

**UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA ELÉTRICA E COMPUTAÇÃO
COMISSÃO DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA**

**Melhorias no Reconhecimento
de Impressões Digitais
Baseado no Método FingerCode**

Autor: Gustavo Ferreira Cardoso de Sá
Orientador: Roberto de Alencar Lotufo

**UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA ELÉTRICA E COMPUTAÇÃO
COMISSÃO DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA
DEPARTAMENTO DE COMPUTAÇÃO E AUTOMAÇÃO**

Melhorias no Reconhecimento de Impressões Digitais Baseado no Método FingerCode

Autor: Gustavo Ferreira Cardoso de Sá
Orientador: Roberto de Alencar Lotufo

Curso: Engenharia Elétrica
Área de Concentração: Engenharia de Computação

Dissertação de mestrado acadêmico apresentada à comissão de Pós Graduação da Faculdade de Engenharia Elétrica e de Computação, como requisito para a obtenção do título de Mestre em Engenharia Elétrica.

Campinas, SP – Brasil

2006

FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DA ÁREA DE ENGENHARIA E ARQUITETURA - BAE - UNICAMP

Sa 11m Sá, Gustavo Ferreira Cardoso de
Melhorias no reconhecimento de impressões digitais
baseado no método FingerCode / Gustavo Ferreira Cardoso
de Sá. --Campinas, SP: [s.n.], 2006.

Orientador: Roberto de Alencar Lotudo
Dissertação (Mestrado) - Universidade Estadual de
Campinas, Faculdade de Engenharia Elétrica e de
Computação.

1. Datiloscopia. 2. Filtros elétricos digitais. 3.
Morfologia matemática. I. Lotufo, Roberto de Alencar. II.
Universidade Estadual de Campinas. Faculdade de
Engenharia Elétrica e de Computação. III. Título.

Título em Inglês: Improvements in fingerprint recognition based on the FingerCode
method

Palavras-chave em Inglês: Fingerprint, FingerCode, Gabor filter, Segmentation,
Open Top-Hat

Área de concentração: Engenharia de Computação

Titulação: Mestre em Engenharia Elétrica

Banca examinadora: Fernando José von Zuben e Pascual Jovino Figueroa Rivero

Data da defesa: 29/06/2006

**UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA ELÉTRICA E COMPUTAÇÃO
COMISSÃO DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA
DEPARTAMENTO DE COMPUTAÇÃO E AUTOMAÇÃO**

DISSERTAÇÃO DE MESTRADO

Melhorias no Reconhecimento de Impressões Digitais Baseado no Método FingerCode

Autor: Gustavo Ferreira Cardoso de Sá

Orientador: Roberto de Alencar Lotufo

A Banca Examinadora composta pelos membros abaixo aprovou esta Dissertação:

Prof. Dr. Roberto de Alencar Lotufo
Universidade Estadual de Campinas (UNICAMP)

Prof. Dr. Fernando José von Zuben
Universidade Estadual de Campinas (UNICAMP)

Prof. Dr. Pascual Jovino Figueroa Rivero
Griaule Tecnologia

Campinas, 29 de junho de 2006

Agradecimentos

Este trabalho não poderia ser terminado sem a ajuda de diversas pessoas às quais presto minha homenagem:

Aos meus pais pelo incentivo em todos os momentos da minha vida.

À minha esposa Joyce pelo apoio.

Ao meu orientador prof. Roberto de Alencar Lotufo pela valiosa orientação.

À Griaule pelos dados fornecidos.

Aos colegas do DCA e do DEB que ajudaram de forma direta e indireta na conclusão deste trabalho.

À CAPES pelo apoio financeiro.

Resumo

SÁ, Gustavo Ferreira Cardoso de, *Melhorias no Reconhecimento de Impressões Digitais Baseado no Método FingerCode*, Campinas: Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, 2006. 96 p. Dissertação (Mestrado).

Neste trabalho são apresentadas melhorias na robustez do método FingerCode para reconhecimento de impressões digitais. No FingerCode a textura dos componentes orientados das impressões digitais são extraídas por um banco direcional de filtros Gabor. Posteriormente, os componentes orientados são setorizados e para cada setor é computado um valor. Este conjunto de valores forma o vetor de atributos. Finalmente, a média da diferença absoluta dos dois vetores de atributos é computada indicando a similaridade entre duas impressões digitais. Foram testadas várias soluções e entre as que apresentaram melhores resultados destacam-se: a substituição dos valores dos atributos através de uma função não-linear, a ponderação dos valores de atributo de acordo com características estatísticas da distribuição espacial dos valores e o cálculo de medidas estatísticas extraídas dos histogramas de distribuição de diferenças. Estas funções apresentaram um ganho significativo, principalmente para o caso dos sensores óticos com uma melhoria de aproximadamente 45% no EER. Outra contribuição apresentada foi uma nova implementação rápida do filtro Gabor 2D, que se constitui de uma onda sinusoidal modulada por um envelope gaussiano. A filtragem 2D da imagem por um banco de filtros Gabor 2D é uma das etapas de maior consumo de tempo no processamento de imagens. Na nova solução proposta, o filtro Gabor 2D é separado em dois filtros Gabor 1D ortogonais, bastando para isto que o envelope gaussiano obedeça a condição de ser circular. O processamento com o filtro separado é mais rápida do que com o filtro não-separado e o ganho na performance aumenta à medida que aumenta o tamanho da imagem ou do filtro. Também foram desenvolvidas novas técnicas de segmentação: baseada em morfologia matemática e baseada em filtros Gabor. Estas segmentações ocorrem ao nível do píxel, com ótimos resultados, principalmente após a uniformização da área através de processos morfológicos.

Palavras Chaves

Impressão Digital, FingerCode, Filtro Gabor, Segmentação, Abertura Top-Hat.

Abstract

SÁ, Gustavo Ferreira Cardoso de, *Melhorias no Reconhecimento de Impressões Digitais Baseado no Método FingerCode*, Campinas: Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, 2006. 96 p. Dissertação (Mestrado).

In this work it is introduced improvements in robustness of FingerCode method to recognize fingerprints. In the FingerCode the texture of fingerprint oriented components are extracted by a bank of directional Gabor filters. After that, the oriented components are tessellated and a value is computed for each sector. This set of values constitutes the attribute vector. Finally, the absolute difference mean between the two attribute vectors is computed that gives the similarity between two fingerprints. New solutions were tested; among them the best results were obtained by: attribute values replacement by a non-linear function, attribute values weighting by statistical characteristics of spatial distribution of values, and the calculus of statistical measures extracted from the difference distribution histograms. These functions presented a significant gain, mainly in the case of optical sensors with an improvement about 45% in EER. Another contribution presented was a new fast implementation of the 2D Gabor filter, which constitutes in a sinusoidal wave modulated by a Gaussian envelope. The 2D image filtering by a bank of 2D Gabor filters is one of the most expensive stage of image processing. In the new solution proposed, the 2D Gabor filter is separated in two orthogonal 1D Gabor filters, for this the Gaussian envelope must obey the condition of being circular. Processing with the separated filter is faster than the non-separated filter, and the gain improves as the size of image or filter increases. Also it was developed new segmentation techniques: based on mathematical morphology, and based on Gabor filters. Those segmentations occur at pixel level, with good results, mostly after the area regularization with morphological processes.

Keywords

Fingerprint, FingerCode, Gabor Filter, Segmentation, Open Top-Hat.

Índice

Lista de Figuras	x
Lista de Tabelas.....	xi
Lista de Artigos	xii
1 Introdução.....	1
1.1 Organização da Tese	2
1.2 Biometria.....	2
1.2.1 Comparação das Biometrias.....	3
1.3 Sistemas de Reconhecimento Biométrico	4
1.3.1 Métodos de Avaliação de Sistemas Biométricos	5
1.4 Impressão Digital	8
1.4.1 Propriedades	8
1.4.2 História	9
1.4.3 Características	11
1.5 Reconhecimento	12
1.5.1 Reconhecimento Baseado em Minúcias.....	13
1.5.2 Reconhecimento Baseado em Texturas (FingerCode).....	16
1.5.3 Método FingerCode Modificado.....	19
1.6 Base de Dados	23
2 Segmentação.....	24
2.1 Segmentação Morfológica.....	24
2.2 Segmentação Baseada em Filtro Gabor	26
2.3 Comparação dos Métodos de Segmentação	27
2.4 Segmentação de Impressões Digitais Baseada em Abertura Top-Hat	31
3 Filtro Gabor	41

3.1	Filtro Gabor 2D Separável com Envelope Gaussiano Circular.....	41
4	Novas Funções de Escore.....	51
4.1	Função Escore do FingerCode Melhorado.....	51
4.2	Histograma de Distribuição do FingerCode.....	65
4.3	Métodos Combinados.....	77
5	Conclusões	79
	Bibliografia.....	83

Lista de Figuras

Figura 1.1. Curvas de Comparação.	7
Figura 1.2. Estrutura da impressão digital.....	9
Figura 1.3. Singularidades e Minúcias.	12
Figura 1.4. Etapas de um sistema de reconhecimento baseado em minúcias.	13
Figura 1.5. Máscara de Sobel 3x3.	14
Figura 1.6. Estágios do método FingerCode.	16
Figura 1.7. Vizinhança do ponto (i, j)	17
Figura 1.8. Setorização da imagem.	20
Figura 1.9. Máscara da imagem.	20
Figura 1.10. Componentes orientados.....	21
Figura 1.11. Comparação das texturas.	22
Figura 2.1. Etapas do processo de segmentação morfológica.....	25
Figura 2.2. Etapas do processo de segmentação Gabor.	27
Figura 2.3. Direções do ponto p	29
Figura 2.4. Curvas ROC utilizando diferentes métodos de segmentação.	30
Figura 4.1. Fusão de classificadores.....	77
Figura 4.2. Curvas ROC do método do voto majoritário.	78

Lista de Tabelas

Tabela 1.1. Comparação das biometrias.....	4
Tabela 1.2. Possíveis respostas do sistema de reconhecimento	6
Tabela 1.3. Base de dados da FVC2002.....	23
Tabela 4. Equal Error Rate (EER) para os diferentes métodos de segmentação	30

Lista de Artigos

Artigo 1. Segmentação de Impressões Digitais Baseada em Abertura Top-Hat.....	32
Artigo 2. Separable 2D Gabor Filter with a Circular Gaussian Envelope	42
Artigo 3. Improved FingerCode Matching Function (Aceito SIBGRAPI2006).....	53
Artigo 4. FingerCode Matching Histogram	66

Capítulo 1

1 Introdução

Devido à crescente necessidade de identificação pessoal, o uso de sistemas biométrico automáticos está se tornando cada vez mais popular. Entre as biometrias utilizadas a de maior abrangência é a impressão digital. Além da sua relevância em aplicações forenses, onde é usada há mais de um século, o seu uso em aplicações comerciais e pessoais está se tornando comum. Apesar da tecnologia de reconhecimento de impressões digitais ser bem madura, os requisitos por sistemas mais robustos é uma necessidade crescente, principalmente nos sistemas AFIS que possuem milhares ou mesmo milhões de amostras.

A maioria dos métodos de reconhecimento de impressões digitais é baseada em minúcias que possuem características altamente discriminantes no nível local. No entanto os métodos exclusivamente locais, baseados apenas nas minúcias, deixam de aproveitar a rica informação encontrada no nível global. Métodos correlacionais, baseados apenas em características globais [1] e [8], são poucos utilizados no reconhecimento. A principal aplicação destes métodos é na classificação de impressões digitais. Um modo de superar as deficiências de cada um dos métodos é o uso de métodos híbridos, que usam tanto informações locais como globais [48].

Por serem pouco pesquisados os métodos baseados em textura não tiveram todo o seu potencial explorado. Devido a isso, o principal objetivo deste estudo foi obter melhorias no método FingerCode. As melhorias visaram: diminuir a velocidade de processamento e aumentar a robustez do reconhecimento.

1.1 Organização da Tese

No Capítulo 1 é feita uma introdução às biometrias e em particular à impressão digital. São descritos os sistemas de reconhecimento de impressões digitais baseados em características locais ou globais. Também são mostradas as principais ferramentas para a comparação de sistemas biométricos. No Capítulo 2 novos métodos de segmentação da impressão digital são mostrados: um baseado em morfologia matemática e outro em filtro Gabor. Um bom processo de segmentação evita a extração de características de áreas não pertencentes à impressão digital. No Capítulo 3 é desenvolvida uma implementação rápida para o filtro Gabor 2D através da separação em dois filtros 1D ortogonais. No Capítulo 4 são apresentadas novas funções para o cálculo do score no método FingerCode, usando as diversas características do vetor de atributos. Finalmente, no Capítulo 5 são apresentadas as conclusões e a análise final das contribuições.

1.2 Biometria

Biometria (do grego antigo: *bios* = “vida” e *metron* = “medida”) é o estudo de métodos de reconhecimento humano baseado em uma ou mais características físicas ou comportamentais do indivíduo [107]. Todas as biometrias possuem tanto características físicas quanto comportamentais em algum grau, algumas são predominantemente físicas tais como impressões digitais, íris, retina e mão, outras são predominantemente comportamentais, incluindo assinatura e padrão de digitação, e outras possuem um equilíbrio entre as características, como a voz.

O reconhecimento biométrico é feito pelas pessoas no seu dia a dia, em um encontro ao reconhecer o outro indivíduo pela face ou em uma conversa ao telefone, em que o reconhecimento é feito pela voz. Este reconhecimento é baseado apenas na memória do indivíduo. No entanto, com o passar do tempo a necessidade de fazer um reconhecimento mais confiável dos indivíduos foi crescendo e outros métodos foram surgindo. Uma das primeiras biometrias a ser utilizada nos países de cultura ocidental foi a assinatura. Na China, no século XIV, comerciantes imprimiam as impressões palmares e plantares das crianças para distingui-las [107], este sistema ainda é utilizado nos dias atuais.

Na ciência forense a identificação era feita usando apenas o sistema de “memória fotográfica”, no qual policiais de memória excepcional faziam o reconhecimento dos criminosos. Mesmo com o surgimento da fotografia o reconhecimento ainda era muito difícil. Em 1883 um antropologista da polícia francesa, Alphonse Bertillion, desenvolveu o sistema antropométrico (conhecido também como Bertillionage). O sistema utilizava certas medidas do corpo, tais como o comprimento do dedo médio e do pé, fotos de frente e de perfil e registros de marcas individuais, tais como tatuagens e cicatrizes [107]. Este foi o primeiro sistema científico utilizado para identificar criminosos e fez surgir o campo de estudo da biometria. Este sistema foi largamente utilizado no Ocidente até ser rapidamente obscurecido por um sistema mais preciso e prático, a impressão digital.

Nos últimos anos, os métodos de identificação deixaram de ser baseados em apenas uma biometria (impressão digital), para utilizarem várias outras características biométricas do indivíduo. Com o melhoramento e barateamento da tecnologia de reconhecimento biométrico o uso de biometrias cresceu. Apesar de nenhuma outra biometria ter alcançado o nível de aceitação da impressão digital, algumas estão sendo utilizadas não apenas na área legal como também na comercial.

1.2.1 Comparação das Biometrias

Para uma biometria ser útil em situações práticas, ela deve ter precisão no reconhecimento e velocidade com o uso de equipamentos de boa performance e baixo custo, ser aceita pela população atendida e ser suficientemente robusta contra fraudes. Na Tabela 1.1 as propriedades de vários tipos de biometrias são comparadas [101] e [59]:

- Universal: indica se a biometria é encontrada em todos os indivíduos;
- Distinção: indica o quanto uma biometria distingue um indivíduo do outro;
- Permanência: indica se a biometria mantém-se inalterada no envelhecimento;
- Coleta: indica a facilidade de adquirir os atributos da biometria para comparação;
- Desempenho: indica a precisão, a velocidade e a robustez do sistema;
- Aceitação: indica o grau de aprovação do uso da biometria em seu cotidiano;
- Fraude: indica a dificuldade de se enganar o sistema.

As propriedades estão categorizadas como: alta (A), média (M) ou baixa (B). Uma classificação alta indica uma performance boa da biometria no critério avaliado, enquanto que uma classificação baixa indica um performance ruim.

Tabela 1.1. Comparação das biometrias

Identificador Biométrico	Univer sal	Distin ção	Perma nência	Coleta	Desem penho	Aceita ção	Fraude
Assinatura	B	B	B	A	B	A	B
Caminhada	M	B	B	A	B	A	M
Dinâmica de Digitação	B	B	B	M	B	M	M
DNA	A	A	A	B	A	B	B
Face	A	B	M	A	B	A	B
Geometria da Mão	M	M	M	A	M	M	M
Impressão Digital	M	A	A	M	A	M	A
Íris	A	A	A	M	A	B	A
Odor	A	A	A	B	B	M	A
Orelha	M	M	A	M	M	A	M
Retina	A	A	M	B	A	B	A
Termograma Facial	A	A	B	A	M	A	A
Veias da Mão	M	M	M	M	M	M	A
Voz	M	B	B	M	B	A	B

1.3 Sistemas de Reconhecimento Biométrico

Na tecnologia de informação, sistemas de reconhecimento biométrico se referem às tecnologias utilizadas para medir e analisar as biometrias de modo a reconhecer o indivíduo. Dependendo do contexto os sistemas biométricos podem ser divididos em [59]:

- **Sistemas de autenticação:** verifica a identidade do indivíduo comparando a amostra biométrica recolhida no momento da autenticação com a amostra na base de dados previamente recolhida.
- **Sistemas de identificação:** reconhece a identidade de um indivíduo comparando sua característica biométrica com todas as amostras da base de dados.

Os sistemas de identificação operam com grandes quantidades de dados biométricos, as biometrias muitas vezes são adquiridas em condições adversas e sendo utilizadas principalmente em aplicações forenses. Os sistemas de autenticação operam com pequena quantidade de dados biométricos, a aquisição ocorre em ambientes controlados e são utilizadas principalmente em aplicações comerciais. Os sistemas biométricos podem ainda ter as seguintes propriedades:

- Cooperativo ou não cooperativo: o indivíduo tem a intenção de facilitar a identificação;
- Claro ou velado: o indivíduo tem consciência de estar sendo submetido ao reconhecimento;
- Habitual ou não habitual: o quanto o indivíduo é submetido ao reconhecimento
- Auxiliado ou não auxiliado: o sistema de aquisição ou reconhecimento da biometria é observado, guiado ou supervisionado por um humano;
- Ambiente padrão ou não padrão: o sistema opera em um ambiente controlado, tais como: iluminação, temperatura, etc.
- Público ou privado: os usuários do sistema são clientes (acesso a conta corrente em um banco) ou empregados (acesso ao servidor de dados da empresa) de uma companhia ou governo;
- Aberto ou fechado: a base dos dados biométricos é usada em apenas uma ou várias aplicações;
- Positivo ou negativo: o sistema estabelece se o indivíduo é quem ele afirma ou nega ser.

1.3.1 Métodos de Avaliação de Sistemas Biométricos

Em um sistema biométrico típico duas amostras da biometria são comparadas: a amostra de referência, previamente armazenada na base de dados e a amostra de entrada, recolhida no momento do reconhecimento. O sistema gera um valor discriminatório (score) resultante da comparação, sem perda de generalidade, variando de [0 1]. Quanto mais próximo de 0 for o score, mais semelhantes são as amostras, indicando que devem pertencer ao mesmo indivíduo. Deste modo, arbitrando certo valor de limiar obtém-se:

- $\text{Score} < \text{Limiar}$: Identificação positiva, as amostras pertencem ao mesmo indivíduo;
- $\text{Score} > \text{Limiar}$: Identificação negativa, as amostras pertencem a indivíduos diferentes.

A Tabela 1.2 mostra as possíveis situações que ocorrem dadas as respostas do sistema de reconhecimento biométricos.

Tabela 1.2. Possíveis respostas do sistema de reconhecimento

	As duas amostras biométricas pertencem ao mesmo indivíduo?	
	Resposta do Sistema	Resposta Esperada
Positivo	SIM	SIM
Falso Positivo	SIM	NÃO
Negativo	NÃO	NÃO
Falso Negativo	NÃO	SIM

A partir das respostas do sistema são computadas as taxas de erros:

- FAR (False Acceptance Rate – Taxa de Falsa Aceitação): é a porcentagem de erro do sistema ao responder que as amostras pertencem ao mesmo indivíduo, ou seja, o número casos de falsos positivos entre todos os casos positivos (positivos e falsos positivos);
- FRR (False Rejection Rate – Taxa de Falsa Rejeição): é a porcentagem de erro no sistema ao responder que as amostras são de indivíduos diferentes, ou seja, casos de falsos negativos entre todos os casos negativos (negativos e falsos negativos).

A Figura 1.1 mostra a distribuição dos escores resultantes das comparações das amostras, as taxas de erros para os diferentes escores e o desempenho do sistema nos diferentes limiares (curva ROC – Receiver Operating Characteristic). A Figura 1.1.a mostra a distribuição dos escores nos casos de comparação das amostras biométricas do mesmo indivíduo e de indivíduos diferentes. O trecho de interseção entre as duas curvas é a área de conflito em que não é possível determinar, dado o escore, se a comparação é entre amostras do mesmo indivíduo ou não. A Figura 1.1.b é a curva de distribuição acumulada dos escores, a FAR e a FRR são funções do limiar escolhido. A FAR é a distribuição acumulada das comparações de indivíduos diferentes, computada da esquerda para a direita. A FRR é a distribuição acumulada das comparações do mesmo indivíduo, computada da direita para a esquerda. Ao aumentar o limiar o sistema se torna mais permissivo diminuindo o número de falsas rejeições e aumentando o número de falsas aceitações e ao diminuir o limiar o inverso acontece, o sistema se torna mais restritivo. A curva

ROC é construída a partir das curvas de FAR e FRR. Para cada ponto do limiar é obtido um valor de FAR e de FRR que resulta em um ponto na curva ROC. A curva ROC é desenhada em escala logxlog para facilitar a visualização, pois nesta escala a região que realmente possui interesse (entre FAR0 e a FRR0) é destacada.

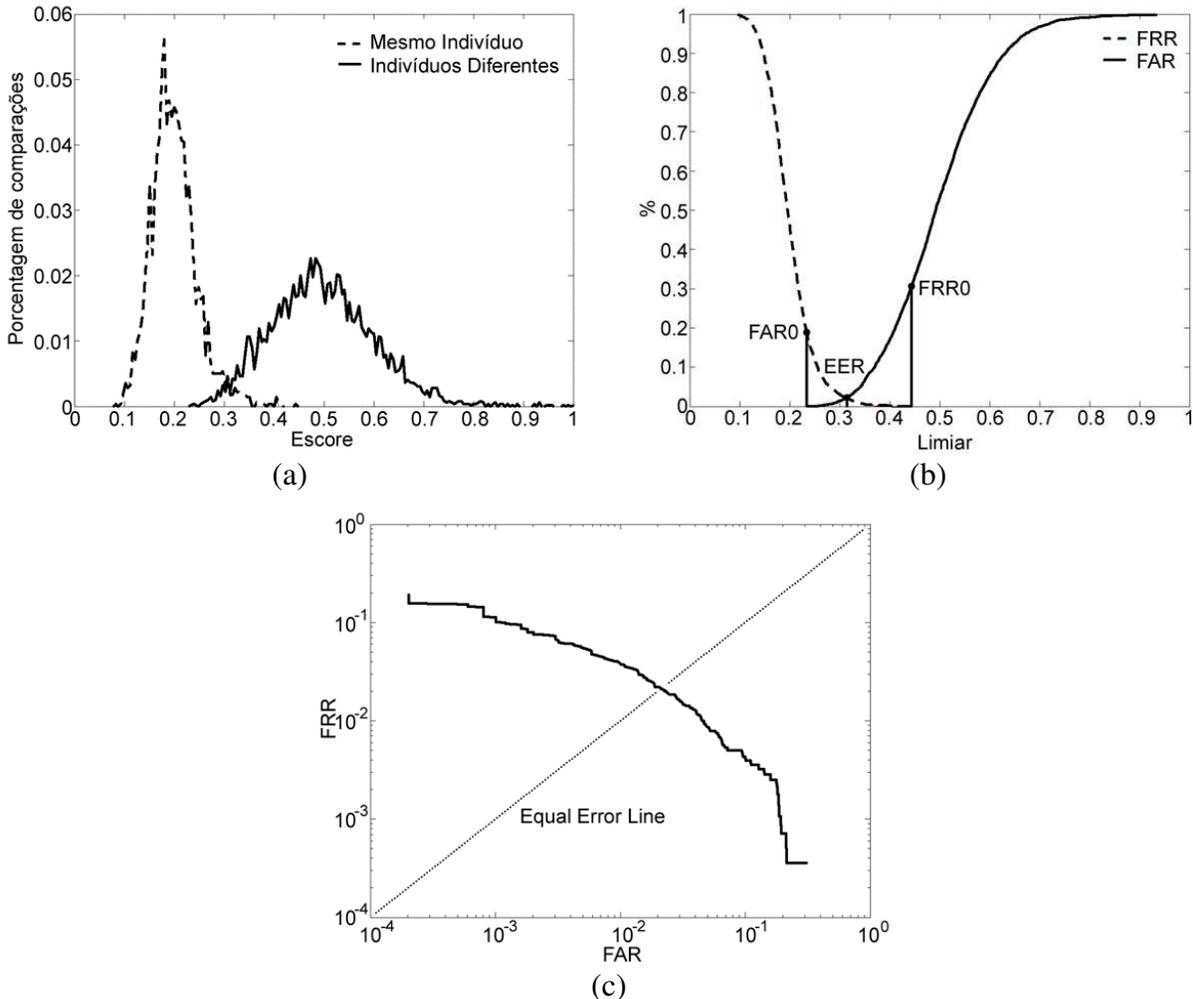


Figura 1.1. Curvas de Comparação: (a) Distribuição do escore das comparações entre amostras do mesmo indivíduo e amostras de indivíduos diferentes; (b) Curvas das taxas de erro (FAR e FRR), distribuição acumulada dos escores; (c) Curva ROC.

A robustez do sistema pode ser medida pela análise das curvas da Figura 1.1. Na Figura 1.1.a observa-se que os erros do sistema ocorrem na faixa de escores em que existem tanto comparações do mesmo indivíduo quanto de indivíduos diferentes. Quanto menor o número de casos que ocorrerem nesta faixa melhor é o sistema. O mesmo ocorre na Figura 1.1.b em que

quanto menor a área formada pelas curvas FAR e FRR entre os pontos FAR0 e o FRR0 mais discriminante é o sistema. No entanto, a melhor maneira de se comparar os sistemas biométricos é através da curva ROC (Figura 1.1.c), quanto menores os valores na curva mais robusto é o sistema. Na avaliação da robustez dos sistemas usam-se, também, alguns pontos de interesse:

- EER (Equal Error Rate): ponto em que as taxas de falso positivos e falso negativos são iguais (FAR = FRR);
- FAR0: ponto em que a taxa de falso positivo torna-se igual a zero;
- FRR0: ponto em que a taxa de falso negativo torna-se igual a zero.

Pode ocorrer de um sistema ser melhor que outro em determinada faixa de valores e pior em outra faixa de valores. Neste caso a escolha do sistema dependerá do ponto de trabalho desejado. O ponto de trabalho (limiar) de um sistema irá depender da aplicação em que será empregado. Em uma aplicação forense em que o objetivo é descobrir um suspeito a taxa de falsa rejeição deve ser o mais baixa possível para evitar que o sistema descarte um suspeito, mesmo que com isto outros indivíduos sejam encontrados. Nestes casos, os erros são eliminados posteriormente, manualmente. No acesso em uma área de alta segurança o objetivo é evitar o máximo que indivíduos não autorizados tenham acesso, mesmo que para isto os indivíduos autorizados possam sofrer dificuldades para serem reconhecidos. Neste caso, trabalha-se com uma baixíssima taxa de falsa aceitação. Para sistemas comerciais de modo geral utiliza-se um ponto intermediário próximo ao EER, de modo a evitar muitas fraudes e/ou muito transtorno aos usuários.

1.4 Impressão Digital

As impressões digitais oferecem um meio confiável de identificação, sendo usadas há mais de um século na ciência forense. Comparada às outras biometrias, as impressões digitais possuem um bom balanço das propriedades (Tabela 1.1). Devido a isto a impressão digital ainda é a biometria mais utilizada, tanto em aplicações legais como comerciais.

