para calcular os coeficientes de difusão. Substituímos a média anisotrópica pela difusão anisotrópica robusta [Black et al., 1998]. Esta técnica pode ser usada para processar fMRI tanto periódico como relacionado a eventos.

Modelo Linear Geral

O modelo linear geral é simplesmente uma equação que relaciona o que se observa com o que se esperaria observar, expressando as observações como uma combinação linear dos componentes esperados e algum erro residual. O modelo linear geral pode ser escrito como [Friston et al., 1995]:

$$Y = X\beta + \varepsilon .$$

Geralmente, todas as variáveis envolvidas na equação acima são matrizes. Porém, para simplificar a exposição, iremos supor que *Y* é um vetor coluna das observações, β é um vetor coluna dos parâmetros, e ε é um vetor coluna dos erros. *X* é a matriz de projeto (design matrix) com uma linha por observação e duas colunas: A primeira coluna é o parâmetro do modelo e a segunda coluna é "fantoche", inteiramente preenchido com 1, cuja finalidade é corrigir a média das observações. O modelo linear geral assume que os erros ε_j são variáveis aleatórias normais, independentes e identicamente distribuídas.

A seguinte equação executa a estimação de mínimos quadrados dos parâmetros:

$$\hat{\beta} = (X^T X)^{-1} X^T Y .$$

$\hat{\beta}$ é na verdade um vetor coluna com duas linhas. Porém, somente a primeira coluna $\hat{\beta}_1$ é útil, pois $\hat{\beta}_2$ é "fantoche". Chamaremos a imagem obtida dispondo espacialmente os parâmetros obtidos $\hat{\beta}_1$ como EPM (estimated parameters map). Um EPM *B* pode ser transformado num SPM (statistical parametric map) através de alguns cálculos. Por exemplo, dividindo o valor de um voxel de *B* pelo seu respectivo erro padrão, a estatística *t* de Student é obtida. O mapa paramétrico estatístico das estatísticas *t* de Student é denotado como SPM{t}.

O seguinte exemplo numérico clarifica essas idéias:

$$\begin{bmatrix} 50 \\ 51 \\ 60 \\ 62 \\ 51 \\ 52 \\ 62 \\ 63 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 1 \\ 1 & 1 \\ 1 & 1 \\ 0 & 1 \\ 0 & 1 \\ 1 & 1 \\ 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} \hat{\beta}_1 \\ \hat{\beta}_2 \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \\ \varepsilon_6 \\ \varepsilon_7 \\ \varepsilon_8 \end{bmatrix}$$

A primeira matriz *Y* é uma série temporal dos valores de um voxel. Vamos supor que os valores foram adquiridos a cada 5 segundos. A segunda matriz *X* é a matriz de projeto. A primeira coluna indica que gostaríamos de detectar uma ativação que se repete a cada 20 segundos e dura 10 segundos. A segunda coluna é "fantoche", colocada na matriz de projeto para corrigir a média dos valores de *Y*. Estimando os parâmetros, obtemos $\hat{\beta}_1 = 10{,}75$ e $\hat{\beta}_2 = 51{,}00$. O alto valor de $\hat{\beta}_1$ indica que muito provavelmente este voxel está ativado.

Para transformar $\hat{\beta}_1$ numa distribuição *t* de Student t_1 com *r* graus de liberdade, aplicamos a equação abaixo:

$$t_1 = \frac{\hat{\beta}_1}{\sqrt{\varepsilon^T \varepsilon / r}} = 10{,}947$$

onde $r = 7$ é a quantidade de observações menos um. Isto significa que, se o voxel considerado fosse completamente não relacionado com a matriz de projeto, existe uma probabilidade $5{,}9 \times 10^{-6}$ de resultar um parâmetro maior que 10,75 por acaso.

Média Anisotrópica

A difusão anisotrópica foi aplicada com sucesso para MRI estrutural [Gerig et al., 1992]. Porém, esta técnica não pode ser aplicada diretamente em fMRI, principalmente devido à falta de bordas claras entre as regiões ativadas e não-ativadas. Também não pode ser aplicada diretamente em SPM, pois SPM é normalmente muito ruidoso e filtrá-lo pela difusão anisotrópica somente aumentaria as imperfeições do SPM.

Fazer uma convolução da fMRI com uma máscara para suavização geralmente aumenta a razão sinal a ruído. Porém, este procedimento também borra as arestas entre as regiões ativadas e não-ativadas. Portanto, é desejável efetuar somente a suavização intra-região, evitando a suavização inter-região.

Solé et al. [Solé et al., 2001] propuseram esta idéia e chamaram-na de média anisotrópica (anisotropic averaging). Consiste em calcular uma média seletiva da vizinhança do sinal. Seja *I* uma imagem fMRI e seja $I(s,n)$ o valor do voxel na posição espacial *s* e no volume (isto é, na aquisição ou na observação) $n \in [1...N]$. A média anisotrópica irá aos poucos modificar *I*. Vamos denotar imagem fMRI na iteração $t \geq 0$ como $I(s,n,t)$, sendo $I(s,n,0) = I(s,n)$. Então, a média aritmética consiste em calcular:

$$I(s,n,t+1) = \frac{1}{\sum_{p \in \eta_s} w(s,p)} \sum_{p \in \eta_s} w(s,p) I(p,n,t),$$

para todos os voxels *s*, todos os volumes *n* e as iterações $t \geq 0$. Como antes, η_s representa o conjunto dos vizinhos espaciais do voxel *s*.

Seja $I(s)$ a série temporal do sinal no voxel *s*. A medida de similaridade Ψ determina os pesos $w(s,p)$, utilizando a série temporal $I(s)$ que sofrerá o processo de média anisotrópica e a série temporal do seu voxel vizinho $I(p)$:

$$w(s,p) = \Psi(I(s), I(p)).$$

Esta medida de similaridade permite-nos distinguir voxels ativados daqueles não-ativados. Permite-nos calcular uma média seletiva, combinando somente os sinais da mesma classe. Solé et al. propuseram calcular um conjunto inicial Ω de voxels claramente ativados escolhendo os voxels com alto coeficiente de correlação com a matriz de projeto. Depois, os espectros de Fourier dos voxels em Ω são calculados para definir a função de medida de similaridade Ψ. O espectro de Fourier da série temporal de cada voxel *s* é também calculado para avaliar a similaridade entre *s* e os voxels em Ω. O procedimento todo é altamente complexo e os leitores são encaminhados a [Solé et al., 2001] para maiores detalhes. Gostaríamos de perguntar: “todos esses cálculos são realmente necessários?” Mais, “como esta técnica pode ser aplicada no protocolo de aquisição de fMRI relacionado a eventos?”

Algoritmos

Propomos uma abordagem diferente, motivada diretamente pela difusão anisotrópica robusta, para filtrar fMRI. O nosso método é mais simples, pode ser aplicado a fMRI relacionado a eventos, e tem gerado SPMs surpreendentemente nítidos. Este método também aumentou a significância estatística do SPM, o que nos permite decidir com mais confiança se um voxel está ativado ou não. A nossa técnica está descrita abaixo.

Seja dada uma fMRI *I* e uma matriz de projeto *X*. Seja $I(s,n)$ o valor de *I* na posição espacial *s* e no volume $n \in [1...N]$. Vamos denotar a imagem fMRI na iteração $t \geq 0$ como $I(s,n,t)$, com $I(s,n,0) = I(s,n)$.

Usando a fMRI *I* e a matriz de projeto *X*, estime os parâmetros $\hat{\beta}$ para cada voxel *s*, como descrevemos anteriormente. Estes parâmetros, dispostos espacialmente, formam o EPM *B*. Vamos denotar como $B(s,t)$ o valor de *B* no voxel *s* e iteração *t*. A magnitude do gradiente de $B(s,0)$ será usada como o argumento da função "parada-na-aresta" *g* para calcular os coeficientes de difusão $g\left(\left|\nabla B_{s,p}(0)\right|\right)$ no instante $t = 0$, onde:

$$\nabla B_{s,p}(t) = B(p,t) - B(s,t), \quad p \in \eta_s.$$

Estes coeficientes são usados para executar a difusão em fMRI $I(s,n,0)$, gerando a fMRI difundida $I(s,n,1)$ no instante $t = 1$. $I(s,n,1)$ é então usada para estimar o novo EPM $B(s,1)$ na iteração $t = 1$. Estes passos são repetidos até que a média do valor difundido esteja abaixo de algum limiar predefinido. É também possível especificar o número de iterações desejado, em vez de definir um limite para o valor médio difundido. A seguinte equação descreve este processo:

$$I(s,n,t+1) = I(s,n,t) + \frac{\lambda}{|\eta_s|} \sum_{p \in \eta_s} g\left(\left|\nabla B_{s,p}(t)\right|\right) \nabla I_{s,p}(t),$$

para todos os voxels *s*, todos os volumes *n* e as iterações $t \geq 0$.

A melhor função "parada na aresta" *g* é a função biweight de Tukey. Note que a escolha correta do parâmetro de escala σ da função de Tukey é essencial para gerar bons resultados.

Quando o processo de difusão terminar em alguma iteração t_f, o EPM $B(s,t_f)$ pode ser transformado em SPM{t} usando o procedimento descrito anteriormente.

Resultados e Dados Experimentais

A figura 3.11 mostra uma parte de um fantom fMRI simulado com 10×10×3 voxels por volume e 64 volumes. Todos valores dos voxels eram 500 originalmente. O ruído gaussiano com média zero e desvio-padrão 10 foi somado aos valores originais. Os volumes 3, 4, 7, 8, 11, 12, ... têm um quadrado 6×6 ativado no centro do volume, com dois furos não-ativados de 4 voxels cada um. Os voxels ativados tiveram seus valores aumentados de 20.

O SPM{t} obtido pelo modelo linear geral sem filtragem é apresentado na coluna à esquerda da figura 3.12. Entre os voxels ativados, o menor valor foi 1,48 e o maior 2,78. Usando a distribuição *t* de Student com 63 graus de liberdade, podemos inferir que existe uma probabilidade 7,2% de um voxel não-ativado assumir valores maiores que 1,48 por acaso. Entre os voxels não-ativados, o menor e o maior valores foram -0,93 e 0,74, respectivamente.

O SPM{t} obtido filtrando fMRI com o método proposto (usando a função de Tukey com $\sigma = 10$) é mostrado na coluna direita da figura 3.12. Note que o SPM filtrado é completamente sem ruído, e as arestas estão perfeitamente preservadas. Todos os voxels ativados apresentaram valores aproximadamente 12,1, significando que a confiança estatística melhorou consideravelmente. Virtualmente, é impossível que um voxel não-ativado assuma valores tão altos por acaso (probabilidade menor que 10^{-16}). Os valores dos voxels não-ativados ficaram no intervalo de -0,22 a 0,45.

A figura 3.13 mostra fMRI real com 79×95×68 voxels por volume e 12 volumes. Os volumes foram realinhados para corrigir o movimento da cabeça do paciente. Depois,

os volumes foram embaralhados aleatoriamente para remover qualquer sinal de ativação que possa estar presente. Ativamos artificialmente pequenas regiões esféricas nos volumes 3, 4, 7, 8, 11 e 12 aumentando o valor dos voxels em 3%. A coluna esquerda da figura 3.13 mostra três fatias do volume 1, sem ativação. A coluna direita da figura 3.13 mostra três fatias do volume 3, com regiões ativadas artificialmente. Note que as áreas ativadas são completamente invisíveis a olho nu.

A coluna esquerda da figura 3.14 mostra o SPM{t} obtido sem filtragem. O voxel ativado com o menor valor tinha valor 1,3. Assumindo a distribuição *t* de Student com 11 graus de liberdade, um voxel não-ativado pode assumir valor maior que 1,3 com probabilidade 11%. A coluna direita da figura 3.14 é obtida limiarizando SPM{t} na altura 2,2. Note que muitas áreas não-ativadas foram falsamente detectadas como ativadas (e vice-versa).

A figura 3.15 mostra o SPM obtido filtrando fMRI com o método proposto (coluna esquerda) e imagens limiarizadas correspondentes (coluna direita). A maioria do ruído foi removida e a imagem limiarizada está perfeita: não há nem voxels não-ativados falsamente detectados como ativados, nem voxels ativados falsamente detectados como não-ativados. O voxel ativado com o menor valor tem valor 2,4, uma melhoria considerável sobre 1,3 anterior.

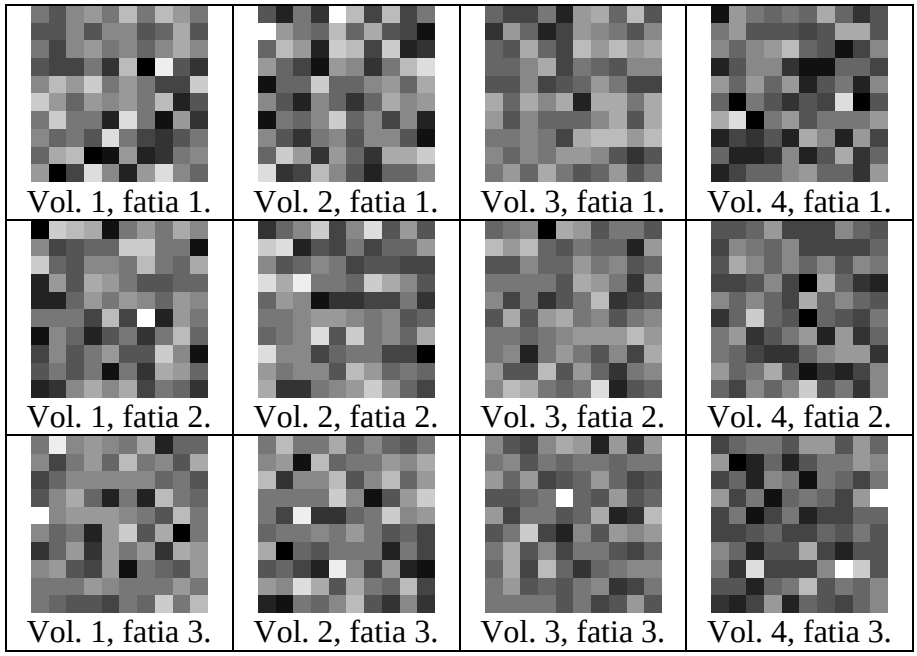


Fig. 3.11: Um fantom fMRI simulado com 10×10×3 voxels por volume e 64 volumes. Somente os 4 primeiros volumes estão mostrados. Os valores de todos os voxels são 500, somados a um ruído gaussiano com média zero e desvio-padrão 10. Volumes 3, 4, 7, 8, 11, 12, ... tiveram alguns voxels ativados onde 20 foi somado aos valores originais.

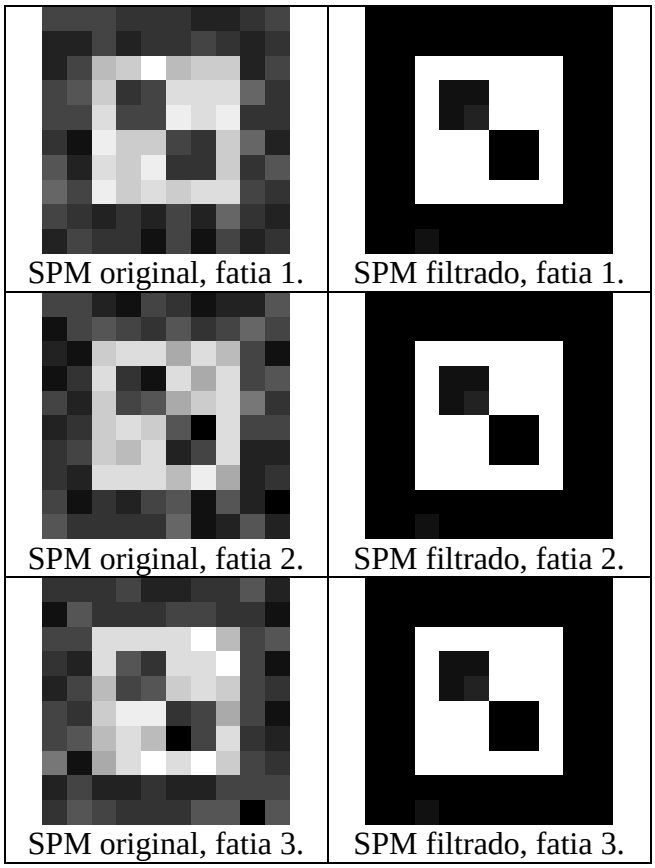


Fig. 3.12: SPM{t} obtido da fMRI da figura 3.11. Coluna esquerda: SPM{t} original. O menor valor de um voxel ativado foi 1,48. Um voxel não-ativado pode assumir um valor maior que 1,48 ao acaso com a probabilidade 7,2%. Coluna direita: SPM{t} obtido pela técnica proposta. Todos os voxels ativados apresentaram valores aproximadamente 12,1. Virtualmente é impossível que um voxel não-ativado assuma valores tão altos por acaso (probabilidade menor que 10^{-16}).

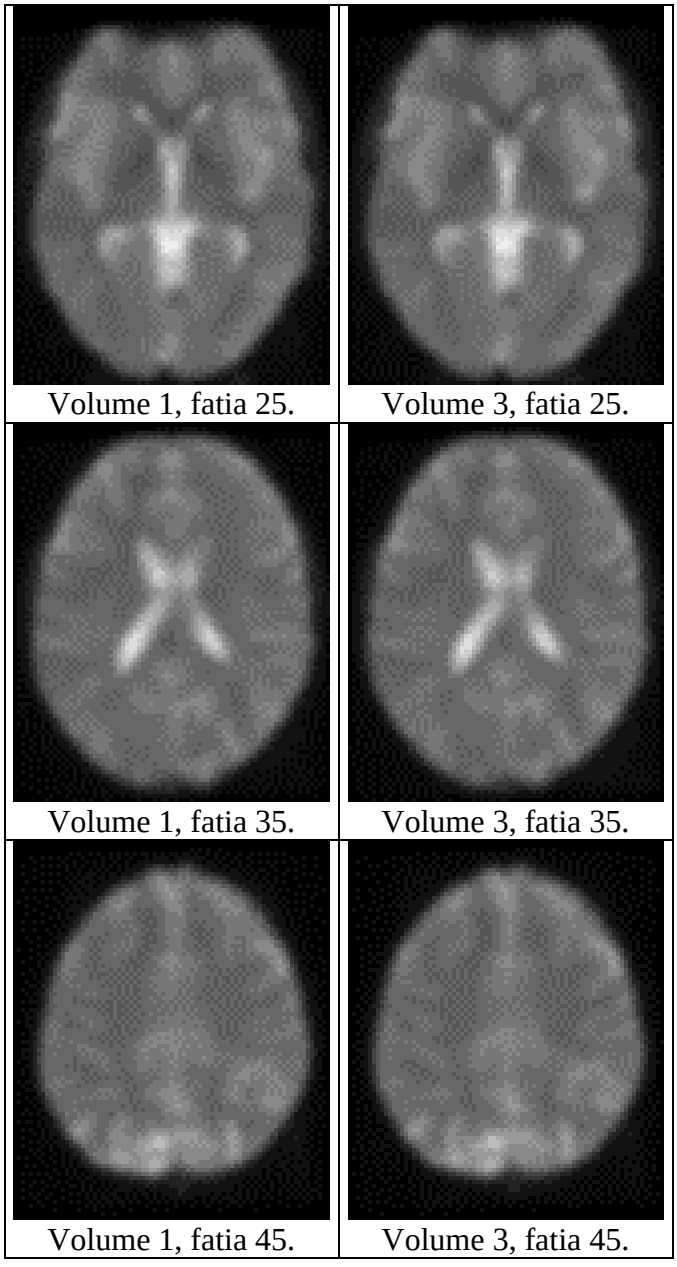


Fig. 3.13: Imagens fMRI reais com áreas artificialmente ativadas. Ativamos artificialmente pequenas regiões esféricas nos volumes 3, 4, 7, 8, 11 e 12, aumentando o valor do voxel em 3%.

