UNIVERSIDADE ESTADUAL PAULISTA

"Júlio de Mesquita Filho"

Pós-Graduação em Ciência da Computação

Guilherme José da Costa Kami

Análise de Técnicas de Reconhecimento de Padrões para a Identificação Biométrica de Usuários em Aplicações WEB Utilizando Faces a Partir de Vídeos

Bauru

2011

Guilherme José da Costa Kami

Análise de Técnicas de Reconhecimento de Padrões para a Identificação Biométrica de Usuários em Aplicações WEB Utilizando Faces a Partir de Vídeos

Dissertação apresentada para obtenção do título de Mestre em Ciência da Computação, área de concentração "Sistemas de Computação" na linha de pesquisa "Processamento de Imagens e Visão Computacional", junto ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Biociências, Letras e Ciências Exatas da Universidade Estadual Paulista "Júlio de Mesquita Filho", Campus de São José do Rio Preto.

Orientador: Prof. Dr. Aparecido Nilceu Marana

Bauru, Agosto de 2011

Kami, Guilherme José da Costa.
Análise de técnicas de reconhecimento de padrões para a identificação biométrica de usuários em aplicações web utilizando faces a partir de vídeos / Guilherme José da Costa Kami. - São José do Rio Preto: [s.n.], 2011.
95 f. : il.; 30 cm.

Orientador: Aparecido Nilceu Marana
Dissertação (mestrado) - Universidade Estadual Paulista, Instituto de Biociências, Letras e Ciências Exatas

1. Computação. 2. Biometria. 3. Reconhecimento de padrões. 4. Processamento de imagens. I. Marana, Aparecido Nilceu. II. Universidade Estadual Paulista, Instituto de Biociências, Letras e Ciências Exatas. III. Título.

CDU – 57.087.1

Ficha catalográfica elaborada pela Biblioteca do IBILCE
Campus de São José do Rio Preto - UNESP

Guilherme José da Costa Kami

Análise de Técnicas de Reconhecimento de Padrões para a Identificação Biométrica de Usuários em Aplicações WEB Utilizando Faces a Partir de Vídeos

Dissertação apresentada para obtenção do título de Mestre em Ciência da Computação, área de concentração "Sistemas de Computação" na linha de pesquisa "Processamento de Imagens e Visão Computacional", junto ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Biociências, Letras e Ciências Exatas da Universidade Estadual Paulista "Júlio de Mesquita Filho", Campus de São José do Rio Preto.

BANCA EXAMINADORA

Prof. Dr. Aparecido Nilceu Marana
Professor Doutor
UNESP – Bauru
Orientador

Prof. Dr. Hélio Pedrini
Professor Doutor
UNICAMP - Campinas

Prof. Dr. Aledir Silveira Pereira
Professor Doutor
UNESP – São José do Rio Preto

Bauru, 5 de Agosto de 2011

Aos meus pais, Aparecida Lourdes e Koitiro.

Agradecimentos

Agradeço, acima de tudo, a Deus pela oportunidade de realização deste trabalho, pelos meios para conseguir realizá-lo e também...

... aos meus pais e família em geral, pelo grande apoio em todos os aspectos possíveis;

... à minha namorada, Aline, pela compreensão, atenção e carinho em todos os momentos;

... à minha irmã, Camila, pela ajuda relacionada à confecção de textos;

... ao Igor Bonadio, pelas discussões sobre várias idéias utilizadas neste trabalho;

... ao Bruno Penteado, por fornecer o sistema para o qual foi dado continuidade neste trabalho e também por ajudar em vários aspectos técnicos.

... à MSTech, pela oportunidade de conciliar os estudos à profissão e a todos os funcionários que colaboraram prontamente para a confecção da base de dados de vídeos;

... ao João Paulo Papa, pela enorme ajuda no entendimento de vários conceitos e prontidão para atendimento nos horários mais adversos;

... ao prof. Nilceu, pelo real trabalho da orientação com total empenho e dedicação, além da grande paciência em todos momentos durante toda a fase de elaboração deste trabalho;

... a todos os outros que contribuíram de alguma forma para a elaboração deste trabalho.