1.4.1 Propriedades

Impressão digital é o padrão de cristas e vales encontrados na superfície dos dedos. Praticamente todo ser humano possui impressões digitais. Elas permanecem imutáveis por toda a

vida, apesar de crescerem durante a infância, seus detalhes não mudam. Cortes ou queimaduras podem mudar temporariamente a impressão digital. No entanto, exceto no caso de formação de cicatrizes, elas se regeneram idênticas ao que eram antes.

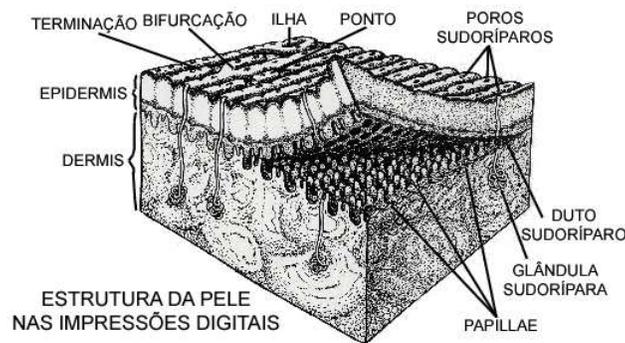


Figura 1.2. Estrutura da impressão digital.

As impressões digitais são formadas no estágio embrionário sendo determinada pelos genes e pelo ambiente intra-uterino. O processo de diferenciação é ativado pelo crescimento das almofadas volares das palmas, solas e dedos. Devido às pequenas variações no fluxo do líquido amniótico as células dos dedos crescem em micro-ambientes ligeiramente diferentes o que faz com que as impressões digitais sejam diferentes em cada um dos dedos. Esta diferença também é encontrada nos gêmeos monozigóticos. Apesar da enorme variedade, sendo praticamente impossível duas impressões digitais serem iguais, elas não são padrões totalmente aleatórios, pois são inicialmente determinadas pelos genes.

1.4.2 História

As impressões digitais podem ser encontradas em um grande número de artefatos arqueológicos e históricos [59]. Na Nova Escócia, foi encontrado um desenho de uma mão com os padrões de cristas. No Egito, foi descoberta uma das primeiras impressões dos pés e das mãos, datadas de 4000 anos atrás. Na Babilônia antiga, as impressões digitais eram usadas em blocos de argila para transações comerciais. Na China, era prática comum usar impressões digitais em documentos oficiais, venda de terras, contratos, empréstimos e reconhecimento de débitos [108], sendo encontrados documentos desde o século III A.C. Na Pérsia do século XIV, vários documentos oficiais possuíam impressões digitais e um oficial do governo observou que quaisquer duas impressões digitais não eram exatamente iguais [109].

Apenas no século XVI os primeiros estudos modernos começaram a ser feitos [109] e [110]:



- Marcello Malpighius (1628-1694): Professor de anatomia na Universidade de Bolonha descreveu em seu tratado de 1686 a existência de padrões nas cristas, como curvas e redemoinhos;



- Dr. Nehemiah Grew (1641-1712): Membro da Sociedade Real e da Faculdade de Físicos descreveu e mostrou desenhos acurados dos padrões de cristas dos dedos e da palma no Philosophical Transactions de 1684;



- Joannes Evangelista Purkinje (1787-1869): Professor de anatomia da Universidade de Breslau publicou em sua tese 9 tipos de padrões de cristas (curva transversa, estria central longitudinal, listra oblíqua, curva oblíqua, redemoinho amêndoa, redemoinho espiral, eclipse, círculo e redemoinho duplo);



- William Herschel (1791-1871): Magistrado Chefe do distrito de Jungipoor (Índia) foi o primeiro europeu a reconhecer o valor das impressões digitais com o objetivo de identificação, usando-as nos contratos locais. Ele escreveu “Skin Furrows of the Hand”, publicado na Nature em 1880 e “The Origin of Fingerprinting” [33];



- Dr. Henry Faulds (1843-1930): Cirurgião do Hospital Tsukiji em Tóquio (Japão) começou seu estudo em impressões digitais após notá-las em cerâmicas antigas, reconhecendo sua importância na identificação, criando um método para classificá-las. Foi o primeiro a sugerir que as impressões digitais poderiam auxiliar na investigação de crimes. Publicou um artigo na Nature em 1880 [23];



- Sir Francis Galton (1822-1911): Foi capaz de provar cientificamente o que Herschel e Faulds suspeitavam: as impressões digitais não mudam com a idade e não existem duas impressões digitais exatamente iguais. Introduziu as minúcias publicadas no livro “Finger Prints” de 1892 [26];



- Sir Edward Henry (1850-1931): Inspetor Geral de Polícia da Província de Bengala (Índia) com a ajuda de dois assistentes indianos desenvolveu um sistema de classificação permitindo que as impressões digitais fossem facilmente arquivadas e recuperadas, este sistema ainda é utilizado nos dias atuais. Em 1900 publicou o livro “The Classification and Uses of Finger Prints” [32];



- Juan Vucetich (1858-1925): Oficial da Polícia Argentina foi responsável pela primeira identificação criminal utilizando impressões digitais em 1892, também desenvolveu um sistema de classificação.

1.4.3 Características

A estrutura mais característica das impressões digitais é o padrão de cristas e vales intercalados. A espessura das cristas varia de 100 μm a 300 μm , sendo que o período da estrutura crista/vale é de aproximadamente 500 μm [59]. As impressões digitais podem ser definidas tanto por suas características globais quanto por suas características locais. No nível global o padrão assume formas distintas em uma ou mais regiões, tais como frequência, orientação e curvatura. As principais características globais são as singularidades: núcleo e delta (Figura 1.3). Um ponto de interesse é o núcleo [13], definido como o ponto superior da crista mais interna. O principal método de classificação de impressões digitais, o sistema de Henry, utiliza as singularidades, no qual as impressões digitais são divididas em cinco classes principais [98], [99] e [95]. A classificação de imagens permite direcionar a busca [41], [42] e [62]. No nível local são encontradas outras características chamadas minúcias, que são descontinuidades das cristas. As principais minúcias são as terminações e as bifurcações (Figura 1.3), os outros tipos podem ser descritos a partir destas duas, tais como: lago, duas bifurcações voltadas uma para a outra; ilha, crista limitada por duas terminações uma próxima da outra. No caso de imagens com boa qualidade e alta resolução outra característica possível de ser usada são os poros sudoríparos (Figura 1.3). O número, a disposição e a forma destes poros são altamente distintivos.



Figura 1.3. Singularidades e Minúcias.

1.5 Reconhecimento

O sistema de reconhecimento de impressões digitais compara duas amostras e retorna um grau de similaridade, que dependendo do limiar escolhido indica se as duas amostras pertencem ou não ao mesmo dedo. As dificuldades de reconhecer as impressões digitais se devem a grande variabilidade em diferentes amostras do mesmo dedo. Os principais fatores responsáveis por estas variações são [59]:

- Deslocamento: o mesmo dedo pode ser colocado em diferentes posições do sensor;
- Rotação: o mesmo dedo pode ser colocado em diferentes ângulos no sensor;
- Sobreposição parcial: diferentes áreas do mesmo dedo são capturadas pelo sensor;
- Distorções não-lineares: devido à plasticidade da pele a área de contato do dedo no sensor se dilata de maneira não uniforme;
- Condições da pele: a diferença de pressão, a umidade, o suor, a gordura e a sujeira alteram o contato do dedo no sensor, levando a diferenças na qualidade das impressões digitais;
- Ruído: introduzido geralmente pelo sensor devido, principalmente, a resíduos deixados por capturas anteriores.

Dependendo das características que utilizam os sistemas de reconhecimento de impressões digitais, eles podem ser divididos em: baseado em correlação [36] e [54] ou baseado em minúcias. A grande maioria destes sistemas atualmente utiliza o método baseado em minúcias, enquanto que os baseados em correlação são utilizados principalmente para a classificação das impressões digitais [10] e [94]. Um dos principais métodos de correlação utilizados para o reconhecimento é o FingerCode que utiliza a textura da impressão digital extraída por um banco

de filtros Gabor. Os sistemas baseados em minúcias são caracterizados pelo uso de atributos invariantes em relação às variações globais, como translação e rotação. No entanto estes dois tipos de sistemas, baseados em correlação e em minúcias, não levam em conta a rica informação encontrada no outro nível. Os sistemas baseados em minúcias não aproveitam informações do nível global enquanto os baseados em correlação não utilizam informações do nível local. Um modo de solucionar este problema é o uso de sistemas híbridos [62], [76] e [81].

1.5.1 Reconhecimento Baseado em Minúcias

Nos sistemas baseados em minúcias dois conjuntos de minúcias, da imagem de entrada e a de referência, são comparados [77]. Caso a quantidade de minúcias que combinam for acima de um determinado número é dito que as duas impressões digitais pertencem ao mesmo dedo, caso contrário é dito que elas pertencem a dedos diferentes. A Figura 1.4 mostra as principais etapas de um sistema baseado em minúcias.

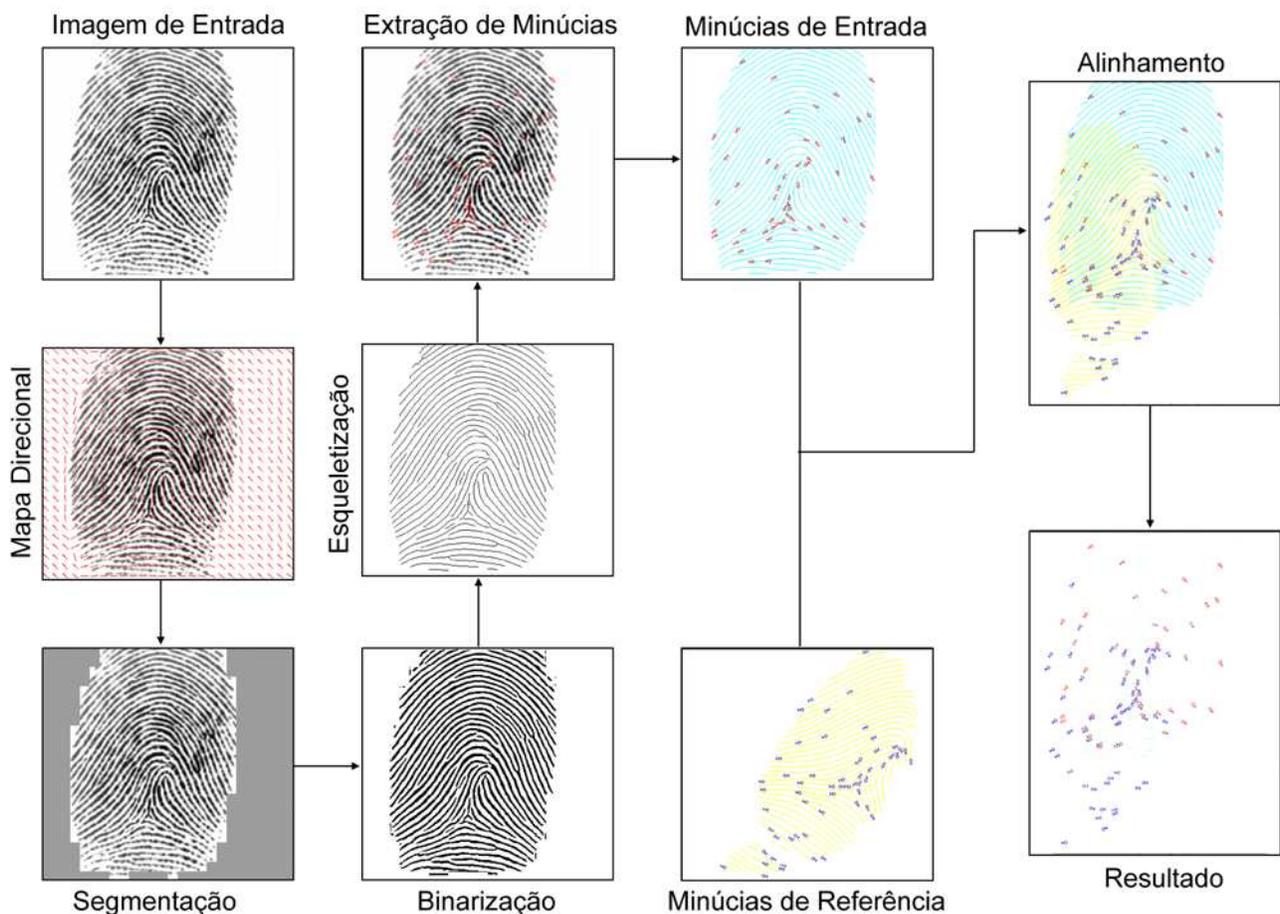


Figura 1.4. Etapas de um sistema de reconhecimento baseado em minúcias [39].

- Aquisição: a imagem de entrada é coletada [103]. A coleta pode ser feita digitalmente através de sensores específicos, que podem ser óticos, capacitivos, etc., ou pode primeiro ser coletada utilizando-se tinta e papel e depois a digitalização é feita. As imagens obtidas podem ser submetidas a um processamento para melhorar a sua qualidade [29], [35], [70] e [96].
- Mapa Direcional: a direção indica a orientação de um determinado ponto na crista [14]. Um dos métodos mais populares, proposto por Kass e Witkin [49], consiste em utilizar o gradiente da imagem. O gradiente é obtido convoluindo a imagem com a máscara de Sobel (Figura 1.5). A direção é dada pela média do dobro dos ângulos dos pontos dentro do bloco. Este método é utilizado em [79], [80] e [7]. A direção de cada bloco (i, j) é dada pela Equação 1.1.

$$\begin{array}{cc} \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} & \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \\ \text{(a)} & \text{(b)} \end{array}$$

Figura 1.5. Máscara de Sobel 3x3: (a) Operador horizontal; e (b) Operador vertical.

$$\theta(i, j) = \begin{cases} \phi(i, j) - \frac{\pi}{4}, & \text{se } \phi(i, j) \geq 0 \\ \phi(i, j) + \frac{\pi}{4}, & \text{se } \phi(i, j) < 0 \end{cases}, \quad (1.1.a)$$

$$\phi(i, j) = \begin{cases} \frac{1}{2} \arctan\left(\frac{G_n(i, j)}{2G_d(i, j)}\right), & \text{se } G_n(i, j) < 0 \text{ e } G_d(i, j) \neq 0 \\ \frac{1}{2} \arctan\left(\frac{G_n(i, j)}{2G_d(i, j)}\right) + \frac{\pi}{2}, & \text{se } G_n(i, j) > 0 \text{ e } G_d(i, j) > 0 \\ \frac{1}{2} \arctan\left(\frac{G_n(i, j)}{2G_d(i, j)}\right) - \frac{\pi}{2}, & \text{se } G_n(i, j) > 0 \text{ e } G_d(i, j) < 0 \\ 0, & \text{caso contrário} \end{cases}, \quad (1.1.b)$$

$$G_n(i, j) = \sum_h \sum_k \nabla_x(x_i + h, y_j + k) \square \nabla_y(x_i + h, y_j + k), \quad (1.1.c)$$

$$G_d(i, j) = \sum_h \sum_k \nabla_x(x_i + h, y_j + k)^2 - \nabla_y(x_i + h, y_j + k)^2, \quad (1.1.d)$$

onde ∇_x e ∇_y são os gradientes horizontal e vertical computados através da convolução da imagem com os operadores de Sobel horizontal e vertical, respectivamente; e $x_i + h$ e $y_j + k$ dão as posições de todos os pixels dentro do bloco (i, j) . Em [4] é apresentado outro método para determinar o campo direcional.

- Segmentação: o fundo da imagem é separado da área de interesse da imagem [37], evitando a extração de falsas minúcias do fundo e de áreas muito ruidosas. Mais detalhes do processo de segmentação será visto no Capítulo 2.
- Binarização: a imagem em tons de cinza é convertida para uma imagem binária. A binarização melhora o contraste entre as cristas e os vales e, conseqüentemente, facilita a extração das minúcias. Um dos métodos proposto em [71] usa um filtro contextual para melhorar a imagem para então fazer a binarização. Alguns métodos como o proposto em [58] extraem as minúcias diretamente da imagem em tons de cinza.
- Esqueletização: é uma operação morfológica que erode sucessivamente as cristas até que elas tenham um pixel de espessura [89]. Esta operação forma uma versão do esqueleto da imagem binária preservando a conectividade.
- Extração de Minúcias: após a esqueletização uma simples busca na imagem permite a localização das minúcias [2]. As minúcias são caracterizadas como os pontos que possuem um número de vizinhos diferente de dois. Uma terminação possui apenas um vizinho e uma bifurcação possui três ou mais vizinhos. Na extração é definido o tipo, a localização e a direção da minúcia [27]. Em muitos casos é feito um pós-processamento para eliminar falsas minúcias. Os dois tipos básicos de pós-processamento são baseados na estrutura do esqueleto ou na filtragem da imagem em tons de cinza.
- Alinhamento: todas as sobreposições dos dois conjuntos de minúcias, o de entrada e o de referência, são testadas [40]. A melhor sobreposição gera os parâmetros de rotação e translação. Um dos métodos mais populares é a da transformada Hough que converte o problema de correspondência do padrão de minúcias para o problema de detectar picos no espaço de Hough.
- Resultado: a partir do número de correspondências entres os dois conjuntos de minúcias é gerado um escore. De acordo com um limiar escolhido o sistema responde se as duas amostras, entrada e referência, pertencem ao mesmo dedo ou não.

1.5.2 Reconhecimento Baseado em Texturas (FingerCode)

Baseado no modelo para reconhecimento de íris [17], Jain et al. [43] desenvolveram um método para reconhecimento de impressões digitais denominado FingerCode. Este método usa a textura extraída por um banco de filtros Gabor para fazer a comparação das impressões digitais [75]. As etapas do método são mostradas na Figura 1.6.

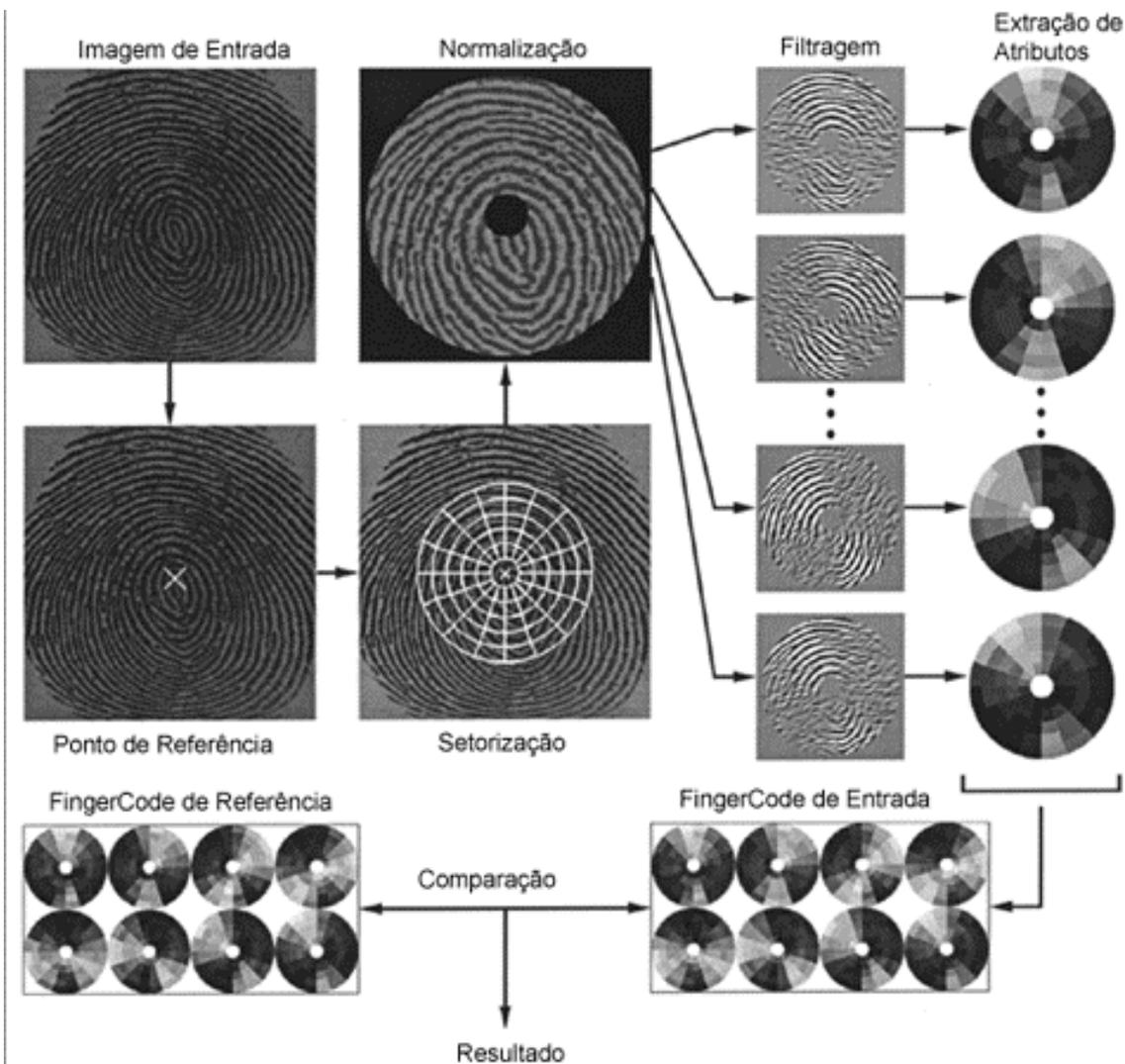


Figura 1.6. Estágios do método FingerCode.

- Aquisição: a imagem de entrada é coletada. A coleta pode ser feita digitalmente através de sensores específicos, que podem ser óticos ou capacitivos, ou pode primeiro ser coletada utilizando-se tinta e papel e depois a digitalização é feita;

- **Ponto de Referência:** o núcleo é localizado [93]. Um dos métodos para localizar o núcleo é baseado no índice de Poincaré proposto em [50]. Considerando o mapa direcional da imagem obtido pela Equação 1.1 o índice de Poincaré, P , no ponto (i, j) é dado pela soma das diferenças dos ângulos dos vizinhos, $N = \{d_0, d_1, d_2, d_3, d_4, d_5, d_6, d_7\}$ (Figura 1.7), em torno do ponto (i, j) (Equação 1.2).

	d_2	d_3	d_4	
	d_1	(i, j)	d_5	
	d_0	d_7	d_6	

Figura 1.7. Vizinhança do ponto (i, j) .

$$P(i, j) = \sum_{k=0}^7 d_k - d_{(k+1) \bmod 8} \quad (1.2)$$

$$\text{Se } P(i, j) = \begin{cases} 0^\circ, & (i, j) \text{ não pertence a nenhuma singularidade} \\ 360^\circ, & (i, j) \text{ pertence a um core do tipo redemoinho} \\ 180^\circ, & (i, j) \text{ pertence a um core do tipo curva} \\ -180^\circ, & (i, j) \text{ pertence a um delta} \end{cases}$$

O núcleo também pode ser obtido através da partição do mapa direcional da imagem usando um algoritmo de clustering iterativo com um conjunto de máscaras dinâmicas, este método é proposto em [57] e [11]. Jain et al. [44] propõe um método que usa a integração dos senos em duas regiões adjacentes definidas para capturar a máxima curvatura. Os pontos que maximizam a diferença da integração das duas regiões são selecionados como candidatos e examinados em um nível mais detalhado. Outros métodos para localização do núcleo são apresentados em [6], [69], [73] e [82].

- **Setorização:** a região em torno do núcleo é dividida em setores semicirculares. Esta é a região utilizada no processo de reconhecimento;
- **Normalização:** a região de interesse é normalizada. Este passo é necessário para minimizar os efeitos dos ruídos e diferenças de pressão no processo de aquisição da imagem. Em [34] é proposto um método de normalização local. Neste método cada setor é normalizado separadamente para uma média, M_o , e variância, V_o , pré-determinada. Seja I um pixel localizado na posição (x, y) , S_i o setor que contém o pixel I , L_i o nível de

cinza pixel I , e M_i e V_i a média e a variância, respectivamente, do setor S_i . O novo nível de cinza normalizado, N_i , do pixel I , é dado pela Equação 1.3.

$$N_i = \begin{cases} M_o + \sqrt{\frac{V_o \times (L_i - M_i)^2}{V_i}}, & \text{se } L_i > M_i \\ M_o - \sqrt{\frac{V_o \times (L_i - M_i)^2}{V_i}}, & \text{caso contrário} \end{cases} \quad (1.3)$$

- Filtragem: a imagem é convoluída com um banco direcional de filtros Gabor 2D (ver Capítulo 3) resultando em oito componentes orientados. O banco é formado por oito filtros nas direções $\Theta = \{0^\circ, 22.5^\circ, 45^\circ, 67.5^\circ, 90^\circ, 112.5^\circ, 135^\circ, 157.5^\circ\}$. A frequência escolhida foi baseada na distância média aproximada entre as cristas de 10 pixels para uma imagem de 500 dpi ($f_0 = 0.1 \text{ pixel}^{-1}$). Há métodos que calculam a frequência entre cristas o que permite uma escolha mais precisa da frequência do filtro Gabor [47] e [53].
- Extração de Atributos: para cada setor de cada componente orientado é computado um valor de atributo de textura, $F_{\rho\theta}$, dado pela Equação 1.4.

$$F_{\rho\theta} = \sqrt{\frac{1}{K_{\rho\theta}} \sum_{(x,y) \in C_{\rho\theta}} (C_{\rho\theta}(x,y) - \bar{C}_{\rho\theta})^2}, \quad (1.4)$$

$$\bar{C}_{\rho\theta} = \frac{1}{K_{\rho\theta}} \sum_{(x,y) \in C_{\rho\theta}} C_{\rho\theta}(x,y)$$

onde $C_{\rho\theta}$ é o setor na posição ρ no componente orientado θ , $C_{\rho\theta}(x,y)$ é o valor do pixel na posição (x,y) dentro do setor $C_{\rho\theta}$, $K_{\rho\theta}$ é o número de pixels do setor $C_{\rho\theta}$, e $\bar{C}_{\rho\theta}$ é a média dos valores dos pixels dentro do setor $C_{\rho\theta}$. O FingerCode é o vetor composto pelo conjunto dos atributos de textura de todos os setores nos oito componentes orientados.

- Comparação: dois FingerCodes são comparados e a distância Euclideana é computada. O escore, dado pela distância Euclideana, é obtido pela soma dos setores do vetor diferença (Equação 1.5):

$$S = \sum_{\rho \in P} \sum_{\theta \in \Theta} E_{\rho\theta} \quad (1.5)$$

E é o vetor de diferenças formado pela diferença absoluta entre os setores correspondentes dos vetores de atributo de entrada, F , e referência, T (Equação 1.6).

$$E_{\rho\theta} = |F_{\rho\theta} - T_{\rho\theta}| \quad (1.6)$$

onde $E_{\rho\theta}$, $F_{\rho\theta}$ e $T_{\rho\theta}$ são os valores dos setores na posição ρ e na direção θ dos vetores de diferenças, de entrada e de referência, respectivamente.

- Resultado: se o escore obtido for abaixo de um pré-determinado limiar o sistema considera que as impressões digitais vieram do mesmo dedo, caso contrário o sistema considera que elas vieram de dedos diferentes.

1.5.3 Método FingerCode Modificado

O método FingerCode modificado foi proposto em [83], este método combina minúcia e textura, ou seja usa informações locais e globais. Este método apresenta melhor robustez que os métodos baseados apenas em minúcias ou FingerCode. O método FingerCode modificado foi utilizado nos experimentos para a avaliação das melhorias propostas. A implementação destas melhorias no método FingerCode é simples e direta.

A principal diferença em relação ao método FingerCode é o uso das minúcias para o alinhamento das impressões digitais ao invés do núcleo. A extração das minúcias é mais confiável do que a extração do núcleo, principalmente em imagens da baixa qualidade. Além do que impressões digitais do tipo arco não possuem núcleo. Também há casos em que o núcleo não se encontra dentro da área capturada da impressão digital, isto ocorre principalmente em sensores capacitivos que possuem área de captura reduzida. Outras diferenças são: o uso de toda a área da impressão digital e não apenas da área em torno do núcleo e a setorização retangular em vez de arcos [46].