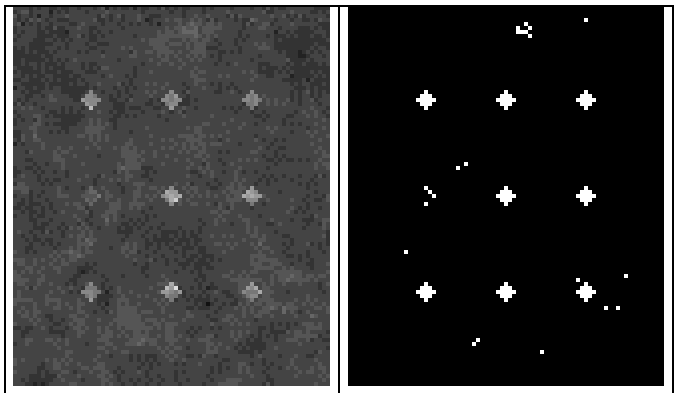


Fig. 3.14: Coluna esquerda: SPM{t} obtido da fMRI da figura 3.13 sem filtragem. O menor valor ativado foi 1,3. Coluna direita: Áreas ativadas detectadas limiarizando SPM na altura 2,2. Um voxel não-ativado pode assumir valor acima de 2,2 com probabilidade 2,5%.

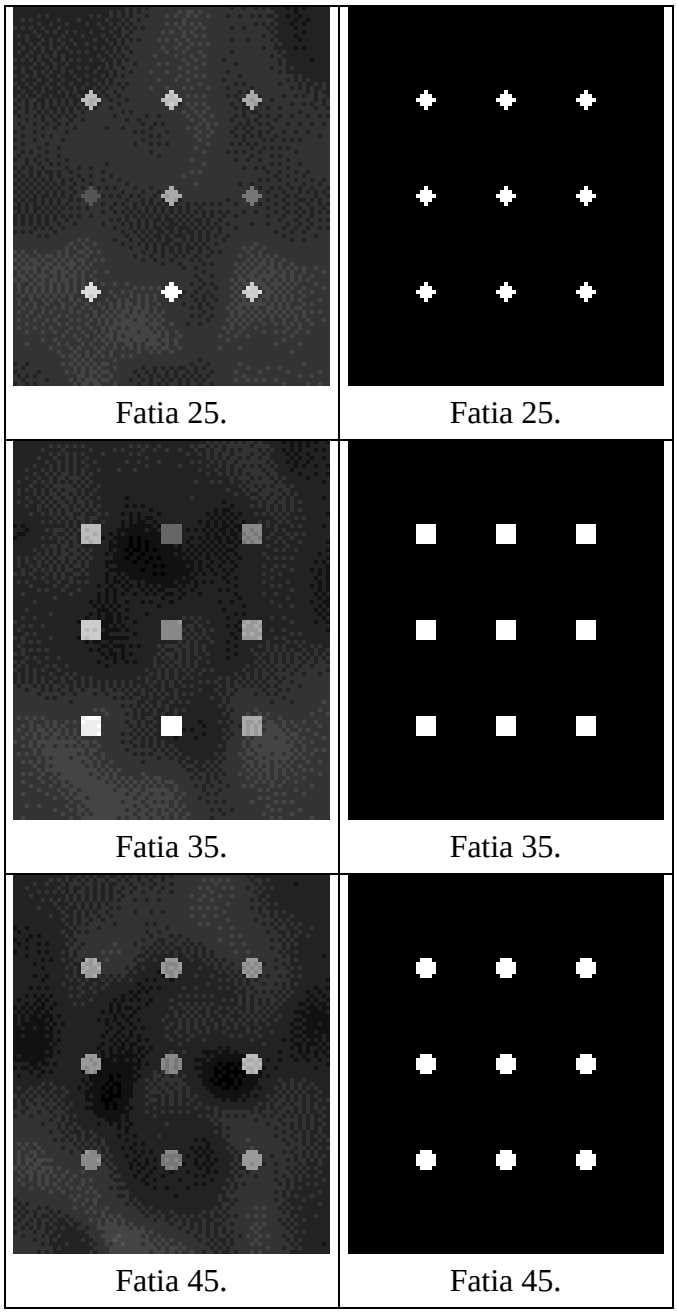


Fig. 3.15: Coluna esquerda: SPM{t} obtido da fMRI da figura 3.13 usando a técnica proposta. O menor valor ativado foi 2,4, uma melhoria significativa sobre 1,3 anterior. Coluna direita: Áreas ativadas detectadas limiarizando o SPM em 2,2.

3.6 Conclusões

O objetivo deste capítulo foi apresentar as nossas contribuições científicas na área das aplicações da difusão anisotrópica.

Para isso, expusemos a teoria do espaço de escala linear obtida através da difusão isotrópica e o espaço de escala não-linear obtida pela difusão anisotrópica. Para ilustrar o processo de difusão anisotrópica, mostramos o seu uso na filtragem dos sinais de sensor de aceleração. Demonstramos que a difusão anisotrópica robusta (RAD) é superior às difusões propostas por Perona e Malik na restauração desse tipo de sinal. Depois, ilustramos o uso da difusão anisotrópica na detecção multi-escala das arestas de uma imagem. Também para esta aplicação, concluímos que a RAD é melhor que as técnicas de Perona e Malik.

Em seguida, mostramos a aplicação da RAD no melhoramento da reconstrução tomográfica. Especificamente, o algoritmo MENT-estendido foi melhorado, resultando num algoritmo que denominamos de MENT reconstrução-difusão. Mostramos através dos resultados experimentais que o novo algoritmo gera as imagens mais nítidas que o algoritmo MENT-estendido original. Comparamos também o novo algoritmo com o algoritmo amplamente conhecido retro-projeção filtrada, constatando novamente a superioridade da nossa proposta.

Por fim, aplicamos a RAD para melhorar a detecção das áreas ativadas do cérebro através da análise de imagens fMRI. Mostramos que o uso da RAD pode aumentar a confiabilidade na detecção das áreas ativadas.

Capítulo 4:
Marcas d'Água de Autenticação

Resumo e nossas contribuições

Uma marca d'água é um sinal portador de informação embutido no dado digital que pode ser extraído mais tarde para fazer alguma asserção sobre o dado hospedeiro. As marcas d'água digitais são normalmente classificadas em robustas e frágeis. As marcas robustas são projetadas para resistirem a maioria dos procedimentos de manipulação de imagens e normalmente são usadas para atestar a propriedade da imagem. As marcas frágeis são facilmente corrompidas por qualquer processamento na imagem. Porém, as marcas para checar a integridade das imagens devem ser frágeis, para que qualquer alteração seja detectada. Este capítulo descreve as teorias que fundamentam as marcas d'água de autenticação e as nossas contribuições científicas nesta área.

Primeiro, descrevemos o conceito de assinatura digital, amplamente utilizada nas marcas de autenticação de chave pública. Em segundo lugar, descrevemos as principais marcas de autenticação para as imagens estáticas de tonalidade contínua (isto é, as imagens em níveis de cinza e coloridas): Yeung-Mintzer e Wong. Explicamos os principais ataques contra estas marcas e os meios para se defender contra eles. Em terceiro lugar, descrevemos as marcas de autenticação para as imagens binárias e meio-tom.

As nossas contribuições na área de marcas d'água de autenticação são:

1) *Hash block chaining:* Esta contribuição foi publicada em [Ri04; Ci04; Cn07; Cn05]. Nesta tese, ela está documentada na subseção 4.3.2. O principal responsável por esta contribuição foi o meu ex-orientando de doutorado Paulo S. L. M. Barreto, com a cooperação do Dr. Vincent Rijmen (da empresa Cryptomathic, Bélgica).

   Resumo: As nossas pesquisas foram sobre as fraquezas criptográficas das marcas d'água para autenticação de imagem orientada a blocos. O algoritmo original de Wong [Wong, 1997; Wong, 1998], assim como vários outros algoritmos variantes, não são seguros contra um simples ataque recortar-e-colar ou o bem conhecido ataque de aniversário. Para torná-los seguros, foram propostos alguns esquemas para tornar a assinatura de cada bloco depender do conteúdo dos seus blocos vizinhos. Procuramos maximizar a resolução de localização das alterações, utilizando somente uma dependência por bloco através de um esquema que denominamos de "hash block chaining" versão 1 (HBC1). Mostramos que HBC1, assim como qualquer outro esquema dependente do contexto dos blocos vizinhos, é suscetível a uma outra técnica de falsificação que denominamos de ataque de transplante. Também mostramos um novo tipo de ataque de aniversário que consegue atacar HBC1. Para impedir esses ataques, propomos utilizar uma assinatura digital não-determinística junto com o esquema dependente de assinatura (HBC2). Finalmente, discutimos as vantagens de se utilizar assinaturas de logaritmo discreto em vez de RSA nas marcas de autenticação.

2) *Marcas de autenticação para imagens binárias:* Esta contribuição foi publicada em [Cn14] e está submetida em [Su02; Su04]. Nesta tese, elas estão documentadas nas subseções 4.4.2 e 4.4.3. Eu fui o principal responsável por esta contribuição, com a colaboração do meu orientando de mestrado Amir Afif e do prof. Ricardo de Queiroz da UnB.

   Resumo: Na literatura, apenas um pequeno número de marcas de autenticação está disponível para as imagens binárias. Propomos duas novas marcas de autenticação para as imagens binárias: AWST e AWSF. A marca AWST é a-

propriada para as imagens meio-tom pontos dispersos e pode detectar qualquer alteração, mesmo uma alteração de um único pixel. A AWSF é apropriada para as imagens binárias em geral e pode detectar qualquer alteração visualmente significativa, ao mesmo tempo em que se mantém uma boa qualidade visual da imagem marcada. Esses algoritmos podem ser utilizados juntamente com a criptografia de chave secreta ou chave pública/privada. A segurança desses algoritmos baseia-se somente no segredo da chave. Na versão chave pública/privada, somente o dono da chave privada pode inserir a marca correta, enquanto que qualquer um pode verificar a autenticidade através da chave pública correspondente. Uma possível aplicação das técnicas propostas é na transmissão de fax pela internet, isto é, para a autenticação legal de documentos roteados fora da rede telefônica.

3) Temos outras contribuições em criptografia de chave pública [Ci08; Ci06; Ci03] que não estão documentadas nesta tese. Além disso, publicamos um curso tutorial sobre a marca de autenticação e a esteganografia [Rn01] cujo texto adaptado foi aproveitado nesta tese.

4.1 Introdução

O espetacular crescimento dos sistemas de multimídia interligados pela rede de computadores nos últimos anos (particularmente com o advento da World Wide Web) tem apresentado um enorme desafio nos aspectos tais como propriedade, integridade e autenticação dos dados digitais (áudio, vídeo e imagens estáticas). Para enfrentar tal desafio, o conceito de marca d'água digital foi definido.

Uma marca d'água é um sinal portador de informação, visualmente imperceptível, embutido numa imagem digital. Quando não houver perigo de confusão, utilizaremos a palavra "marca" como sinônimo de "marca d'água".

A imagem que contém uma marca é dita imagem marcada ou hospedeira. Apesar de muitas técnicas de marca d'água poderem ser aplicadas diretamente para diferentes

tipos de dados digitais, este capítulo irá tratar somente das marcas para imagens digitais 2-D estáticas.

Esteganografia

O primeiro passo no estudo das marcas d'água é o estudo das técnicas utilizadas para embutir a informação numa imagem, conhecidas como esteganografia (information hiding ou steganography, em inglês). Nesta área de pesquisa, estuda-se como inserir a maior quantidade possível de informações com uma mínima deterioração na qualidade da imagem hospedeira, sem se preocupar com a utilidade da informação escondida ou se a informação escondida é fácil ou difícil de ser removida. Algumas dificuldades especiais para inserir dados escondidos aparecem em tipos especiais de imagens, como nos formatos de imagens compactadas com perdas ou nas imagens binárias.

Marcas robustas e frágeis

As marcas d'água digitais são classificadas de acordo com a dificuldade em removê-las em robustas, frágeis e semifrágeis. Esta classificação também normalmente determina a finalidade para a qual a marca será utilizada.

As marcas robustas são projetadas para resistirem a maioria dos procedimentos de manipulação de imagens. A informação embutida numa imagem através de uma marca robusta deveria ser possível de ser extraída mesmo que a imagem hospedeira sofra rotação, mudança de escala, mudança de brilho/contraste, compactação com perdas com diferentes níveis de compressão, corte das bordas (cropping), etc. Uma boa marca d'água robusta deveria ser impossível de ser removida a não ser que a qualidade da imagem resultante deteriore a ponto de destruir o seu conteúdo visual. Isto é, a correlação entre uma imagem marcada e a marca robusta nela inserida deveria permanecer detectável mesmo após um processamento digital, enquanto a imagem resultante do processamento continuar visualmente reconhecível e identificável como a imagem original. Por esse motivo, as marcas d'água robustas são normalmente utilizadas para a verificação da propriedade (*copyright*) das imagens. Para dar um exemplo, se uma agência de notícias colocasse uma marca robusta numa fotografia, ne-

nhum adulterador malicioso deveria ser capaz de remover essa marca. Apesar de muitas pesquisas, parece que ainda não foi possível obter uma marca d'água robusta realmente segura.

As marcas frágeis são facilmente removíveis e corrompidas por qualquer processamento na imagem [Yeung and Mintzer, 1997; Wong, 1997; Wong, 1998; Wu and Liu, 1998; Li et al., 2000; Holliman and Memon, 2000]. Este tipo de marca d'água é útil para checar a integridade e a autenticidade da imagem, pois possibilita detectar alterações na imagem. Em outras palavras, uma marca d'água frágil fornece uma garantia de que a imagem marcada não seja despercebidamente editada ou adulterada. Neste sentido, o termo "frágil" é infeliz para qualificar esses algoritmos, sendo mantido por razões históricas. Talvez o termo mais apropriado seja "marca d'água de autenticação".

As marcas frágeis de autenticação detectam *qualquer* alteração na imagem. Às vezes, esta propriedade é indesejável. Por exemplo, ajustar brilho/contraste para melhorar a qualidade da imagem pode ser um processamento válido, que não deveria ser detectado como uma tentativa de adulteração maliciosa. Ou então, compactar uma imagem com perdas (como JPEG ou JPEG2000) em diferentes níveis de compressão deveria ser uma operação permitida. Ainda, imprimir e escanear uma imagem não deveria levar à perda da autenticação. Assim, foram criadas as marcas d'água semifrágeis. Uma marca semifrágil também serve para autenticar imagens. Só que estas procuram distinguir as alterações que modificam uma imagem substancialmente daquelas que não modificam o conteúdo visual da imagem. Uma marca semifrágil normalmente extrai algumas características da imagem que permanecem invariantes através das operações "permitidas" e as insere de volta na imagem de forma que a alteração de uma dessas características possa ser detectada.

Tipos de marcas de autenticação

Podemos subdividir as marcas de autenticação (tanto frágeis como semifrágeis) em três subcategorias: sem chave, com chave secreta (cifra simétrica) e com chave pública/privada (cifra assimétrica):

Uma marca de autenticação sem chave é útil para detectar as alterações não-intencionais na imagem tais como um erro de transmissão ou de armazenamento. Funciona como uma espécie de "check-sum". Se o algoritmo de autenticação sem chave estiver disponível publicamente, qualquer pessoa pode inserir este tipo de marca em qualquer imagem e qualquer pessoa pode verificar se uma imagem contém uma marca válida.

A marca de autenticação com chave secreta (cifra simétrica) é usada para detectar uma alteração que pode ser inclusive intencional ou maliciosa. Este tipo de marca é similar aos códigos de autenticação de mensagem, sendo que a única diferença é que o código de autenticação é inserido na imagem em vez de ser armazenado separadamente. Os algoritmos para inserção e detecção deste tipo de marca podem ser disponibilizados publicamente, e uma chave secreta é usada em ambas as fases. Vamos supor que Alice administra um grande banco de dados de imagens, onde cada imagem está assinada com uma chave secreta k que somente Alice conhece. Vamos supor que Mallory, um hacker malicioso, modifique uma imagem neste banco de dados. Mallory não consegue inserir a marca correta na imagem adulterada pois ele não conhece a chave k. Além disso, Alice será capaz de detectar todas as imagens alteradas pelo Mallory usando o algoritmo de detecção de marca d'água e sua chave secreta k.

As marcas de autenticação com chave pública (cifra assimétrica) utilizam a criptografia de chave pública para inserir uma assinatura digital na imagem. Usando uma cifra de chave pública, a autenticidade de uma imagem pode ser julgada sem a necessidade de se tornar pública qualquer informação privada.

Marca de autenticação em imagens contones e binárias

Existe uma forma "natural" de embutir as marcas de autenticação em imagens de tonalidade contínua (contone) não compactadas. É inserir os dados nos bits menos significativos (LSBs). Alterar os LSBs afeta muito pouco a qualidade da imagem, ao mesmo tempo em que se conhece exatamente os bits que serão afetados pela inserção da marca.

Não ocorre o mesmo com as imagens binárias. Numa imagem binária, cada pixel consiste de um único bit, de forma que não existe LSB. Isto traz dificuldades especiais para projetar marcas de autenticação para este tipo de imagem.

Inserir uma marca de autenticação em imagens contone compactadas com perdas também apresenta dificuldades especiais. Porém, este assunto não será tratado nesta tese.