"Dicionário é o único lugar no qual o sucesso vem antes do trabalho. O trabalho árduo é o preço a ser pago pelo sucesso. Você pode realizar qualquer tarefa se estiver disposto a pagar o preço."

Vince Lombardi

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Siglas

AAM	*Active Appearance Model*
ANN	*Artificial Neural Network*
CAR1	*Clustering by Attratction and Repulsion*
CAR2	*Clustering by Attraction and Repulsion global Optimization*
DNA	*Deoxyribonucleic Acid*
EBGM	*Elastic Bunch Graph Matching*
EER	*Equal Error Rate*
EGM	*Elastic Graph Matching*
EM	*Expectation Maximization*
FANN	*Fast Artificial Neural Network*
FAR	*False Acceptance Rate*
FBG	*Face Bunch Graph*
FLD	*Fisher's Linear Discriminant*
FN	Falso Negativo
FP	Falso Positivo
FRR	*False Rejection Rate*
FTC	*Failure To Capture*
FTE	*Failure To Enroll*
GMM	*Gaussian Mixture Model*
HMM	*Hidden Markov Model*
HTK	*Hidden Markov Model Toolkit*
IFT	*Image Foresting Transform*
KNN	*K-Nearest Neighbors*
MAP	*Maximum a Posteriori*
MLP	*Multi Layer Perceptron*

MRTD	*Machine Readable Travel Documents*
MSM	*Mutual Subspace Method*
NN	*Nearest Neighbor*
OPF	*Optimum Path Forest*
PCA	*Principal Component Analysis*
RAM	*Random Acess Memory*
RBF	*Radial Basis Function*
ROC	*Receiver Operating Characteristic*
SOM	*Self Organizing Maps*
SVM	*Support Vector Machines*
UCSD	*University of California San Diego*

Resumo

As técnicas para identificação biométrica têm evoluído cada vez mais devido à necessidade que os seres humanos têm de identificar as pessoas em tempo real e de forma precisa para permitir o acesso a determinados recursos, como por exemplo, as aplicações e serviços WEB. O reconhecimento facial é uma técnica biométrica que apresenta várias vantagens em relação às demais, tais como: uso de equipamentos simples e baratos para a obtenção das amostras e a possibilidade de se realizar o reconhecimento em sigilo e à distância. O reconhecimento de faces a partir de vídeo é uma tendência recente na área de Biometria. Esta dissertação tem por objetivo principal comparar diferentes técnicas de reconhecimento facial a partir de vídeo para determinar as que apresentam um melhor compromisso entre tempo de processamento e precisão. Outro objetivo é a incorporação dessas melhores técnicas no sistema de autenticação biométrica em ambientes de *E-Learning*, proposto em um trabalho anterior. Foi comparado o classificador vizinho mais próximo usando as medidas de distância Euclidiana e Mahalanobis com os seguintes classificadores: Redes Neurais MLP e SOM, K Vizinhos mais Próximos, Classificador Bayesiano, Máquinas de Vetores de Suporte (SVM) e Floresta de Caminhos Ótimos (OPF). Também foi avaliada a técnica de Modelos Ocultos de Markov (HMM). Nos experimentos realizados com a base *Recogna Video Database*, criada especialmente para uso neste trabalho, e *Honda/UCSD Video Database*, os classificadores apresentaram os melhores resultados em termos de precisão, com destaque para o classificador SVM da biblioteca SVM Torch. A técnica HMM, que incorpora informações temporais, apresentou resultados melhores do que as funções de distância, em termos de precisão, mas inferiores aos classificadores.

Palavras-chave: Biometria, reconhecimento facial, vídeo, aplicações WEB, classificadores de padrões, Modelos Ocultos de Markov.