O primeiro estágio do método FingerCode modificado consiste no alinhamento das impressões digitais, de entrada e de referência, através das minúcias, conseqüentemente todas as etapas do reconhecimento baseado em minúcias, exceto o cálculo do resultado, são aplicados: aquisição, mapa direcional, segmentação, binarização, esqueletização, extração de minúcias e alinhamento. Após o alinhamento segue as etapas do método do FingerCode: setorização,

normalização, filtragem, segmentação, extração de atributos, comparação e resultado. Na setorização toda imagem é dividida em setores retangulares (Figura 1.8).



Figura 1.8. Setorização da imagem.

Após a normalização usando a Equação 1.3, a imagem é segmentada utilizando o método descrito na Seção 2.1. Os setores que possuem ao menos um pixel do fundo são considerados pertencentes ao fundo, caso contrário pertencentes à área de interesse da impressão digital (Figura 1.9)

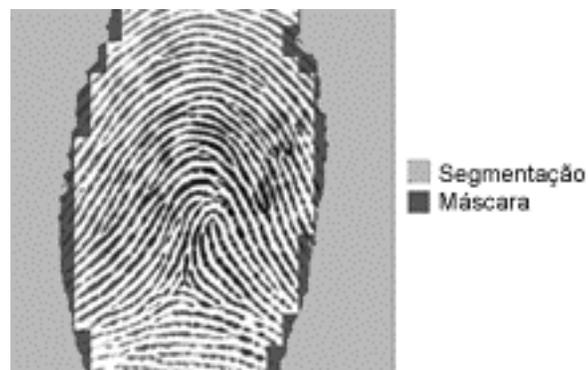


Figura 1.9. Máscara da imagem.

Os componentes orientados da imagem são obtidos pela filtragem da imagem por um banco direcional de filtros Gabor 2D [92] nas direções $\Theta = \{0^\circ, 22.5^\circ, 45^\circ, 67.5^\circ, 90^\circ, 112.5^\circ, 135^\circ, 157.5^\circ\}$ (Figura 1.10). Para cada setor de cada componente orientado é computado usando a Equação 1.4 resultando em um valor de atributo. O vetor de atributos de textura é formado pelos valores de atributo de todos setores de todos componentes orientados.

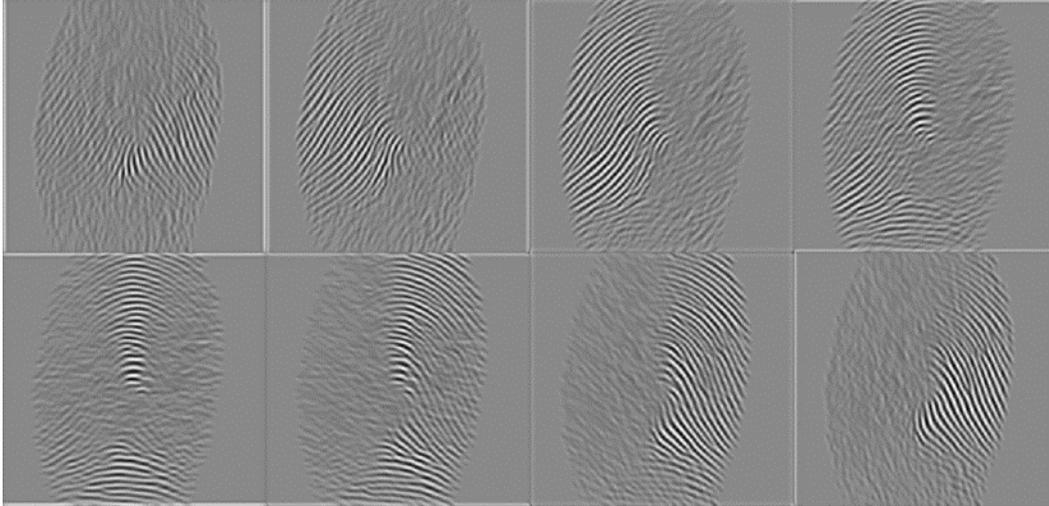


Figura 1.10. Componentes orientados.

O resultado é dado comparando o escore obtido com o limiar escolhido, se o escore for menor que limiar o sistema responde que as impressões pertencem ao mesmo dedo, caso contrário responde que pertencem a dedos diferentes. O escore é calculado como a soma da diferença absoluta dos dois vetores atributos de textura normalizada pelo número de setores válidos (Equação 1.7). Os setores válidos são dados pela interseção das máscaras de cada uma das impressões digitais obtidas no processo de segmentação. A interseção das máscaras é replicada em todos os componentes orientados.

$$S_M = \frac{1}{V} \sum_{\rho \in P} \sum_{\theta \in \Theta} V_{\rho\theta} E_{\rho\theta}, \quad (1.7)$$

onde $V_{\rho\theta}$ indica se o setor na posição ρ na direção θ é válida ($V_{\rho\theta} = 1$) ou não-válida ($V_{\rho\theta} = 0$), e V é o número de setores válidos. A Figura 1.11 mostra a comparação entre amostras do mesmo dedo e amostras de dedos diferentes.

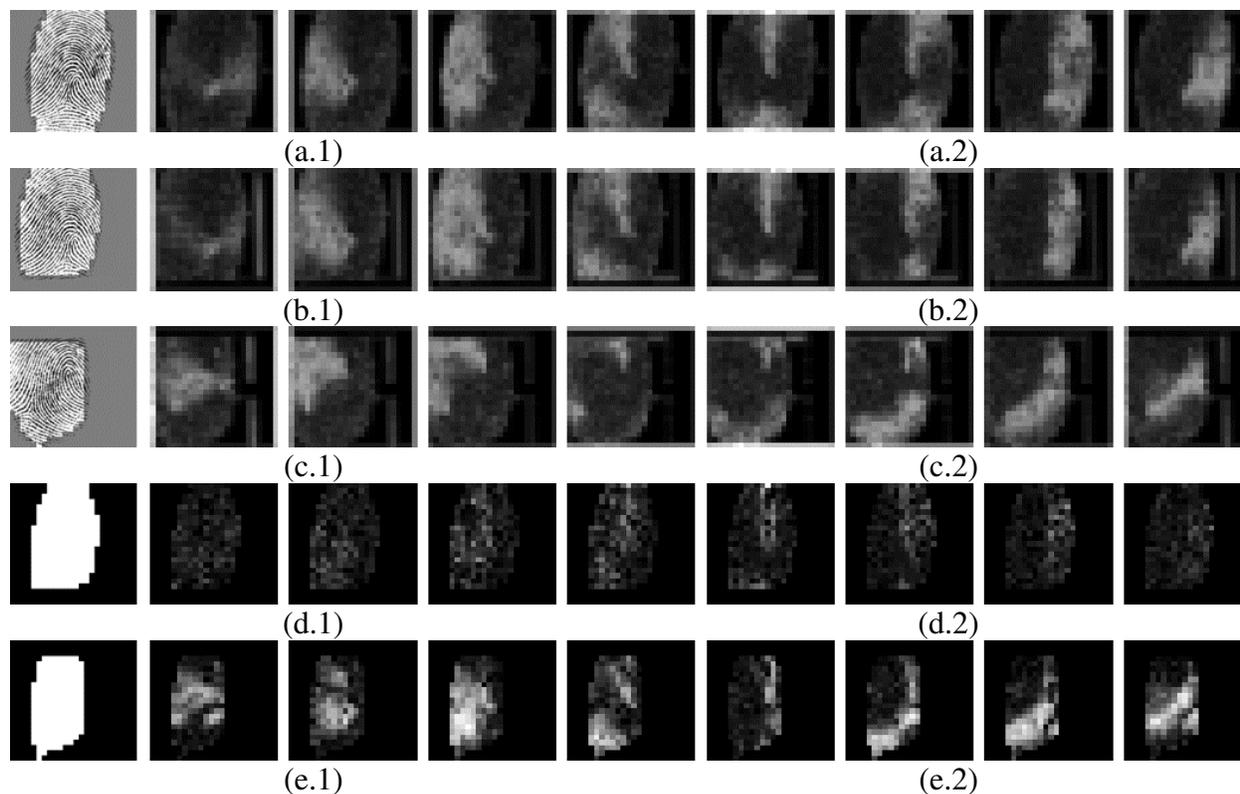


Figura 1.11. Comparação das texturas: (a.1) e (a.2) Amostra de referência e seu vetor de atributos de textura, respectivamente, já segmentados; (b.1) e (b.2) Amostra de entrada do mesmo dedo da referência e seu vetor de atributos de textura, respectivamente, já segmentados e alinhados em relação à referência; (c.1) e (c.2) Amostra de entrada de um dedo diferente da referência e seu vetor de atributos de textura, respectivamente, já segmentados e alinhados em relação à referência; (d.1) Interseção das máscaras das amostras do mesmo dedo, a.1 e b.1; (d.2) Vetor de diferença (Equação 1.6) resultante da comparação entre as amostras do mesmo dedo, a.2 e b.2; (e.1) Interseção das máscaras das amostras de dedos diferentes, a.1 e c.1; e (e.2) Vetor de diferença (Equação 1.6) resultante da comparação entre amostras de dedos diferentes, a.2 e c.2.

Obs.: Os parâmetros de alinhamento pelas minúcias foram gentilmente cedidos pela Griaule [105].

1.6 Base de Dados

Nos experimentos foi utilizada a base de dados da Fingerprint Verification Competition 2002 (FVC2002) para comparar o ganho obtido com as novas propostas. A base de dados do FVC2002 consiste de quatro bancos diferentes de impressões digitais (Db1-Db4), cada um contendo imagens coletadas utilizando um sensor/tecnologia diferente (Tabela 1.3).

Tabela 1.3. Base de dados da FVC2002

	Sensor	Image Size	Resolution
Db1	Sensor Ótico “TouchView II” da Identix	388x374	500 dpi
Db2	Sensor Ótico “FX2000” da Biometrika	296x560	569 dpi
Db3	Sensor Capacitivo “100 SC” da Precise Biometrics	300x300	500 dpi
Db4	Impressões Digitais Sintéticas “SFinGe v2.51”	288x384	cerca de 500 dpi

Cada banco de imagens é composto por 8 amostras de impressões digitais de cada um dos 100 dedos distintos. Nos experimentos, assim como na metodologia de comparação dos sistemas competidores do FVC, foram comparadas entre si as oito amostras do mesmo dedo e, também, foi comparada a primeira amostra de cada dedo com a primeira amostra dos outros dedos. Estas combinações resultam em 2800 comparações entre diferentes amostras do mesmo dedo e 4950 comparações entre as primeiras amostras de dedos diferentes.

O FVC possui um grande prestígio nas comunidades de biometria, tanto acadêmicas como industriais. Pois, além de fornecer um critério comum de comparação dos algoritmos, também provê uma visão geral do estado da arte no reconhecimento de impressões digitais. Desta forma os testes realizados nos experimentos seguiram os padrões do FVC2002 de forma a permitir, também, uma análise comparativa com outros sistemas de reconhecimento de impressão digital.

Capítulo 2

2 Segmentação

A segmentação é o processo de separação da área de interesse e do fundo da imagem [9] e [22]. Este processo é útil para evitar a extração de características de áreas que não fazem parte da impressão digital ou áreas muito ruidosas. O que diferencia a área de interesse da impressão digital do fundo é a presença do padrão cristas e vales. No caso de imagens de boa qualidade a segmentação é um processo simples e pode ser obtido baseado na intensidade local [59], no entanto impressões digitais de má qualidade ou presença de ruído requerem técnicas mais robustas.

Alguns métodos para a segmentação de impressão digital consistem em setorizar a imagem e calcular a variância da intensidade dos tons de cinza de cada um dos setores. Caso a variância esteja abaixo de certo limiar o setor pertence ao fundo, caso contrário pertence a área de interesse [63] e [64]. Utilizando o mapa direcional dado pela Equação 1.1, apresentado em [39], é calculado o nível de certeza local da direção de cada bloco. Outra proposta é apresentada em [5] em que um classificador linear determina se um pixel pertence ao fundo ou não baseado em três índices (coerência do gradiente e média e variância da intensidade). Após a classificação, o algoritmo realiza um pós-processamento morfológico para eliminar furos e regularizar a silhueta. Outros métodos usam filtro Gabor para detectar áreas com determinadas características na imagem [45]. No método apresentado em [86] a imagem é convoluída com um banco direcional de filtros Gabor e a variância da resposta nas direções é utilizada para segmentar a imagem.

2.1 Segmentação Morfológica

Esta nova proposta de segmentação é baseada na morfologia matemática. A morfologia matemática é bastante usada para correções em imagens [21]. No reconhecimento de impressões

digitais a morfologia matemática é utilizada principalmente na esqueletização, sendo utilizada também em alguns estágios de pré-processamento ou pós-processamento [28]. Em [5] a morfologia matemática é usada para uniformizar a área da impressão digital.

A proposta apresentada se utiliza do padrão estriado da impressão digital no processo de segmentação. Para detectar este padrão foi utilizado o operador *open top-hat*. A subtração da imagem aberta pela imagem original detecta áreas claras circundadas por áreas escuras [21]. Deste modo escolhendo um elemento estruturante com raio maior que a largura entre duas cristas seleciona o padrão de cristas. Esta operação além de selecionar, realiza uma normalização das cristas. Aplicando um limiar, a imagem em tons de cinza é convertida para uma imagem binária, com as cristas em branco e o restante da imagem em preto. O próximo passo consiste em expandir as cristas, através de uma operação de dilatação, criando uma zona uniforme e contínua. As etapas desta proposta são mostradas na Figura 2.1. Este foi o método utilizado nos experimentos para segmentar as imagens.

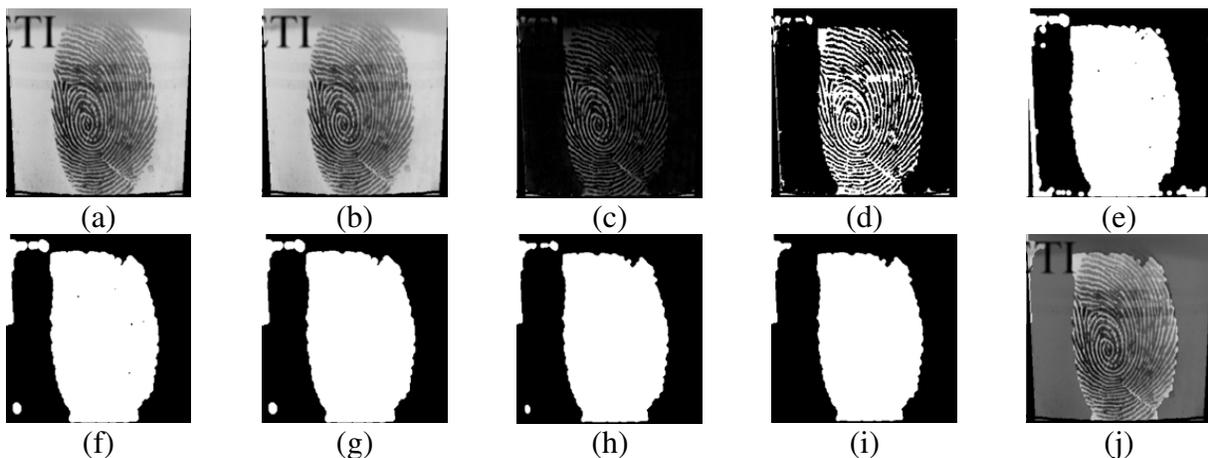


Figura 2.1. Etapas do processo de segmentação morfológica: (a) Imagem de entrada; (b) Fechamento com disco de raio = 1; (c) Abertura Top-Hat com disco de raio = 10; (d) Limiar = 20; (e) Dilatação com disco de raio = 5; (f) Abertura com disco de raio = 8; (g) Preenche Furos; (h) Erosão com disco de raio = 4; (i) Abertura por área com área = 500; e (j) Máscara sobreposta à imagem.

As etapas após a binarização são realizadas apenas para ajustar a área segmentada e eliminar pequenos ruídos, várias alternativas podem ser utilizadas sem perda de qualidade. As etapas do método de segmentação morfológica são dadas abaixo:

- Fechamento: elimina pequenos ruídos;
- Abertura Top-Hat: seleciona as cristas, áreas escuras cercadas por áreas claras;
- Limiar: binariza a máscara;
- Dilatação: expande as cristas, criando uma área uniforme;
- Abertura: diminui a quantidade de ruídos do fundo;
- Preenchimento de Furos: elimina ruídos da área da impressão digital;
- Erosão: ajusta a máscara;
- Abertura por Área: elimina ruídos do fundo.

2.2 Segmentação Baseada em Filtro Gabor

A segmentação Gabor é uma nova proposta de segmentação derivada da segmentação apresentada em [86]. Neste método a imagem é dividida em setores e para cada setor um ponto é amostrado. A imagem é convoluída com um banco direcional de filtros Gabor no ponto amostrado, que resulta em um conjunto de componentes orientados formados pelos pontos amostrados dos setores. Para cada posição amostrada é calculado o desvio padrão dos pontos na mesma posição nos componentes orientados. As cristas por possuírem uma direção bem definida possuem uma alta resposta ao filtro Gabor perpendicular e uma baixa resposta ao filtro Gabor na mesma direção enquanto que a região do fundo não possui uma direção predominante e sua resposta aos filtros Gabor são semelhantes em todas as direções. Portanto o desvio padrão da área com cristas é grande e a região do fundo possui um desvio padrão pequeno. Se o desvio padrão for abaixo de certo limiar o setor pertence ao fundo caso contrário pertence á área de interesse [24].

Na variação proposta a segmentação é feita ao nível do pixel e não ao nível de setor. A imagem é convoluída com um banco direcional de oito filtros Gabor, obtendo oito componentes orientados. Para cada posição da imagem é calculado o desvio padrão dos pixels na mesma posição nos oito componentes orientados (Equação 2.1). Após esta etapa é aplicado um processo morfológico para uniformizar a área de interesse [21]. Na Figura 2.2 é mostrado todas as etapas desta nova proposta.

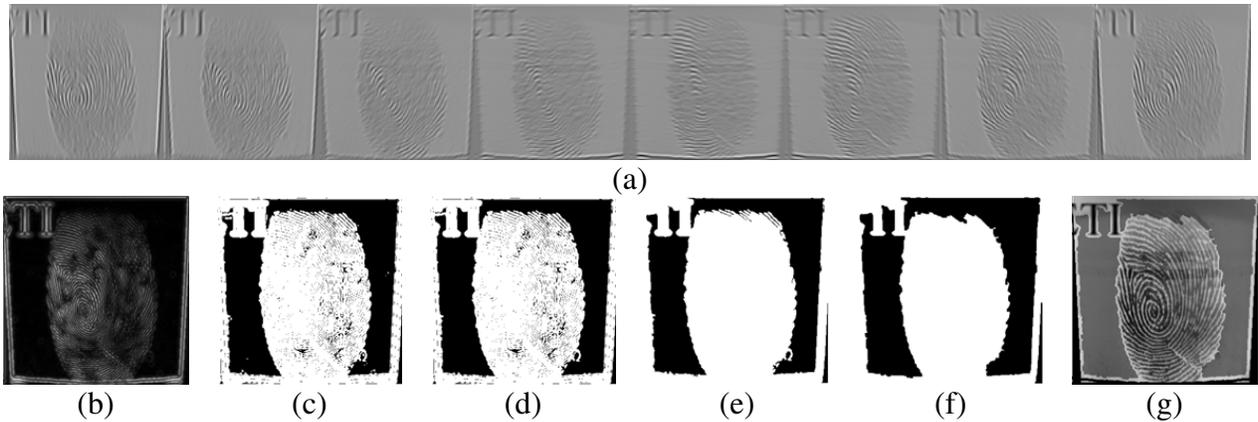


Figura 2.2. Etapas do processo de segmentação Gabor: (a) Componentes orientados obtidos pelos oito filtros Gabor; (b) Desvio padrão dos pixels na mesma posição; (c) Limiar = 100; (d) Abertura por área com área = 500; (e) Fechamento por área com área = 250; (f) Erosão com disco de raio = 3; e (g) Máscara sobreposta à imagem.

Devido à grande quantidade de ruído a etapa de fechamento por área é necessária para criar uma zona uniforme na região da impressão digital. As outras etapas morfológicas apesar de não serem estritamente necessárias ajudam a obter uma melhor qualidade na segmentação. As etapas do método de segmentação baseada em filtro Gabor são dadas abaixo:

- Filtragem: convolução da imagem com filtros Gabor nas direções $\Theta = \{0^\circ, 22.5^\circ, 45^\circ, 67.5^\circ, 90^\circ, 112.5^\circ, 135^\circ, 157.5^\circ\}$, gerando oito componentes orientados;
- Desvio Padrão: calculado para os pixels na mesma posição em cada um dos componentes orientados;
- Limiar: binariza a máscara;
- Abertura por Área: elimina ruídos do fundo;
- Fechamento por Área: elimina ruídos da área da impressão digital;
- Erosão: ajusta a máscara.

2.3 Comparação dos Métodos de Segmentação

Para comparar a qualidade das segmentações propostas foi realizado um teste com vários métodos de segmentação. Para os testes foi utilizada a base da FVC2002. O escore foi obtido usando a metodologia descrita no Artigo 4, em que apenas a máscara é alterada de acordo com o método de segmentação. A pequena diferença entre os valores obtidos no Artigo 4 e os

apresentados nesta Seção com a abertura top-hat se deve às diferenças nas etapas de uniformização da área segmentada. Todos os métodos sofreram um pós-processamento morfológico para uniformizar a área obtida. Os métodos utilizados são brevemente descritos abaixo:

1. Abertura Top-Hat: método proposto na Seção 2.1, utiliza o operador morfológico open top-hat para detectar a estrutura crista-vale. Primeiramente é feito um fechamento com um disco de raio = 1, para eliminar pequenos ruídos. Para detectar a região da impressão digital aplica-se o operador de abertura top-hat com um disco de raio = 10. O diâmetro = 20 corresponde a 1,5 da estrutura crista-vale (duas cristas e um vale ou dois vales e uma crista) em uma imagem de 500dpi. Após esta etapa a imagem é binarizada usando um limiar = 20. Para uniformizar a área segmentada é feita uma dilatação com um disco de raio = 5, um preenchimento dos furos e uma abertura com um disco de raio = 8.
2. Filtro Gabor por Ponto: método proposto na Seção 2.2, utiliza um banco de filtros Gabor para segmentar a imagem. Primeiramente a imagem é convoluída com um banco de filtros Gabor em 8 direções, extraindo os componentes orientados. Posteriormente, é calculado o desvio padrão dos pontos na mesma posição nos 8 componentes orientados. Após esta etapa, a imagem é binarizada usando um limiar = 70. Para uniformizar a área segmentada é realizado um preenchimento dos furos, uma abertura com disco de raio = 8 e uma abertura por área = 500.
3. Transformada de Fourier: neste método a imagem é dividida em blocos de 16x16. Para cada bloco a região de 32x32 pixels centralizada no bloco é selecionada. A transformada de Fourier e a intensidade média desta região são obtidas. Caso o componente principal da região no domínio da frequência seja de baixa frequência (próxima ao centro, no caso utilizado a um pixel de distância) e a intensidade média seja maior que o limiar = 60, o bloco pertence ao fundo. O sinal contínuo não é considerado na obtenção do componente principal. Para uniformizar a região é feito um fechamento com um disco de raio = 1 e um preenchimento dos furos.

4. Diferença de Intensidade: neste método a imagem é dividida em blocos de 16x16. Para cada bloco são obtidos os valores acumulados resultante da soma das intensidades em 8 direções para cada um dos pontos (Figura 2.3) do bloco. A equação é dada por $S_{Bd} = \sum_{p \in B} (I_{pd} + I_{p\bar{d}})$; onde S_{Bd} é o valor acumulado de um bloco, B , em uma direção, d ; e I_{pd} é a intensidade do pixel na direção, d , relativo ao ponto, p , do bloco, B . Após a obtenção dos 8 valores acumulados para cada uma das direções, o maior é selecionado e subtraído do valor na direção perpendicular $S_{Bd} - S_{B(d+90^\circ)}$. Posteriormente, a imagem é binarizada utilizando um limiar = 300. Para uniformizar a área segmentada é feito um fechamento com disco de raio = 1, um preenchimento dos furos, uma abertura com disco de raio = 1 e uma abertura por área = 20.

\bar{d}_5	\bar{d}_6	\bar{d}_7	\bar{d}_8	d_1
\bar{d}_4				d_2
\bar{d}_3		p		d_3
\bar{d}_2				d_4
\bar{d}_1	d_5	d_7	d_6	d_8

Figura 2.3. Direções do ponto p .

5. Filtro Gabor por Bloco: neste método o desvio padrão dos valores de atributo de textura na mesma posição em cada um dos componentes orientados é calculado. Posteriormente a imagem é binarizada utilizando um limiar = 40. Para uniformizar a área segmentada é feito um preenchimento de furos e uma abertura com disco de raio = 1.
6. Desvio Padrão do Gradiente: neste método os gradientes da imagem na horizontal e na vertical são obtidos utilizando as máscaras de Sobel. O gradiente na horizontal é somado ao gradiente da vertical. A imagem resultante é dividida em blocos de 16x16. Para cada bloco é calculado o desvio padrão. Posteriormente, a imagem é binarizada utilizando um limiar = 50. Para uniformizar a área segmentada foi feito um preenchimento de furos e uma abertura com disco de raio = 1.
7. Desvio Padrão da Intensidade: neste método a imagem é dividida em blocos de 16x16. Para cada bloco é calculado o desvio padrão da intensidade. Posteriormente a imagem é binarizada utilizando um limiar = 10. Para uniformizar a área segmentada foi feita uma abertura com disco de raio = 1 e um preenchimento de furos.

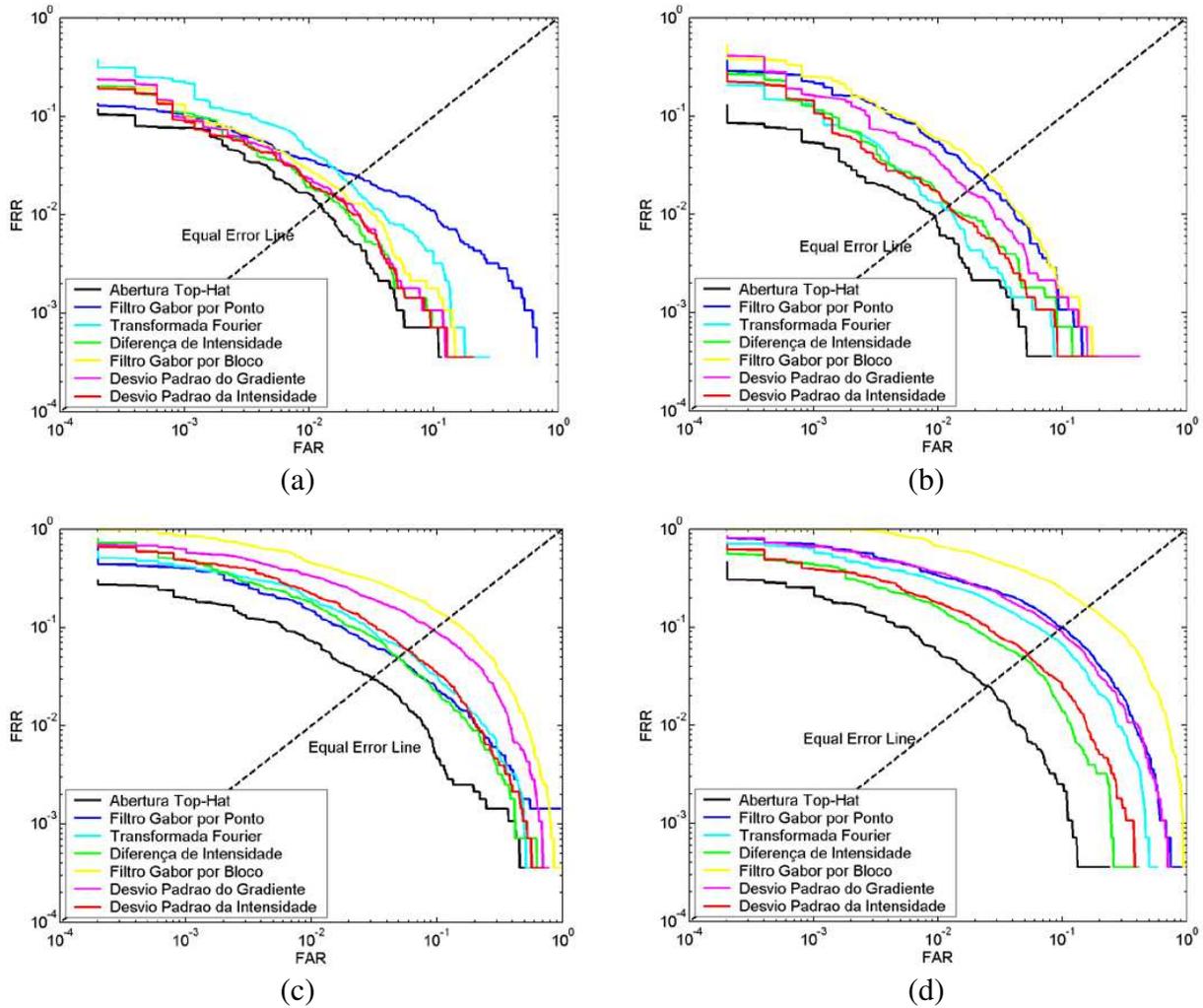


Figura 2.4. Curvas ROC dos métodos de segmentação: (a) Db1; (b) Db2; (c) Db3; e (d) Db4.