Exemplos de uso de marcas de autenticação de chave pública

Entre os três tipos de marca de autenticação, a de chave pública é a que oferece mais recursos. Os possíveis usos de uma marca de autenticação de chave pública são enormes. Abaixo, citamos três exemplos:

1) *Câmera digital segura.* Costuma-se citar o artigo [Friedman, 1993] como o trabalho que inspirou os primeiros trabalhos de marca d'água de autenticação. Na câmera digital proposta, a câmera produz dois arquivos de saída para cada imagem capturada: a primeira é a própria imagem digital capturada pela câmera em algum formato; e a segunda é uma assinatura digital produzida aplicando a chave privada da câmera (que deve estar armazenada de forma segura num circuito integrado dentro da câmera). O usuário deve tomar cuidado para guardar os dois arquivos, para que se possa autenticar a imagem mais tarde. Uma vez que a imagem digital e a assinatura digital são geradas pela câmera e armazenadas no computador, a integridade e a autenticidade da imagem pode ser verificada usando um programa para decodificar a assinatura digital, que pode ser

distribuído livremente aos usuários. O programa de verificação recebe como entrada a imagem digital, a assinatura digital e a chave pública da câmera. Ele calcula a função "hash" da imagem digital, decriptografa a assinatura digital e verifica se as duas "impressões digitais" obtidas são iguais. O esquema proposto por Friedman poderia ser melhorado de duas formas. A primeira seria embutir a assinatura digital no arquivo da imagem, o que eliminaria a necessidade de armazenar dois arquivos para cada imagem. Alguns formatos de imagem permitem armazenar alguns dados adicionais no cabeçalho ou rodapé do arquivo. Mas o mais interessante seria embutir a assinatura digital na própria imagem. A segunda seria permitir a localização da região alterada. Isto poderia ser interessante, por exemplo, para descobrir a intenção do falsificador ao adulterar a imagem. A marca d'água de autenticação de chave pública pode ser usada para incorporar essas melhorias à câmera de Friedman.

2) *Autenticação de imagens distribuídas pela rede.* Vamos supor que uma agência de notícias chamada Alice deseja distribuir pela internet uma fotografia jornalística, com alguma prova de autenticidade de que a foto foi distribuída pela Alice e que ninguém introduziu alterações maliciosas na foto. Alice utiliza a sua chave privada para inserir marca d'água de autenticação na imagem e distribui a foto marcada. Vamos supor que Bob recebe a foto marcada. Bob usa a chave pública da Alice para verificar que a foto está assinada pela Alice e que ninguém introduziu qualquer alteração depois de Alice assiná-la. Se Mallory, um hacker malicioso, alterar a foto, ele não será capaz de inserir a marca correta na imagem falsificada porque ele não conhece a chave privada da Alice. Além disso, Mallory não poderá distribuir uma foto sua como sendo da Alice porque ele não conseguirá assiná-la por desconhecer a chave privada da Alice.

3) *Fax confiável.* Uma "máquina de FAX confiável" poderia conter internamente uma chave privada e inserir uma marca d'água em todos os documentos transmitidos por ela. O receptor de FAX, usando a chave pública da máquina trans-

missora, poderia verificar que o documento foi originado de uma máquina específica de FAX e que o documento não foi manipulado.

Organização deste capítulo

O restante deste capítulo está organizado como segue. A seção 4.2 apresenta o conceito de assinatura digital, amplamente utilizado nas marcas de autenticação. A seção 4.3 apresenta algumas marcas de autenticação para imagens contone, subdividida em duas subseções. A subseção 4.3.1 apresenta a marca de Yeung-Mintzer e a subseção 4.3.2 descreve a marca de Wong, os ataques contra esta marca e a nossa proposta para robustecer esta marca denominada "hash block chaining". A seção 4.4 apresenta as marcas de autenticação para as imagens binárias e meio-tom, subdividida em 3 subseções. A subseção 4.4.1 é a introdução, a subseção 4.4.2 apresenta a marca de autenticação AWST (apropriada para as imagens meio-tom pontos dispersos) e a subseção 4.4.3 descreve marca de autenticação AWSF (apropriada para as imagens binárias em geral, exceto as imagens meio-tom pontos dispersos).

4.2 Assinatura Digital

Vamos apresentar nesta seção um conceito que é bastante utilizado nas marcas d'água de autenticação: a assinatura digital. Para isso, seguiremos de perto a redação didática de [Friedman, 1993] e [Barreto, 2003].

Criptografia simétrica

A criptografia de chave secreta ou simétrica requer que tanto o transmissor quanto o receptor da mensagem possuam a mesma chave secreta: o transmissor utiliza a chave para transformar a mensagem original em texto cifrado, e o receptor utiliza a mesma chave para executar a transformação inversa, recuperando o texto original. O defeito histórico deste esquema é a distribuição segura das chaves: a chave deve ser transmitida através de um caro meio seguro alternativo.

Criptografia assimétrica

O conceito de criptografia de chave pública foi inventado pelo W. Diffie e M. Hellman, e independentemente por R. Merkle [Schneier, 1996, chap. 19]. A criptografia de chave pública ou cifra assimétrica utiliza duas chaves: uma chave privada e outra pública. Conhecendo a chave privada, é fácil e rápido calcular a chave pública correspondente. Porém, o contrário é uma tarefa extremamente difícil computacionalmente (levaria talvez séculos utilizando os supercomputadores atuais).

Para enviar uma mensagem secreta que somente o receptor Bob possa ler, Bob primeiro torna a sua chave pública conhecida publicamente. Qualquer pessoa que queira enviar uma mensagem secreta a Bob deve criptografar a mensagem usando esta chave pública e enviá-la a Bob. Bob, sendo único possuidor da chave privada, é a única pessoa capaz de decriptografar a mensagem. Note que a necessidade de se combinar uma chave secreta entre o transmissor e o receptor foi eliminada.

Assinatura digital

O processo descrito acima pode ser implementado "ao contrário". Neste caso, a transmissora de mensagem Alice guarda uma chave privada, e a chave pública correspondente é disponibilizada publicamente a qualquer receptor que queira decriptografar. Este procedimento não mais executa a função tradicional de criptografia, que é permitir uma comunicação confidencial entre as duas partes. Porém, fornece um meio para assegurar que as mensagens não foram forjadas: somente Alice, que possui a chave privada, poderia ter codificado uma mensagem que é decifrável pela correspondente chave pública.

As assinaturas digitais estão construídas sobre as técnicas de criptografia de chave pública. Elas permitem autenticar o conteúdo da mensagem e a identidade do emissor.

As assinaturas são produzidas através de uma função *hash*. Intuitivamente, uma função *hash* calcula, rápida, segura e univocamente, representantes adequadamente curtos para as mensagens arbitrariamente longas (chamadas "impressões digitais" das mensagens). Essas "impressões digitais" são criptografadas utilizando a chave privada, em lugar das próprias mensagens. Isto acelera tanto o processo de criar a assinatura como o processo de verificá-la. O resultado é um dado (chamada assinatura digital e abreviada como DS) que acompanha a mensagem original. Desta forma, a mensagem original pode ser lida por todos, porém se um receptor chamado Bob desejar autenticá-la, Bob pode decriptografar a assinatura digital da mensagem usando a chave pública da Alice, recuperando a "impressão digital" da mensagem. Esta impressão digital deve ser idêntica à *hash* da mensagem original, se a mensagem não tiver sido adulterada.

Assinatura não-determinística

Uma assinatura digital diz-se *determinística* se o seu valor for exclusiva e univocamente determinado pelos dados assinados e pela chave privada do signatário. Em contraste, uma assinatura digital é *não-determinística* se depender também de algum

parâmetro aleatório, chamado *sal* ou *nonce*. Assume-se que esse parâmetro seja estatisticamente único (irrepetível) e uniformemente distribuído. Além disso, o seu valor deve ser imprevisível para um adversário do sistema. Alguns esquemas de assinatura (por exemplo, DSA e Schnorr [Schneier, 1996]) são naturalmente não-determinísticos. Outros esquemas (por exemplo, RSA) precisam de construções especiais para que se tornem não-determinísticos.

Algoritmo RSA

Descreveremos resumidamente, aquele que é provavelmente o algoritmo de criptografia de chave pública mais amplamente utilizado atualmente, o algoritmo RSA (Rivest, Shamir e Adleman [Rivest et al., 1978]). Sejam p e q dois números primos distintos de tamanhos aproximadamente iguais, seja $n = pq$, e seja e um inteiro inversível módulo $(p-1)(q-1)$, com inverso $d \equiv e^{-1} \bmod (p-1)(q-1)$, isto é, $ed \equiv 1 \pmod{(p-1)(q-1)}$. A chave pública é o par (e,n), e a chave privada é o inteiro d (os primos p e q são também mantidos secretos, e podem até ser descartados, pois o conhecimento deles não é essencial para as operações do RSA). Seja $M \in \mathbb{Z}_n$ a mensagem a ser assinada. Uma assinatura RSA para M é definida como $C = M^d \bmod n$. A verificação da assinatura procede com a recuperação de M a partir de C: $M = C^e \bmod n$.

4.3 Marcas de Autenticação para Imagens Contone

4.3.1 Marca de Autenticação de Yeung-Mintzer

Introdução

Yeung e Mintzer [Yeung and Mintzer, 1997] propuseram uma das primeiras técnicas de marca d'água de autenticação. A marca é inserida pixel a pixel, de forma que a alteração pode ser localizada com precisão. Porém, como há apenas 1 bit de marca para autenticar cada pixel, há 50% de chance de uma alteração de um único pixel passar despercebida. Porém, se uma região de tamanho razoável for alterada, muito dificilmente essa alteração passará despercebida. Este esquema funciona com chave secreta, isto é, a inserção e a detecção da marca devem ser feitas utilizando a mesma chave. Demonstrou-se mais tarde que este esquema é completamente inseguro. Isto é, um falsificador poderia inserir a marca válida em qualquer imagem dispondo apenas de um pequeno conjunto de imagens validamente marcadas.

Inserção de marca de Yeung-Mintzer

Seja *B* uma imagem-logotipo binária a ser inserida na imagem-original *I* para produzir a imagem-marcada I'. A imagem-original *I* pode ser tanto em níveis de cinza (neste caso, vamos supor 8 bits por pixel ou 256 níveis de cinza) como colorida (neste caso, vamos supor 3 bytes por pixel no formato RGB). Vamos supor que a imagem-logotipo *B* seja do mesmo tamanho que a imagem-original *I*. Se os tamanhos das duas imagens forem diferentes, a imagem-logotipo *B* deve ser replicada ou redimensionada para que seja do mesmo tamanho que *I*.

Vamos descrever primeiro o caso em níveis de cinza. Tanto a inserção, quanto a extração da marca depende de uma look-up-table (LUT) $k:\{0...255\}\rightarrow\{0,1\}$. Uma LUT *k* aleatória pode ser gerada sorteando 256 valores booleanos. Esta LUT *k* funciona como uma chave secreta e deve ser mantida em segredo.

A inserção processa os pixels numa determinada ordem. Vamos supor que a ordem "raster" seja utilizada (isto é, processar os pixels linha por linha de cima para baixo e, dentro de uma determinada linha, da esquerda para direita).

Para inserir a marca no primeiro pixel (1, 1), calcula-se $k(I(1,1))$. Se este valor for igual ao valor $B(1,1)$ da imagem-logotipo, não há nada que fazer. Se $k(I(1,1)) \neq B(1,1)$, o valor de $I(1,1)$ deve ser alterado para um nível de cinza próximo $I'(1,1)$ para obtermos $k(I'(1,1)) = B(1,1)$.

O erro cometido ao aproximar $I(1,1)$ para $I'(1,1)$ (isto é, $I(1,1) - I'(1,1)$) é espalhado para os pixels vizinhos, de forma semelhante ao bem conhecido algoritmo de difusão de erro utilizado para gerar imagens meio-tom. Isto assegura que o nível de cinza médio não é alterado localmente, o que garante uma alta qualidade visual à imagem marcada. Os autores usaram os pesos de difusão de erro abaixo, mas outros valores poderiam ser utilizados:

- $W(i+1, j) = 0{,}5$;
- $W(i+1, j+1) = 0{,}0$;
- $W(i, j+1) = 0{,}5$.

Após a difusão de erro usando os pesos acima, obtemos novos valores de I na vizinhança do pixel (1, 1). Denotaremos a imagem obtida após o espalhamento de erro de $\bar{I}$. Assim, usando os pesos acima, obtemos os seguintes valores de $\bar{I}$ para a vizinhança de $I(1,1)$:

- $\bar{I}(2,1) = 0{,}5\big(I(1,1) - I'(1,1)\big) + I(2,1)$;
- $\bar{I}(1,2) = 0{,}5\big(I(1,1) - I'(1,1)\big) + I(1,2)$.

Agora, estamos prontos para processar o segundo pixel, digamos (1, 2), calculando a cor $I'(1,2)$ semelhante a $\bar{I}(1,2)$ de forma que $k(I'(1,2)) = B(1,2)$. O novo erro obtido é espalhado aos vizinhos. Este processo se repete até processar a imagem toda.

Para inserir a marca d'água numa imagem colorida *I*, necessita-se de uma LUT pra cada plano de cor. Vamos denotá-las como k_R, k_G e k_B, respectivamente as LUTs dos planos de cores vermelho, verde e azul. Para inserir a marca d'água num pixel (i, j), calcula-se a expressão booleana:

$$k_R(\bar{I}_R(i, j)) \otimes k_G(\bar{I}_G(i, j)) \otimes k_B(\bar{I}_B(i, j))$$

onde:

- $\otimes$ indica ou-exclusivo;
- $\bar{I}_R(i, j)$, $\bar{I}_G(i, j)$ e $\bar{I}_B(i, j)$ indicam os valores do pixel (i, j) da imagem $\bar{I}$ obtida difundindo o erro, nos planos de cores vermelho, verde e azul, respectivamente.

Se o valor da expressão acima for igual a $b(i, j)$, nada a fazer. Se for diferente, os valores $\bar{I}_R(i, j)$, $\bar{I}_G(i, j)$ e/ou $\bar{I}_B(i, j)$ devem ser alterados para os valores próximos $I'_R(i, j)$, $I'_G(i, j)$ e $I'_B(i, j)$ para que a expressão abaixo se torne igual a $b(i, j)$:

$$k_R(I'_R(i, j)) \otimes k_G(I'_G(i, j)) \otimes k_B(I'_B(i, j)).$$

Extração da marca de Yeung-Mintzer

Dada uma imagem em níveis de cinza I' marcada com a marca de Yeung-Mintzer e a LUT *k* utilizada na inserção da marca, a imagem binária de checagem *C* pode ser extraída facilmente. Basta calcular:

$$C(i, j) \leftarrow k(I'(i, j))$$

para todos os pixels (i, j). Da mesma forma, dada uma imagem colorida marcada I', e três LUTs k_R, k_G e k_B, basta calcular:

$$C(i, j) \leftarrow k_R(I'_R(i, j)) \otimes k_G(I'_G(i, j)) \otimes k_B(I'_B(i, j)), \text{ para todos os pixels } (i, j).$$

Se a imagem de checagem *C* for igual à imagem inserida *B*, a imagem marcada I' não foi alterada. Caso contrário, houve a alteração na região onde as imagens *C* e *B* forem diferentes.

Ataque de falsificação

Holliman e Memon [Holliman and Memon, 2000] apresentaram o ataque de falsificação (counterfeiting attack) que pode subverter completamente a marca de Yeung-Mintzer. Isto é, tendo algumas poucas imagens marcadas utilizando uma LUT *k*, é possível marcar validamente uma imagem qualquer sem conhecer a tabela *k* ou a imagem-logotipo. Além disso, é possível calcular a chave secreta *k* a partir de algumas imagens marcadas com a tabela *k*, conhecendo a imagem-logotipo *B*.

Para forjar uma marca d'água de Yeung-Mintzer, aproveita-se do fato de que cada pixel é autenticado independentemente de qualquer outro. Vamos expor o ataque somente para o caso níveis de cinza, porém a mesma idéia vale para o caso colorido.

Vamos supor que Mallory, um hacker malicioso, gostaria de inserir uma marca válida numa imagem *J* qualquer, sem conhecer a LUT *k*. Vamos supor que Mallory de alguma forma conheça a imagem-logotipo *B* e tenha à disposição uma imagem I' onde a imagem *B* foi embutida utilizando a LUT *k*. O ataque torna-se mais fácil se Mallory dispuser de uma quantidade grande de imagens onde a imagem-logotipo *B* foi inserida utilizando a mesma LUT *k*. Porém, para simplificar a notação, assumiremos disponível uma única imagem hospedeira I' (na verdade, se houver várias imagens hospedeiras, todas podem ser grudadas uma na outra para formar uma única imagem).

Para marcar a imagem *J*, Mallory divide os pixels de I' em dois subconjuntos disjuntos: o primeiro subconjunto S_0 de pixels com valor zero na imagem-logotipo *B* e o segundo S_1 de pixels com valor um em *B*. Como só existem 256 níveis de cinza, e uma imagem de tamanho usual possui centenas de milhares de pixels, provavelmente haverá exemplos de praticamente todos os níveis de cinza. Em cada pixel $J(i, j)$, deve ser embutido o bit $B(i, j)$. Para isso, Mallory procura, no subconjunto S_0 ou S_1 correspondente ao bit $B(i, j)$, o pixel com o nível de cinza mais próximo possível do $J(i, j)$. Daí, coloca esse valor no pixel (i, j) da imagem falsificada J'. Basta repetir este processo para todos os pixels da imagem *J* para obter a imagem corretamente marcada

J'. Aliás, se quisesse otimizar a qualidade visual da imagem forjada J', seria até possível executar um algoritmo de difusão de erro semelhante ao utilizado no algoritmo de inserção de marca d'água.

Se o tamanho da imagem I' for suficientemente grande para conter um pixel exemplar para cada nível de cinza (o que costuma acontecer na prática), a LUT secreta *k* pode ser completamente descoberta a partir dos subconjuntos S_0 e S_1. Basta associar a cada nível de cinza em S_0 o bit 0 e a cada nível de cinza em S_1 o bit 1.

4.3.2 Marca de Wong e Hash Block Chaining

Introdução

Esta subseção descreve uma contribuição científica original nossa. O principal responsável pelas descobertas foi o meu ex-orientando de doutorado Paulo S. L. M. Barreto.

Wong [Wong, 1997] propôs uma outra marca d'água de autenticação, desta vez baseada em criptografia simétrica. Esse artigo foi melhorado em [Wong, 1998] para utilizar a criptografia de chave pública, tornando-se o primeiro trabalho de marca de autenticação de chave pública. O esquema de Wong consiste, basicamente, em dividir uma imagem em blocos e assinar cada bloco independentemente. Assim, é possível localizar o bloco onde a imagem foi alterada. Quanto menor o tamanho dos blocos, melhor a resolução de localização da alteração. A marca d'água é inserida nos bits menos significativos (LSBs - least significant bits) da imagem. Assim, nas imagens em níveis de cinza, é possível inserir um bit em cada pixel. Nas imagens coloridas, é possível inserir três bits em cada pixel. Para uma imagem com 256 níveis de cinza (8 bits por pixel), a alteração dos LSBs é visualmente imperceptível, pois equivale a somar ou subtrair um do nível de cinza.