Tabela 4. Equal Error Rate (EER) para os diferentes métodos de segmentação

	Db1	Db2	Db3	Db4
Abertura Top-Hat	1.26%	0.95%	3.04%	2.50%
Filtro Gabor por Ponto	2.46%	2.24%	4.96%	9.94%
Transformada Fourier	2.08%	1.13%	5.93%	8.05%
Diferença de Intensidade	1.48%	1.23%	4.97%	4.96%
Filtro Gabor por Bloco	1.82%	2.46%	12.39%	16.16%
Desvio Padrão do Gradiente	1.60%	1.71%	9.46%	9.37%
Desvio Padrão da Intensidade	1.56%	1.24%	5.86%	5.25%

Neste experimento, através da análise da Figura 2.4 e da Tabela 4, fica mostrado que a segmentação com abertura top-hat possui uma qualidade superior aos outros métodos. O ganho no método fingercode foi em torno de 15% para as bases Db1 e Db2 e atingiu 40% na base Db3. No caso da base Db3 que possui grande quantidade de ruído o método de segmentação baseado em filtro Gabor também apresentou boa qualidade.

2.4 Segmentação de Impressões Digitais Baseada em Abertura Top-Hat

Artigo submetido para II Workshop de Visão Computacional (WVC2006).

Nesta seção é apresentada a comparação da segmentação utilizando o operador morfológico abertura top-hat em relação a outros métodos encontrados na literatura. A comparação foi realizada em dois tipos de sistemas de reconhecimento de impressões digitais: um baseado em minúcias e outro baseado no FingerCode.

Artigo 1. Segmentação de Impressões Digitais Baseada em Abertura Top-Hat

Gustavo de Sá¹, Pascual Figueroa¹, and Roberto Lotufo²

¹Griaule Tecnologia, r. Bernardo Sayão 100, sala 209, 13083-866, Campinas, SP, Brasil

²Faculdade de Engenharia Elétrica e Computação, Universidade de Campinas (UNICAMP),
C.P. 6101, 13083-970, Campinas, SP, Brasil

gustavosa@griaule.com, pascual.figueroa@griaule.com, and www.dca.fee.unicamp.br/~lotufo

Abstract

Em muitos sistemas de reconhecimento automático de impressões digitais a segmentação é uma etapa muito importante. Na segmentação a área de interesse da impressão digital é separada do fundo da imagem. Neste artigo é apresentado um método de segmentação baseado em operadores morfológicos. Os dois principais passos consistem em: extrair a estrutura de cristas e vales através de uma abertura top-hat e expandir os vales através de uma dilatação. Para mostrar a robustez do método são feitas comparações com outros métodos.

1. Introdução

A segmentação é o processo de separação da área de interesse do fundo da imagem. Este processo é útil para evitar a extração de características de áreas que não fazem parte da impressão digital ou áreas muito ruidosas. Após a segmentação, a área a ser analisada é menor, permitindo a redução do tempo de processamento. O que diferencia a área de interesse da impressão digital do fundo é a presença do padrão estriado de cristas e vales. Devido a isto, o uso de limiares globais ou locais não é efetivo para realizar a segmentação.

No caso de imagens de boa qualidade a segmentação é um processo simples e pode ser obtida baseada na intensidade local [59]. Neste método a imagem é dividida em blocos e a variância de cada bloco é calculada. A variância nas regiões que possuem cristas e vales é alta, enquanto que na região do fundo é baixa. A principal vantagem deste método é a sua simplicidade e velocidade. Outros métodos propõem o uso da média do gradiente [58] ou da variância da coerência [1] e [3] por blocos. O primeiro método se vale do aspecto da área de interesse ser rica em bordas. Enquanto o segundo método mede a força da orientação dos pixels

em cada bloco. Em [80] a direção ortogonal à direção do bloco é utilizada para calcular a variância da intensidade.

Impressões digitais de má qualidade ou com presença de muito ruído requerem técnicas mais robustas. Em [14] o espectro de Fourier é analisado para se obter a segmentação. O método proposto por Shen et al. [86] utiliza um banco de filtros Gabor em oito direções para realizar a segmentação. Este método assume que os blocos de boa qualidade possuem alta resposta a uma das direções e baixa resposta nas outras direções e os blocos de má qualidade possuem respostas similares em todas as direções. Em [24] uma variação do método anterior é proposta. As diferenças são: a sobreposição dos blocos e a estimativa da frequência local. Bernard et al. [9] utilizam um banco de filtros Gabor em oito direções e três frequências para obter as regiões de fundo, ruidosa e de interesse. Em [10] é descrito o método do contraste direcional proposto por [2]. O contraste direcional é obtido pela diferença absoluta entre a soma das intensidades na direção predominante e na direção ortogonal. Um método proposto por Bazen e Gerez [5] utiliza um classificador linear para determinar se um pixel pertence ao fundo ou não baseado em três índices: coerência do gradiente e média e variância da intensidade. Após a classificação, o algoritmo realiza um pós-processamento morfológico para eliminar furos e regularizar a silhueta. Em [6] um método similar utilizando um classificador linear baseado em três índices (grau do cluster e média e variância de intensidade) é proposto. A principal diferença é que o primeiro segmenta por pixel e o segundo por bloco. Por realizar as operações por bloco, o segundo método é mais rápido do que o método apresentado em [5]. Apesar da boa qualidade dos dois métodos, há a necessidade constante de treinamento para cada base de imagens.

O pós-processamento morfológico é bastante utilizado por vários métodos a fim de eliminar pequenos ruídos, minimizar erros e diminuir a fragmentação da máscara. Por isto, o pós-processamento morfológico resulta em melhorias na segmentação. Este artigo propõe utilizar operadores morfológicos em todo o processo de segmentação e não apenas no pós-processamento. As duas principais etapas neste processo consistem em: localizar a estrutura crista-vale, através do operador de abertura top-hat e uniformizar a região de interesse, através de uma dilatação dos vales. Em [13] um algoritmo para binarização baseado em operações

morfológicas é apresentado, no entanto não são apresentados resultados experimentais e análise de desempenho.

Este artigo está organizado da seguinte maneira: na Seção 2 o método de segmentação morfológica é apresentado; na Seção 3 é feita a análise experimental e a comparação da segmentação proposta com outros métodos; finalmente na Seção 4 são apresentadas as conclusões.

2. Segmentação Morfológica

Esta nova proposta de segmentação é baseada na morfologia matemática. A morfologia matemática é bastante usada para correções em imagens e detecção de padrões [21]. No reconhecimento de impressões digitais a morfologia matemática é utilizada principalmente na esqueletização, sendo utilizada também em alguns estágios de pré-processamento ou pós-processamento.

A proposta apresentada se aproveita do padrão estriado da impressão digital no processo de segmentação. Este padrão possui uma alternância de áreas claras e escuras com frequência quase constante. Deste modo, um elemento estruturante plano com raio maior que a largura de uma crista é capaz de detectar este padrão. Para selecionar este padrão foi utilizado o operador de abertura top-hat. A abertura top-hat é definida como a subtração da imagem aberta da imagem original [21]: $A \hat{\ominus} B = A - (A \circ B)$, onde A é a imagem original, B o elemento estruturante e $(A \circ B)$ é a imagem A aberta pelo elemento estruturante B . Esta operação resulta na seleção de regiões que possuam regiões claras circundadas por regiões escuras. Além de selecionar as cristas, esta operação faz uma normalização da imagem. Isto possibilita a utilização de um threshold global na imagem resultante. Após a aplicação do threshold, a imagem é convertida em uma imagem binária, com os vales em branco e o restante da imagem em preto. O próximo passo consiste em expandir os vales em branco, através de uma operação de dilatação, criando uma zona uniforme e contínua. Após esta etapa, são aplicadas outras operações morfológicas para regularizar a região de interesse e do fundo, no caso: um fechamento de furos e uma abertura. As etapas desta proposta são mostradas na Figura 5.



Figura 5. Etapas do processo de segmentação morfológica: (a) Imagem de entrada; (b) Abertura Top-Hat; (c) Binarização; (d) Dilatação; (e) Fechamento de furos e abertura; e (f) Máscara sobreposta à imagem.

Este método por ser feito a nível do pixel apresenta precisão maior que os métodos feitos por blocos. Esta melhor precisão não acarreta em um grande custo no tempo de processamento. Alguns exemplos de segmentação de imagens da base FVC2002 são mostrados na Figura 6. Vários tipos de falhas que podem ocorrer na captura das impressões digitais são bem resolvidos pelo método proposto: diferenças de pressão, ruídos no fundo, etc.

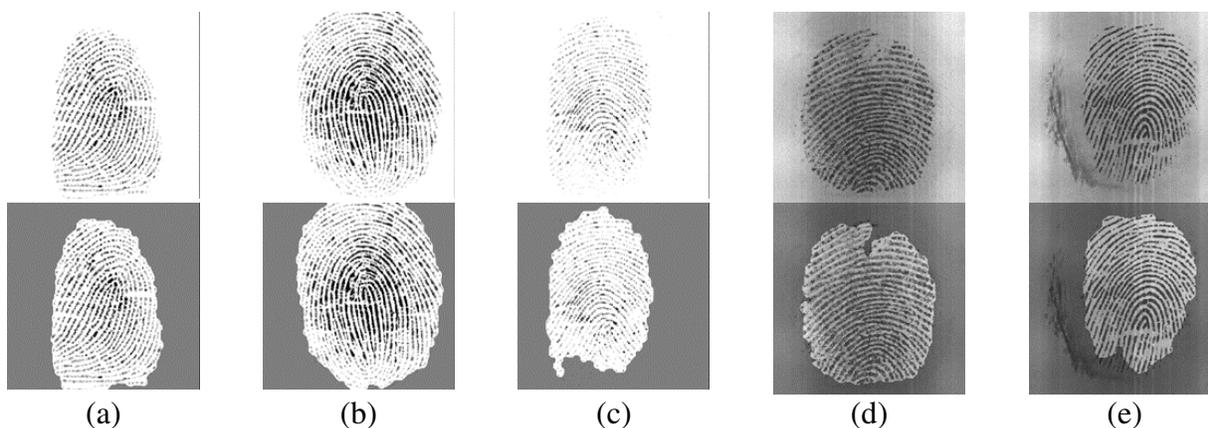


Figura 6. Exemplos de segmentação na base FVC2002: (a) 12_8 da Db1; (b) 16_8 da Db1; (c) 27_8 da Db1; (d) 33_7 da Db3; e (e) 15_1 da Db3.

3. Análise Experimental

Para os testes foi utilizada a base da FVC2002. A base de dados do FVC2002 consiste de quatro bancos diferentes de impressões digitais (Db1-Db4), cada um contendo imagens coletadas utilizando um sensor/tecnologia diferente. Cada banco de imagens é composto por 8 amostras de impressões digitais de cada um dos 100 dedos distintos. Nos experimentos, assim como na metodologia de comparação dos sistemas competidores do FVC, foram comparadas entre si as oito amostras do mesmo dedo e, também, foi comparada a primeira amostra de cada dedo com a primeira amostra dos outros dedos. Estas combinações resultam em 2800 comparações entre

diferentes amostras do mesmo dedo e 4950 comparações entre as primeiras amostras de dedos diferentes.

Para comparar a qualidade das segmentações propostas foram realizados testes tanto em um sistema baseado em minúcias quanto em um sistema baseado no FingerCode [15] utilizando diferentes métodos de segmentação. O uso dos sistemas para testes se deve ao fato das minúcias serem mais sensíveis a ruídos e menos sensíveis à segmentação. Uma região do fundo com pouco ruído, erroneamente segmentada, não altera muito o resultado. Além disto ocorre uma etapa no pós-processamento para eliminar as minúcias falsas. O sistema FingerCode apresenta comportamento inverso ao sistema de minúcias, é pouco sensível a ruídos e muito sensível à segmentação.

Todos os métodos testados sofreram um pós-processamento morfológico para uniformizar a área obtida a fim de melhorar o resultado obtido. Os métodos utilizados são brevemente descritos abaixo:

- Abertura Top-Hat: método proposto neste artigo, utiliza o operador morfológico de abertura top-hat para detectar a estrutura crista-vale.
- Contraste Direcional: neste método a imagem é dividida em blocos de 8x8. Para cada bloco são obtidos os valores acumulados resultante da soma das intensidades em 8 direções para cada um dos pontos do bloco. Após a obtenção dos 8 valores acumulados para cada uma das direções, o maior é selecionado e subtraído do valor na direção perpendicular.
- Desvio Padrão da Intensidade: neste método a imagem é dividida em blocos de 16x16. Para cada bloco é calculado o desvio padrão da intensidade.
- Transformada de Fourier: neste método a imagem é dividida em blocos de 8x8. Para cada bloco a região de 32x32 pixels centralizada no bloco é selecionada. A transformada de Fourier e a intensidade média desta região são obtidas. Caso o componente principal da região no domínio de Fourier seja de baixa frequência e a intensidade média seja alta, o bloco pertence ao fundo.

Nas Figuras Figura 7 e Figura 8 são mostradas a ROC (Receive Operating Curve) comparando o método proposto a outros métodos utilizados na segmentação de impressões digitais. No primeiro caso a comparação foi realizada utilizando um sistema baseado em minúcias. Na Tabela 5 as taxas de erros iguais (EER- Equal Error Rate) correspondentes à Figura 7 são exibidas. No segundo caso a comparação foi realizada utilizando um sistema baseado no FingerCode. Na Tabela 6 as taxas de erros iguais (EER) correspondentes à Figura 8 são exibidas.

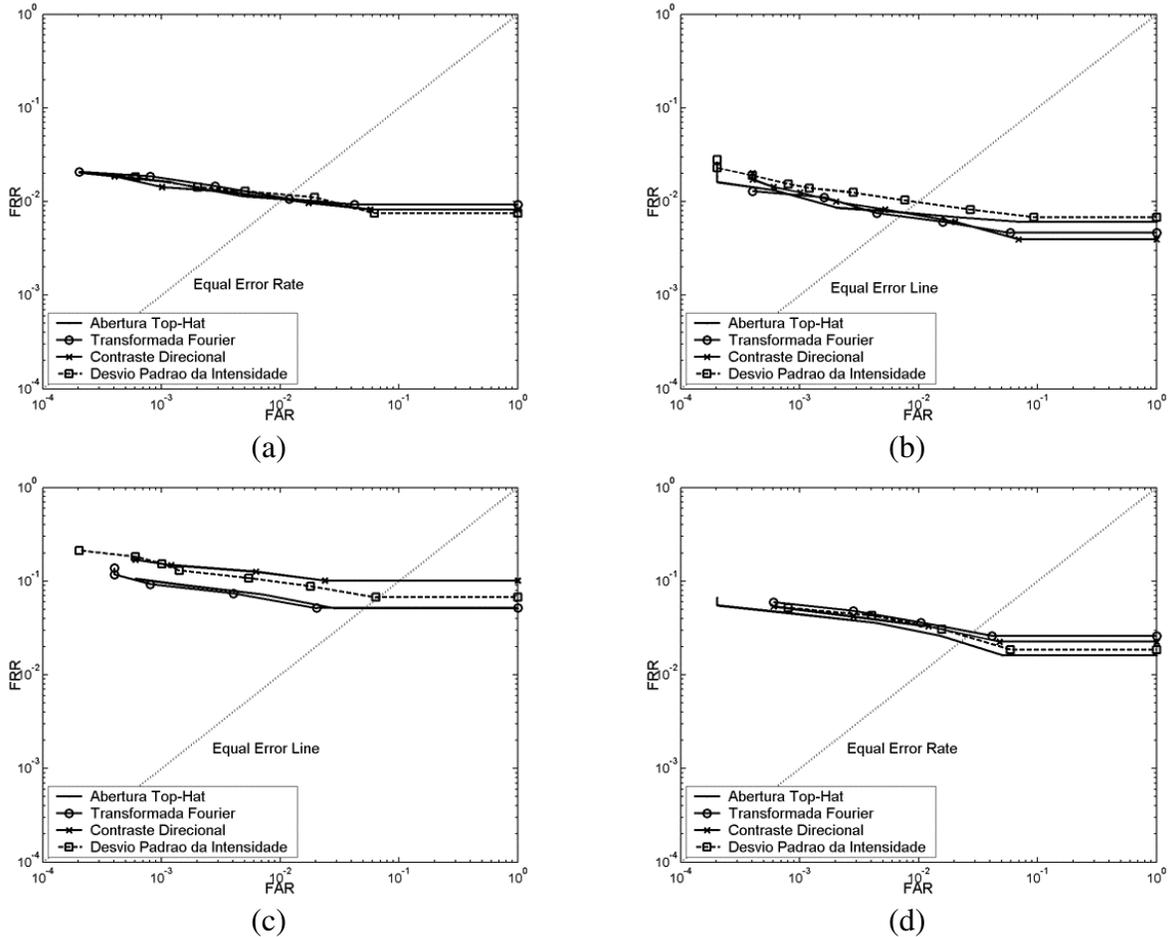


Figura 7. Curvas ROC utilizando Minúcias: (a) Db1; (b) Db2; (c) Db3; e (d) Db4.

Tabela 5. Equal Error Rate (EER) utilizando Minúcias.

	Db1	Db2	Db3	Db4	Média
Abertura Top-Hat	1.08%	0.77%	5.14%	2.37%	2.34%
Transformada Fourier	1.11%	0.72%	5.18%	2.99%	2.50%
Contraste Direcional	1.10%	0.78%	10.14%	2.82%	3.71%
Desvio Padrão da Intensidade	1.20%	1.01%	6.75%	2.75%	2.93%

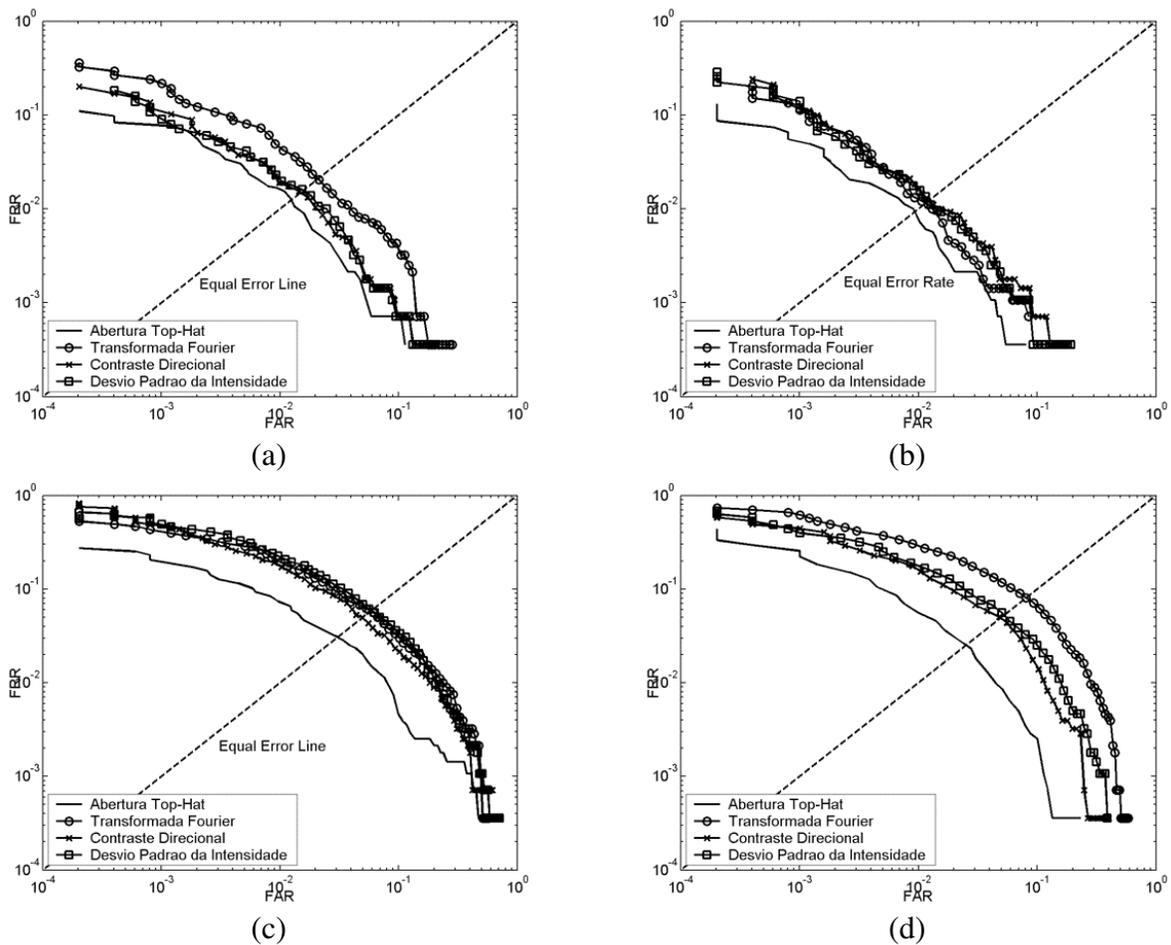


Figura 8. Curvas ROC utilizando FingerCode: (a) Db1; (b) Db2; (c) Db3; e (d) Db4.

Tabela 6. Equal Error Rate (EER) utilizando FingerCode.

	Db1	Db2	Db3	Db4	Média
Abertura Top-Hat	1.26%	0.95%	3.04%	2.50%	1.94%
Transformada Fourier	2.08%	1.13%	5.93%	8.05%	4.30%
Contraste Direcional	1.48%	1.23%	4.97%	4.96%	3.16%
Desvio Padrão da Intensidade	1.56%	1.24%	5.86%	5.25%	3.48%

4. Conclusão

Este artigo apresenta uma nova proposta de segmentação. No processo são utilizados apenas operadores morfológicos. Estes operadores detectam o padrão característico das impressões digitais. Este padrão se constitui na alternância de regiões claras e escuras. Além de detectar a região de interesse o operador top-hat possui a propriedade de normalizar a imagem.

Isto possibilita realizar uma binarização da imagem utilizando um threshold global. Finalmente, a região de interesse é obtida dilatando as áreas brancas, correspondentes aos vales. Após a dilatação é realizado um pós-processamento para eliminar pequenos fragmentos tanto na área de interesse quanto no fundo. Para remover estas pequenas falhas um fechamento de furos e uma abertura.

Através da análise das Figuras Figura 7 e Figura 8 e das Tabelas Tabela 5 e Tabela 6, fica mostrado a boa qualidade da segmentação com abertura top-hat comparada a outros métodos. O ganho médio no método baseado no FingerCode foi em torno de 40%. No sistema baseado em minúcias o ganho é menor. O ganho médio foi de 6,5%. Este menor ganho é devido principalmente à eliminação de minúcias falsas.

O método baseado em morfologia matemática é um método atraente, devido aos seguintes fatores: apresenta uma boa qualidade da segmentação, possui precisão maior que os métodos por blocos e é mais simples e mais rápido que a maioria dos métodos por pixel.

Bibliografia

- [1] F. A. Afsar, M. Arif e M. Hussain, “Fingerprint Identification and Verification System Using Minutiae Matching”, National Conf. on Emerging Technologies (NCET), pp. 141-146, 2004.
- [2] M. Ballan, F. A. Sakarya, e B. L. Evans, “A Fingerprint Classification Technique Using Directional Images”, 31th Asilomar Conf. on Signals, Systems and Computers, vol. 1, pp. 101-104, 1997.
- [3] A. M. Bazen e S. Gerez, “Directional Field Computation for Fingerprints Based on the Principal Component Analysis of Local Gradients”, Proc. Workshop on Circuits, Systems and Signal Processing (ProRISC2000), pp. 1-7, 2000.
- [4] A. M. Bazen e S. H. Gerez, “Segmentation of Fingerprint Images”, Proc. Workshop on Circuits, Systems and Signal Processing (ProRISC2001), pp. 276-280, 2001.
- [5] S. Bernard, N. Boujemaa, D. Vitale e C. Bricot, “Fingerprint Segmentation Using the Phase of Multiscale Gabor Wavelets”, Proc. Asian Conference on Computer Vision (ACCV), 2002.

- [6] X. Chen, J. Tian., J. Cheng e X. Yang, “Segmentation of Fingerprint Images Using Linear Classifier”, *EURASIP Journal on Applied Signal Processing*, pp. 480-494, 2004.
- [7] S. Chikkerur, C. Wu e V. Govindaraju, “A Systematic Approach for Feature Extraction in Fingerprint Images”, *First Int. Conf. on Biometric Authentication (ICBA)*, pp. 344-350, 2004.
- [8] E. R. Dougherty e R. A. Lotufo, “Hands-On Morphological Image Processing”, *SPIE-International Society for Optical Engine*, 2003.
- [9] F. A. Fernadez, J. F. Aguilar e J. O. Garcia, “An Enhanced Gabor Filter-Based Segmentation Algorithm for Fingerprint Recognition Systems”, *Int. Symposium on Image and Signal Processing and Analysis*, pp. 239-244, 2005.
- [10] B. Lee, J. Moon e H. Kim, “A Novel Measure of Fingerprint Image Quality Using Fourier Spectrum”, *Proc. of SPIE Biometric Technology for Human Identification II*, Vol. 5779, 2005.
- [11] D. Maio e D. Maltoni, “Direct Gray-Scale Minutiae Detection in Fingerprints”, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 19, no. 1, pp. 27-40, 1997.
- [12] D. Maltoni, D. Maio, A. K. Jain e S. Prabhakar, “Handbook of Fingerprint Recognition”, Springer, 2003.
- [13] V. Morosov, “Using Tophat Transformation For Image Fingerprint Segmentation”, *Int. Conf. on Signal and Electronic Systems (ICSES2001)*, pp. 241-246, 2001.
- [14] N. K. Ratha, S. Chen e A. K. Jain, “Adaptative Flow Orientation-Based Feature Extraction in Fingerprint Images”, *Pattern Recognition*, vol. 28, no. 11, pp. 1657-1672, 1995.
- [15] A. Ross, A. Jain e J. Reisman, “A Hybrid Fingerprint Matcher”, *Pattern Recognition* vol. 36, pp. 1661-1673, 2003.
- [16] L. Shen, A. Kot e W. Koo, “Quality Measures of Fingerprint Images”, *Proc. Int. Conf. on Audio- and Video-Based Biometric Person Authentication (AVBPA)*, pp. 266-271, 2001.

Capítulo 3

3 Filtro Gabor

O filtro Gabor é composto de uma onda sinusoidal em determinada frequência e orientação modulada por um envelope gaussiano. Estudos sugerem que o córtex visual realiza uma decomposição similar ao filtro Gabor. Devido a estas propriedades o filtro Gabor foi utilizado para a extração dos componentes orientados das impressões digitais. Para a extração foi utilizado um banco de 8 filtros Gabor nas direções $\theta = \{0^\circ; 22,5^\circ; 45^\circ; 67,5^\circ; 90^\circ; 112,5^\circ; 135^\circ; 157,5^\circ\}$ em uma frequência pré-determinada. A frequência escolhida corresponde a 1,5 da largura média da estrutura cristálica. No caso de imagens a 500dpi isto resulta em uma frequência de $f = 1/10$. O desvio padrão do envelope gaussiano indica o detalhamento da informação a ser capturada. Quanto maior for o desvio padrão mais robusto contra ruídos é o filtro, no entanto a informação é menos detalhada. Experimentalmente o valor escolhido foi $\sigma = 4.1$.

3.1 Filtro Gabor 2D Separável com Envelope Gaussiano Circular

Artigo submetido para IEEE Image Processing.

Nesta seção é apresentada uma nova proposta de implementação rápida da filtragem pelo filtro Gabor, bastante utilizado para a extração de texturas [30], [66] e [90]. Esta melhoria na velocidade ganha mais relevância quando considerado o caso da utilização de um banco de filtros Gabor o que ocorre na maioria dos métodos [68] e [78]. O banco orientado de filtros Gabor é utilizado para a extração dos componentes orientados da imagem em uma determinada frequência, processo realizado para a extração dos atributos de textura pelo método FingerCode. Para possibilitar esta implementação rápida é necessário que o filtro Gabor obedeça a condição de que o envelope gaussiano seja circular. No caso do método FingerCode e de outros casos de extração de textura esta condição é obedecida.

Artigo 2. Separable 2D Gabor Filter with a Circular Gaussian Envelope

Gustavo de Sá, and Roberto Lotufo

School of Electrical and Computer Engineering, University of Campinas (UNICAMP),

C.P. 6101, 13083-970, Campinas, SP, Brazil

gsa@dca.fee.unicamp.br, and www.dca.fee.unicamp.br/~lotufo

Abstract

The 2D Gabor filter is used in a variety of image applications on account of its directional and frequency selective properties. Usually, a bank of Gabor filters is applied to an image to extract the oriented components. This requires multiple 2D filtering, which is computationally very expensive. Here we show that if the Gaussian envelope is circular (i.e., the standard deviations of the envelope on the x and y -axes are equal), the 2D Gabor filter can be separated into two orthogonal 1D Gabor filters, one on the x -axis and the other on the y -axis, in any direction. Through calculations of the number of complex MADD operations required for the separated and nonseparated filters, and a simple illustrative example, we confirm that the 2D filtering by two 1D Gabor filters is faster than using a single 2D Gabor filter.

1. Introduction

Filters are used in many signal and image processing applications. The widespread use of filters has prompted the inclusion of fast convolution and Fourier transform algorithms into many processors and programs. In image processing, however, there are still many cases where filtering is one of the most computationally expensive stages. Thus, methods are needed to improve the filtering performance for particular cases and specific filters. One such case is the Gabor filter, which is used for image enhancement [1]-[2], extraction of contours [3], pattern recognition [4], texture segmentation [5], image compression [6], and other tasks.

Usually, an oriented Gabor filter bank is applied to an image in order to obtain the space oriented image components in a specific frequency range. Various approaches can be taken to improve the performance of the Gabor filter. Areekul et al. [7] separated the Gabor filter into one

1D band-pass and one 1D low-pass filter. In this implementation, the convolution is performed toward direction θ with the low-pass filter and toward direction $\theta+90^\circ$ with the band-pass filter. The pixel sequence needs to be defined and interpolated to form a new image along the selected orientation. An alternative approach to Gabor filtering was taken by Young et al. [8], who developed a method consisting of a recursive forward function followed by a recursive backward function. Although their method is one of the fastest implementations available ($O(N^2)$ complex operations), it is an approximation of the original Gabor filter, with the difference increasing towards the borders.

The 2D Gabor function can only be separated when the direction θ is equal to 0° or 90° . This paper presents the decomposition of a 2D Gabor filter into two orthogonal 1D Gabor filters in any direction. When the standard deviations of the Gaussian kernel on the x - and y -axes are equal (i.e., $\sigma_x = \sigma_y$), the envelope of the Gabor function is circular. This symmetry permits the decomposition of the 2D Gabor function into two 1D Gabor functions, one on the x -axis and the other on the y -axis. This separation results in faster filtering. Although this is a simple solution, we could not find it in the literature or in existing implementations. This paper is organized as follows: in Section 2 the Gabor filter is presented; the decomposition of a 2D Gabor filter with a circular Gaussian envelope into two 1D Gabor filters is demonstrated in Section 3; the number of complex operations for the separated and nonseparated 2D Gabor filter implementations are compared in section 4; and finally, conclusions are presented in Section 5.

2. Gabor Filter

In 1946, Gabor showed that the effective width of a function in direct space multiplied by the effective width of the same function transformed into Fourier space has a lower limit. He also proved that the family of Gabor filters is defined by a complex sinusoidal function modulated by a Gaussian envelope. Gabor filters are similar to Fourier filters; however, they are limited to certain frequency bands [9]. The 1D Gabor function is defined as:

$$G(t, \sigma, \omega_0) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{t^2}{2\sigma^2}} e^{j\omega_0 t} \quad (1)$$

where $\omega_0 = 2\pi f_0$, f_0 is the frequency of the sinusoidal wave, and σ is the standard deviation of the Gaussian envelope. Figure 1 shows the real and imaginary parts from the complex 1D Gabor filter.

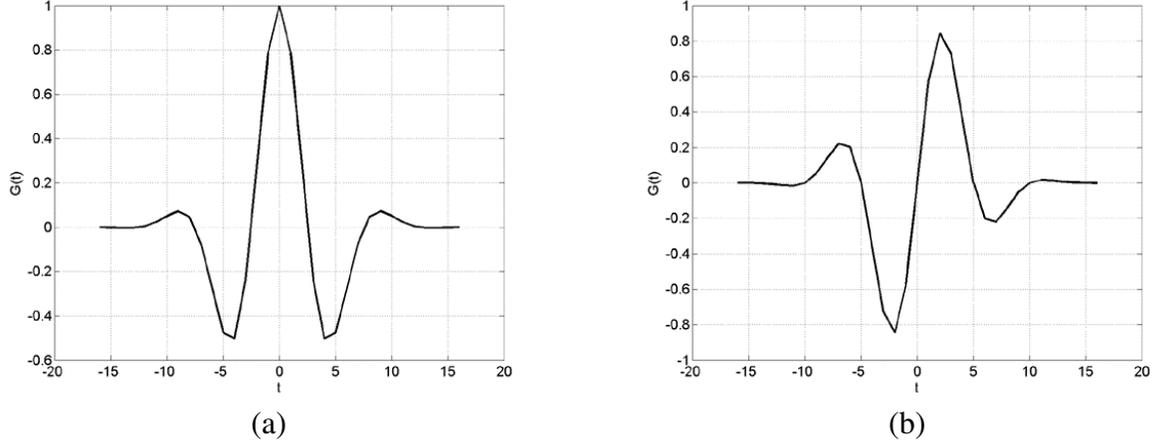


Figure 1. 1D Gabor Filter ($\sigma = 4$ and $\omega_0 = 20\pi$): (a) Real part; (b) Imaginary part.

Studies in biology have suggested that the primary visual cortex performs a similar oriented and Fourier space decomposition. Daugman [10] used 2D Gabor filters to explain the orientation-, space-, and frequency-selective properties of neurons in the visual cortical receptive field, and subsequently demonstrated the mathematical function [11]. The 2D Gabor function is a harmonic oscillator, composed of a sinusoidal plane wave of a particular frequency and orientation modulated by a Gaussian envelope. The 2D Gabor function is defined as:

$$G(x, y, \sigma_x, \sigma_y, \omega_0, \theta) = \frac{1}{2\pi\sigma_x\sigma_y} e^{-\frac{1}{2}\left(\frac{x_\theta^2}{\sigma_x^2} + \frac{y_\theta^2}{\sigma_y^2}\right)} e^{j(\omega_x x + \omega_y y)} \quad (2)$$

where

$$\omega_x = \omega_0 \cos \theta \quad (3)$$

$$\omega_y = \omega_0 \sin \theta$$

and

$$x_\theta = x \cos \theta + y \sin \theta \quad (4)$$

$$y_\theta = -x \sin \theta + y \cos \theta$$

here, σ_x and σ_y are the standard deviations of the Gaussian envelope on the x - and y -axes, respectively, and θ is the filter orientation. Figure 2 shows the real and imaginary parts of the complex 2D Gabor filter.

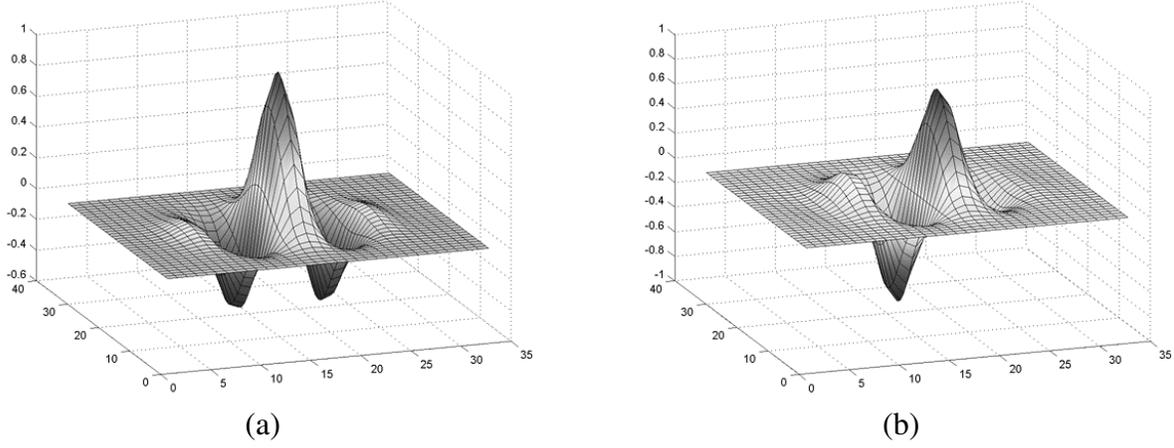


Figure 2. 2D Gabor Filter ($\theta = 135^\circ$, $\sigma_x = \sigma_y = 4$ and $\omega_0 = 20\pi$): (a) Real part; (b) Imaginary part.

3. Separable Gabor Filter with a Circular Gaussian Envelope

The standard deviations on the x - and y -axes of the Gabor filter can be represented by the following relation:

$$\sigma = \sigma_x = \frac{1}{\lambda} \sigma_y \quad (5)$$

Substituting (4) and (5) into the 2D Gabor filter equation (2) yields:

$$G_S(x, y, \sigma, \omega_0, \theta) = \frac{1}{2\pi\lambda\sigma^2} e^{-\frac{1}{2} \left(\frac{\lambda(x\cos\theta + y\sin\theta)^2 + (-x\sin\theta + y\cos\theta)^2}{\sigma^2} \right)} e^{j(\omega_x x + \omega_y y)} \quad (6)$$

and then expanding the terms in (6) results in:

$$G_S(x, y, \sigma, \omega_0, \theta) = \frac{1}{2\pi\lambda\sigma^2} e^{-\frac{1}{2} \left(\frac{x^2(\lambda\cos^2\theta + \sin^2\theta) + 2xy(\lambda\cos\theta\sin\theta - \sin\theta\cos\theta) + y^2(\lambda\sin^2\theta + \cos^2\theta)}{\sigma^2} \right)} e^{j(\omega_x x + \omega_y y)} \quad (7)$$

Because of the term $2xy(\lambda\cos\theta\sin\theta - \sin\theta\cos\theta)$, the Gabor function cannot be separated into two independent functions on the x - and y -axes, except when $\theta = 0^\circ$ or 90° .

There are many applications where the Gaussian envelope is circular. In those cases, $\sigma_x = \sigma_y$ (i.e., $\lambda = 1$), and hence Eq. (7) becomes:

$$G_S(x, y, \sigma, \omega_0, \theta) = \frac{1}{2\pi\sigma^2} e^{-\frac{1}{2} \left(\frac{x^2(\cos^2\theta + \sin^2\theta) + 2xy(\cos\theta\sin\theta - \sin\theta\cos\theta) + y^2(\sin^2\theta + \cos^2\theta)}{\sigma^2} \right)} e^{j(\omega_x x + \omega_y y)} \quad (8)$$

Equation (8) can be simplified to:

$$G_S(x, y, \sigma, \omega_0, \theta) = \frac{1}{2\pi\sigma^2} e^{-\frac{1}{2}\left(\frac{x^2+y^2}{\sigma^2}\right)} e^{j(\omega_x x + \omega_y y)} \quad (9)$$

which can be rewritten as:

$$G_S(x, y, \sigma, \omega_0, \theta) = \underbrace{\frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{x^2}{2\sigma^2}} e^{j\omega_x x}}_{\text{1D Gabor Filter in the } x \text{ axis}} \underbrace{\frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{y^2}{2\sigma^2}} e^{j\omega_y y}}_{\text{1D Gabor Filter in the } y \text{ axis}} \quad (10)$$

Thus, for cases with a circular Gaussian envelope, the 2D Gabor filter can be separated into 1D Gabor filters in the x - and y -axes, allowing filtering first on one axis and then the other. The advantage of performing the 2D filtering using two 1D filters instead of one single 2D filter lies in the sizes of the filters [13]: For a filter of size $M \times M$, the nonseparated 2D filter has M^2 complex points, whereas the separated 2D filter has $2M$ complex points.

4. Comparisons

If a 2D Gabor filter, G , of size $M \times M$ is applied to an image, I , of size $N \times N$, the output, O , will have a size of $W \times W$, where $W = M + N - 1$. The corresponding filtering in the frequency domain using the discrete Fourier transform (DFT) also gives an output of size $W \times W$.

The number of complex multiply-and-add (MADD) operations to filter the image I by the filter G using a 2D convolution to the nonseparated filter is given by:

$$M^2 W^2 \quad (11)$$

If, by contrast, the filter G is separable, the 2D convolution can be done using 1D convolution on the x -axis followed by 1D convolution on the y -axis. First, for each column along the x -axis of I , a convolution is performed using the 1D x -axis Gabor filter. Then, for each row on the y -axis of the resulting output, a convolution is performed using the 1D y -axis Gabor filter. The number of complex MADD operations is given by 1D convolution on W columns and then on W rows:

$$MW \times W + MW \times W = 2MW^2 \quad (12)$$

By comparison, to perform the filtering using the FFT approach when the filter is not separated, it is necessary to carry out one 2D transform of the Gabor filter with size W^2 , one 2D transform of the image with size W^2 , one multiplication point-to-point of the Gabor filter by the image in the frequency domain, and one 2D inverse transform of the multiplication product of size W^2 . Hence, the number of complex MADD operations is:

$$\begin{aligned} & W^2 \log_2 W^2 + W^2 \log_2 W^2 + W^2 + W^2 \log_2 W^2 \\ & = 3W^2 \log_2 W^2 + W^2 \end{aligned} \quad (13)$$

In the case of the separated 2D Gabor filter using FFT, the operations on the x -axis are performed first: one 1D transform of size W for the x -axis Gabor filter, one 1D transform of size W for each column of the image, one multiplication point-to-point of size W for each column of the image using the x -axis Gabor filter in the frequency domain, and one 1D inverse transform of each column output product. Then, the same process is applied on the y -axis for each row. The number of complex MADD operations in this case is:

$$\begin{aligned} & 2 \times (W \log_2 W + W \times (W \log_2 W + W + W \log_2 W)) \\ & = 2W^2 \log_2 W^2 + W \log_2 W^2 + 2W^2 \end{aligned} \quad (14)$$

Comparing the nonseparated and separated filters, we find that for the convolution case, when $M > 2$, $M^2 W^2 > 2MW^2$, and for the Fourier transform case, when $W > 2$, $3W^2 \log_2 W^2 + W^2 > 2W^2 \log_2 W^2 + W \log_2 W^2 + 2W^2$. Thus, the number of complex operations in the separated filtering is smaller than that in the nonseparated filtering in almost all cases.

To illustrate the improvement in performance, we performed a test using Matlab 6.5 and its functions `conv`, `conv2`, `fft`, `fft2`, `ifft`, and `ifft2` from the `datafun` library. The computer used had an AMD 2.1 Mhz processor with 256 MB of RAM, with Microsoft Windows XP as the operating system. The average processing times for the 2D Gabor filter with a circular Gaussian envelope are listed in Table 1. Four methods were used: OriCON2D, convolution with a nonseparated filter; SepCON2D, convolution with a separated filter; OriFFT2D, Fourier transform with a nonseparated filter; and SepFFT2D, Fourier transform with a separated filter. These methods were applied to images of size $N=32$ to 1024 using Gabor filters of size $M=17$ to 129.

Table 1. Average Processing Time

METHOD	FILTER SIZE (M)	IMAGE SIZE (N)					
		32	64	128	256	512	1024
OriCON2D	17	2.3	8.0	31.2	127.7	512.6	2028.1
SepCON2D	17	5.0	10.0	26.8	101.6	367.1	1387.7
OriFFT2D	17	1.3	3.0	10.4	115.0	304.8	1156.2
SepFFT2D	17	1.9	4.1	13.8	84.7	207.5	942.0
OriCON2D	33	-	24.9	96.7	388.0	1599.9	6245.3
SepCON2D	33	-	13.4	36.1	125.8	442.4	1642.2
OriFFT2D	33	-	5.2	14.1	71.6	517.1	1595.4
SepFFT2D	33	-	5.3	15.8	52.7	323.6	1085.8
OriCON2D	65	-	-	380.5	1529.8	6081.0	24481.1
SepCON2D	65	-	-	55.0	178.0	609.6	2254.8
OriFFT2D	65	-	-	28.3	92.2	504.7	2937.3
SepFFT2D	65	-	-	19.7	54.8	274.8	1676.6
OriCON2D	129	-	-	-	7953.1	31959.5	127406.3
SepCON2D	129	-	-	-	315.5	1023.2	3665.7
OriFFT2D	129	-	-	-	250.0	693.9	3298.5
SepFFT2D	129	-	-	-	101.7	384.5	1412.5

5. Conclusions

In this paper we have shown that if the Gaussian envelope is circular, the 2D Gabor filter can be separated into two orthogonal 1D filters in any direction. This separation will improve the performance of filters based either on convolution or Fourier transform. The degree of improvement was demonstrated by comparing the number of complex MADD operations required for the separated and nonseparated filters. Although the recursive Gabor filtering [8] entails a smaller number of complex operations than the separated filtering described here, the former method is an approximate approach, whereas the separated Gabor filtering gives outputs identical to those of the original 2D Gabor filtering. This simple solution can be easily implemented in any application that uses the Gabor filter. Moreover, it can take advantage of the

optimized convolution and FFT implementations included in many processors and compilers. The improvement obtained by separating the Gabor filter was confirmed by comparing the process time for both separated and nonseparated filtering across a range of image and filter sizes. The results showed that the larger the filter or image size, the greater the process time gain for the separated compared to the nonseparated filter.

Extracting many oriented components from an image is an expensive process because it entails filtering of the image by a bank of Gabor filters in many directions. Separating a 2D filter into two 1D filters on the x - and y -axes is a well-known method to reduce the image filtering process time. Hence the ability to separate the 2D Gabor filter into two orthogonal 1D filters in any direction is very beneficial, because the gain obtained in a single filtering is multiplied by the number of directions of the Gabor filter bank.

References

- [1] G. Cristobal and R. Navarro, "Space and Frequency Variant Image Enhancement Based on a Gabor Representation", *Patt. Recog. Lett.*, 15, 273-277, 1994.
- [2] J. Yang, L. Liu, T. Jiang, and Y. Fan, "A Modified Gabor Filter Design Method for Fingerprint Image Enhancement", *Pattern Recognition Letters* 24, 1805-1817, 2003.
- [3] C. Grigorescu, N. Petkov, and M. A. Westenberg, "Contour Detection Based on Nonclassical Receptive Field Inhibition", *IEEE Transactions on Image Processing* 12, 729-739, 2003.
- [4] B. S. Manjunath, C. Shekhar, and R. Chellappa, "A New Approach to Image Feature Detection with Applications", *Pattern Recognition* 29, 627-640, 1996.
- [5] A. K. Jain, and F. Farrokhnia, "Unsupervised Texture Segmentation Using Gabor Filters", *Pattern Recognition* 24, 1167-1186, 1991.
- [6] J. G. Daugman, "Complete Discrete 2-D Gabor Transforms by Neural Networks for Image Analysis and Compression", *IEEE Trans. Acoust., Speech, Signal Processing* 36, 1169-1179, 1998.
- [7] V. Areekul, U. Watchareeruetai, K. Suppasriwasuseth, and S. Tantaratana, "Separable Gabor Filter Realization for Fast Fingerprint Enhancement", *IEEE International Conference on Image Processing (ICIP)*, 253-256, 2005.

- [8] I. T. Young, L. J. van Vliet, and M. van Ginkel, "Recursive Gabor Filtering", *IEEE Transactions on Signal Processing* 50, 2798-2805, 2002.
- [9] D. Gabor, "Theory of communication", *Journal of the Institute for Electrical Engineers* 93, 429-439, 1946.
- [10] J. G. Daugman, "Two-Dimensional Spectral Analysis of Cortical Receptive Field Profiles", *Vision Research* 20, 847-856, 1980.
- [11] J. G. Daugman, "Uncertainty Relation for Resolution in Space, Spatial Frequency, and Orientation Optimized by Two Dimensional Visual Cortical Filters", *J. Optical Soc. Amer.* 2 (7), 1160-1169, 1985.
- [12] A. V. Oppenheim, R. W. Schaffer, and J. R. Buck, "Discrete-Time Signal Processing", Prentice Hall, 1999.
- [13] J. S. Lim, "Two-Dimensional Signal and Image Processing", Prentice-Hall, 1990
- [14] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, "Handbook of Fingerprint Recognition", Springer, 2003.

Capítulo 4

4 Novas Funções de Escore

Após a extração dos componentes orientados pelo banco de filtros Gabor, o vetor atributo de textura de cada impressão digital é calculado. A diferença absoluta ponto a ponto entre os dois vetores atributos resulta no vetor diferença. Neste capítulo são apresentadas novas maneiras de se calcular o escore. Através de análise experimental foram testadas várias funções de escore utilizando diversas metodologias: substituição direta dos valores de atributos através de diversos tipos de funções; ponderação estatística das características espaciais dos valores de atributos; e análise dos histogramas de distribuição dos valores de atributo. Os melhores resultados obtidos são apresentados nas seções seguintes.

4.1 Função Escore do FingerCode Melhorado

Artigo aceito no Simpósio Brasileiro de Computação Gráfica e Processamento de Imagens (SIBGRAPI-2006).

O uso de funções mais elaboradas no cálculo do escore pode resultar em uma maior robustez no reconhecimento de impressões digitais. Para obter esta melhoria várias funções foram analisadas e testadas. Dois tipos de métodos foram utilizados nos testes: a substituição dos valores do vetor de diferença através de funções não-lineares e a aplicação de peso aos valores do vetor diferença, obtido através da análise estatística. No primeiro caso os melhores resultados foram obtidos com funções em que na faixa de valores do vetor diferença possui uma forma semelhante à função exponencial. Isto indica que os pontos de maior diferença entre os vetores atributos de textura são mais discriminantes do que os pontos que se assemelham.

No segundo caso foram utilizadas duas formas de se analisar estatisticamente o ponto: utilizando a região vizinha ao ponto analisado em um mesmo componente orientado e utilizando

os pontos na mesma posição do ponto analisado em todos os componentes orientados. Nas duas situações os melhores resultados foram obtidos com o desvio padrão dos pontos no vetor diferença. Quanto maior o desvio padrão mais discriminante é a região analisada. Um alto desvio padrão na vizinhança de ponto de um mesmo componente orientado indica que em alguns pontos desta região uma das impressões digitais possui cristas na direção do componente orientado enquanto que a outra impressão digital possui cristas em outra direção. Um alto desvio padrão dos pontos de todos os componentes orientados em uma mesma posição indica que no ponto analisado as impressões digitais possuem cristas em direções diferentes.

Artigo 3. Improved FingerCode Matching Function

Gustavo de Sá, and Roberto Lotufo

School of Electrical and Computer Engineering, University of Campinas (UNICAMP),

C.P. 6101, 13083-970, Campinas, SP, Brazil

gsa@dca.fee.unicamp.br, and www.dca.fee.unicamp.br/~lotufo

Abstract

FingerCode is a fingerprint correlation matching scheme that relies on texture information. In this scheme, the oriented components are extracted from a fingerprint image using a bank of Gabor filters, and a directional texture feature vector is computed for each oriented component. The feature vectors from the input and template images are compared and a matching score is obtained. Here we explore ways to improve the matching score for the FingerCode method by using more complex matching functions. The best results were obtained by applying a nonlinear function to the texture values and weighting the texture vectors based on the spatial distribution.

1. Introduction

The use of biometric systems, in particular fingerprint-based systems, is increasing. Because fingerprints are unique, permanent, and easy to collect identifiers of individuals, systems based on fingerprint patterns show good performance and reliability [1]. These characteristics, which underpin the push towards exploiting fingerprint systems in civilian applications, are the same as have been used for more than a century in forensic applications.

A fingerprint is the pattern of alternating ridges and valleys on the finger surface. At the global level this pattern has distinct shapes and characteristics called singularities, such as high curvature and predominant orientation. At the local level, anomalies in the pattern, called minutiae, can be found; examples of minutiae are terminations and bifurcations. Fingerprint systems can be broadly categorized into correlation-based techniques [2-4] and minutiae-based techniques [5-6] depending on whether they consider singularities or minutiae respectively. The majority of fingerprint applications use minutiae in the matching process, while the correlation-

based technique is mostly used to classify fingerprints [7-9]. Each of these techniques does not account for information on the other level: minutiae based methods do not utilize the full information from the global level and correlation based methods do not use the detailed local level information. Hence, systems that combine the two techniques could potentially be of great utility [9].

Based on a model for iris recognition [11-13], Jain et al. [14] developed a texture-based method for fingerprint recognition called FingerCode. In their method, the pair of fingerprints under consideration is first aligned according to their core points, then oriented texture vector sets are extracted by a bank of Gabor filters, and finally the Euclidean distance of the oriented texture vector sets is computed. Subsequently, Jain et al. [15] proposed a hybrid FingerCode method that combines minutiae and texture (i.e. local and global information). In this variation, the minutiae points are used to align the fingerprints and the texture vector sets are used to obtain the final match score [16].

Motivated by our perception from past work that certain aspects of fingerprints have greater discriminating power than others, in the present study we explored the use of more complex FingerCode matching functions. The best results were obtained using two types of function: a nonlinear function (e.g., an exponential) as a substitute for the texture values; and applying weights to the texture vectors using statistical measures extracted from the set of texture vectors (e.g., mean and standard deviation) based on the spatial distribution. The statistical measures were obtained in two ways: from an area around a cell in a specific oriented component, or from a cell in the same position in all oriented components.

In our experiments, we use the hybrid FingerCode of Ross et al. [16] because it is more accurate than the FingerCode method alone. This paper is organized as follows: Section 2 presents the FingerCode method and its hybrid variation; the alternative matching score functions are described in Section 3; the experimental results obtained using the new matching score functions are given in Section 4; and finally, our conclusions are presented in Section 5.

2. FingerCode

The FingerCode method is a correlation based technique, where a small circular area around the core point is tessellated in an arc fashion and filtered by an oriented Gabor filter bank. For each cell in the oriented components, a value is computed that constitutes the texture feature vector. Two fingerprints are aligned by their core points and their texture vectors are compared, generating a score that indicates whether they belong to the same finger. The FingerCode hybrid variation differs from the FingerCode method in that it uses the entire fingerprint area, which is tessellated in a square rather than an arc fashion, and, most importantly, the minutia points are used to align the fingerprints.

In the hybrid variation, the fingerprint image is tessellated in rectangular cells. The block size used in the present work was 16x16 pixels, which corresponds to approximately 1.5 times the width of the ridge-valley structure in a 500 dpi image. Then, the fingerprint is separated from the background in the segmentation process, while avoiding extraction of features from invalid cells (cells in which one or more pixels belong to the background). In this process, a morphological scheme is used to establish the fingerprint mask. To reduce intensity variations due to differences in finger pressure, the fingerprint image is normalized using the approach presented in [17].