Assim como o trabalho de Yeung-Mintzer, o trabalho de Wong possui defeitos sérios de segurança. Nesta subseção, estudaremos apenas a versão chave-pública da marca de Wong. A versão chave-secreta é inteiramente análoga.

Inserção da marca de Wong

A inserção de marca d'água numa imagem em níveis de cinza, usando o esquema de Wong chave-pública, pode ser resumida como segue.

Passo 1: Seja I uma imagem em níveis de cinza a ser marcada, com $\boldsymbol{N} \times M$ pixels. Particione I em n blocos I_t $(0 \le t < n)$ de 8×8 pixels (no máximo, os blocos nas bordas podem ser menores). Cada bloco I_t será marcado independentemente.

Passo 2: Seja B uma imagem-logotipo binária a ser utilizada como marca d'água. Esta imagem é replicada periodicamente ou redimensionada para obter uma imagem suficientemente grande para cobrir I. Para cada bloco I_t, existe um bloco binário correspondente B_t.

Passo 3: Seja I_t^* o bloco obtido de I_t zerando o bit menos significativo de todos os pixels. Usando uma função hash H criptograficamente segura, calcule a impressão digital $H_t = H(M, N, I_t^*)$. Aqui, M e N entram na função hash para detectar cortes das bordas da imagem (cropping).

Passo 4: Calcule o ou-exclusivo de H_t com B_t, obtendo a impressão digital marcada $\hat{H}_t$.

Passo 5: Criptografe $\hat{H}_t$ com a chave privada, gerando assim a assinatura digital S_t do bloco t.

Passo 6: Insira S_t nos LSBs de I_t^*, obtendo o bloco marcado I_t'.

Extração da marca de Wong

O algoritmo de verificação da marca d'água correspondente é direto:

Passo 1: Seja I' uma imagem $\boldsymbol{N} \times M$ em níveis de cinza com marca d'água inserida. Particione I' em n blocos I_t', como na inserção.

Passo 2: Seja I_t^* o bloco obtido de I_t' limpando os LSBs de todos os pixels. Usando a mesma função *hash* escolhida para a inserção, calcule a impressão digital $H_t = H(M, N, I_t^*)$.

Passo 3: Retire os LSBs de I_t' e decriptografe o resultado usando a chave pública, obtendo o bloco decriptografado D_t.

Passo 4: Calcule o ou-exclusivo de H_t com D_t, obtendo o bloco de checagem C_t.

Passo 5: Se C_t e B_t (o bloco t da imagem-logotipo) forem iguais, a marca d'água está verificada. Caso contrário, a imagem marcada I' foi alterada no bloco t.

Aqui e no resto desta subseção, o operador * indica limpar os LSBs e a marca ′ indica um bloco ou uma imagem com a assinatura embutida.

Observe que, teoricamente, a imagem-logotipo B deveria estar disponível publicamente para efetuar a verificação da marca d'água. Na prática, porém, B é uma imagem com algum sentido visual (por exemplo, o logotipo da empresa) e qualquer alteração em I_t' irá muito provavelmente gerar um bloco de checagem C_t parecido com ruído aleatório, que não pode ser confundido com B_t mesmo que B não esteja disponível. A imagem B poderia ser até completamente preta (ou branca) e neste caso torna-se muito fácil disponibilizar B publicamente.

Li *et al.* [Li et al., 2000] sugerem uma ligeira variação do esquema acima. O seu método particiona cada bloco em duas metades. Depois, a metade à direita do bloco I_t^* é trocada com a metade à direita do próximo bloco $I_{(t+1) \bmod n}^*$ seguindo a ordem em zig-zag (figura 4.1c) de forma que os blocos vizinhos estão relacionados pelos dados fundidos. Cada bloco combinado é então assinado e inserido nos LSBs do bloco I_t^*. A mesma operação deve ser executada na verificação da marca d'água.

Ataque recortar-e-colar e ataque de falsificação

Mostraremos a seguir algumas fraquezas criptoanalíticas dos métodos de Wong e Li e mostraremos os meios para torná-los robustos.

Em primeiro lugar, note que a assinatura RSA de 64 bits, sugerida originalmente para ser usada com o esquema de Wong, é completamente insegura. Uma RSA com chave de 64 bits pode ser fatorada em segundos usando um computador pessoal atual.

Um esquema de autenticação que consegue detectar quaisquer alterações na imagem marcada deve ser considerado mais seguro que um outro que não consegue detectar algumas formas de alterações, mesmo que estas alterações aparentemente não possam ser utilizadas para propósitos maliciosos. A mera existência de tais falhas indica uma fraqueza do esquema. Elas podem ser usadas no futuro para atacar a marca d'água, mesmo que neste momento ninguém saiba como fazê-lo.

Por exemplo, Wong [Wong, 1998] sugeriu que a sua marca d'água em níveis de cinza fosse generalizada para as imagens coloridas simplesmente aplicando o método independentemente aos três planos de cores. Neste caso, a verificação da marca não irá detectar a troca dos planos de cores. Embora possa ser difícil imaginar como este ataque poderia ser usado maliciosamente, é mais seguro que mesmo este tipo de alteração não passe despercebida. Este problema em concreto pode ser facilmente resolvido alimentando os três planos de cores em conjunto na função de hash.

Existe um outro ataque muito simples, indetectável pelo esquema de Wong, que pode realmente ser utilizado com intenções maliciosas. Denominamos esse ataque de "recortar-e-colar". Suponha que Mallory, um hacker malicioso, possui uma coleção de imagens legitimamente marcadas, todas elas do mesmo tamanho e contendo a mesma imagem embutida *B*. Como cada bloco é marcado separadamente sem qualquer informação sobre a imagem hospedeira exceto as suas dimensões, é possível que Mallory selecione alguns blocos das imagens autênticas e construa com eles uma nova imagem cuja marca d'água será falsamente verificada como legítima. Aqui assumimos que as coordenadas originais de cada bloco são mantidas na imagem falsificada. Porém, em alguns casos (por exemplo, se o tamanho da imagem *B* for 4×4, 4×8, 8×4, 8×8, 8×16, etc.) pode até ser possível recortar um bloco de uma imagem e colá-lo dentro da mesma imagem mantendo a marca d'água inalterada. A figura 4.2 mostra um exemplo deste ataque.

Este ataque também se aplica para a marca de Li: o atacante deve somente copiar os conteúdos sem os LSBs dos dois semi-blocos de dois blocos vizinhos, digamos I_t^* e I_{t+1}^*, e colá-los juntos com a assinatura digital que se encontra nos LSBs do bloco I_t'.

Se o ataque recortar-e-colar for aplicado repetidamente, uma imagem inteira falsificada mas com uma marca válida pode ser construída. Esta é exatamente a idéia do ataque de falsificação (counterfeiting) de Holliman-Memon. Vamos supor que Mallory deseja marcar uma imagem *J* tendo em mãos um banco de dado de imagens protegidas pela marca de Wong. Mallory primeiro particiona *J* em blocos J_t. Vamos supor que B_t é a imagem-logotipo que deve ser inserido no bloco J_t. Mallory procura, entre os blocos do banco de dados contendo a marca B_t, o bloco D_t' visualmente mais parecido ao bloco J_t. Então, insere o bloco D_t' no lugar de J_t. Repetindo este processo para todos os blocos de *J*, uma imagem falsificada (mas com uma marca d'água válida) pode ser construída. Este ataque pode ser aplicado com sucesso mesmo usando um banco de dados relativamente pequeno. Holliman e Memon pegaram duas imagens de impressões digitais da NIST (750×750 pixels, em níveis de cinza), inseriram a marca de Wong num deles, e então construíram uma aproximação convincente da segunda imagem e corretamente marcada utilizando a primeira como o banco de dados, isto é, utilizando somente 9000 blocos validamente marcados como banco de dados. Um ataque similar também pode ser efetuado contra a marca de Li. Mostraremos adiante que HBC1 torna impossíveis os ataques recortar-e-colar e falsificação.

Ataque de aniversário simples

O ataque de aniversário [Menezes, 1997, seção 9.7] constitui um meio bem conhecido e poderoso para subverter assinaturas digitais. O atacante procura pelas colisões, isto é, pares de blocos que são levados a mesmo valor pela função de hash, portanto que têm a mesma assinatura. Usando função uma de hash que produz *m* valores possíveis, existe mais de 50% de chance de se achar uma colisão toda vez que aproxi-

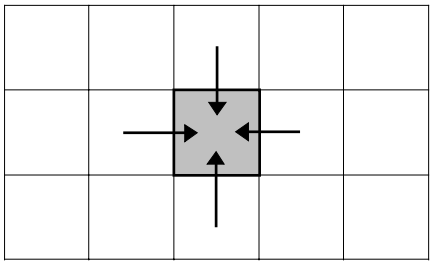
madamente $\sqrt{m}$ blocos estiverem disponíveis. O esquema de Wong utiliza uma função de hash de não mais que 64 bits. Daí, espera-se que as colisões ocorram quando o atacante tiver coletado somente 2^{32} blocos. Em geral, a única proteção contra o ataque de aniversário é aumentar o tamanho da função hash. Isto diminuiria a resolução de localização das alterações, pois os blocos devem ser maiores para hospedar mais dados inseridos. Mostraremos na próxima subsubseção que o ataque de aniversário clássico também se torna impossível sob HBC1.

Um cenário possível para o ataque de aniversário é uma companhia de seguros que mantém um banco de dados de imagens de incidentes usando a marca d'água de Wong para a proteção da integridade e da autenticidade das imagens. Um banco de dado típico de uma grande companhia de seguros pode conter mais de um milhões de imagens com, digamos, 640×480 pixels, de forma que cada imagem é particionada em 4800 blocos (de 8×8 pixels) individualmente assinados. Isto resulta em aproximadamente 2^{32} assinaturas, o suficiente para um ataque de aniversário.

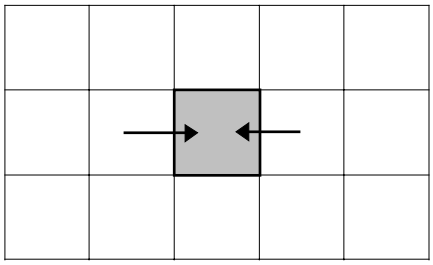
Mallory, um hacker malicioso, deseja substituir um bloco assinado I_t' por um outro bloco J e prepara $r \approx 2^{32}$ variantes visualmente equivalentes $J_1, \ldots, J_r$ de J. Isto pode ser feito variando o segundo bit menos significativo de cada um dos 32 pixels arbitrariamente escolhidos de J (os LSBs não podem ser usados, uma vez que a marca d'água será armazenada lá). Mallory então procura por um bloco D' no banco de dados que é levado ao mesmo valor que J_j pela função de hash, isto é

$$H(M, N, J_j^*) = H(M, N, D^*).$$

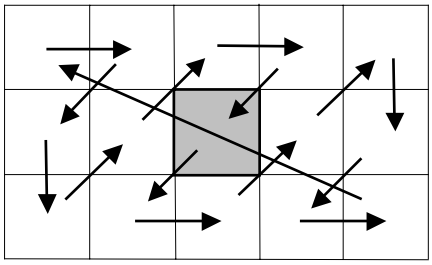
A probabilidade de sucesso é maior que 0,5 por causa do paradoxo do aniversário. O bloco J_j (com a assinatura pega de D') pode substituir o bloco I_t' sem ser detectado pelo esquema de Wong. Se este processo for repetido um número suficiente de vezes, uma imagem inteira falsificada pode ser gerada. Um ataque similar também pode ser executado contra a marca de Li.



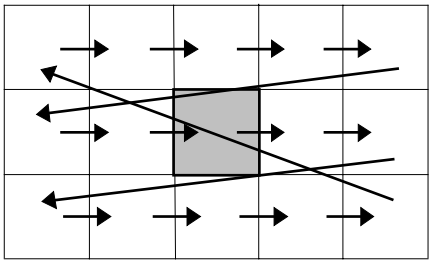
(4.1a) 4 dependências por bloco



(4.1b) 2 dependências por bloco



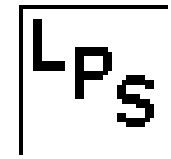
(4.1c) 1 dependência por bloco (zig-zag)



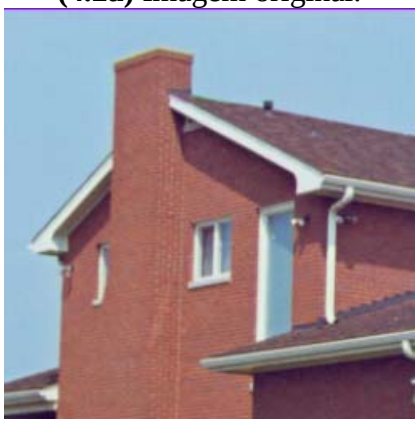
(4.1d) 1 dependência por bloco (raster)

Fig. 4.1: Uso da informação contextual. Para calcular a assinatura de um bloco I_t (mostrado em cinza), o conteúdo do bloco I_t e de seus blocos vizinhos são levados em conta. O HBC utiliza 1 dependência por bloco, em ordem zig-zag ou raster.

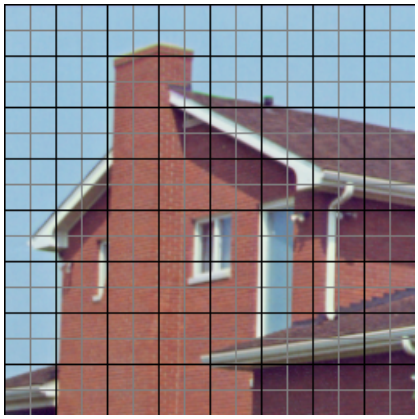
(4.2a) Imagem original.



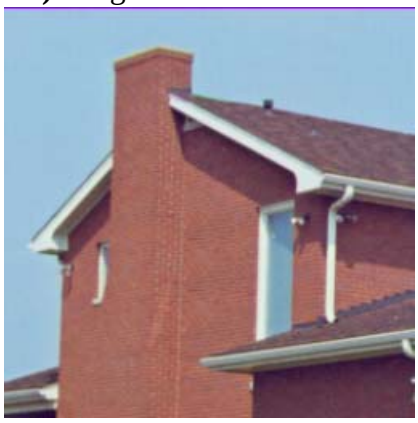
(4.2b) Imagem-logotipo 32×32.



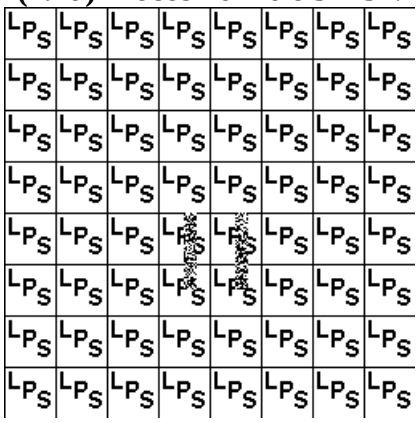
(4.2c) Imagem marcada com HBC2.



(4.2d) Blocos 16×16 e 32×32.



(4.2e) Ataque recortar-e-colar.



(4.2f) Delimitação das alterações.

Fig. 4.2: Impedindo o ataque "recortar-e-colar" com HBC2. Uma imagem colorida 256×256 original (a) foi marcada usando a chave privada e uma imagem logotipo 32×32 (b), gerando a imagem marcada (c). A imagem (d) mostra os seus blocos constituintes. A imagem marcada (c) sofreu um ataque "recordar e colar" (e), indetectável pelo esquema de Wong. Usando o HBC2, os blocos alterados podem ser localizados (f). Note que o HBC2 detecta somente as bordas dos blocos 16×16 alterados.

Hash block chaining versão 1

Conforme mostrado em [Cn05; Cn07; Ri04; Holliman and Memon, 2000], a solução para impedir os ataques descritos anteriormente é introduzir uma informação contextual. Isto é, no cálculo da impressão digital H_t, alimentar a função de hashing *H* com os blocos vizinhos de I_t^*, além do próprio bloco I_t^* (veja a figura 4.1). Neste caso, se um bloco I_t' for alterado, a verificação da assinatura irá falhar em todos aqueles blocos que dependem de I_t', além do próprio bloco I_t'. Portanto, um número tão pequeno quanto possível de dependências é desejável para uma localização acurada da alteração na imagem. Idealmente, uma única dependência por bloco. O seguinte esquema implementa esta idéia:

$$H_t \equiv H(M, N, I_t^*, I_{(t-1) \bmod n}^*, t).$$

O índice do bloco *t* foi inserido para detectar a rotação bloco a bloco. Assim como no esquema de Wong, os tamanhos *M* e *N* da imagem são inseridos para detectar cortes na imagem. Chamamos esta construção de *hash block chaining*, versão 1 (HBC1). Repetimos que se um bloco I_t' for alterado, o HBC1 irá reportar que o bloco $I'_{(t+1) \bmod n}$ é inválido (além do próprio I_t').

Usando o HBC1, o ataque recortar-e-colar simples não mais pode ser executado, pois se um bloco espúrio for colado no lugar de I_t', com probabilidade muito alta esta alteração irá introduzir uma alteração em $H_{(t+1) \bmod n}$. A probabilidade de que tal mudança não acontecer é de apenas $O(m^{-1})$. Esta alteração invalida a assinatura do bloco $I'_{(t+1) \bmod n}$. Assim, o ataque recortar-e-colar (e conseqüentemente, o ataque de falsificação) não pode ser mais executado.

De forma similar, se um ataque de aniversário for executado, o conteúdo alterado de I_t' induz com alta probabilidade uma mudança em $H_{(t+1) \bmod n}$. Assim, o atacante terá de forjar a assinatura do bloco $I'_{(t+1) \bmod n}$ também, perpetrando um outro ataque.

Mas isto induz uma mudança no bloco $I'_{(t+2) \bmod n}$. Portanto, o atacante irá defrontar com o problema de assinaturas inválidas propagarem ciclicamente sobre todos os blocos, eventualmente destruindo a assinatura forjada do primeiro bloco falsificado.