After extracting the minutiae points from the input image, the rotation and translation parameters are generated that give the best correspondence with the set of minutiae points from the template image. The aligned image is filtered by an oriented Gabor filter bank. The Gabor filter is frequency- and orientation-selective and consists of a sinusoidal wave modulated by a Gaussian envelope. The bank is formed by eight Gabor filters in directions $\Theta = \{0^\circ, 22.5^\circ, 45^\circ, 67.5^\circ, 90^\circ, 112.5^\circ, 135^\circ, 157.5^\circ\}$. The chosen frequency ($f_0 = 0.1 \text{ pixel}^{-1}$) was based on the mean inter-ridge distance of approximately 10 pixels for a 500 dpi image. The real part of the Gabor filter is given by Equation (1):

$$G(x, y) = \exp\left(-\frac{1}{2}\left(\frac{x_\theta^2}{\sigma_x^2} + \frac{y_\theta^2}{\sigma_y^2}\right)\right) \cos(\omega_0 x_\theta) \quad (1)$$

$$x_\theta = x \sin \theta + y \cos \theta$$

$$y_\theta = x \cos \theta - y \sin \theta$$

where σ_x and σ_y are the standard deviations of the Gaussian envelope along the x- and y-axes respectively, θ is the direction of the filter, and $\omega_0 = 2\pi f_0$. The filtering extracts eight oriented image components in each direction. For each cell, C , the feature value, $F_{\rho\theta}$, is computed:

$$F_{\rho\theta} = \sqrt{\frac{1}{K_{\rho\theta}} \sum_{(x,y) \in C_{\rho\theta}} (C_{\rho\theta}(x,y) - \bar{C}_{\rho\theta})^2} \quad (2)$$

$$\bar{C}_{\rho\theta} = \frac{1}{K_{\rho\theta}} \sum_{(x,y) \in C_{\rho\theta}} C_{\rho\theta}(x,y)$$

where $C_{\rho\theta}$ is the cell at position ρ in the oriented component θ , $C_{\rho\theta}(x,y)$ is the pixel value at position (x,y) in cell $C_{\rho\theta}$, $K_{\rho\theta}$ is the number of pixels from cell $C_{\rho\theta}$, and $\bar{C}_{\rho\theta}$ is the mean value of the pixels from cell $C_{\rho\theta}$. FingerCode is the set of eight feature vectors composed of all cell values, $F_{\rho\theta}$. The difference vector, E , is formed by the absolute difference between corresponding cells from the input, F , and template, T , FingerCode feature vectors:

$$E_{\rho\theta} = |F_{\rho\theta} - T_{\rho\theta}| \quad (3)$$

where $E_{\rho\theta}$, $F_{\rho\theta}$ and $T_{\rho\theta}$ are the cell values at position ρ in direction θ from the difference, input and template feature vectors, respectively. The matching score S is expressed as the sum of all differences normalized by the number of valid cells, V , given by the intersection of the two fingerprint masks:

$$S = \frac{1}{V} \sum_{\rho \in P} \sum_{\theta \in \Theta} V_{\rho\theta} E_{\rho\theta} \quad (4)$$

where $V_{\rho\theta}$ indicates whether the cell at position ρ in direction θ is valid ($V_{\rho\theta} = 1$) or invalid ($V_{\rho\theta} = 0$), and P is the set of all cell positions in one oriented component. If the score S is below a predetermined threshold, the two fingerprints are considered to belong to the same finger; otherwise they are considered to belong to different fingers.

3. Matching Functions

The use of more complex functions to compute the difference feature vector can improve the discrimination of fingerprint matching. In preliminary testing, we found that among the various approaches that can be used to calculate the matching score, the best results were obtained by the following two methods: replacing the difference feature cell values with new values given by a nonlinear function; and applying weights to the texture or difference feature cells using statistical measures extracted from them. In the second case the statistical measures pick up characteristics of spatial distribution of the feature vector.

3.1. Nonlinear Functions

One way to improve the matching score is to apply a nonlinear function, f_α , to the difference feature values so as to create new ones. The new matching score is:

$$S_f = \frac{1}{V} \sum_{\rho \in P} \sum_{\theta \in \Theta} V_{\rho\theta} f_\alpha (E_{\rho\theta}) \quad (5)$$

The functions chosen give more importance of differences than of similarities between the features vectors. Two of the best functions, f_α , identified in the present work were a polynomial function of the form:

$$f_\alpha (x) = x^\alpha \quad (6)$$

where α is the polynomial degree; and an exponential function of the form:

$$f_\alpha (x) = \frac{e^{\alpha x} - 1}{e^\alpha - 1} \quad (7)$$

where α is the coefficient degree. In these equations, $f_\alpha(x)$ gives the new values from the difference cells.

3.2. Statistical Weighting

Statistical measures were used to weight the feature vectors based on the spatial distribution of the features. These measures can be extracted from and applied to each of the difference feature vector, Equation (8).

$$E'_{\rho\theta} = R_{\rho\theta} |F_{\rho\theta} - T_{\rho\theta}| \quad (8)$$

Two ways to extract the measures from the features vectors were explored: from the neighborhood of a cell in a specific oriented component, or from cells in the same position across all oriented components. The neighborhood, N_ρ , of cell ρ is given by the eight cells surrounding that cell and by cell ρ itself in a specific oriented component $\theta \in \Theta$. Thus, as shown in Figure 1, the neighborhood of a cell is $N_\rho = \{\rho, \rho_1, \rho_2, \rho_3, \rho_4, \rho_5, \rho_6, \rho_7, \rho_8\}$. The measures extracted from cells at the same position across all directions are obtained from the values of P_ρ , where $P_\rho = \{\rho_{\theta 1}, \rho_{\theta 2}, \rho_{\theta 3}, \rho_{\theta 4}, \rho_{\theta 5}, \rho_{\theta 6}, \rho_{\theta 7}, \rho_{\theta 8}\}$ as shown in Figure 2.

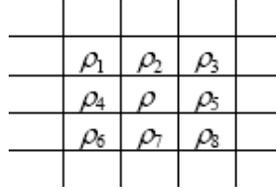


Figure 1. Neighborhood of cell ρ , $N_\rho = \{\rho, \rho_1, \rho_2, \rho_3, \rho_4, \rho_5, \rho_6, \rho_7, \rho_8\}$.

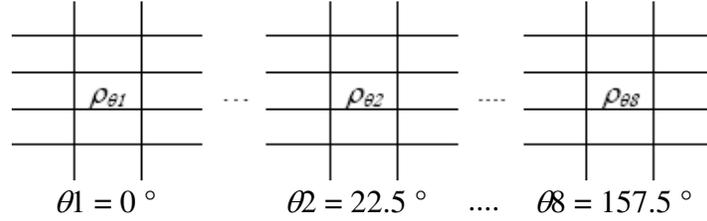


Figure 2. All cells ρ across all directions θ , $P_\rho = \{\rho_{\theta 1}, \rho_{\theta 2}, \rho_{\theta 3}, \rho_{\theta 4}, \rho_{\theta 5}, \rho_{\theta 6}, \rho_{\theta 7}, \rho_{\theta 8}\}$.

The weights, R , can be given by the mean or standard deviation of the feature values from the neighborhood of a cell (Equations (9) and (10), respectively); or the mean or standard deviation of the feature values from the cells in the same position (Equations (11) and (12), respectively).

$$\bar{R}_{\rho\theta} = \frac{1}{9} \sum_{r \in N_\rho} E_{r\theta} \quad (9)$$

$$\hat{R}_{\rho\theta} = \sqrt{\frac{1}{9} \sum_{r \in N_\rho} (E_{r\theta} - \bar{R}_{\rho\theta})^2} \quad (10)$$

$$\bar{R}_\rho = \frac{1}{8} \sum_{q \in \Theta} E_{\rho q} \quad (11)$$

$$\hat{R}_\rho = \sqrt{\frac{1}{8} \sum_{q \in \Theta} (E_{\rho q} - \bar{R}_\rho)^2} \quad (12)$$

Other measures tested were the median, maximum, minimum, and combinations of two of these measures. When the measures are extracted from cells in the same position across all directions, the weight R is the same for all oriented components.

4. Experimental Results

We used the databases of the Fingerprint Verification Competition 2002 (FVC2002) to test the gain obtained using the proposed matching functions. The FVC2002 databases consist of four distinct databases (Db1–Db4), each of which contains data collected using a different sensor/technology.

Each database contains 8 fingerprint samples for each of the 100 distinct fingers. In the present work, we compared the eight samples of the same finger with each other, and also compared the first sample of each finger with the first sample of all others fingers. These combinations result in 2800 comparisons between different samples of the same finger and 4950 comparisons between the pairs of first samples of different fingers.

To compare the new matching functions, we used the Receiver Operating Characteristic curve (ROC), which took the form of a plot of the False Accept Rate (FAR) against the False Reject Rate (FRR) for various thresholds. The FAR is given by the percentage of comparisons between different fingers where the matching score is below the threshold, i.e., false matching. The FRR is given by the percentage of comparisons between different samples of the same finger where the matching score is above the threshold, i.e., false non-matching. The matching scores were normalized to $[0, 1]$. The equal error rate (EER) is the point where the FRR and FAR are equal.

The first experiment replaces the difference cell values using nonlinear functions. Among the various functions tested, the best results were obtained using functions for which the effective replacement values had an exponential shape. These functions emphasize the contrast of the difference feature vector. Table 1 shows the Equal Error Rate (EER) for the hybrid FingerCode method, as well as for the polynomial and exponential replacing functions (Equations (6) and (7), respectively). Figure 3 shows the ROC curves obtained for the functions listed in Table 1.

Table 1. Equal Error Rate (EER) for non-linear functions, where α is the polynomial degree or exponential coefficient degree used.

	Original	Polynomial	Exponential
Db1	2.12%	1.17% ($\alpha = 4$)	1.19% ($\alpha = 17$)
Db2	1.76%	0.99% ($\alpha = 3$)	1.01% ($\alpha = 14$)
Db3	3.43%	3.04% ($\alpha = 4$)	3.05% ($\alpha = 14$)
Db4	3.06%	2.66% ($\alpha = 3$)	2.67% ($\alpha = 12$)

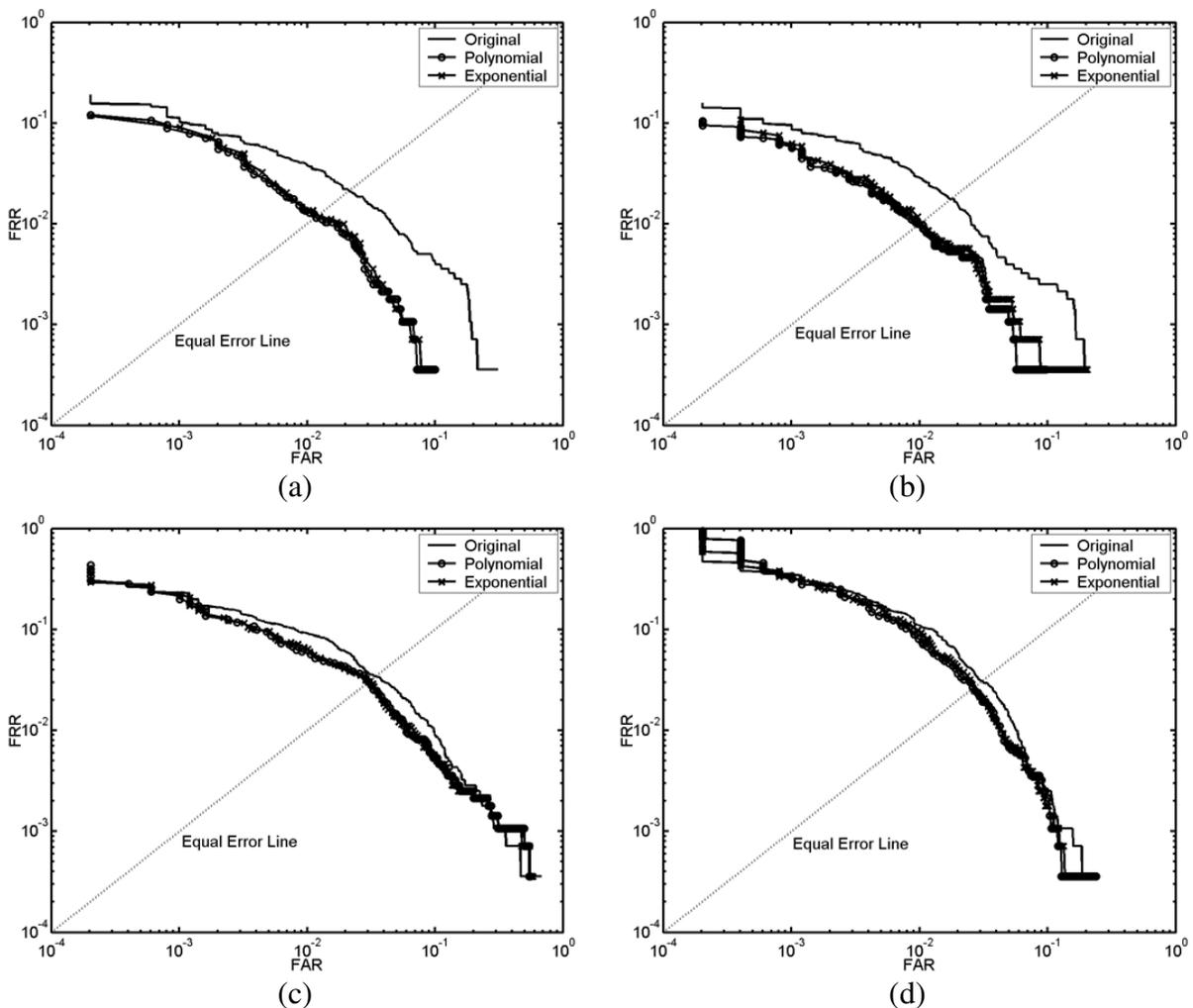


Figure 3. ROC curves for non-linear functions: (a) Db1; (b) Db2; (c) Db3; and (d) Db4.

In the second experiment, we tested several approaches to weighting the cells using statistical measures extracted from the texture or difference feature vectors: mean, standard

deviation, median, maximum, minimum, and combinations of two of these measures. These measures were used directly, either by giving more weight to cells with high values of the measure, or inversely, giving more weight to cells with low values. For Db1, Db2, and Db3, the statistical measures for cell weighting that gave the best responses were the standard deviation of the neighborhood around a cell in a specific oriented component (Equation 10) and the standard deviation of the cells in the same position across all oriented components (Equation 12), extracted from and applied to the difference feature vector (Equation 8), used in the direct way. The first one analyzes the uniformity of the surrounding area of the point, while the second one analyzes the direction force in the point. Table 2 shows the EERs for the hybrid FingerCode method (Original), the standard deviation of the neighborhood of a cell in a specific direction (Neighborhood), the standard deviation of the cells in the same position across all directions (Across Directions). Figure 4 shows the ROC curves for the cases listed in Table 2.

In Db4, the Neighborhood case worsened the response, and only the Across Directions case worked well. For Db4, the best results (EER = 1.80%) were obtained by weighting the cells using the maximum cell value from the neighborhood of a cell in a specific direction, which was extracted from and applied to each of the texture vectors; by contrast, the gain obtained using this measure was very small for the other databases. However, given that Db4 is composed of synthetic fingerprints, we do not recommend the use of this measure.

Table 2. Equal Error Rate (EER) for statistical weighting.

	Original	Neighborhood	Across Direction
Db1	2.12%	1.19%	1.35%
Db2	1.76%	0.99%	1.01%
Db3	3.43%	2.69%	3.11%
Db4	3.06%	3.17%	2.61%

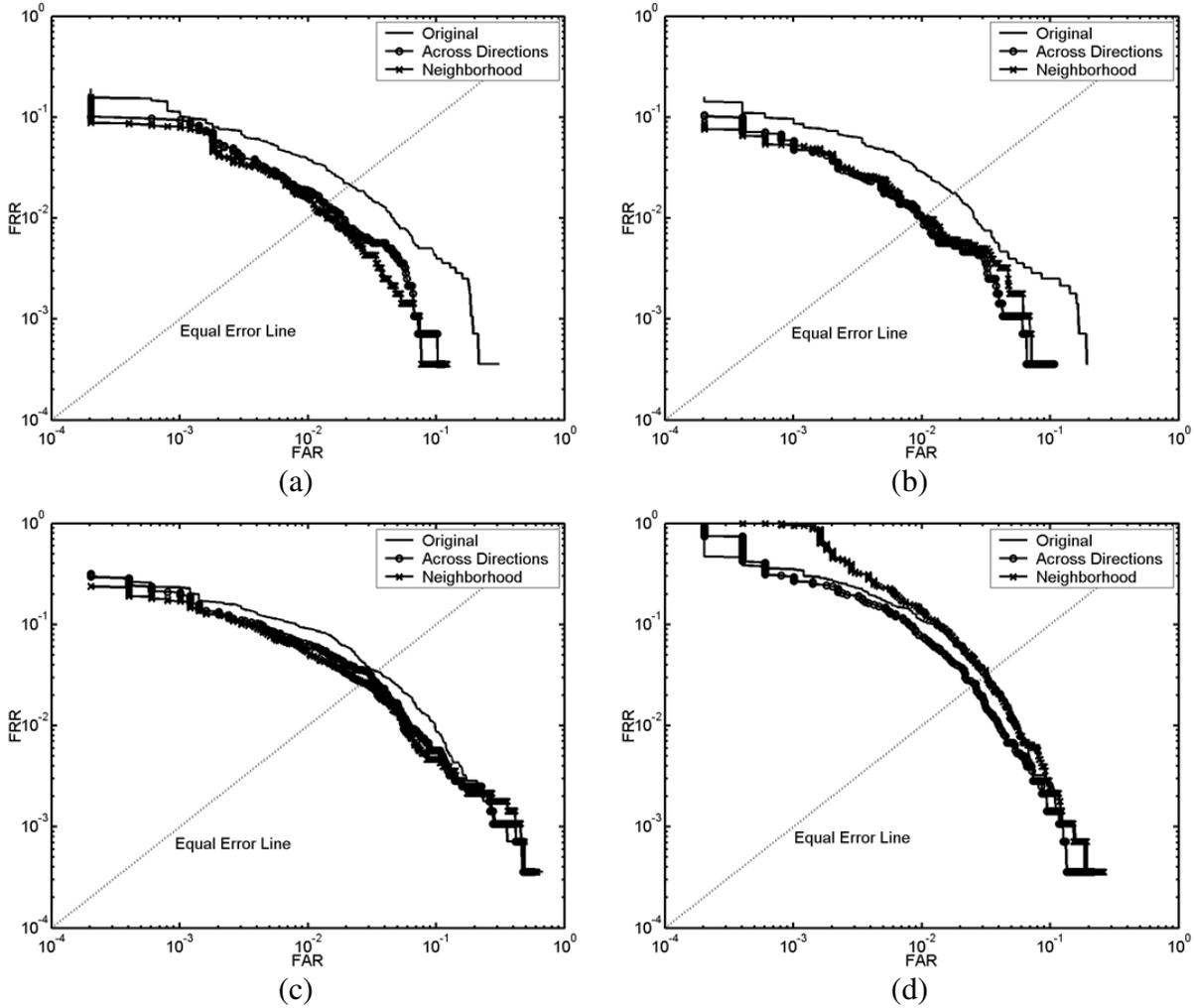


Figure 4. ROC curves for statistical weighting: (a) Db1; (b) Db2; (c) Db3; and (d) Db4.

5. Conclusions

In the present work we have established that using more complex functions to match the texture feature vectors can improve the discrimination of the FingerCode method. A large range of matching functions could potentially be used for this purpose; here we have presented two such functions: replacing the cell values of the difference vector by a non-linear function; and weighting the cells using statistical measures extracted from the feature vector. Application of the method to the four data sets within the FVC2002 database showed that the gain achieved depends on the database used. Db1 and Db2 (optical sensors) showed the greatest gain (~40% improvement in the EER) for both approaches. Db3 (capacitive sensor) showed a lower gain than Db1 and Db2 (~10% improvement in the EER). Db4 (synthetic fingerprints) showed behavior

that was very different from that observed for the other databases in the experiments using statistical measures.

The process of replacing cell values with new values given by a nonlinear function is faster and easier than the process of weighting cells based on statistical measures. In the nonlinear function approach, however, it is necessary to find the optimum parameters of the function to be used, which requires a good representative sample of the database. If new input fingerprints diverge from this representative sample, the response of the function will diminish. Here we found that the best fitting functions had exponential-like characteristics that cause low values to be decreased to near zero, and high values to be markedly increased.

Compared to the nonlinear function approach, the use of statistical measures has the advantage of being less sensitive to the specifics of the database. The same measures work well for almost all types of fingerprints. Here we found that the two best statistical measures were the standard deviation from the neighborhood of a cell in a specific direction, and the standard deviation from the cells in the same position across all oriented components, extracted from and applied to the difference feature vector. The former measure gave superior results for all databases except the database of synthetic fingerprints. The gain achieved using a combination of statistical measures was small compared to that achieved using a single measure, and the use of multiple measures has the disadvantage of being more complex.

6. References

- [1] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, "Handbook of Fingerprint Recognition", Springer, 2003.
- [2] M. U. Munir and M. Y. Javed, "Fingerprint Matching Using Gabor Filters", National Conference on Emerging Technologies, pp. 147-151, 2004.
- [3] K. Venkataramani, and B. V. K. V. Kumar, "Fingerprint Verification Using Correlation Filters", Proc. Int. Conf. on Audio- and Video-Based Biometric Person Authentication (AVBPA), pp. 886-894, 2003.

- [4] M. Tico and P. Kuosmanen, "Fingerprint Matching Using an Orientation-Based Minutia Descriptor", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 8, pp. 1009-1014, 2003.
- [5] X. Jiang and W. Y. Yau, "Fingerprint Minutiae Matching Based on the Local and Global Structures", *Proc. Int. Conf. Pattern Recognition*, vol. 2, pp. 1038-1041, 2000.
- [6] D. Maio and D. Maltoni, "Direct Gray-Scale Minutiae Detection in Fingerprints", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 1, pp. 27-40, 1997.
- [7] J. K. Min, J. H. Hong, and S. B. Cho, "Effective Fingerprint Classification by Localized Models of Support Vector Machine", *International Conference on Biometrics (ICB)*, pp. 287-293, 2006.
- [8] Q. Zhang, K. Huang, and H. Yan, "Fingerprint Classification Based on Extraction and Analysis of Singularities and Pseudoridges", *Pan-Sydney Area Workshop on Visual Information Processing (VIP)*, 2001.
- [9] A. Senior, "A Combination Fingerprint Classifier", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 10, pp. 1165-1174, 2001.
- [10] A. K. Jain, S. Prabhakar, L. Hong, and S. Pankanti, "FingerCode: A Filterbank for Fingerprint Representation and Matching", *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, vol. 2, pp. 187-193, 1999.
- [11] J. G. Daugman, "High Confidence Recognition of Persons by a Test of Statistical Independence", *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 15, no. 11, pp. 1148-1161, 1993.
- [12] J. Daugamn, "Iris Recognition", *American Scientist*, vol. 89, pp. 326-333, 2001.
- [13] J. Daugamn, "How Iris Recognition Works", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 1, pp. 21-30, 2004.
- [14] A. K. Jain, S. Prabhakar, L. Hong, and S. Pankanti, "Filterbank-Based Fingerprint Matching", *IEEE Trans. on Image Processing*, vol. 9, no. 5, pp. 846-859, 2000.
- [15] A. Jain, A. Ross, and S. Prabhakar, "Fingerprint Matching Using Minutiae and Texture Features", *Proc. Int. Conf. on Image Processing (ICIP)*, pp. 282-285, 2001.
- [16] A. Ross, A. Jain, and J. Reisman, "A Hybrid Fingerprint Matcher", *Pattern Recognition*, vol. 36, pp. 1661-1673, 2003.
- [17] L. Hong, Y. Wan, and A. K. Jain, "Fingerprint Image Enhancement: Algorithms and Performance Evaluation", *IEEE Transactions on PAMI*, vol. 20, pp. 777-789, 1998.

4.2 Histograma de Distribuição do FingerCode

Artigo submetido para IEEE Pattern Recognition.

Através da análise dos histogramas de distribuição dos valores dos vetores diferença, foi possível detectar várias de suas propriedades. Nesta análise pôde-se observar que outras características estatísticas, além da média dos valores, poderiam ser utilizadas para o cálculo do score. Experimentos realizados mostraram que o desvio padrão e os percentis acima de 75 são mais robustos do que a média (utilizada para calcular o score no método FingerCode original). O melhor resultado foi obtido utilizando o desvio padrão, apresentando um ganho significativo sem aumento na complexidade.

Artigo 4. FingerCode Matching Histogram

Gustavo de Sá, and Roberto Lotufo

School of Electrical and Computer Engineering, University of Campinas (UNICAMP),

C.P. 6101, 13083-970, Campinas, SP, Brazil

gsa@dca.fee.unicamp.br, and www.dca.fee.unicamp.br/~lotufo

Abstract

The FingerCode method is a correlation based scheme for matching fingerprints. In this scheme, an oriented bank of Gabor filters extracts directional texture information from the fingerprint image. The extracted texture vectors of the template and input fingerprints are compared and a matching score is generated by computing the mean of the absolute differences of the vector values. In this paper we propose new ways to compute the matching score. The use of other statistical measures extracted from the feature vectors, besides the mean, is explored. Among the various statistical measures tested, we found that the standard deviation gave the best results. The use of the standard deviation resulted in better discrimination, thus improving the FingerCode method. Tests using the FVC2002 database confirmed the advantages of the proposed matching score function.

1. Introduction

Fingerprints are the pattern of alternating ridges and valleys on the finger surface. Because fingerprints are unique, permanent, and easy to collect identifiers of individuals, identification systems based on fingerprint patterns show good performance and reliability [1]. These characteristics, which underpin the push towards exploiting fingerprint systems in civilian applications, are the same as have been used for more than a century in forensic work. Fingerprint recognition systems can be broadly categorized as either correlation [2-4] or minutiae [5-6] based techniques, depending on the scale of the characteristics used. Correlation-based techniques use global information such as curvature and ridge orientation, whereas minutiae-based techniques use local anomalies (called minutiae) such as terminations and bifurcations. The majority of fingerprint applications use minutiae in the matching process, with correlation-based

techniques mostly being used to classify fingerprints [7-9]. Each of these techniques does not account for information on the other level: minutia-based techniques do not use the full information from the global level, and correlation-based techniques do not use the detailed information from the local level.

Based on a model for iris recognition [11-12], Jain et al. [14] developed a texture-based method for fingerprint recognition called FingerCode. In this method, the two fingerprints being compared are first aligned by their core points, which are extracted by a bank of Gabor filters, and then the normalized Euclidean distance of the oriented texture vectors is computed. Subsequently Jain et al. [15] proposed a hybrid FingerCode method that combines minutiae and texture (i.e. local and global information), and showed that it performed better than minutiae or FingerCode methods. In this variation, the minutiae points are used to align the fingerprints and the texture vector sets are used to obtain the final match score [16].

Here we propose a new technique for comparing the FingerCode feature vectors of a pair of fingerprints. In our method, instead of computing the matching score as the sum of the differences between the texture vectors, we compute the score based on a statistical measure of the distribution of the difference cell values. Several statistical measures were tested, including the standard deviation and percentiles. This technique is based on an analysis of the histogram of the difference vector. Through experimental tests, we show that the proposed technique works better than FingerCode.

This paper is organized as follows: Section 2 explains the FingerCode method and its hybrid variation; Section 3 presents the new technique for computing the matching score; Section 4 outlines the experimental analysis of the difference vector and the gain obtained using the proposed modification; and finally, Section 5 present the conclusions drawn from this work.