Ataque de transplante

O HBC1 é efetivo contra ataques recortar-e-colar, falsificação e aniversário. Mas não é seguro contra uma forma melhorada do ataque recortar-e-colar descrita abaixo. De fato, o HBC1 ou qualquer outra técnica de partição que aumenta a função de hashing com contexto determinístico e limitado dos blocos vizinhos são suscetíveis ao que chamamos um ataque de transplante. Para isto, sejam X' e $\overline{X}'$ duas imagens com marcas d'água tipo HBC1. Vamos denotar o fato da impressão digital de um bloco X'_B depender do conteúdo do bloco X'_A (isto é, X^*_A) de $X'_A \to X'_B$. Suponha que as imagens X' e $\overline{X}'$ possuam os blocos conforme mostrados abaixo:

$$\cdots \to X'_A \to X'_D \to X'_B \to X'_C \to \cdots,$$

$$\cdots \to \overline{X}'_A \to \overline{X}'_E \to \overline{X}'_B \to \overline{X}'_C \to \cdots,$$

onde $X^*_A = \overline{X}^*_A$, $X^*_B = \overline{X}^*_B$, $X^*_C = \overline{X}^*_C$ mas $X^*_D \neq \overline{X}^*_E$. Então, o par de blocos (X'_D, X'_B) pode ser trocado com o par $(\overline{X}'_E, \overline{X}'_B)$, sem ser detectado pelo esquema HBC1:

$$\cdots \to X'_A \to \overline{X}'_E \to \overline{X}'_B \to X'_C \to \cdots,$$

$$\cdots \to \overline{X}'_A \to X'_D \to X'_B \to \overline{X}'_C \to \cdots.$$

As imagens de documentos normalmente apresentam amplas áreas brancas, o que as torna muito suscetíveis a ataques de transplante. Por exemplo, se X'_A, X'_B, X'_C, $\overline{X}'_A$, $\overline{X}'_B$ e $\overline{X}'_C$ fossem todos blocos brancos sem ruído, o ataque teria sucesso facilmente. Note que simplesmente aumentar o número de dependências não consegue evitar o ataque de transplante. Se existirem duas dependências por bloco, como ilustrado abaixo, a tripla de blocos (X'_B, X'_E, X'_C) poderia ser trocada com a tripla $(\overline{X}'_B, \overline{X}'_F, \overline{X}'_C)$.

$$\cdots \leftrightarrow X'_A \leftrightarrow X'_B \leftrightarrow X'_E \leftrightarrow X'_C \leftrightarrow X'_D \leftrightarrow \cdots,$$

$$\cdots \leftrightarrow \overline{X}'_A \leftrightarrow \overline{X}'_B \leftrightarrow \overline{X}'_F \leftrightarrow \overline{X}'_C \leftrightarrow \overline{X}'_D \leftrightarrow \cdots.$$

Ataques semelhantes também podem ser executados contra 4 dependências ou 8 dependências.

Ataque de aniversário melhorado

O esquema HBC1 também não consegue resistir a um ataque de aniversário mais sofisticado. Este ataque substitui dois blocos consecutivos X'_t e X'_{t+1} pelos blocos forjados J_t e J_{t+1} (omitiremos "mod *n*" nos índices para simplificar a notação). Três impressões digitais são afetadas por estas substituições: H_t (que depende de X'_t), H_{t+1} (que depende de ambos X'_t e X'_{t+1}), e H_{t+2} (que depende de X'_{t+1}). Suponha que o banco de dados tenha *s* blocos assinados.

O atacante prepara *p* variantes visualmente equivalentes para J_t. Então, provavelmente $P \cong ps/m$ colisões para H_t serão encontradas (veja [Nishimura and Sibuya, 1990]). Mais explicitamente, P pares (J_t^1, D_t^1), ..., (J_t^P, D_t^P) serão encontrados, onde J_t^1, ..., J_t^P são as variantes visualmente equivalentes de J_t e D_t^1, ..., D_t^P são os blocos do banco de dados tais que a impressão digital de D_t^i é o mesmo que a impressão digital de J_t^i. Isto é:

$$H(M, N, D_t^{i*}, X_{t-1}^*, t) = H(M, N, J_t^{i*}, X_{t-1}^*, t), \text{ para } 1 \le i \le P.$$

Conseqüentemente, a assinatura do bloco *t* permanecerá válida se X_t for substituído por qualquer bloco J_t^{i*} junto com a assinatura obtida dos LSBs de D_t^i. Porém, quase certamente esta substituição irá tornar inválida a assinatura do bloco *t*+1.

De forma semelhante, o atacante prepara *q* variantes de J_{t+1}, provavelmente gerando $Q \cong qs/m$ colisões para H_{t+2}. Sejam (J_{t+1}^1, D_{t+2}^1), ..., (J_{t+1}^Q, D_{t+2}^Q) os pares que se colidem, isto é:

$$H(M, N, X^{*}_{t+2}, J^{j*}_{t+1}, t+2) = H(M, N, X^{*}_{t+2}, D^{j*}_{t+2}, t+2), \text{ para } 1 \le j \le Q.$$

A assinatura do bloco t+2 irá permanecer válida se X_{t+1} for substituída por quaisquer J^{j*}_{t+1} juntamente com a assinatura obtida dos LSBs de D^{j}_{t+2}. Mas esta substituição irá provavelmente tornar inválida a assinatura do bloco t+1.

Combinando todas as variantes de J_t e J_{t+1} que colidem irá gerar aproximadamente $(ps/m)(qs/m) = pqs^2/m^2$ pares (J^{i}_{t}, J^{j}_{t+1}), visualmente equivalentes a (J_t, J_{t+1}). Agora, o atacante deve achar uma colisão para H_{t+1}, isto é, deve achar um par variante (J^{i}_{t}, J^{j}_{t+1}) e um bloco do banco de dados D_{t+1} tais que:

$$H(M, N, J^{j*}_{t+1}, J^{i*}_{t}, t+1) = H(M, N, D^{*}_{t+1}, J^{i*}_{t}, t+1).$$

Então, se X_t e X_{t+1} forem substituídos pelos blocos falsificados J^{i*}_{t} e J^{j*}_{t+1} e, ao mesmo tempo, as assinaturas dos blocos t, t+1 e t+2 forem substituídos pelas assinaturas obtidas dos LSBs de D^{i}_{t}, D_{t+1} e D^{j}_{t+2}, a adulteração passará despercebida pelo HBC1.

Quais devem ser os tamanhos p e q para que a chance de sucesso seja maior que 50%? Como existem pqs^2/m^2 pares de blocos e s blocos de banco de dados, uma colisão para H_{t+1} irá provavelmente ocorrer quando $(pqs^2/m^2)s \approx m$, isto é, quando $pq \approx (m/s)^3$. Portanto, se o banco de dados possuir $s \approx \sqrt{m}$ assinaturas válidas, provavelmente dois blocos falsificados podem substituir dois blocos consecutivos válidos quando $p \approx q \approx m^{3/4}$ blocos variantes, visualmente equivalentes a cada bloco falsificado, são preparados.

Hash block chaining versão 2

Melhoramos o esquema HBC1 para resistir aos ataques de transplante e de aniversário melhorado. Esta versão melhorada foi denominada HBC2 e faz uso de assinatura não-determinística. Alguns esquemas de assinaturas (por exemplo, DSA e Schnorr

[Menezes, 1997, seção 11.5]) são não-determinísticos no sentido que cada assinatura individual depende não somente da função de hashing, mas também de algum parâmetro escolhido aleatoriamente. Usando uma assinatura não-determinística, mesmo as assinaturas de duas imagens idênticas serão diferentes. Esta propriedade efetivamente previne os ataques de transplante. Uma assinatura determinística (como RSA) pode ser convertida numa não-determinística acrescentando "sal" (isto é, um dado arbitrário, estatisticamente único) à mensagem sendo assinada. O esquema HBC2 é definido como segue:

$$H_t \equiv H(M, N, I_t^*, I_{(t-1) \bmod n}^*, t, S_{t-1}),$$

onde S_{t-1} é a assinatura não-determinística do bloco I_{t-1}, e $S_{-1} \equiv \varnothing$. Note que não podemos usar $S_{(t-1) \bmod n}$ porque quando a impressão digital H_0 estiver sendo calculada, a assinatura S_{-1} ainda não será conhecida.

O ataque de aniversário melhorado é completamente ineficaz contra o HBC2, pois no HBC2 a assinatura de um bloco depende não somente do conteúdo do bloco vizinho, mas também da sua assinatura não-determinística. Vamos supor que um atacante tenha conseguido substituir dois blocos consecutivos válidos X_t e X_{t+1} por dois blocos falsificados J_t e J_{t+1}, e três assinaturas S_t, S_{t+1} e S_{t+2} por três assinaturas falsificadas (mas válidas) L_t, L_{t+1}, L_{t+2} enquanto mantém intacto o conteúdo do bloco X_{t+2}. Note que esta substituição é muito mais difícil no HBC2 do que no HBC1 devido à assinatura não-determinística e a dependência da assinatura. Mesmo neste cenário improvável, o HBC2 irá reportar uma alteração, pois H_{t+3} depende não somente do conteúdo de X_{t+2}, que não se altera, mas também da sua assinatura, que quase certamente muda.

O uso do HBC2 tem um surpreendente e agradável efeito colateral. Tipicamente, o ataque de aniversário pode ser executado contra uma função hashing de comprimento m com um esforço de $O(\sqrt{m})$ passos. Porém, para o HBC2, nenhum ataque que leva menos de $O(m)$ passos é conhecido. Portanto, parece que, num cenário otimista, o comprimento da função hashing poderia ser cortado pela metade mantendo o nível de segurança original. Porém não recomendamos reduzir o comprimento da hashing até

que esta conjectura seja analisada em maior profundidade, pois tal redução poderia afetar a segurança do próprio algoritmo de assinatura.

O HBC2 é capaz de detectar se algum bloco foi modificado, rearranjado, apagado, inserido, ou transplantado de uma imagem legitimamente assinada. Além disso, indica ou quais blocos foram alterados ou, se uma grande região validamente marcada for copiada, onde ficam as bordas da região alterada. Chamamos atenção que a capacidade de localização é perdida se um bloco (ou uma linha ou uma coluna) for inserido ou apagado, embora mesmo neste caso o HBC2 irá reportar corretamente a presença de alguma alteração.

Discussões

Tipicamente, o comprimento de uma assinatura de logaritmo discreto é aproximadamente duas vezes o tamanho da função hashing utilizada [Menezes, 1997, seção 11.5]. Isto é melhor que as assinaturas RSA, cujo comprimento é sempre o da chave pública. Por exemplo, as assinaturas DSA têm comprimento 320 bits, enquanto que as assinaturas RSA com o nível de segurança equivalente devem ter aproximadamente 1024 bits. Neste sentido, as assinaturas Schnorr são as melhores para o HBC2 [Menezes, 1997, seção 11.5.3], uma vez que elas conseguem a redução máxima no tamanho da assinatura e portanto na quantidade de dados a serem incorporados na imagem hospedeira.

As experiências com o HBC2 utilizando a criptografia de curva elíptica resultaram em tempos de assinatura e verificação de aproximadamente 10 segundos num Pentium-500, para as imagens em níveis de cinza 512×512. A incerteza de localização de alteração foi menor que 0,2% da área da imagem.

Marca d'água de Wong-Memon

Wong e Memon [Wong and Memon, 2001] propuseram um esquema de marca d'água muito semelhante ao HBC2. O nosso trabalho reportado em [Ri04] foi desenvolvido independentemente do trabalho de Wong-Memon.

A diferença essencial entre os esquemas HBC2 e Wong-Memon é que o último utiliza um identificador $\mathscr{I}_I$ único para cada imagem I (por exemplo, um número seqüencial) que deve ser armazenado de alguma forma, fora da imagem. A existência desse identificador simplifica a construção de Wong-Memon, porém traz o desconforto ao usuário de ter que armazenar esse identificador de alguma forma (se é necessário armazenar esse número serial, por que não armazenar a própria assinatura num arquivo independente?). A função hash de Wong-Memon torna-se:

$$H_t \equiv H(\mathscr{I}_I, M, N, I_t^*, t).$$

Note que desta forma não é necessário mais alimentar a função hash com a informação de contexto do bloco I_t^*.

4.4 Marcas de Autenticação para Imagens Binárias e Meio-Tom

4.4.1 Introdução

Uma vez estudadas algumas técnicas de autenticação de imagens em tonalidade contínua sem compactação, naturalmente aparece a curiosidade de querer estendê-las para as imagens binárias e para os formatos de imagens compactadas com perdas. Nesta tese, estudaremos somente o primeiro caso.

Nas seções anteriores, vimos que é praticamente impossível que uma marca de autenticação seja realmente segura sem estar apoiada na sólida teoria criptográfica. De fato, aquelas marcas d'água que não estavam fundadas em criptografia [Zhao and Koch, 1995; Yeung and Mintzer, 1997] or aquelas que aplicaram as técnicas criptográficas sem o devido cuidado [Wong, 1997; Wong, 1998; Li et al., 2000] tiveram mais tarde as suas fraquezas descobertas [Holliman and Memon, 2000; Ri04].

Numa marca de autenticação baseada em criptografia, o código de autenticação de mensagem (MAC) ou a assinatura digital (DS) de toda a imagem é computado e inserido na própria imagem. Porém, a inserção do código MAC/DS altera a imagem e conseqüentemente altera o próprio MAC/DS, invalidando a marca. Para evitar este problema, para as imagens em níveis de cinza ou coloridas, normalmente os bits menos significativos (LSBs) são apagados, calcula-se o MAC/DS da imagem com os LSBs apagados, e então o código é inserido nos LSBs. Em outras palavras, aqueles bits onde o código será inserido não são levados em conta ao se calcular o MAC/DS.

Para as imagens binárias ou meio-tom, esta idéia falha completamente, porque cada pixel possui um único bit. Modificando qualquer pixel para embutir a marca, a impressão digital da imagem é alterada, invalidando a marca. Conseqüentemente, embora haja muitos artigos sobre as técnicas para esconder dados em imagens binárias [Deseilligny and Le Men, 1998; Baharav and Shaked, 1998; Chen et al., 2000; Wu et

al., 2000; Fu and Au, 2000; Fu and Au, 2002a], conhecemos poucas marcas de autenticação baseadas em criptografia para as imagens binárias e meio-tom. Fu e Au [Fu and Au, 2002b] apresentam uma marca para detectar as alterações não-intencionais em imagens meio-tom, mas esta não pode ser considerada uma marca de autenticação porque não resiste a um ataque intencional ou malicioso.

De acordo com o paradigma criptográfico amplamente aceito, a segurança de uma marca de autenticação deve estar apoiada somente no segredo da chave. O fato de que uma imagem foi marcada, assim como o algoritmo utilizado para marcar a imagem devem poder se tornar públicos sem comprometer a segurança do esquema. Nas próximas subsubseções, propomos duas marcas de autenticação para as imagens binárias que satisfazem este requerimento, que denominamos de AWST (authentication watermarking by self toggling) e AWSF (authentication watermarking by shuffling and flipping). As marcas AWST e AWSF são apropriadas respectivamente para as imagens meio-tom pontos dispersos e as imagens binárias em geral. A AWSF não é adequada para as imagens meio-tom pontos dispersos, mas pode ser utilizada em imagens meio-tom pontos aglutinados. Assim, as duas técnicas podem ser usadas de forma complementar para proteger qualquer imagem binária.

As marcas AWST e AWSF podem ser usadas com criptografias de chave secreta ou pública. A AWSF de chave pública necessita de cuidados especiais para evitar um ataque que denominamos de "ataque de paridade". Um uso possível de AWST/AWSF é em FAX seguro. Utilizando o FAX seguro, o receptor de um documento pode se certificar quem foi o gerador do documento, e que o documento não foi alterado (acidental ou maliciosamente) durante a transmissão.

Estas técnicas foram projetadas somente para autenticar as imagens digitais ortográficas. Para autenticar as imagens impressas, mais pesquisas são necessárias.

Esteganografias para imagens binárias e meio-tom

Existem três formas básicas de embutir dados em imagens binárias e meio-tom: alterar os valores dos pixels individuais, mudar as características de um grupo de pixels e mudar as características dos blocos da imagem.

A primeira abordagem troca as cores de determinados pixels [Fu and Au, 2000; Fu and Au, 2002; Tseng et al., 2002]. A técnica DHST, que será descrita adiante, pertence a esta categoria.

A segunda abordagem modifica as características tais como a posição do pixel superior esquerdo de cada componente conexo, a largura da pincelada, a curvatura, etc. [Maxemchuk and Low, 1997]. Esta abordagem normalmente depende do tipo de imagem e a quantidade de dados que pode ser inserida é limitada.

A terceira abordagem divide uma imagem em blocos e embute as informações através de alguma característica dos blocos da imagem. Por exemplo, poderia dividir uma imagem binária em blocos, digamos 8×8. Em cada bloco, um bit é embutido forçando o número de pixels brancos do bloco a ser par ou ímpar. Se o número de pixels brancos do bloco for par, convenciona-se que o bit zero está embutido naquele bloco [Wu et al., 2000]. Se for ímpar, o bit um está embutido. Se um bloco já representar o bit que se deseja inserir, não há nada a fazer. Caso contrário, procura-se pelo pixel que causará a menor degradação visual segundo algum critério perceptual e troca-se o seu valor. Evidentemente, é possível estender a idéia para inserir dois ou mais bits por bloco. Uma outra técnica orientada a blocos (mas que desta vez só se aplica para as imagens meio-tom) é alternar a matriz de pesos utilizada na difusão de erro de um bloco para outro [Pei and Guo, 2003; Hel-Or, 2001]. A imagem meio-tom é dividida em blocos e, dentro de cada bloco, utiliza-se uma determinada matriz de pesos (Floyd-Steinberg, Jarvis ou Stucki) para efetuar a difusão de erro. A matriz de pesos utilizada na geração da imagem meio-tom de um bloco pode ser determinada calculando a transformada de Fourier do bloco. Conforme a matriz utilizada, convenciona-se que está embutido o bit zero ou um.

Para as imagens meio-tom, podemos citar ainda uma quarta abordagem: uma imagem é escondida em duas imagens meio-tom de forma que ela torna-se visível quando as duas são sobrepostas [Wang, 1998; Fu and Au, 2001; Pei and Guo, 2003].

4.4.2 Marca de Autenticação AWST

Esta subseção descreve uma contribuição científica original nossa. Eu fui o principal responsável pelas pesquisas descritas, contando com a colaboração do meu orientando de mestrado A. Afif.

Técnica esteganográfica DHST

DHST (data hiding by self toggling) é a técnica esteganográfica que se enquadra na primeira das quatro categorias listadas na subseção anterior [Fu and Au, 2000; Fu and Au, 2002a]. Ela é especialmente interessante pela sua simplicidade. Essa técnica foi projetada originariamente para embutir bits em imagens meio-tom pontos dispersos.

Na DHST, um gerador de números pseudo-aleatórios com uma semente conhecida é usado para gerar um conjunto de posições pseudo-aleatórias não repetidas dentro da imagem. Um bit é embutido em cada posição forçando-a a ser preta ou branca. Com a probabilidade de 50%, o pixel na imagem original tem o valor desejado e portanto nenhuma mudança é necessária. Com a probabilidade de 50%, o pixel tem o valor oposto ao desejado, e o pixel deve ser alterado. É importante garantir que não haja repetições de posições pseudo-aleatórias, pois neste caso o algoritmo tentaria inserir dois ou mais bits de informação num único pixel, o que evidentemente levaria a erro. Para ler o dado escondido, deve-se simplesmente gerar novamente as mesmas posições pseudo-aleatórias não-repetidas e ler os valores nessas localizações.

Evidentemente, a DHST pode também ser usada para qualquer imagem binária. Porém, neste caso, um ruído sal-e-pimenta se tornará visível. Neste artigo, iremos converter DHST numa marca de autenticação criptograficamente segura.

Como DHST muda os valores dos pixels individuais nas posições pseudo-aleatórias selecionadas, a intensidade local média pode ser afetada severamente. Para resolver este problema, Fu e Au [Fu and Au, 2000] apresentam Data Hiding by Pair-Toggling (DHPT). A idéia desse algoritmo é, na posição escolhida pseudo-aleatoriamente, a mudança de valor de um pixel ser acompanhada, sempre que possível, pela mudança complementar de um vizinho. Por exemplo, se um pixel mestre é forçado mudar de 0 para 255, então os pixels vizinhos (na vizinhança 3×3) com valor 255 são identificados e um deles é escolhido aleatoriamente para mudar o seu valor para 0. Este pixel é chamado de pixel escravo. No mesmo artigo, Fu e Au apresentaram Data Hiding by Smart Pair Toggling (DHSPT). Consiste basicamente em estabelecer algumas regras para escolher o pixel escravo, entre os candidatos, de forma a perturbar o menos possível a qualidade visual da imagem meio-tom.

Marca de autenticação AWST

Numa marca de autenticação segura utilizando alguma técnica para embutir dados em imagens binárias, deve-se calcular a função de hashing da imagem binária *B*, obtendo a impressão digital $H = H(B)$. A impressão digital *H*, depois de efetuar ou-exclusivo e encriptar, torna-se a assinatura digital *S*. Esta assinatura digital deve ser inserida na própria imagem *B*, obtendo a imagem marcada *B*'. O problema é que, com a inserção da marca, a imagem *B* muda e conseqüentemente a sua impressão digital se altera. Isto é, teremos $H(B) \neq H(B')$. Como podemos superar esta dificuldade?

Apresentamos uma solução bem simples utilizando a DHST. Diferentemente da maioria de outras técnicas para embutir dados em imagens binárias, na DHST somente uns poucos bits são modificados e as posições desses bits são conhecidas tanto na fase de inserção como na fase de extração. Conseqüentemente, estes pixels podem ser zerados antes de calcular a função de hashing, da mesma forma que LSBs são zerados para marcar imagens em níveis de cinza. Vamos chamar a técnica assim obtida de AWST (authentication watermarking by self toggling). O algoritmo de inserção da marca AWST é:

1. Seja *B* a imagem binária a ser marcada e seja *A* a imagem logotipo binária a ser inserida em *B*.
2. Use um gerador de números pseudo-aleatórios com uma semente conhecida para gerar um conjunto de posições pseudo-aleatórias não-repetidas *L* dentro da imagem *B*.
3. Zere todos os pixels de *B* que pertencem a *L*, obtendo *B**.
4. Calcule a impressão digital $H = H(B^*)$.
5. Calcule o ou-exclusivo de *H* com *A*, obtendo a impressão digital marcada $\hat{H}$.
6. Criptografe $\hat{H}$ com a chave secreta (criptografia simétrica) ou privada (criptografia assimétrica), gerando a assinatura digital *S*.
7. Insira *S* no conjunto de pixels *L*, gerando a imagem marcada *B*'.

O algoritmo de verificação da marca AWST é:

1. Seja *X*' a imagem marcada. Usando o mesmo gerador de números pseudo-aleatórios, gere novamente o mesmo conjunto de posições pseudo-aleatórias não-repetidas *L* onde a marca foi inserida.
2. Seja *X** a imagem obtida de *X*' zerando todos os pixels de *L*. Usando a mesma função de hashing, calcule a impressão digital $H = H(X^*)$.
3. Extraia a marca de *X*' lendo os pixels de *L* e decriptografando-os com a chave secreta (criptografia simétrica) ou pública (criptografia assimétrica), obtendo os dados decriptografados *D*.
4. Calcule o ou-exclusivo de *H* com *D*, obtendo a imagem de checagem *C*.
5. Se *C* e *A* são iguais, a marca está verificada. Caso contrário, a imagem marcada *X*' foi alterada.

A figura 4.3 ilustra o uso da marca d'água de autenticação AWST. Vamos supor que a imagem *B* (figura 4.3a) seja uma imagem suscetível a ataques a ser transmitida através de um canal pouco confiável, onde as alterações maliciosas podem ocorrer. Para proteger *B*, a imagem logotipo *A* (figura 4.3b) foi inserida em *B* usando o algoritmo AWST. A imagem *B*' (figura 4.3c) é a imagem marcada onde 1024 bits foram inseridos. Isto é suficiente para embutir uma assinatura digital RSA [Schneier, 1996].

Se executar o algoritmo de verificação, obtemos a imagem de checagem *C* (figura 4.3d) exatamente igual à imagem logotipo *A*. Mesmo que um único pixel de *B*' seja alterado, a imagem extraída será completamente ruidosa (figura 4.3f).

A figura 4.4 mostra a qualidade de um documento marcado com a AWST. Uma página de uma revista foi escaneada em 300 dpi, resultando numa imagem binária com 3318 linhas e 2536 colunas (figura 4.4a). As figuras 4.4b, 4.4c e 4.4d mostram respectivamente as imagens com 64, 320 e 1024 bits embutidos. Estas quantidades de bits são suficientes para inserir, respectivamente, um MAC com chave secreta, uma assinatura digital DSA e uma assinatura digital RSA.

Resposta booleana

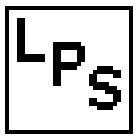
Embora extrair uma imagem logotipo visível da imagem marcada possa ser fascinante, na realidade somente necessitamos receber uma resposta binária à seguinte pergunta: "a imagem marcada contém ou não uma marca válida?" Para obter esta resposta booleana, podemos eliminar o passo 5 do algoritmo de inserção da AWST e o passo 4 do algoritmo de verificação da AWST.

Mantendo inalterado a intensidade média local

A qualidade visual de uma imagem meio-tom pontos dispersos marcada com AWST pode ser melhorada usando as técnicas para embutir dados DHPT ou DHSPT [Fu and Au, 2000; Fu and Au, 2002a], em vez de DHST. Estes melhoramentos procuram manter inalterada a intensidade média local. Nas posições pseudo-aleatórias selecionadas, a alteração de um pixel é acompanhada pela modificação complementar de um pixel vizinho.

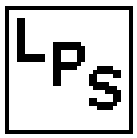
Entretanto, para implementar um desses esquemas, nenhum pixel vizinho dos pixels psedo-aleatórios pode alimentar a função hashing. Conseqüentemente, esses pixels permanecerão desprotegidos, isto é, se uma alteração ocorrer num pixel vizinho de uma posição pseudo-aleatória, essa alteração não será detectada pela marca AWST.

(4.3a) Imagem meio-tom *B* (512×512 pixels) a ser protegida com a marca de autenticação.



(4.3b) Imagem logo *A* (32×32 pixels) a ser inserida em *B.*

(4.3c) Imagem *B*' com marca d'água. 1024 bits foram inseridos.

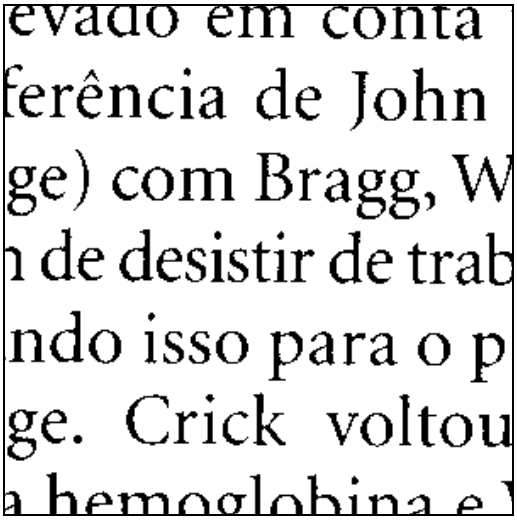


(4.3d) Imagem logo extraída da imagem *B*'.

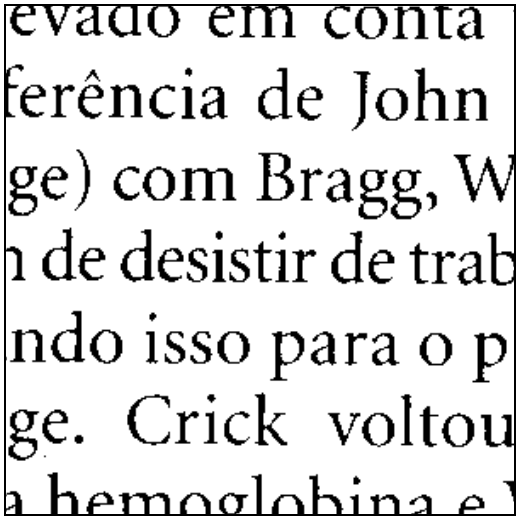
(4.3e) Imagem alterada *X*'.

(4.3f) Imagem logo extraída de *X*'.

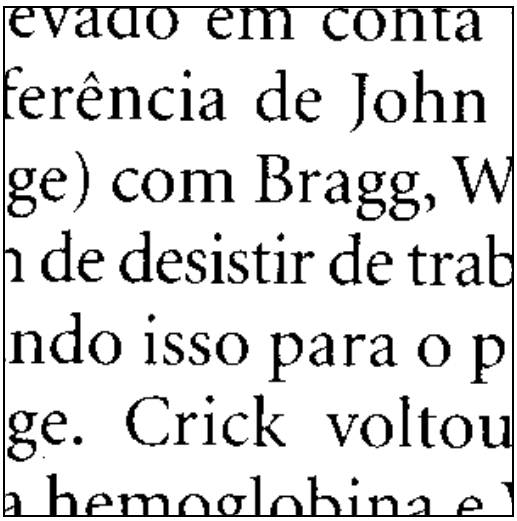
Fig. 4.3: Ilustração da AWST chave pública. A imagem logo *A* (b) foi inserida na imagem *B* (a) usando uma cifra de chave pública. A figura (c) mostra a imagem marcada. Executando o algoritmo de extração da marca, a figura (d) foi obtida. Se a imagem marcada for modificada mesmo que seja ligeiramente (e), uma imagem completamente aleatória é extraída (f).


evado em conta
ferência de John
ge) com Bragg, W
n de desistir de trab
ndo isso para o p
ge. Crick voltou
a hemoglobina e

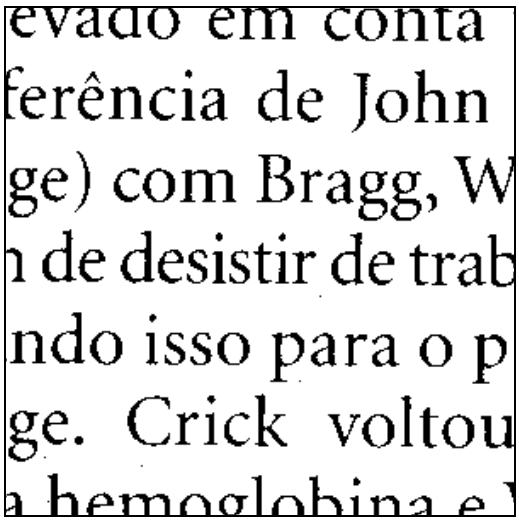
(4.4a) Parte da imagem original.


evado em conta
ferência de John
ge) com Bragg, W
n de desistir de trab
ndo isso para o p
ge. Crick voltou
a hemoglobina e

(4.4b) Imagem com 64 bits embutidos (apropriada para inserir um código de autenticação de mensagem de chave secreta).


evado em conta
ferência de John
ge) com Bragg, W
n de desistir de trab
ndo isso para o p
ge. Crick voltou
a hemoglobina e

(4.4c) Imagem com 320 bits embutidos (apropriada para inserir uma assinatura DSA).


evado em conta
ferência de John
ge) com Bragg, W
n de desistir de trab
ndo isso para o p
ge. Crick voltou
a hemoglobina e

(4.4d) Imagem com 1024 bits embutidos (apropriada para inserir uma assinatura RSA).

Fig. 4.4: Qualidade dos documentos marcados com AWST. (a) Uma página de uma revista foi escaneada em 300 dpi, resultando numa imagem binária com 3318 linhas e 2536 colunas. (b) A marca AWST, utilizando chave secreta, necessita inserir 64 bits na imagem. (c) Usando a assinatura DSA, 320 bits devem ser embutidos na imagem. (d) Usando a assinatura RSA, 1024 bits devem ser embutidos. Note que a degradação de imagem é baixa mesmo embutindo 1024 bits.

4.4.3 Marca de Autenticação AWSF

Esta subseção descreve uma contribuição científica original nossa. Eu fui o principal responsável pelas descobertas descritas, contando com a colaboração do prof. Ricardo de Queiroz da UnB.

Técnica esteganográfica de Wu et al.

Entre as técnicas de esteganografias para as imagens binárias, a de Wu et al. [Wu et al., 2000] é especialmente interessante. Ela pode ser aplicada a maioria das imagens binárias, pode embutir uma quantidade moderada de dados, e a qualidade visual de uma imagem marcada com esta técnica é excelente. Ela pode ser sumarizada como segue:

1) Divida a imagem *Z* a ser marcada em pequenos blocos (digamos, 8×8).
2) A vizinhança de cada pixel (normalmente 3×3) é analisada para calcular a "nota de impacto visual" (VIS - visual impact score). Por exemplo, os pixels na borda de um componente conexo terão VIS's baixas, enquanto que um pixel completamente cercado por pixels brancos (ou pretos) terá VIS alta.
3) Insira um bit em cada bloco, modificando (se necessário) o conteúdo do pixel dentro do bloco com a menor VIS, forçando o bloco a ter um número ímpar de pixels brancos (para inserir o bit 1) ou um número ímpar (para inserir o bit 0).
4) Como diferentes blocos podem ter diferentes quantidades de pixels com VIS's baixas (por exemplo, todos os pixels num bloco completamente branco ou preto terão VIS's altas), Wu et al. sugerem embaralhar a imagem *Z* antes de inserir os dados.

AWSF versão 1

O artigo [Wu et al., 2000] dedicou somente umas poucas linhas para afirmar que a técnica por eles proposta poderia ser usada para detectar alterações em documentos binários, sem dar detalhes técnicos.

A idéia óbvia de calcular o código MAC/DS da imagem toda e inseri-lo na mesma imagem falha porque a inserção irá modificar a impressão digital da imagem. A primeira idéia para inserir MAC/DS, sem modificar a impressão digital da imagem, é dividir a imagem em duas regiões: a primeira (pequena) região onde MAC/DS será inserido, e a segunda (grande) região onde a impressão digital será calculada. Vamos escrever esta idéia de forma algorítmica:

1) Seja dada uma imagem binária *Z*. Usando um gerador de números pseudo-aleatórios com uma semente fixa, construa uma estrutura de dados auxiliar chamada vetor de embaralhamento *V*, de forma que a imagem *Z* possa ser vista como uma seqüência de pixels $\tilde{Z}$ completamente embaralhada. Na versão AWSF chave secreta, a própria chave secreta é usada como a semente do gerador pseudo-aleatório. Na versão chave pública/privada, deve-se tornar a semente conhecida publicamente. Vamos considerar um pequeno exemplo para deixar as idéias mais claras. Considere a seguinte imagem binária *Z* com somente 3×3 pixels:

$$Z = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

Seja o vetor de embaralhamento *V*, onde cada elemento é um índice (linha, coluna) para a imagem *Z*, dado por:

$$V = [(0,0);(1,2);(0,2);(2,1);(2,0);(0,1);(1,1);(1,0);(2,2)]$$

Então, a seqüência embaralhada de pixels $\tilde{Z}$ (a imagem *Z* acessada através dos índices de *V*) é:

$$\tilde{Z} = [1, 1, 1, 0, 1, 0, 1, 0, 0].$$

2) Seja *n* o comprimento do código MAC/DS adotado, e seja *m* o número de pixels de cada bloco. Divida a seqüência embaralhada $\tilde{Z}$ em duas regiões:

- Primeira região $\tilde{Z}_1$ com *n*×*m* pixels, onde o MAC/DS será armazenado. Esta região está subdividida em *n* blocos com *m* pixels cada. Em cada bloco, um bit do MAC/DS será inserido.

- Segunda região $\tilde{Z}_2$ com o restante da seqüência embaralhada $\tilde{Z}$. O algoritmo de inserção irá calcular a impressão digital desta região.

3) Usando uma função de hashing *H* segura do ponto de vista criptográfico, calcule a impressão digital da segunda região $H = H(\tilde{Z}_2)$. Criptografe a impressão digital *H* usando a chave secreta ou privada, obtendo o MAC/DS: $S = K(H)$.

4) Insira *S* na primeira região, obtendo a imagem marcada Z'. Insira um bit de *S* em cada bloco, modificando (se necessário) o conteúdo do pixel do bloco com a menor VIS, para forçar o bloco a ter um número par/ímpar de pixels brancos.

O algoritmo de verificação AWSF1 aplicado a uma imagem marcada Z' é:

1) Calcule o mesmo vetor de embaralhamento *V* usado para a inserção. Note que na versão chave secreta, a chave é também a semente do gerador de números pseudo-aleatórios e conseqüentemente somente o proprietário da chave pode reconstruir o vetor de embaralhamento. Porém, na versão chave pública/privada, a semente é publicamente conhecida e conseqüentemente o vetor de embaralhamento também é publicamente conhecido.

2) Divida Z' em duas regiões Z'_1 e Z'_2. Calcule a impressão digital *H* de Z'_2.

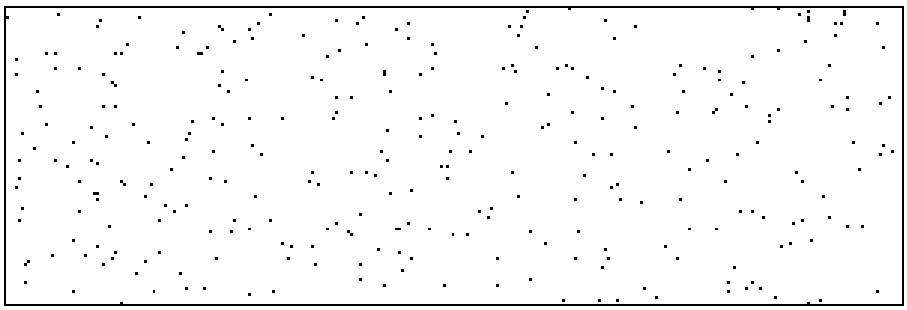
3) Extraia o MAC/DS armazenado em Z'_1 e decriptografe-o usando a chave secreta ou pública, obtendo o dado de checagem *D*.