2. FingerCode

The FingerCode method is a correlation-based technique in which a small circular area around the core of a fingerprint is tessellated in an arc fashion and filtered by an oriented Gabor

filter bank. For each cell in the oriented components, a value is computed; this set of values constitutes the texture feature vectors. Two fingerprints are aligned by their core points and their texture vectors are compared, generating a score that indicates the degree of similarity. The FingerCode hybrid variation differs from the FingerCode method in that it uses the entire fingerprint area, which is tessellated in a square rather than an arc fashion, and, most importantly, the minutiae points are used to align the fingerprints.

In the hybrid variation, the fingerprint image is tessellated into rectangular cells. The block size used in the present work was 16×16 pixels, which corresponds to approximately 1.5 times the width of the ridge-valley structure in a 500 dpi image. Then, the fingerprint is separated from the background in the segmentation process, avoiding the extraction of features from invalid cells (cells in which one or more pixels belong to the background). In this process, a morphological scheme is used to establish the fingerprint mask. To reduce intensity variations due to differences in finger pressure, the fingerprint image is normalized using the approach presented in [17].

After extracting the minutiae points from the input image, the rotation and translation parameters are generated that give the best correspondence with the set of minutiae points from the template image. The aligned image is filtered by an oriented Gabor filter bank. The Gabor filter is frequency- and orientation-selective and consists of a sinusoidal wave modulated by a Gaussian envelope. The bank is formed by eight Gabor filters in directions $\Theta = \{0^\circ, 22.5^\circ, 45^\circ, 67.5^\circ, 90^\circ, 112.5^\circ, 135^\circ, 157.5^\circ\}$. The chosen frequency ($f_0 = 0.1 \text{ pixel}^{-1}$) was based on the mean inter-ridge distance of approximately 10 pixels for a 500 dpi image. The real part of the Gabor filter is given by:

$$G(x, y, \sigma_x, \sigma_y, \omega_0, \theta) = e^{-\frac{1}{2} \left(\frac{x_\theta^2}{\sigma_x^2} + \frac{y_\theta^2}{\sigma_y^2} \right)} \cos(\omega_0 x_\theta) \quad (1)$$

$$x_\theta = x \sin \theta + y \cos \theta$$

$$y_\theta = x \cos \theta - y \sin \theta$$

where σ_x and σ_y are the standard deviation of the Gaussian envelope in the x- and y-axes respectively, θ is the direction of the filter, and $\omega_0 = 2\pi f_0$. The filtering extracts eight oriented

image components in each direction. For each cell, C , the feature value $F_{\rho\theta}$ is computed by equation (2):

$$F_{\rho\theta} = \sqrt{\frac{1}{K_{\rho\theta}} \sum_{(x,y) \in C_{\rho\theta}} (C_{\rho\theta}(x,y) - \bar{C}_{\rho\theta})^2} \quad (2)$$

$$\bar{C}_{\rho\theta} = \frac{1}{K_{\rho\theta}} \sum_{(x,y) \in C_{\rho\theta}} C_{\rho\theta}(x,y)$$

where $C_{\rho\theta}$ is the cell at position ρ in oriented component θ , $C_{\rho\theta}(x,y)$ is the pixel value at position (x,y) in cell $C_{\rho\theta}$, $K_{\rho\theta}$ is the number of pixels from cell $C_{\rho\theta}$, and $\bar{C}_{\rho\theta}$ is the mean value of the pixels from cell $C_{\rho\theta}$. FingerCode is the set of eight feature vectors composed of all cell values, $F_{\rho\theta}$. The difference vector, E , is computed as the absolute difference between corresponding cells from the input, F , and template, T , FingerCode feature vectors:

$$E_{\rho\theta} = |F_{\rho\theta} - T_{\rho\theta}| \quad (3)$$

where $E_{\rho\theta}$, $F_{\rho\theta}$ and $T_{\rho\theta}$ are the cell values at position ρ in direction θ from the difference, input and template feature vectors, respectively. The matching score S_M is expressed as the sum of all differences normalized by the number of valid cells, V , given by the intersection of the two fingerprints masks:

$$S_M = \frac{1}{V} \sum_{\rho \in P} \sum_{\theta \in \Theta} V_{\rho\theta} E_{\rho\theta} \quad (4)$$

where $V_{\rho\theta}$ indicates whether the cell at position ρ in direction θ is valid ($V_{\rho\theta} = 1$) or invalid ($V_{\rho\theta} = 0$), and P is the set of all cell positions in one oriented component. If the score S_M is below a predetermined threshold, the two fingerprints are considered to belong to the same finger, otherwise they are considered to belong to different fingers.

3. New Matching Score

The distribution histogram of the difference cell values was used to analyze the characteristics of the difference vector. The histograms were constructed from all of the difference cell values obtained from all of the fingerprint comparisons for each type of matching (same finger or different finger), normalized by the number of valid cells. This generated one

histogram for all same finger matching and another for all different finger matching. Figure 2 shows the difference histograms of all matching cases for the FVC2002 databases. The same finger histograms are quite different from the different fingers histograms, with the former histograms having a lower mean and standard deviation. For database Db1, for example, the mean and standard deviation are 40.39 and 37.69 for the same finger histograms, and 101.76 and 96.98 for the different finger histograms of all matching cases, respectively.

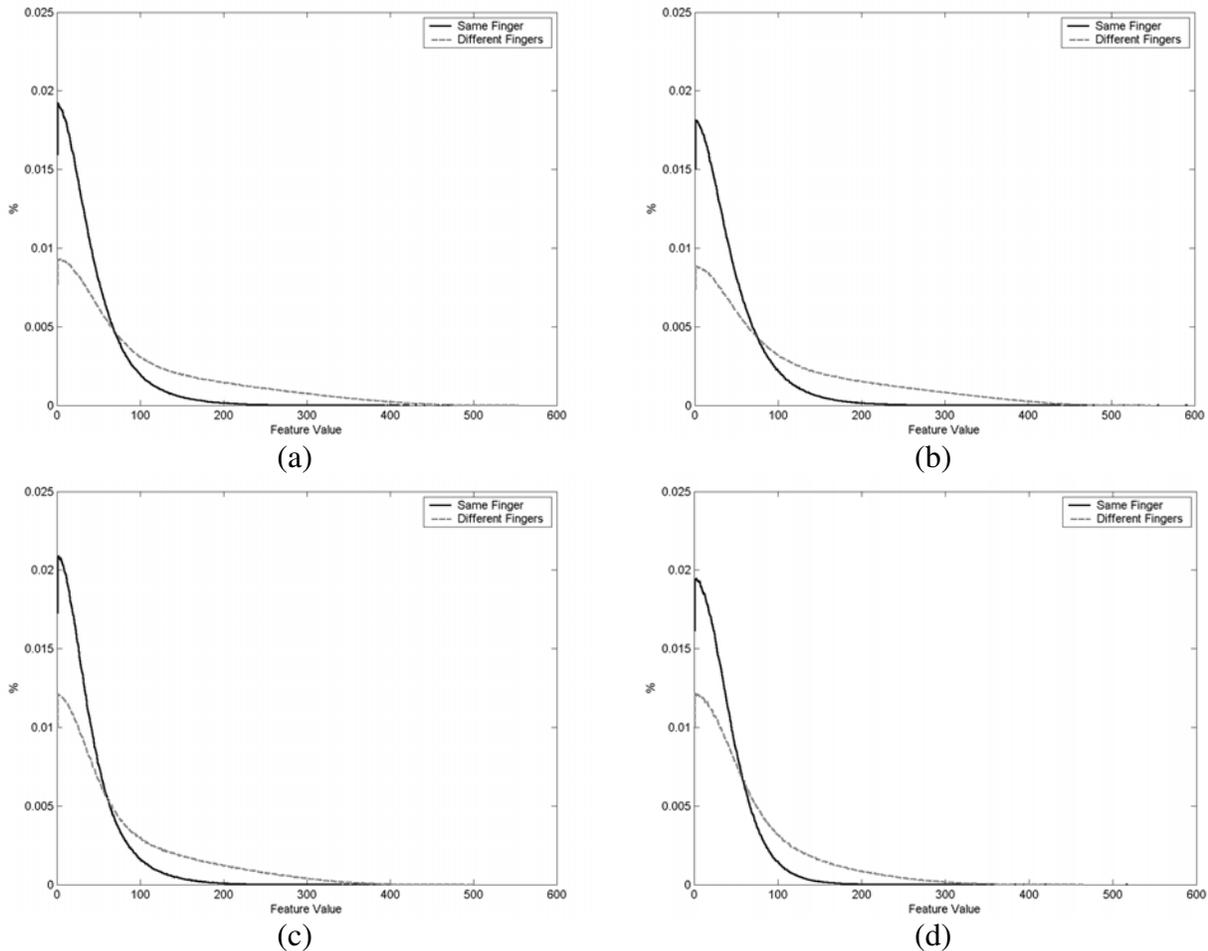


Figure 2. The same finger and different fingers histograms for all matching cases (template histograms): (a) Db1; (b) Db2; (c) Db3; and (d) Db4.

To determine whether a pair of fingerprints match using the histograms, the histogram of the difference cell values for the single pair of fingerprints is generated and compared to the same finger and different fingers template histograms of all matching cases to verify to which one the single histogram fits better.

Figure 3 shows six examples of single distribution histograms from database Db1. The histograms for the same finger (Figures 3.a-3.c) and different fingers (Figures 3.a-3.c) matching have the following characteristics: 3.a, mean = 17.2 and std. = 14.5; 3.b, mean = 40.0 and std. = 36.6; 3.c, mean = 91.1 and std. = 63.9; 3.d, mean = 204.8 and std. = 145.4; 3.e, mean = 100.7 and std. = 97.0; and 3.f, mean = 47.8 and std. = 48.5. The single matching cases 3.b and 3.e are the typical cases; in these cases, the histograms are similar to the same finger and different fingers all bank histograms, respectively. However, the worst case of same finger matching, 3.c, is similar to the typical case of different finger matching, 3.e, and the worst case of different matching, 3.f, is similar to the typical case of same finger matching, 3.b.

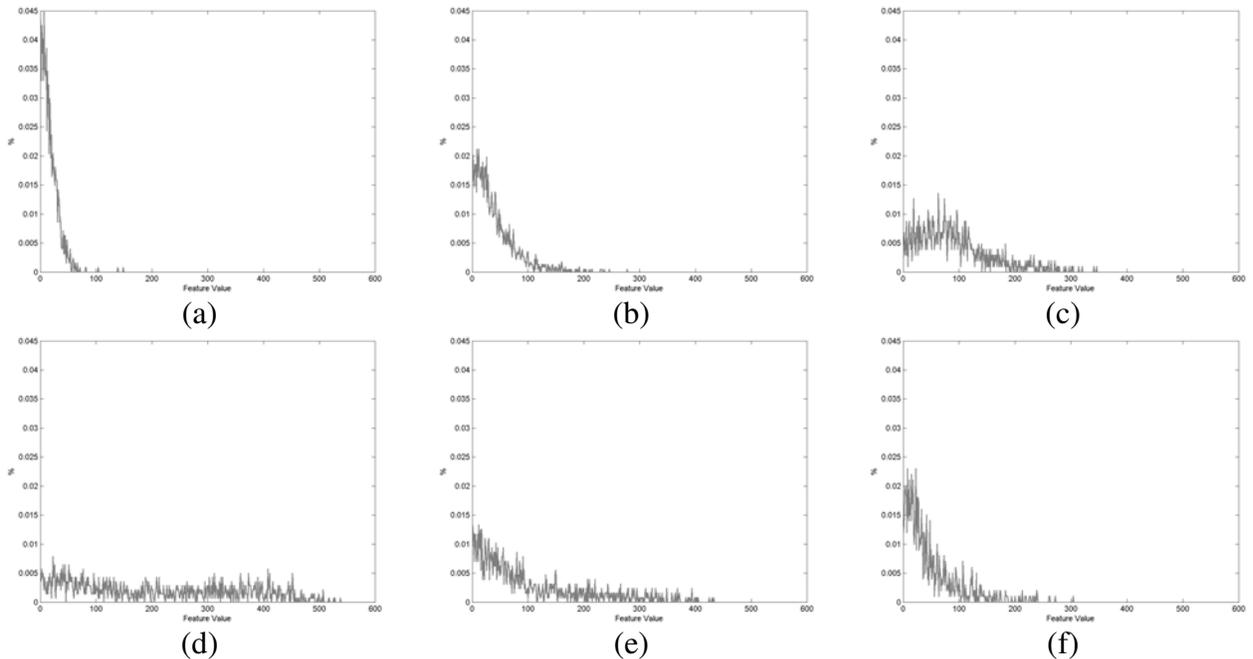


Figure 3. Single distribution histograms from Db1: (a) Best same finger case; (b) Typical same finger case; (c) Worst same finger case; (d) Best different fingers case; (e) Typical different fingers case; and (f) Worst different fingers case.

Motivated by simplicity and speed of computation, we propose to compare the fingerprints by comparing the statistical features of the histograms. Several statistical measures obtained from the histograms were tested, including the mean (original case), standard deviation (Equation 5), skewness, kurtosis, median, and percentiles.

In the original FingerCode method, the matching score (Equation 4) is equal to the mean value of the difference vector (i.e. the sum of the valid cell values divided by the number of valid cells). However, other characteristics can be extracted from the difference vector, including the

distribution of the cell values given by measures such as the standard deviation, percentiles, skewness, and kurtosis. Use of these new measures to compute the matching score could potentially give a better result. In the case of using the standard deviation, the new matching score, S_S , becomes:

$$S_S = \sqrt{\frac{1}{V} \sum_{\rho \in P} \sum_{\theta \in \Theta} V_{\rho\theta} (E_{\rho\theta} - S_M)^2} \quad (5)$$

where S_M is the mean of the difference cell values given by Equation (4), $E_{\rho\theta}$ are the cell values, $V_{\rho\theta}$ indicates whether the cell at position ρ in direction θ is valid ($V_{\rho\theta} = 1$) or invalid ($V_{\rho\theta} = 0$), V is the number of valid cells, P is the set of all cell positions in one oriented component, and Θ is the set of directions.

4. Experimental Analysis

We used the databases of the Fingerprint Verification Competition 2002 (FVC2002) to test the performance gain obtained using the proposed matching functions. The FVC2002 databases consist of four different databases (Db1-Db4), each of which contains data collected using a different sensor/technology (see Table 1).

Table 1. FVC2002 Databases

	Sensor	Image Size	Resolution
Db1	Optical Sensor “TouchView II” by Identix	388x374	500 dpi
Db2	Optical Sensor “FX2000” by Biometrika	296x560	569 dpi
Db3	Capacitive Sensor “100 SC” by Precise Biometrics	300x300	500 dpi
Db4	Synthetic Fingerprint Generation “SFinGe v2.51”	288x384	about 500 dpi

Each database contains 8 fingerprint samples for each of 100 distinct fingers. In the present work, we compared the eight samples of the same finger with each other, and also compared the first sample of each finger with the first sample of all other fingers. These combinations result in 2800 comparisons between different samples of the same finger and 4950 comparisons between the pairs of first samples of different fingers.

Among the various statistical measures tested, the best results were obtained using the standard deviation extracted from the histograms as the matching score. Percentiles 75 and 90 gave better results than the mean, but worse than the standard deviation. To compare the matching functions, we used the Receiver Operating Characteristic curve (ROC), which took the form of a plot of the False Accept Rate (FAR) against the False Reject Rate (FRR) for various thresholds. The FAR is given by the percentage of comparisons between different fingers where the matching score is below the threshold, i.e., false matching. The FRR is given by the percentage of comparisons between different samples of the same finger where the matching score is above the threshold, i.e., false non-matching. The matching scores were normalized to [0 1]. The Equal Error Rate (EER) is the point where the FRR and FAR are equal. Figure 4 shows the ROC curves of the mean and standard deviation matching scores for the four databases. Table 3 shows the EER values for the cases in Figure 4. The results obtained indicate that the standard deviation of FingerCode is a more robust measure than the mean. Among the four databases tested, the optical databases (Db1 and Db2) gave the greatest gain, with error rates of the standard deviation that were much lower than those of the mean for all threshold points. For the synthetic databases (Db3 and Db4), the gain in the robustness was small.

Table 3 – Equal Error Rate (ERR).

	Mean (original)	Standard Deviation
Db1	2.12%	1.14%
Db2	1.76%	0.89%
Db3	3.43%	3.15%
Db4	3.06%	2.68%

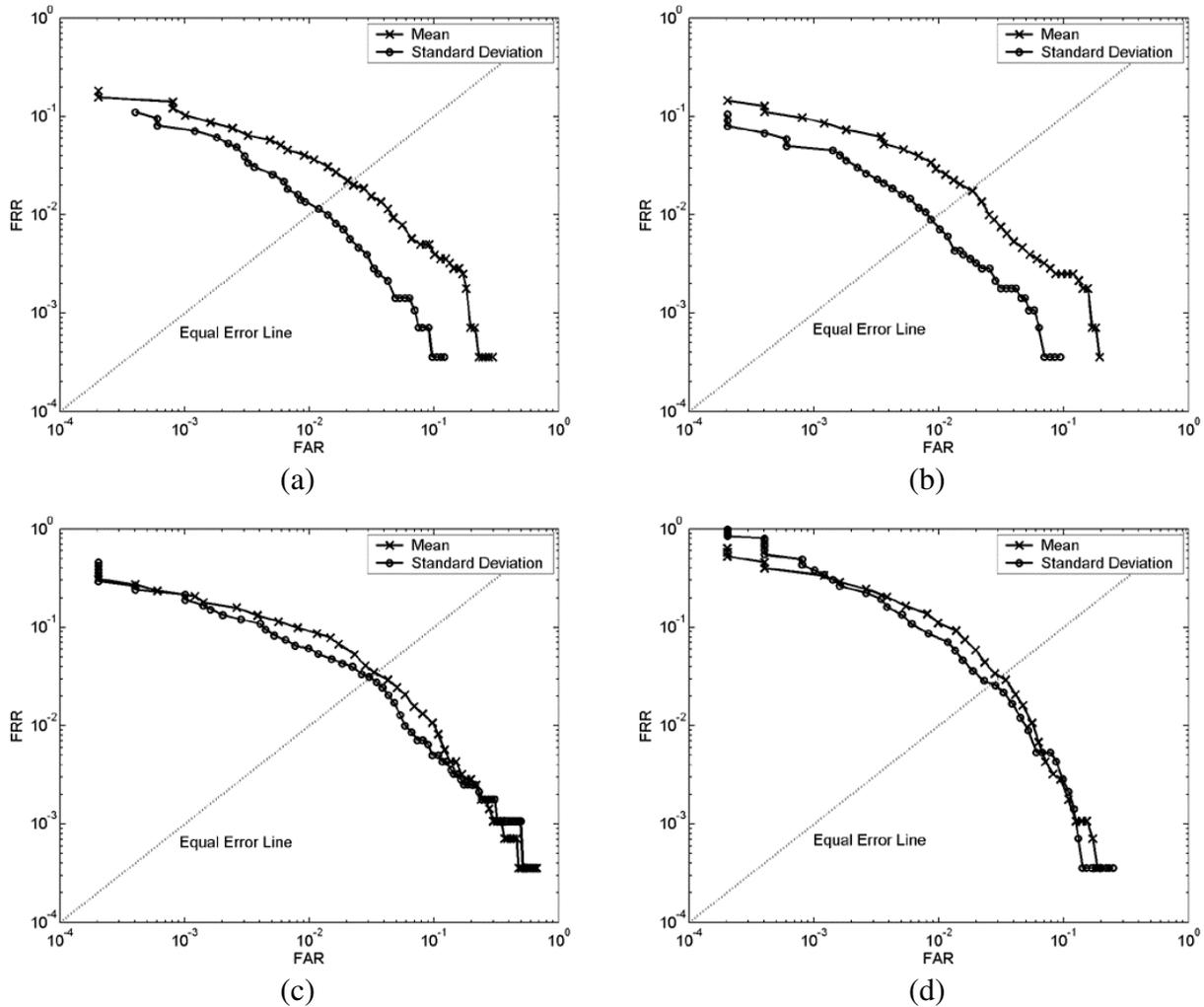


Figure 4. ROC curves: (a) Db1; (b) Db2; (c) Db3; and (d) Db4.

5. Conclusion

In the FingerCode method, the oriented texture vector is extracted from a fingerprint image by a bank of Gabor filters. Texture vectors of two fingerprints are then matched by computing the mean value of the absolute difference between them. Here we propose a new approach that relies on the analysis of the histogram of the difference cell values from the matching cases. Using the difference cell values from a large number of comparisons between fingerprints from the same finger or from different fingers (for a realistic solution it should be a representative sample), the same finger and different fingers template histograms are plotted. Then, to determine whether a particular pair of fingerprints are from the same finger (a match) or from different fingers (a non-

match), the single distribution histogram for that pair of fingerprints is compared with the two templates distribution histograms.

Specific characteristics can be extracted from the histograms, such as the mean, standard deviation, and median. The statistical features of the histograms were compared rather than the complete histograms because this approach is simpler and faster. In the original FingerCode method, the mean of the difference cell values is used. However, the histograms analysis yields a better statistical measure: the standard deviation. The improvement obtained using the standard deviation instead of the mean was significant. For example, the gain in the ERR point was about 46% for Db1 and Db2 (optical sensors), 8% for Db3 (capacitive sensor), and 12% for Db4 (synthetic fingerprints). The other characteristics such as the median and percentiles performed worse than the standard deviation and, in some cases, even worse than the mean.

Bibliography

- [1] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, "Handbook of Fingerprint Recognition", Springer, 2003.
- [2] M. U. Munir e M. Y. Javed, "Fingerprint Matching Using Gabor Filters", National Conference on Emerging Technologies, pp. 147-151, 2004.
- [3] K. Venkataramani, and B. V. K. V. Kumar, "Fingerprint Verification Using Correlation Filters", Proc. Int. Conf. on Audio- and Video-Based Biometric Person Authentication (AVBPA), pp. 886-894, 2003.
- [4] M. Tico and P. Kuosmanen, "Fingerprint Matching Using an Orientation-Based Minutia Descriptor", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25, no. 8, pp. 1009-1014, 2003.
- [5] X. Jiang and W. Y. Yau, "Fingerprint Minutiae Matching Based on the Local and Global Structures", Proc. Int. Conf. Pattern Recognition, vol. 2, pp. 1038-1041, 2000.
- [6] D. Maio e D. Maltoni, "Direct Gray-Scale Minutiae Detection in Fingerprints", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, no. 1, pp. 27-40, 1997.
- [7] J. K. Min, J. H. Hong, and S. B. Cho, "Effective Fingerprint Classification by Localized Models of Support Vector Machine", International Conference on Biometrics (ICB), pp. 287-293, 2006.

- [8] Q. Zhang, K. Huang e H. Yan, “Fingerprint Classification Based on Extraction and Analysis of Singularities and Pseudoridges”, Pan-Sydney Area Workshop on Visual Information Processing (VIP), 2001.
- [9] A. Senior, “A Combination Fingerprint Classifier”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 10, pp. 1165-1174, 2001.
- [10] A. K. Jain, S. Prabhakar, L. Hong e S. Pankanti, “FingerCode: A Filterbank for Fingerprint Representation and Matching”, Proc. IEEE Conf. on Computer Vision and Pattern Recognition, vol. 2, pp. 187-193, 1999.
- [11] J. G. Daugman, “High Confidence Recognition of Persons by a Test of Statistical Independence”, IEEE Trans. Pattern Anal. Machine Intell., vol. 15, no. 11, pp. 1148-1161, 1993.
- [12] J. Daugamn, “How Iris Recognition Works”, IEEE Transactions on Circuits and Systems for Video Technology, vol. 14, no. 1, pp. 21-30, 2004.
- [13] A. K. Jain, S. Prabhakar, L. Hong, and S. Pankanti, “Filterbank-Based Fingerprint Matching”, IEEE Trans. on Image Processing, vol. 9, no. 5, pp. 846-859, 2000.
- [14] A. Jain, A. Ross e S. Prabhakar, “Fingerprint Matching Using Minutiae and Texture Features”, Proc. Int. Conf. on Image Processing (ICIP), pp. 282-285, 2001.
- [15] A. Ross, A. Jain, and J. Reisman, “A Hybrid Fingerprint Matcher”, Pattern Recognition, vol. 36, pp. 1661-1673, 2003.
- [16] L. Hong, Y. Wan, and A. K. Jain, “Fingerprint Image Enhancement: Algorithms and Performance Evaluation”, IEEE Transactions on PAMI, vol. 20, pp. 777-789, 1998.

4.3 Métodos Combinados

As propostas de cálculo do escore apresentadas utilizam diferentes propriedades dos vetores diferenças. Devido a esta diferença de abordagem na classificação dos valores dos vetores diferença foi implementada uma metodologia de fusão dos classificadores (novas funções de escore). A fusão dos classificadores (Figura 4.1) é a combinação de vários classificadores de acordo com uma determinada estratégia [52]. Algumas das estratégias mais comuns de combinação de classificadores são baseadas em regras [51]:

- Regra do máximo
- Regra do mínimo
- Regra da mediana
- Regra do voto majoritário

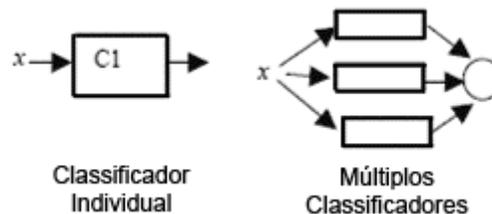


Figura 4.1. Fusão de classificadores.

A estratégia escolhida foi a do voto majoritário [56]. Neste método, a decisão final que indica se duas imagens de impressões digitais pertencem ou não ao mesmo dedo é igual à resposta dada pela maioria dos métodos. Como exemplo, tem-se o caso da combinação de três variações do FingerCode, o sistema indicará que duas impressões digitais pertencem ao mesmo dedo se e somente se duas ou três das variações indicarem que as impressões digitais pertencem ao mesmo dedo. Caso contrário, o sistema indicará que as impressões digitais pertencem a dedos diferentes.

Resultados experimentais mostraram que o ganho com a combinação de variações foi insignificante, como mostra a Figura 4.2. No exemplo da Figura 4.2, foram combinadas três variações: substituição dos valores por uma função não linear (função polinomial), ponderação dos valores na mesma posição através todas as direções e o desvio padrão do valores do vetor de

diferença de texturas. O resultado do voto majoritário corresponde a uma região no gráfico ROC limitada pelas várias curvas geradas.

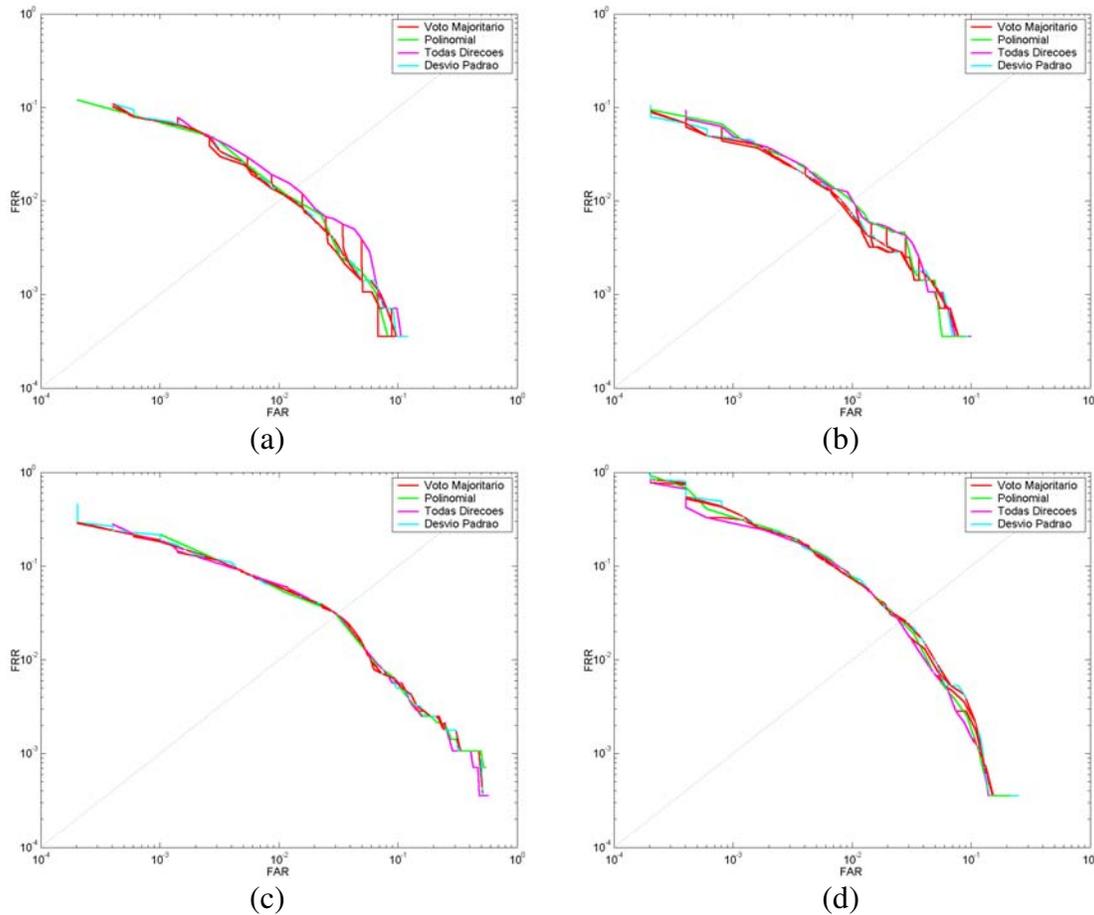


Figura 4.2. Curvas ROC do método voto majoritário: (a) Db1; (b) Db2; (c) Db3; e (d) Db4.