4) Se $D = H$, a marca está verificada. Caso contrário, a imagem Z' foi modificada ou uma chave incorreta foi usada.

A figura 4.5a mostra parte de uma página da revista escaneada em 300 dpi, que poderia ser considerada como um documento binário "típico". A figura 4.5b é a imagem correspondente depois de embutir 1024 bits usando a marca AWSF1, com blocos de 64 pixels.

(4.5a) Parte de uma página de uma revista escaneada em 300 dpi.

(4.5b) Parte da imagem com 1024 bits embutidos com AWSF.



(4.5c) Pixels pretos pertencem à região 1, e pixels brancos à região 2.

Fig. 4.5: Qualidade visual de um documento marcado com a marca AWSF.

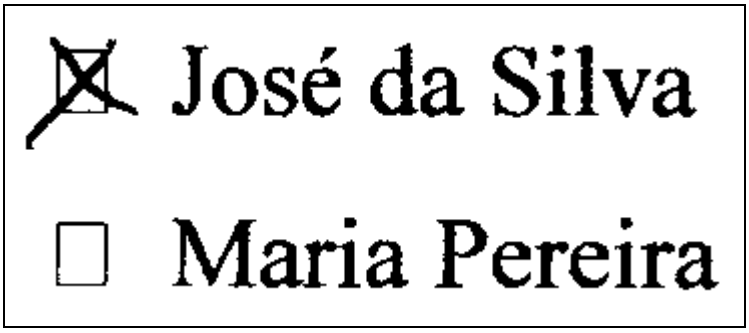
Ataque de paridade

A marca AWSF1 consegue detectar qualquer alteração perpetrada na segunda região da imagem marcada, mesmo a modificação de um único pixel. De fato, a probabilidade de não detectar uma alteração nesta região é somente 2^{-n} (*n* é o comprimento do MAC/DS adotado), o que pode ser desprezado.

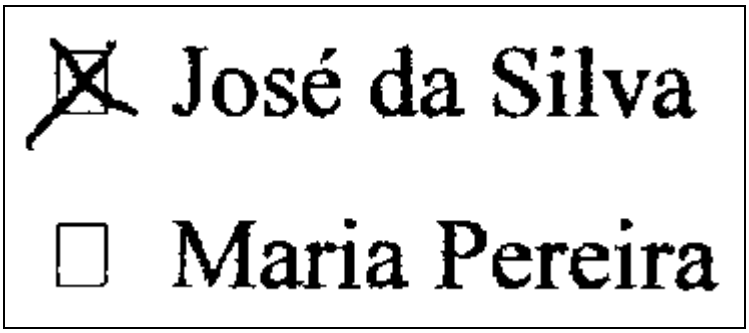
Infelizmente, uma alteração que mantenha a paridade dos blocos na primeira região não pode ser detectada pela AWSF1. Por exemplo, se dois pixels que pertencem ao mesmo bloco mudam os seus valores, a paridade deste bloco não será alterada e conseqüentemente esta alteração passará sem ser detectada. Denominamos este tipo de modificação de "ataque de paridade".

Se a imagem marcada Z' for suficientemente grande, os pixels de Z_1' constituirão pixels isolados dispersos aleatoriamente na imagem Z' e é improvável que Mallory, um hacker malicioso, possa introduzir qualquer alteração visualmente significativa em Z' mudando somente os pixels de Z_1' (enquanto mantém a paridade de cada um dos blocos). Por exemplo, na figura 4.5c, os pixels pretos pertencem à região 1. Esses pixels estão inteiramente dispersos, e nenhuma alteração visualmente significativa poderá resultar modificando somente esses pixels.

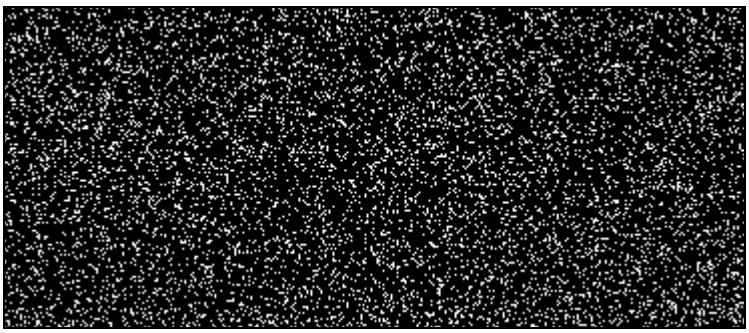
Entretanto, se a imagem Z' for pequena, os pixels de Z_1' podem formar regiões contíguas em Z', o que levanta a possibilidade de que uma modificação visualmente significativa passe sem ser detectada pelo AWSF1. Por exemplo, a figura 4.6a é a imagem de uma cédula de votação e a figura 4.6b é a mesma imagem marcada com AWSF1. A figura 4.6c mostra pixels que pertencem à região 1 em preto. Qualquer pixel da região 1 pode ser modificado, desde que um outro pixel no mesmo bloco também seja modificado. Para obter uma imagem falsificada Mallory, um hacker malicioso, muda um pixel *p* do bloco *i*. Então, ele procura por um outro pixel no bloco *i* com a menor nota VIS e modifica o seu valor. A figura 4.6d mostra uma imagem construída repetindo esta idéia. Esta alteração não será detectada por AWSF1.



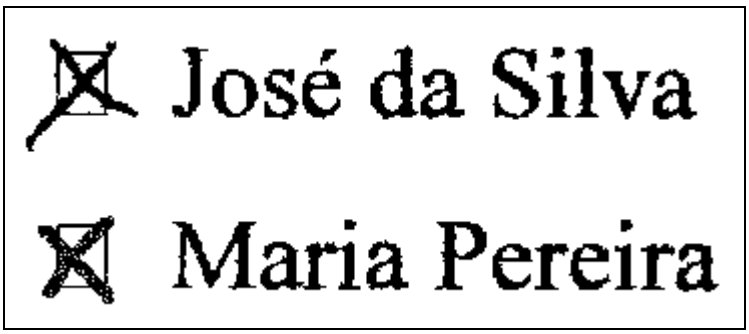
(4.6a) Uma pequena imagem (370×160) a ser marcada.



(4.6b) Imagem com 800 bits embutidos, usando blocos de 64 pixels.



(4.6c) Pixels pretos pertencem à região 1, onde um ataque de paridade pode ocorrer.



(4.6d) Imagem falsificada gerada pelo ataque de paridade, indetectável pela AWSF1.

Fig. 4.6: Falsificação "ataque de paridade", indetectável pela marca AWSF1.

Na verdade, o cenário descrito acima somente se aplica à versão chave pública de AWSF1, onde as localizações das regiões 1 e 2, assim como a subdivisão da região 1 em blocos, são conhecidas publicamente.

Na versão chave secreta da AWSF1, não é necessário preocuparmos *muito* com o ataque de paridade, pois a chave secreta é usada para gerar o vetor de embaralhamento. Assim, Mallory não irá conhecer como a imagem marcada está dividida em regiões 1 e 2, e como a região 1 está subdividida em blocos. Entretanto, devemos nos preocupar *um pouco,* pois Mallory pode ter muitos meios diferentes pra obter pistas sobre as localizações das regiões e blocos. Por exemplo, vamos supor que Mallory tenha acesso a um banco de dados com muitos pares de documentos original e marcado com a AWSF1, todos com o mesmo tamanho e todos marcados usando a mesma chave secreta. Neste caso, ele terá conhecimento de que todos os pixels cujos valores são diferentes nos documentos original e marcado pertencem à região 1.

AWSF versão 2

Para minimizar a possibilidade de um ataque de paridade, sugerimos o seguinte melhoramento no passo 4 do algoritmo de inserção da AWSF1:

4) Insira S na primeira região utilizando o seguinte algoritmo, gerando a imagem marcada Z':

Para i = 0 até n-1 {

Insira bit i de S no bloco i, forçando-o a ter um número par/ímpar de pixels brancos;

Calcule o novo MAC/DS S, alimentando a função de hashing com o conteúdo do bloco i e criptografando-o:

$S \leftarrow K(H(S, \text{pixels do bloco } i))$;

}

Desta forma, o bloco n-1 ainda pode sofrer um ataque de paridade. Porém, se o bloco n-2 for modificado sem modificar a sua paridade, com 50% de chance esta modifica-

ção irá ser detectada. Se o bloco *n*-3 for modificado (mantendo a paridade), há uma probabilidade de 75% de se detectar esta mudança. Se o bloco 0 for alterado (mantendo a sua paridade), existe uma probabilidade de $1-2^{-(n-1)}$ de se detectar esta mudança. Assim, a AWSF2 certamente torna muito mais improvável que Mallory consiga perpetrar um ataque de paridade visualmente significativo com sucesso.

4.5 Conclusões

Neste capítulo, descrevemos as nossas contribuições científicas na área de marca d'água de autenticação.

Para isso, na seção "introdução", definimos os conceitos necessários para a compreensão desta área.

Na seção "assinatura digital", explicamos o funcionamento de uma assinatura digital, um conceito essencial no estudo das marcas de autenticação.

Na seção "marcas de autenticação para imagens contone", descrevemos as marcas de autenticação de Yeung-Mintzer e de Wong, as fraquezas das ambas marcas, e a técnica hash block chaining, proposta por nós para robustecer a marca de Wong.

Na seção "marcas de autenticação para imagens binárias e meio-tom", argumentamos que criar uma marca de autenticação para as imagens binárias possui dificuldades intrínsecas. Descrevemos as duas marcas de autenticação para imagens binárias propostas por nós para as imagens binárias e meio-tom denominadas, respectivamente, de AWST e AWSF.

Capítulo 5:

Referências Bibliográficas

5.1 Publicações do Autor

Dissertação e tese

[T01] H. Y. Kim, *Síntese de Imagem e Rastreamento de Raio*, dissertação de mestrado, Instituto de Matemática e Estatística, Universidade de São Paulo, l992.

[T02] H. Y. Kim, *Construção Automática de Operadores Morfológicos por Aprendizagem Computacional*, tese de doutorado, Escola Politécnica, Universidade de São Paulo, 1997.

Artigos em revistas internacionais

[Ri05] H. Y. Kim, "Binary Halftone Image Resolution Increasing by Decision-Tree Learning," accepted for publication in *IEEE Trans. on Image Processing*.

[Ri04] P. S. L. M. Barreto, H. Y. Kim and V. Rijmen, "Toward a Secure Public-Key Blockwise Fragile Authentication Watermarking," *IEE Proc. Vision, Image and Signal Processing*, vol. 149, no. 2, pp. 57-62, 2002.

[Ri03] H. Y. Kim, "Binary Operator Design by k-Nearest Neighbor Learning with Application to Image Resolution Increasing," *Int. J. Imaging Systems and Technology*, vol. 11, no. 5, pp. 331-339, 2000.

[Ri02] H. Y. Kim, F. A. M. Cipparrone and M. T. C. Andrade, "Technique to Construct Grey-Scale Morphological Operators Using Fuzzy Expert System," *Electronics Letters*, vol. 33, no. 22, pp. 1859-1861, 1997.

[Ri01] H. Y. Kim, "Quick Construction of Efficient Morphological Operators by Computational Learning," *Electronics Letters*, vol. 33, no. 4, pp. 286-287, 13th Feb. 1997.

Trabalhos em congressos internacionais

[Ci11] H. Y. Kim and R. L. Queiroz, "Inverse Halftoning by Decision Tree Learning," in Proc. *IEEE Int. Conf. on Image Processing*, (Barcelona, Spain), 2003.

[Ci10] C. R. P. Dionisio, H. Y. Kim, "A Supervised Shape Classification Technique Invariant Under Rotation and Scaling," in Proc. *Int. Telecommunications Symposium*, (Natal, Brasil), 2002.

[Ci09] H. I. A. Bustos, H. Y. Kim, "Color Image Edge Detection by Robust Anisotropic Diffusion," in Proc. *Int. Telecommunications Symposium*, (Natal, Brasil), 2002.

[Ci08] P. S. L. M. Barreto, H. Y. Kim, B. Lynnm M. Scott, "Efficient Algorithms for Pairing-Based Cryptosystems," Advances in Cryptology – CRYPTO'2002, *Lecture Notes in Computer Science*, vol. 2442, pp. 354–368, Springer-Verlag, 2002.

[Ci07] H. I. A. Bustos, H. Y. Kim, R. T. Lopes, "Image Reconstruction Using Divergent Beams Distributed over Limited Angle," in Proc. *IEEE Int. Conf. on Image Processing*, (Rochester, USA), 2002.

[Ci06] J. Nakahara Jr., P. S. L. M. Barreto, B. Preneel, J. Vanderwalle, H. Y. Kim, "Square Attacks on Reduced-Round PES and IDEA Block Ciphers," in Proc. 23rd *IEEE Symp. Inf. Theory in the BENELUX*, (Louvain-la-Neuve), pp. 187-195, 2002.

[Ci05] H. Y. Kim, "Fast and Accurate Binary Halftone Image Resolution Increasing by Decision-Tree Learning," in Proc. *IEEE Int. Conf. on Image Proc.* (Thessaloniki, Greece), vol. 2, pp. 1093-1096, 2001.

[Ci04] P. S. L. M. Barreto, H. Y. Kim and V. Rijmen, "Toward a Secure Public-Key Blockwise Fragile Authentication Watermarking," in Proc. *IEEE Int. Conf. on Image Proc.* (Thessaloniki, Greece), vol. 2, pp. 494-497, 2001.

[Ci03] P. S. L. M. Barreto, V. Rijmen, J. Nakahara Jr., B. Preneel, J. Vandewalle, H. Y. Kim, "Improved Square Attacks Against Reduced-Round Hierocrypt," Fast Software Encryption Workshop (Yokohama, Japan), *Lecture Notes in Computer Science*, vol. 2355, pp. 165-173, 2001.

[Ci02] H. Y. Kim and Paulo S. L. M. Barreto, "Fast Binary Image Resolution Increasing by k-Nearest Neighbor Learning," in Proc. *IEEE Int. Conf. on Image Proc.* (Vancouver, Canada), vol. 2, TA9.06, pp. 327-330, 2000.

[Ci01] H. Y. Kim and F. A. M. Cipparrone, "Automatic Design of Nonlinear Filters by Nearest Neighbor Learning," in Proc. *IEEE Int. Conf. on Image Proc.* (Chicago, USA), vol. 2, TP7.05, pp. 737-741, 1998.

Revistas brasileiras:

[Rn01] H. Y. Kim, "Marcas d'Água Frágeis de Autenticação para Imagens em Tonalidade Contínua e Esteganografia para Imagens Binárias e Meio-tom," *Revista de Informática Teórica e Aplicada*, Instituto de Informática da UFRGS, vol. 10, no. 1, pp. 97-125, 2003 (artigo sobre curso tutorial ministrado em Sibgrapi'2003).

Trabalhos em congressos nacionais

[Cn14] H. Y. Kim and A. Afif, "Secure Authentication Watermarking for Binary Images," in Proc. *Sibgrapi - Brazilian Symp. on Comp. Graph. and Image Proc.*, pp. 199-206, 2003.

[Cn13] H. I. A. Bustos and H. Y. Kim "Reconstrução-Difusão: Um Algoritmo de Reconstrução MENT Melhorado Baseado em Difusão Anisotrópica Robusta," in Proc. *8º Congresso Brasileiro de Física Médica*, Porto Alegre, Brasil, 2003.

[Cn12] H. Y. Kim and Z. H. Cho, "Robust Anisotropic Diffusion to Produce Clear Statistical Parametric Map from Noisy fMRI," in Proc. *Sibgrapi - Brazilian Symp. on Comp. Graph. and Image Proc.*, pp. 11-17, 2002.

[Cn11] H. I. A. Bustos, H. Y. Kim and R. T. Lopes, "Método de Reconstrução de Imagem Usando Feixe Divergente Distribuído em ângulo Limitado," in Proc. *Simpósio Brasileiro de Telecomunicações*, 2001.

[Cn10] H. Y. Kim, "Complexidade de Amostra para Projetar Operadores para Imagens Binárias pela Aprendizagem de Máquina," in Proc. *V Congresso Brasileiro de Redes Neurais* (Rio de Janeiro), pp. 121-126, 2001.

[Cn09] H. I. A. Bustos, H. Y. Kim and R. T. Lopes, "New Image Reconstruction Method from Fan-Beam Projections in Limited Angle," in Proc. IV *Workshop SIBRATI (Sistema Brasileiro de Tecnologia de Informação)*, 2001.
http://www.lsi.usp.br/~dmi/workshop2001/

[Cn08] H. Y. Kim, "Filtros Nebulosos no Espaço de Escala," in Proc. *Simpósio Brasileiro de Telecomunicações*, paper 4140035, 2000.

[Cn07] P. S. L. M. Barreto, H. Y. Kim and V. Rijmen, "Um Modo de Operação de Funções de Hashing para Localizar Alterações em Dados Digitalmente Assinados," in Proc. *Simpósio Brasileiro de Telecomunicações*, paper 5150124, 2000.

[Cn06] H. Y. Kim, "Segmentation-Free Printed Character Recognition by Relaxed Nearest Neighbor Learning of Windowed Operator," in Proc. *Sibgrapi - Brazilian Symp. on Comp. Graph. and Image Proc.*, pp. 195-204, 1999.

[Cn05] P. S. L. M. Barreto and H. Y. Kim, "Pitfalls in Public Key Watermarking," in Proc. *Sibgrapi - Brazilian Symp. on Comp. Graph. and Image Proc.*, pp. 241–242, 1999.

[Cn04] H. Y. Kim, "Quick Construction of Efficient Morphological Operators by Computational Learning", apresentado em *Brazilian Workshop 1997 on Mathematical Morphology*, DPI-INPE, S. José dos Campos, 1997.
http://www.dpi.inpe.br/~banon/URLib2/workshop/

[Cn03] H. Y. Kim, "Minkowski Operations for Boundary Represented Objects", apresentado em *Brazilian Workshop on Mathematical Morphology II*, IME-USP, 1996.
http://www.dpi.inpe.br/~banon/URLib2/workshop/

[Cn02] H. Y. Kim, "Rastreamento Bidirecional: Um Algoritmo para Gerar as Sombras dos Objetos Transparentes", anais do *XII Congresso da Sociedade Brasileira de Computação*, pp. 188-199, 1992.

[Cn01] H. Y. Kim, "Como Calcular a Probabilidade de Falha do Método de Perturbação", anais da *Quarta Semana de Informática da UFBA*, pp. 13-20, abril de 1992.

Artigos submetidos

[Su06] Marco A. A. de Melo and H. Y. Kim, "Filtragem de Sinais do Acelerômetro pela Difusão Anisotrópica," submitted to *Congresso Brasileiro de Automática*, Gramado, 2004.

[Su05] C. R. P. Dionisio and H. Y. Kim, "New Features for Affine-Invariant Shape Classification," submitted to *Int. Conf Image Processing*, Singapore, 2004.

[Su04] H. Y. Kim and R. L. Queiroz, "A Public-Key Authentication Watermarking For Binary Images," submitted to *Int. Conf Image Processing*, Singapore, 2004.

[Su03] H. I. A. Bustos and H. Y. Kim, "Reconstruction-Diffusion: An Improved MENT Reconstruction Algorithm Based On The Robust Anisotropic Diffusion," submitted to *Int. Conf Image Processing*, Singapore, 2004.

[Su02] H. Y. Kim and A. Afif, "Secure Authentication Watermarking for Binary Images," submitted to *Int. J. Imaging Systems and Technology*.

[Su01] H. Y. Kim and Z. H. Cho, "Robust Anisotropic Diffusion to Produce Clear Statistical Parametric Map from Noisy fMRI," submitted to *Computer Vision and Image Understanding*.

5.2 Referências da Literatura

Referências do capítulo 2: Projeto de operadores pela aprendizagem

[Anthony and Biggs, 1992] M. Anthony and N. Biggs, *Computational Learning Theory - An Introduction*, Cambridge University Press, 1992.

[Bentley, 1975] J. L Bentley, "Multidimensional Binary Search Trees Used for Associative Searching," *Comm. ACM*, vol. 18, no. 9, pp. 509-517, 1975.

[Cormen et al., 1990] T. H. Cormen, C. E. Leiserson and R. L. Rivest, *Introduction to Algorithms*, The MIT Press, 1990.

[Cover and Hart, 1967] T. M. Cover and P. E. Hart, "Nearest Neighbor Pattern Classification," *IEEE T. Information Theory*, vol. IT-13, no. 1, pp. 21-27, 1967.

[Coyle and Lin, 1988] E. J. Coyle and J. H. Lin, "Stack Filters and the Mean Absolute Error Criterion," *IEEE Trans. Ac. Speech Signal Proc.*, vol. 36, no. 8, Aug. 1988.

[Dougherty, 1992a] E. R. Dougherty, "Optimal Mean-Square N-Observation Digital Morphological Filters, Part I - Optimal Binary Filters," *CVGIP: Image Understanding*, vol. 55, no. 1, pp. 36-54, 1992.

[Dougherty, 1992b] E. R. Dougherty, "Optimal Mean-Square N-Observation Digital Morphological Filters, Part II - Optimal Gray-Scale Filters," *CVGIP: Image Understanding*, vol. 55, no. 1, pp. 55-72, 1992.

[Friedman et al., 1977] J. H. Friedman, J. L. Bentley and R. A. Finkel, "An Algorithm for Finding Best Matches in Logarithmic Expected Time," *ACM T. Mathematical Software*, vol. 3, no. 3, pp. 209-226, 1977.

[Gonzalez and Woods, 1992] R. C. Gonzalez and R. E. Woods, *Digital Imagem Processing*, Addison-Wesley Publishing Company, 1992.

[Haussler, 1992] D. Haussler, "Decision Theoretic Generalizations of the PAC Model for Neural Net and Other Learning Applications," *Information and Computation*, vol. 100, pp. 78-150, 1992.

[Jones and Svalbe, 1994] R. Jones and I. Svalbe, "Morphological Filtering as Template Matching," *IEEE T. Pattern Analysis Machine Intelligence*, vol. 16, no. 4, pp. 438-443, 1994

[Knuth, 1987] D. E. Knuth, "Digital Halftones by Dot Diffusion," *ACM T. on Graphics*, vol. 6, no. 4, pp. 245-273, 1987.

[Lee et al., 1997] W. L. Lee, K. C. Fan and Z. M Chen, "Design of optimal stack filter under MAE criterion," in *Proc. IEEE Int. Conf. Image Proc.* (Santa Barbara, USA), vol. 1, pp. 420-423, 1997.

[Loce and Dougherty, 1997] R. P. Loce and E. R. Dougherty, *Enhancement and Restoration of Digital Documents: Statistical Design of Nonlinear Algorithms*, SPIE Press, 1997.

[Loce et al., 1997] R. P. Loce, E. R. Dougherty, R. E. Jodoin and M. S. Cianciosi, "Logically Efficient Spatial Resolution Conversion Using Paired Increasing Operators," *Real-Time Imaging*, vol. 3, no. 1, pp. 7-16, 1997.

[Luo et al., 1998] J. Luo, R. Queiroz, and Z. Fan, "A Robust Technique for Image Descreening Based on the Wavelet Transform," *IEEE. T. Signal Processing*, vol. 46, no. 4, pp. 1179-1184, 1998.

[Mese and Vaidyanathan, 2001] M. Mese and P. P. Vaidyanathan, "Look-Up Table (LUT) Method for Inverse Halftoning," *IEEE T. Image Processing*, vol. 10, no. 10, pp. 1566-1578, October 2001.

[Mese and Vaidyanathan, 2002] M. Mese and P. P. Vaidynathan, "Tree-Structured Method for LUT Inverse Halftoning and for Image Halftoning," *IEEE T. Image Processing*, vol. 11, no. 6, pp. 644-655, June 2002.

[Mitchell, 1997] T. M. Mitchell, *Machine Learning*, WCB/McGraw-Hill, 1997.

[Preparata and Shamos, 1985] F. P. Preparata and M. I. Shamos, *Computational Geometry, an Introduction*, Springer-Verlag, 1985.

[Quinlan, 1986] J. R. Quinlan, "Induction of Decision Trees," *Machine Learning*, 1, pp. 81-106, 1986.

[Robert and Malandain, 1998] L. Robert and G. Malandain, "Fast Binary Image Processing Using Binary Decision Diagrams," *Computer Vision and Image Understanding*, vol. 72, no. 1, pp. 1-9, 1998.

[Roetling and Loce, 1994] P. Roetling and R. Loce, "Digital Halftoning", Chapter 10 in *Digital Image Processing Methods*, E. Dougherty Ed., Marcel Dekker, New York, NY, 1994.

[Ulichney, 1987] R. Ulichney, *Digital Halftoning*, The MIT Press, 1987.

[Vapnik, 1995] V. N. Vapnik, *The Nature of Statistical Learning Theory*, Springer-Verlag, 1995.

[Wong, 1995] P. W. Wong, "Inverse Halftoning and Kernel Estimation for Error Diffusion," *IEEE T. Image Processing*, vol. 4, no. 4, pp. 486-498, 1995.

Referências do capítulo 3: Difusão anisotrópica

[Analog, 2000] Analog Devices., "ADXL202E: Low-Cost 2g Dual Axis Accelerometers with Duty Cycle Output," Data Sheet, Analog Devices, pp. 1-12, 2000.

[Ardekani and Kanno, 1998] B. A. Ardekani and I. Kanno, "Statistical Methods for Detecting Activated Regions in Functional MRI of the Brain," *Magn. Reson. Imag.*, vol. 16, no. 10, pp. 1217-1225, 1998.

[Aurélio, 1999] Aurélio Buarque de Holanda Ferreira, *Dicionário Aurélio Eletrônico Século XXI*, 1999.

[Black et al., 1998] M. J. Black, G. Sapiro, D. H. Marimont, and D. Heeguer, "Robust Anisotropic Diffusion," *IEEE T. Image Processing*, vol. 7, no. 3, pp. 421-432, March 1998.

[Carlson, 1986] A. B. Carlson, *Communication Systems, Probability, Random Variables and Random Signal Principles*, McGraw-Hill, 1986.

[Chuang et al., 1999] K. H. Chuang, M. J. Chiu, C. C. Lin, and J. H. Chen, "Model-Free Functional MRI Analysis Using Kohonen Clustering Neural Network and Fuzzy c-Means," *IEEE Trans. Med. Imag.*, vol. 18, pp. 1117-1128, Dec. 1999.

[Delaney and Bresler, 1998] A. H. Delaney and Y. Bresler, "Globally Convergent Edge-Preserving Regularized Reconstruction: An Application to Limited-Angle Tomography," *IEEE T. Image Processing*, vol. 7, no. 2, pp. 204-221, February 1998.

[Dusassoy and Abdou, 1991] N. J. Dusassoy and I. E. Abdou, "The Extended MENT Algorithm: A Maximum Entropy Type Algorithm Using Prior Knowledge for Computerized Tomography," *IEEE T. Signal Processing*, vol. 39, no. 5, pp. 1164-1180, May 1991.

[Friston et al., 1994] K. J. Friston, P. Jezzard, and R. Turner, "The Analysis of Functional MRI Time-Series," *Human Brain Mapping*, vol. 1, pp. 153-171, 1994.

[Friston et al., 1995] K. J. Friston, A. P. Holmes, K. J. Worsley, J. P. Poline, C. D. Frith and R. S. J. Frackowiak, "Statistical Parametric Maps in Functional Imaging: A General Linear Approach," *Human Brain Mapping*, vol. 2, pp. 189-210, 1995.

[Friston, 1997] K. J. Friston (ed.), *SPM Course – Short Course Notes*, available at site http://www.fil.ion.ucl.ac.uk/spm/ course/notes97/, 1997.

[Gerig et al., 1992] G. Gerig, O. Kübler, R. Kikinis, and F. A Jolesz, "Nonlinear Anisotropic Filtering of MRI Data," *IEEE Trans. Med Imag.*, vol. 11, no. 2, pp. 221-232, June 1992.

[Gold et al., 1998] S. Gold, B. Christian, S. Arndt, G. Zeien, T. Cizadlo, D. L. Johnson, M. Flaum, and N. C. Andreasen, "Functional MRI Statistical Software Packages: A Comparative Analysis," *Human Brain Mapping*, vol. 6, pp. 73-84, 1998.

[Goutte et al., 1999] C. Goutte, P. Toft, E. Rostrup, F. A. Nielsen, and L. K. Hansen, "On Clustering fMRI Time Series," *NeuroImage*, vol. 9, no. 3, pp. 298-310, 1999.

[Jackway and Deriche, 1996] P.T. Jackway and M. Deriche, "Scale-Space Properties of the Multiscale Morphological Dilation-Erosion," *IEEE T. Pattern Analysis and Machine Intell.*, vol. 18, no. 1, pp. 38-51, 1996.

[Jain, 1989] A. K. Jain, *Fundamentals of Digital Image Processing*, Prentice Hall, 1989.

[Kershaw et al., 1999] J. Kershaw, B. A. Ardekani, and I. Kanno, "Application of Bayesian Inference to fMRI Data Analysis," *IEEE Trans. Med. Imag.*, vol. 18, pp. 1138-1153, Dec. 1999.

[Lange et al., 1999] N. Lange, S. C. Strother, J. R. Anderson, F. A. Nielsen, A. P. Holmes, T. Kolenda, R. Savoy, and L. K. Hansen, "Plurality and Resemblance in fMRI Data Analysis," *NeuroImage*, vol. 10, pp. 282-303, 1999.

[Lindeberg, 1994] T. Lindeberg, *Scale-Space Theory in Computer Vision*, Kluwer, 1994.

[Minerbo, 1979] G. Minerbo, "MENT: A Maximum Entropy Algorithm for Reconstructing a Source from Projection Data," *Comput. Graph. Image Processing*, vol. 71, pp. 48-68, 1979.

[Perona and Malik, 1987] P. Perona and J. Malik, "Scale Space and Edge Detection Using Anisotropic Diffusion," in Proc. *IEEE Comp. Soc. Workshop Computer Vision*, pp. 16-27, 1987.

[Perona and Malik, 1990] P. Perona and J. Malik, "Scale-Space and Edge Detection Using Anisotropic Diffusion," *IEEE. Trans. Patt. Anal. and Machine Intell.*, vol. 12, no. 7, pp 629-639, 1990.

[Reis and Roberty, 1992] M. L. Reis and N. C. Roberty, "Maximum-Entropy Algorithms for Image-Reconstruction from Projections," *Inverse Problems*, vol. 8, no. 4, pp. 623-644, 1992.

[Shih and Weinberg, 2001] Peter Shih and Harvey Weinberg, "A Useful Role for the ADXL202 Dual-Axis Accelerometer in Speedometer-Independent Car-Navigation Systems," Vol. 35, No. 4, Auguo.st-September, 2001.

[Solé et al., 2001] A. F. Solé, S. C. Ngan, G. Sapiro, X. P. Hu and A. López, "Anisotropic 2-D and 3-D Averaging of fMRI Signals," *IEEE Trans. Medical Imaging*, vol. 20, no. 2, pp. 86-93, Feb. 2001.

[Velho et al., 2000] L. Velho, R. Teira and J. Gones, *Introdução aos Espaços de Escala*, 12ª Escola de Computação, 2000.

[Witkin, 1983] A. P. Witkin, "Scale-Space Filtering," *Proc. 8th Int. Joint Conf. Art. Intelligence*, vol. 2, pp. 1019-1022, 1983.

Referências do capítulo 4: Marcas d'Água de Autenticação

[Baharav and Shaked, 1998] Z. Baharav and D. Shaked, "Watermarking of Dither Halftone Images", Hewlett-Packard Labs. Tech. Rep. HPL-98-32 (1998).

[Barreto, 2003] P. S. L. M. Barreto, *Criptografia Robusta e Marcas d'Água Frágeis: Construção e Análise de Algoritmos para Localizar Alterações em Imagens Digitais*, tese de doutorado, Escola Politécnica da Universidade de São Paulo, 2003.

[Chen et al., 2000] Y.-Y. Chen, H.-K. Pan and Y.-C. Tseng, "A Secure Data Hiding Scheme for Binary Images," *IEEE Symposium on Computers and Communications*, 2000, pp. 750-755.

[Deseilligny and Le Men, 1998] M. P. Deseilligny and H. Le Men, "An Algorithm for Digital Watermarking of Binary Images, Application to Map and Text Images," available at www-ima.enst.fr/~maitre/tatouage/MPdS_HK.ps, 1998.

[Friedman, 1993] G. L. Friedman, "The Trustworthy Digital Camera: Restoring Credibility to the Photographic Image," *IEEE T. Consumer Electronics*, vol. 39, pp. 905-910, Nov. 1993.

[Fu and Au, 2000] M. S. Fu and O. C. Au, "Data Hiding by Smart Pair Toggling for Halftone Images," *IEEE Int. Conf. Acoustics, Speech and Signal Processing*, vol. 4, pp. 2318-2321, 2000.

[Fu and Au, 2001] M. S. Fu and O. C. Au, "Data Hiding in Halftone Images by Stochastic Error Diffusion," *IEEE Int. Conf. Acoustics, Speech and Signal Processing*, May 2001.

[Fu and Au, 2002a] M. S. Fu and O. C. Au, "Data Hiding Watermarking for Halftone Images," *IEEE Trans. Image Processing*, vol. 11, no. 4, pp. 477- 484, 2002.

[Fu and Au, 2002b] M. S. Fu and O. C. Au, "A Robust Public Watermark for Halftone Images," *IEEE Int. Symp. Circuits and Systems*, vol. 3, pp. 639-642.

[Hel-Or, 2001] H. Z. Hel-Or, "Watermarking and Copyright Labeling of Printed Images," Journal of Electronic Imaging, col. 10, no. 3, pp. 794-803, 2001.

[Holliman and Memon, 2000] M. Holliman and N. Memon "Counterfeiting Attacks on Oblivious Block-wise Independent Invisible Watermarking Schemes," *IEEE Trans. Image Processing*, 2000, vol. 9. no. 3, pp. 432-441.

[Knox, 1998] K. T. Know, "Digital Watermarking Using Stochastic Screen Patterns," United States Patent Number 5,734,752, 1998.

[Knuth, 1987] D. E. Knuth, "Digital Halftones by Dot Diffusion," *ACM Trans. Graph.*, vol. 6, no. 4, Oct. 1987.

[Li et al., 2000] C. T. Li, D. C. Lou and T. H. Chen, "Image Authentication and Integrity Verification via Content-Based Watermarks and a Public Key Cryptosystem," *IEEE Int. Conf. Image Processing*, 2000, vol. 3, pp. 694-697.

[Maxemchuk and Low, 1997] N. F. Maxemchuk and S. Low, "Marking Text Documents," *Int. Conf. Image Processing*, vol. 3, pp. 13-17, 1997.

[Menezes et al., 1997] A. J. Menezes, P. C. Van Oorschot and S. A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1997.

[Nishimura and Sibuya, 1990] K. Nishimura and M. Sibuya, "Probability to Meet in the Middle," *J. Cryptology*, vol. 2, no. 1, pp. 13-22, 1990.

[Pei and Guo, 2003] S. C. Pei and J. M. Guo, "Hybrid Pixel-Based Data Hiding and Block-Based Watermarking for Error-Diffused Halftone Images," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 13, no. 8, pp. 867-884, 2003.

[Rivest et al., 1978] R. L. Rivest, A. Shamir and L. M. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," *Communications of the ACM*, vol. 21, pp. 120-126, 1978.

[Schneier, 1996] B. Schneier, *Applied Cryptography*, second edition, John Wiley & Sons, 1996.

[Tseng et al., 2002] Y.-C. Tseng, Y.-Y. Chen and H.-K. Pan, "A Secure Data Hiding Scheme for Binary Images," *IEEE Trans. on Communications*, Vol. 50, No. 8, Aug. 2002, pp. 1227-31.

[Ulichney, 1987] R. Ulichney, *Digital Halftoning*, The MIT Press, 1987.

[Wang, 1998] S. G. Wang, "Digital Watermarking Using Conjugate Halftone Screens," United States Patent Number 5,790,703, 1998.

[Wong, 1997] P. W. Wong, "A Watermark for Image Integrity and Ownership Verification," *IS&T PIC Conference*, (Portland, OR), May 1998 (also available as *Hewlett-Packard Labs. Tech. Rep.* HPL-97-72, May 1997).

[Wong, 1998] P. W. Wong, "A Public Key Watermark for Image Verification and Authentication," *IEEE Int. Conf. Image Processing*, 1998, vol. 1, pp. 455-459, (MA11.07).

[Wong and Memon, 2001] P. W. Wong and N. Memon, "Secret and Public Key Image Watermarking Schemes for Image Authentication and Ownership Verification," *IEEE Trans. Image Processing*, vol. 10, no. 10, pp. 1593-1601, 2001.

[Wu et al., 2000] M. Wu, E. Tang and B. Liu, "Data Hiding in Digital Binary Image," *IEEE Int. Conf. Multimedia and Expo*, ICME'00, New York, USA, 2000.

[Yeung and Mintzer, 1997] M. M. Yeung and F. Mintzer, "An Invisible Watermarking Technique for Image Verification," *IEEE Int. Conf. Image Processing*, 1997, vol. 1, pp. 680-683.

[Zhao and Koch, 1995] J. Zhao and E. Koch, "Embedding Robust Labels into Images for Copyright Protection," *Proc. Int. Cong. Intellectual Property Rights, Knowledge and New Technologies*, 1995, pp. 242-251.