Estes resultados mostram que o método do voto majoritário não melhora o desempenho do sistema. Também indica a dificuldade de se combinar as variações de forma eficiente. A solução provavelmente seria o uso de métodos de combinação mais complexos.

Capítulo 5

5 Conclusões

As propostas desenvolvidas possibilitaram uma melhoria no processo de reconhecimento de impressões digitais baseados em textura (FingerCode). Apesar de ser pouco utilizado e estudado, o método de reconhecimento de impressões digitais utilizando textura possui grande capacidade discriminatória, podendo ser utilizado sozinho [43] ou combinado com minúcias. O segundo caso (FingerCode modificado) provou ser mais robusto que os métodos baseados apenas em minúcias ou em textura [83]. Em contrapartida o tempo de processamento é maior. Portanto o aperfeiçoamento do método FingerCode torna-se bastante interessante e importante para uma melhor qualidade dos sistemas de reconhecimento de impressões digitais.

Nas propostas para o aperfeiçoamento do reconhecimento baseado em textura, foi utilizado o método FingerCode modificado devido a sua maior robustez. Através de testes experimentais foram obtidas três formas distintas de melhorar o escore da comparação: substituindo os valores de atributos através de uma função não-linear, dando peso aos atributos de acordo com medidas extraídas da distribuição espacial dos valores e utilizando medidas estatísticas extraídas dos histogramas de distribuição dos valores.

Na primeira proposta, os valores dos atributos das diferenças são modificados apenas de acordo com os seus valores. Esta modificação é definida por uma função não-linear. Foram testadas várias funções e as que apresentaram melhores resultados possuíam, na faixa dos valores dos atributos, uma forma similar à exponencial. Neste tipo de função, os valores baixos são levados próximos a zero, enquanto os valores altos crescem acentuadamente. Apesar da simplicidade do método, é necessário fazer os ajustes dos parâmetros da função para otimizar a

resposta do sistema. Como mostrado na Seção 4.1, estes parâmetros possuem uma faixa de valores estreita.

Na segunda proposta, os valores dos atributos são ponderados de acordo com medidas estatísticas extraídas dos setores próximos. Esta proximidade é dada de duas formas: os setores vizinhos do setor no mesmo componente orientado ou os setores na mesma posição do setor em todos os componentes orientados. No primeiro caso, para cada posição em cada componente orientado é computado um valor baseado nos valores dos setores circunvizinhos e do próprio setor. Este valor é usado para ponderar o valor do atributo nesta determinada posição e neste determinado componente orientado. No segundo caso, os componentes orientados são empilhados e para cada posição nesta pilha é computado um valor baseado nos valores dos setores que possuem a mesma posição nos componentes orientados. Este valor é idêntico para uma mesma posição independente do componente orientado. Este valor é usado para ponderar o valor dos atributos nesta determinada posição em todos os componentes orientados.

Na terceira proposta, os valores dos atributos das diferenças são analisados utilizando um histograma. Este histograma de distribuição dos valores permite uma melhor compreensão das características dos casos de comparações de impressões digitais. Da observação dos histogramas, conclui-se que medidas estatísticas poderiam ser obtidas e utilizadas como critério de discriminação. O método FingerCode original utiliza a média dos valores dos atributos para calcular o escore. No entanto, testes mostraram que o desvio padrão destes valores produz um resultado melhor. Algumas medidas produziram resultados superiores à média, mas inferiores ao desvio padrão.

As três variações propostas apresentaram ganhos equivalentes. Esta melhoria na robustez do sistema foi significativa, principalmente nos casos de sensores óticos (Db1 e Db2), em que o ganho ficou em torno de 45% no EER. Por sua maior simplicidade, o uso da terceira variação é o mais recomendável. A combinação de duas ou mais variações do FingerCode usando o método de voto majoritário não apresentou melhoria, o que indica a dificuldade de combinação das variações e a necessidade de métodos mais complexos para obter um ganho com a combinação das variações.

Grande parte das aplicações de processamento de imagens utilizam o filtro Gabor para extrair componentes em determinada orientação e frequência. Este processo envolve a filtragem da imagem por um banco direcional de filtros Gabor 2D. A filtragem por um banco de filtros é uma das etapas mais caras no processamento da imagem. Portanto, qualquer ganho obtido nesta etapa se torna extremamente importante.

Normalmente o filtro Gabor 2D não pode ser separado em dois componentes 1D independentes, exceto nas direções de 0° e 90° . No entanto, quando o filtro Gabor 2D em qualquer direção obedece à condição de possuir um envelope gaussiano circular ele pode ser separado em dois filtros 1D, um no eixo x e outro no eixo y . Esta condição é comum em várias aplicações que utilizam o filtro Gabor, como é o caso de extração de textura das impressões digitais. A filtragem de uma imagem por dois filtros 1D, geralmente, é mais rápida que a filtragem por um filtro 2D equivalente. Neste trabalho, foi mostrado a diminuição no tempo teórico de filtragem, além de mostrar o ganho obtido nas implementações em Matlab do filtro Gabor 2D separado em relação ao não-separado.

Outra contribuição deste estudo foi o desenvolvimento de novos processos de segmentação de impressões digitais: utilizando morfologia matemática e utilizando filtro Gabor. No primeiro caso, o processo de segmentação se vale do padrão característico das impressões digitais de estrias claras e escuras alternadas. Para detectar este padrão, foi utilizado o operador open top-hat. Após a detecção do padrão, são aplicadas outras operações morfológicas para uniformizar a área selecionada. No segundo caso, o processo de segmentação deriva do método proposto em [86]. No método original, a segmentação é feita por setores. A imagem é convoluída com um banco de filtros Gabor em pontos amostrados em cada um dos setores. O desvio padrão nos pontos amostrados nos componentes orientados determina se o bloco pertence ou não à área da impressão digital. Na modificação proposta, a imagem é convoluída com o banco de filtros Gabor em todos os pontos da imagem e, posteriormente, é calculado o desvio padrão para esses pontos. Após esta etapa, são utilizadas operações morfológicas para uniformizar a área de segmentação. As novas propostas apresentadas possuem a vantagem de fazer a segmentação ao nível do pixel, o que permite uma maior precisão.

As contribuições apresentadas neste estudo tornaram o método FingerCode mais robusto, pelo uso de novas funções para o cálculo do escore e de novas técnicas de segmentação. Além do aumento da robustez, o uso da nova implementação do filtro Gabor 2D separado permitiu a diminuição no tempo de processamento. Estas melhorias fazem com que o uso do método FingerCode, e em particular do FingerCode modificado seja uma opção atraente no processo de reconhecimento de impressões digitais.

Bibliografia

- [1] T. B. J. Andrew e N. C. L. David, “Integrated Wavelet and Fourier-Mellin Invariant Feature in Fingerprint Verification System”, Workshop on Biometrics Methods and Applications (WBMA), pp. 82-88, 2003.
- [2] C. Arcelli e G. S. D. Baja, “A Width Independent Fast Thinning Algorithm”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 4, no. 7, pp. 463-474, 1984.
- [3] V. Areekul, U. Watchareeruetai, K. Suppasriwasuseth e S. Tantaratana, “Separable Gabor Filter Realization for Fast Fingerprint Enhancement”, IEEE International Conference on Image Processing (ICIP), pp. 253-256, 2005.
- [4] A. M. Bazen e S. H. Gerez, “Directional Field Computation for Fingerprints Based on the Principal Component Analysis of Local Gradients”, Proc. Workshop on Circuits, Systems and Signal Processing (ProRisc2000), pp. 1-7, 2000.
- [5] A. M. Bazen e S. H. Gerez, “Segmentation of Fingerprint Images”, Proc. Workshop on Circuits, Systems and Signal Processing (ProRISC2001), pp. 276-280, 2001.
- [6] A. M. Bazen e S. H. Gerez, “Extraction of Singular Points from Directional Fields of Fingerprints”, Annual CTIT Workshop, 2001.
- [7] A. M. Bazen e S. H. Gerez, “Systematic Methods for the Computation of the Directional Fields and Singular Points of Fingerprints”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no 7, pp. 905-919, 2002.
- [8] A. M. Bazen e R. N. J. Veldhuis, “Likelihood Ratio-Based Biometric Verification”, IEEE Trans. Circuits and Systems for Video Technology, vol. 14, no. 1, pp. 86-94, 2004.
- [9] S. Bernard, N. Boujemaa, D. Vitale e C. Bricot, “Fingerprint Segmentation Using the Phase of Multiscale Gabor Wavelets”, Proc. Asian Conference on Computer Vision (ACCV), 2002.

- [10] J. de Boer, A. M. Bazen e S. H. Gerez, “Indexing Fingerprint Databases Based on Multiple Features”, Proc. Workshop on Circuits, Systems and Signal Processing (ProRISC2001), pp. 300-306, 2001.
- [11] R. Cappelli, A. Lumini, D. Maio e D. Maltoni, “Fingerprint Classification by Directional Image Partitioning”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 21, no. 5, pp. 402-421, 1999.
- [12] S. S. Chikkerur, “Online Fingerprint Verification System”, Msc. Thesis, State University of New York at Buffalo, 2005.
- [13] S. Chikkerur, S. Pankanti, N. Ratha e V. Govindaraju, “Singular Point Detection in Fingerprint Images Using Linear Phase Portraits”.
- [14] S. Chikkerur, C. Wu e V. Govindaraju, “A Systematic Approach for Feature Extraction in Fingerprint Images”, First International Conference on Biometric Authentication (ICBA), pp. 344-350, 2004.
- [15] G. Cristobal and R. Navarro, “Space and Frequency Variant Image Enhancement Based on a Gabor Representation”, Pattern Recognition Letters, vol. 15, pp. 273-277, 1994.
- [16] J. G. Daugman, “Two-Dimensional Spectral Analysis of Cortical Receptive Field Profiles”, Vision Research, vol. 20, pp. 847–856, 1980.
- [17] J. G. Daugman, “Uncertainty Relation for Resolution in Space, Spatial Frequency, and Orientation Optimized by Two Dimensional Visual Cortical Filters”, Journal Optical Society American, vol. 2, pp. 1160–1169, 1985.
- [18] J. G. Daugman, “High Confidence Recognition of Persons by a Test of Statistical Independence”, IEEE Transactions on Pattern Analysis Machine Intelligence, vol. 15, no. 11, pp. 1148-1161, 1993.
- [19] J. G. Daugman, “Complete Discrete 2-D Gabor Transforms by Neural Networks for Image Analysis and Compression”, IEEE Trans. Acoustic, Speech, and Signal Processing, vol. 36, pp. 1169-1179, 1998.
- [20] J. Daugman, “How Iris Recognition Works”, IEEE Transactions on Circuits and Systems for Video Technology, vol. 14, no. 1, pp. 21-30, 2004.
- [21] E. R. Dougherty e R. A. Lotufo, “Hands-On Morphological Image Processing”, SPIE-International Society for Optical Engine, 2003.
- [22] R. O. Duda, P. E. Hart e D. G. Stork, “Pattern Classification”, Wiley, 2001.

- [23] H. Faulds, “A Manual of Practical Dactylography”, The Police Review, 1923.
- [24] F. A. Fernandez, J. F. Aguilar e J. O. Garcia, “An Enhanced Gabor Filter-Based Segmentation Algorithm for Fingerprint Recognition Systems”, International Symposium on Image and Signal Processing and Analysis, pp. 239-244, 2005.
- [25] D. Gabor, “Theory of communication”, Journal of the Institute for Electrical Engineers 93, 429–439, 1946.
- [26] F. Galton, “Finger Prints”, Macmillan, 1892.
- [27] M. Gamassi, V. Piuri e F. Scotti, “Fingerprint Local Analysis for High-Performance Minutiae Extraction”, International Conference on Image Processing (ICIP), pp. 265-268, 2005.
- [28] R. C. Gonzalez e R. E. Woods, “Digital Image Processing”, Addison-Wesley, 1992.
- [29] S. Greenberg, M. Aladjem, D. Kogan e I. Dimitrov, “Fingerprint Image Enhancement Using Filtering Techniques”, Proc. Int. Conf. on Pattern Recognition, vol. 3, pp. 326-329, 2000.
- [30] S. E. Grigorescu, N. Petkov e P. Kruizinga, “Comparison of Texture Features Based on Gabor Filters”, IEEE Transactions on Image Processing, vol. 11, no. 10, pp. 1160-1167, 2002.
- [31] C. Grigorescu, N. Petkov e M. A. Westnberg, “Contour Detection Based on Nonclassical Receptive Field Inhibition”, IEEE Transactions on Image Processing, vol. 12, no. 7, pp. 729-739, 2003.
- [32] E. R. Henry, “Classification and Uses of Finger Prints”, George Routledge, 1900.
- [33] W. J. Herschel, “The Origin of Finger-Printing”, Humphrey Milford, 1916.
- [34] L. Hong, Y. Wan e A. K. Jain, “Fingerprint Image Enhancement: Algorithms and Performance Evaluation”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 20, no. 8, pp. 777-789, 1998.
- [35] C. T. Hsieh, E. Lai e Y. C. Wang, “An effective Algorithm for Fingerprint, Image Enhancement Based on Wavelet Transform”, Pattern Recognition, vol. 36, pp. 303-312, 2003.
- [36] K. Ito, A. Morita, T. Aoki, T. Higuchi, H. Nakajima e K. Kobayashi, “A Fingerprint Recognition Algorithm Using Phase-Based Image Matching for Low-Quality Fingerprints”, International Conference on Image Processing (ICIP), pp. 33-36, 2005.

- [37] A. K. Jain, "Fundamentals of Digital Image Processing", Prentice Hall, 1989.
- [38] A. K. Jain e F. Farrokhnia, "Unsupervised Texture Segmentation Using Gabor Filters", Pattern Recognition, vol. 24, no. 12, pp. 1167-1186, 1991.
- [39] A. Jain, L. Hong e R. Bolle, "On-Line Fingerprint Verification", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, no. 4, pp. 302-313, 1997.
- [40] A. Jain, L. Hong, S. Pankanti e R. Bolle, "An Identity Authentication System Using Fingerprints", Proceedings of IEEE, vol. 85, no. 9, pp. 1365-1388, 1997.
- [41] A. K. Jain, M. N. Murty e P. J. Flynn, "Data Clustering: A Review", ACM Computing Surveys, vol. 31, no. 3, 1999.
- [42] A. Jain e S. Pankanti, "Fingerprint Classification and Matching", 1999.
- [43] A. K. Jain, S. Prabhakar, L. Hong e S. Pankanti, "FingerCode: A Filterbank for Fingerprint Representation and Matching", Proc. IEEE Conf. on Computer Vision and Pattern Recognition, vol. 2, pp. 187-193, 1999.
- [44] A. K. Jain, S. Prabhakar, L. Hong e S. Pankanti, "Filterbank-Based Fingerprint Matching", IEEE Transactions on Image Processing, vol. 9, no. 5, pp. 846-859, 2000.
- [45] A. K. Jain, N. K. Ratha e S. Lakshmanan, "Object Detection Using Gabor Filters", Pattern Recognition, vol. 30, pp. 295-309, 1997.
- [46] A. Jain, A. Ross e S. Prabhakar, "Fingerprint Matching Using Minutiae and Texture Features", Proc. Int. Conf. on Image Processing (ICIP), pp. 282-285, 2001.
- [47] X. Jiang, "Fingerprint Image Ridge Frequency Estimation by High Order Spectrum", Proc. Int. Conf. on Image Processing, vol. 1, pp. 462-465, 2000.
- [48] X. Jiang e W. Y. Yau, "Fingerprint Minutiae Matching Based on the Local and Global Structures", Proc. Int. Conf. on Pattern Recognition, vol. 2, pp. 1042-1045, 2000.
- [49] M. Kass e A. Witkin, "Analyzing Oriented Patterns", Computer Vision, Graphics, and Image Processing, vol. 37, no. 3, pp. 362-385, 1987.
- [50] M. Kawagoe e A. Tojo, "Fingerprint Pattern Classification", Pattern Recognition, vol. 17, pp. 295-303, 1984.
- [51] J. Kittler, M. Hatef, R. P. W. Dui e J. Matas, "On Combining Classifiers", IEEE PAMI, vol. 20, no. 3, 1998.
- [52] J. Z. Kolter e M. A. Maloof, "Dynamic Weighted Majority: A New Ensemble Method for Tracking Concept Drift", Proc. Int. Conf. on Data Mining, pp. 123-130, 2003.

- [53] Z. M. Kovács-Vajna, R. Rovatti e M. Frazzoni, “Fingerprint Ridge Distance Computation Methodologies”, *Pattern Recognition*, vol. 33, no. 1, pp. 69-80, 2000.
- [54] C. J. Lee, S. D. Wang, “Fingerprint Feature Extraction Using Gabor Filters”, *Electronics Letters*, vol. 35, no. 4, pp. 288-290, 1999.
- [55] J. S. Lim, “Two-Dimensional Signal and Image Processing”, Prentice-Hall, 1990.
- [56] N. Littlestone e M. Warmuth, “The Weighted Majority Algorithm”, *Information and Computation*, vol. 108, pp. 212–261, 1994.
- [57] D. Maio e D. Maltoni, “A Structural Approach to Fingerprint Classification”, *Proc. Int. Conf. on Pattern Recognition*, 1996.
- [58] D. Maio e D. Maltoni, “Direct Gray-Scale Minutiae Detection in Fingerprints”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 1, pp. 27-40, 1997.
- [59] D. Maltoni, D. Maio, A. K. Jain e S. Prabhakar, “Handbook of Fingerprint Recognition”, Springer, 2003.
- [60] B. S. Manjunath, C. Shekhar e R. Chellappa, “A New Approach to Image Feature Detection with Applications”, *Pattern Recognition*, vol. 29, pp. 627-640, 1996.
- [61] G. L. Marcialis e F. Roli, “High Security Fingerprint Verification by Perceptron-Based Fusion of Multiple Matchers”, *Multiple Classifier Systems*, pp. 364-373, 2004.
- [62] G. L. Marcialis, F. Roli e P. Frasconi, “Fingerprint Classification by Combination of Flat and Structural Approaches”, *Proc. Int. Conf. on Audio- and Video-Based Biometric Person Authentication (AVBPA)*, pp. 241-246, 2001.
- [63] B. M. Mehre, “Fingerprint Image Analysis for Automatic Identification”, *Machine Vision and Applications*, vol. 6, no.2-3, pp. 124-139, 1993.
- [64] B. M. Mehre e B. Chatterjee, “Segmentation of Fingerprint Images – A Composite Method”, *Pattern Recognition*, vol. 22, no. 4, pp. 381-385, 1989.
- [65] J. K. Min, J. H. Hong, and S. B. Cho, “Effective Fingerprint Classification by Localized Models of Support Vector Machine”, *International Conference on Biometrics (ICB)*, pp. 287-293, 2006.
- [66] J. R. Movellan, “Tutorial on Gabor Filters”, kolmogorov.sourceforge.net, 2002.
- [67] M. U. Munir e M. Y. Javed, “Fingerprint Matching Using Gabor Filters”, *National Conference on Emerging Technologies*, pp. 147-151, 2004.

- [68] O. Nestares, R. Navarro, J. Portilla e A. Taberero, “Efficient Spatial-Domain Implementation of a Multiscale Image Representation Based on Gabor Functions”, *J. Electronic Imaging*, vol. 7, pp. 166-173, 1998.
- [69] K. Nilsson e J. Bigun, “Localization of Corresponding Points in Fingerprints by Complex Filtering”, *Pattern Recognition Letters*, vol. 24, no. 13, pp. 2135-2144, 2003.
- [70] L. O’Gorman e J. V. Nickerson, “Matched Filter Design for Fingerprint Image Enhancement”, *Proc. Int. Conf. on Acoustic, Speech and Signal Processing*, pp. 916-919, 1988.
- [71] L. O’Gorman e J. V. Nickerson, “An Approach to Fingerprint Filter Design”, *Pattern Recognition*, vol. 22, no. 1, pp. 29-38, 1989.
- [72] A. V. Oppenheim, R. W. Schafer e J. R. Buck, “Discrete-Time Signal Processing”, Prentice Hall, 1999.
- [73] C. H. Park, S. K. Oh, D. M. Kwak, B. S. Kim, Y. C. Song e K. H. Park, “A New Reference Point Detection Algorithm Based on Orientation Pattern Labeling in Fingerprint Images”, *Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA)*, pp. 697-703, 2003.
- [74] P. Perona, “Orientation Diffusions”, *IEEE Transactions on Image Processing*, vol. 7, no. 3, pp. 457-467, 1998.
- [75] S. Prabhakar, “Fingerprint Classification and Matching Using a Filterbank”, PhD Thesis, Michigan State University, 2001.
- [76] S. Prabhakar, A. K. Jain, “Decision-Level Fusion in Fingerprint Verification”, *Pattern Recognition*, vol. 35, pp. 861-874, 2002.
- [77] S. Prabhakar, A. K. Jain e S. Pankanti, “Learning Fingerprint Minutiae Location and Type”, *Pattern Recognition*, vol. 36, no. 1, pp. 1847-1857, 2003.
- [78] S. Qiu, F. Zhou e P. E. Crandall, “Discrete Gabor Transforms with Complexity $O(M\log N)$ ”, *Signal Processing*, vol. 77, pp. 159-170, 1999.
- [79] A. R. Rao, “A Taxonomy for Texture Description and Identification”, Springer, 1990.
- [80] N. K. Ratha, S. Chen e A. K. Jain, “Adaptative Flow Orientation-Based Feature Extraction in Fingerprint Images”, *Pattern Recognition*, vol. 28, no. 11, pp. 1657-1672, 1995.

- [81] N. K. Ratha, V. D. Pandit, R. M. Bolle e V. Vaish, “Robust Fingerprint Authentication Using Local Structure Similarity”, Proc. Workshop on Applications of Computer Vision, pp. 29-34, 2000.
- [82] K. Rerkrai e V. Areekul, “A New Reference Point for Fingerprint Recognition”, Proc. Int. Conf. on Image Processing, vol. 2, pp. 499-502, 2000.
- [83] A. Ross, A. Jain e J. Reisman, “A Hybrid Fingerprint Matcher”, Proc. Int. Conf. on Pattern Recognition (ICPR), pp. 11-15, 2002.
- [84] A. Ross, A. Jain e J. Reisman, “A Hybrid Fingerprint Matcher”, Pattern Recognition vol. 36, pp. 1661-1673, 2003.
- [85] A. Senior, “A Combination Fingerprint Classifier”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 10, pp. 1165-1174, 2001.
- [86] L. Shen, A. Kot e W. Koo, “Quality Measures of Fingerprint Images”, Proc. Int. Conf. on Audio- and Video-Based Biometric Person Authentication (AVBPA), pp. 266-271, 2001.
- [87] S. Tan, J. L. Dale e A. Johnston, “Performance Evaluation of Three Recursive Algorithms for Fast-Variant Gaussian Filtering”, Real-Time Imaging, vol. 9, pp. 215-228, 2003.
- [88] M. Tico and P. Kuosmanen, “Fingerprint Matching Using an Orientation-Based Minutia Descriptor”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25, no. 8, pp. 1009-1014, 2003.
- [89] R. Thai, “Fingerprint Image Enhancement and Minutiae Extraction”, Msc. Thesis, University of Western Australia, 2003.
- [90] B. Trigs e M. Sdika, “Boundary Conditions for Young – van Vliet Recursive Filtering”, IEEE Transactions on Signal Processing, 2005.
- [91] K. Venkataramani, and B. V. K. V. Kumar, “Fingerprint Verification Using Correlation Filters”, Proc. Int. Conf. on Audio- and Video-Based Biometric Person Authentication (AVBPA), pp. 886-894, 2003.
- [92] B. Vidakovic e P. Mueller, “Wavelet for Kids – A Tutorial Introduction”, Duke University, 1991.
- [93] S. Wang e Y. Wang, “Fingerprint Enhancement in The Singular Point Area”, IEEE Signal Processing Letters, vol. 11, no. 1, 2004.
- [94] S. Wang, W. W. Zhang e Y. S. Wang, “Fingerprint Classification by Directional Fields”, IEEE International Conference on Multimodal Interfaces (ICMI), pp. 395-399, 2002.

- [95] C. L. Wilson, G. T. Candela e C. I. Watson, “Neural Network Fingerprint Classification”, *Journal of Artificial Neural Networks*, vol. 1, no. 2, pp. 203-228, 1994.
- [96] C. Wu, Z. Shi e V. Govindaraju, “Fingerprint Image Enhancement Method Using Directional Median Filter”, *Proc. SPIE*, vol. 5404, pp. 66-75, 2004.
- [97] J. Yang, L. Liu, T. Jiang e Y. Fan, “A Modified Gabor Filter Design Method for Fingerprint Image Enhancement”, *Pattern Recognition Letters*, vol. 24, pp. 1805-1817, 2003.
- [98] Y. Yao, P. Frasconi e M. Pontil, “Fingerprint Classification with Combinations of Support Vector Machines”, *Proc. Int. Conf. on Audio- and Video-Based Biometric Person Authentication (AVBPA)*, pp. 253-258, 2001.
- [99] Y. Yao, G. L. Marcialis, M. Pontil, P. Frasconi e F. Roli, “Combining Flat and Structured Representations for Fingerprint Classification with Recursive Neural Networks and Support Vector Machines”, *Pattern Recognition*, 36, pp. 397-406, 2003.
- [100] I. T. Young, L. J. van Vilet e M. van Ginkel, “Recursive Gabor Filtering”, *IEEE Transactions on Signal Processing*, vol. 50, no. 11, pp. 2798-2805, 2002.
- [101] Y. W. Yun, “The ‘123’ of Biometric Technology”, 2003.
- [102] Q. Zhang, K. Huang e H. Yan, “Fingerprint Classification Based on Extraction and Analysis of Singularities and Pseudoridges”, *Pan-Sydney Area Workshop on Visual Information Processing (VIP)*, 2001.
- [103] D. S. Zorita, J. O. Garcia, J. F. Aguilar e J. G. Rodriguez, “Image Quality and Position Variability Assessment in Minutiae-Based Fingerprint Verification”, *IEE Proceedings Vision, Image and Signal Processing*, vol. 150, no. 6, pp. 402-408, 2003.
- [104] Homepage da Fingerprint Verification Competition 2002 (FVC2002), <http://bias.csr.unibo.it/fvc2002>, março, 2006.
- [105] Homepage da Griaule, <http://www.griaule.com.br>, março, 2006.
- [106] Homepage da Galton.org, <http://galton.org>, março, 2006.
- [107] Biometrics e Fingerprint, <http://www.wikipedia.org>, março, 2006.
- [108] History of Fingerprint, <http://www.policensw.com>, março, 2006.
- [109] History of Fingerprints – A Time Line, <http://www.scro.police.uk>, março, 2006.
- [110] Early Fingerprint Pioneers, <http://www.ridgesandfurrows.homestead.com>, março, 2006.