Abstract

The biometric identification techniques have evolved increasingly due to the need that humans have to identify people in real time to allow access to certain resources, such as applications and Web services. Facial recognition is a biometric technique that has several advantages over others. Some of these advantages are the use of simple and cheap equipment to obtain the samples and the ability to perform the recognition in covert mode. The face recognition from video is a recent approach in the area of Biometrics. The work in this dissertation aims at comparing different techniques for face recognition from video in order to find the best rates on processing time and accuracy. Another goal is the incorporation of these techniques in the biometric authentication system for E-Learning environments, proposed in an earlier work. We have compared the nearest neighbor classifier using the Euclidean and Mahalanobis distance measures with some other classifiers, such as neural networks (MLP and SOM), k-nearest neighbor, Bayesian classifier, Support Vector Machines (SVM), and Optimum Path Forest (OPF). We have also evaluated the Hidden Markov Model (HMM) approach, as a way of using the temporal information. In the experiments with *Recogna Video Database*, created especially for this study, and *Honda/UCSD Video Database*, the classifiers obtained the best accuracy, especially the SVM classifier from the SVM Torch library. HMM, which takes into account temporal information, presented better performance than the distance metrics, but worse than the classifiers.

Keywords: Biometrics, facial recognition, video, Web application, pattern classifiers, Hidden Markov Models.

Capítulo 1

Introdução

Com o desenvolvimento contínuo da Internet, a mesma se torna cada vez mais veloz, com mais recursos e mais acessível à população. Diante disso, empresas aproveitam para fornecer os mais diferenciados serviços *online*. Muitas vezes, esses serviços precisam ser restringidos para um determinado grupo de indivíduos, o que é em geral realizado por meio de autenticação baseada em senhas.

Um dos serviços mais importantes que tem sido disponibilizado à sociedade por meio da Internet é o curso para educação à distância. A proliferação dos sistemas de *e-Learning* é conseqüência das vantagens da educação baseada na Web, tais como: custos de distribuição diminuídos, aprendizado auto-dirigido, aprendizado geograficamente independente, atualização simples de materiais, gerenciamento simplificado de grandes grupos de estudantes e assim por diante (CANTONI et al., 2004). No entanto, a falta de mecanismos adequados para assegurar a identidade remota, e correta, dos alunos desses cursos a distância tem sido apontada como uma séria deficiência (RABUZIN et al., 2006).

Com a popularização e, portanto, a queda nos preços dos dispositivos de hardware que possibilitam a captura das informações biométricas, como leitores de impressões digitais, microfones e web câmeras, o uso de técnicas e sistemas biométricos de identificação de usuários para aplicações Web são cada vez mais factíveis.

O desenvolvimento de aplicações Web exige mais atenção em alguns aspectos do que o desenvolvimento de aplicações que funcionam *offline*, tais como: distribuição de carga entre cliente e servidor, interceptação indesejada de dados que trafegam na rede, baixa velocidade de comunicação entre cliente e servidor, entre outros. Um sistema biométrico criado para ser utilizado *online* deve ser projetado levando em consideração tais fatores além de utilizar algo-

ritmos que tenham tempo de processamento aceitável para essa finalidade e que possam lidar da melhor forma possível com problemas decorrentes da interação do usuário com o computador, durante a utilização da aplicação Web. No caso de reconhecimento de faces, por exemplo, pode ocorrer grande variação nas poses e nas condições de iluminação, além de oclusões parciais da face.

Penteado e Marana (2009) propuseram uma arquitetura para sistemas de autenticação biométrica de usuários em ambientes de e-Learning, baseada em reconhecimento de faces a partir de vídeo e avaliaram o desempenho desta tecnologia biométrica. De acordo com a arquitetura proposta, o sistema de autenticação biométrica pode ser acoplado a qualquer ambiente de educação a distância, independentemente da linguagem na qual o ambiente tenha sido desenvolvido. A arquitetura proposta tem também como objetivo a exploração das características intrínsecas de aplicações para Internet, buscando eficiência no tráfego da rede, distribuição de carga entre clientes e servidores e independência de plataforma de e-Learning adotada.

Apesar da motivação inicial ter sido a autenticação de usuários em sistemas de e-Learning, a arquitetura proposta por Penteado e Marana (2009) mostrou-se genérica a qualquer tipo de sistema Web, pois não requer nenhuma configuração adicional no sistema no qual a autenticação será aplicada.

A estratégia de identificação de usuários proposta por Penteado e Marana (2009) utiliza a maioria de votos para determinar a identidade do usuário da aplicação a partir de um determinado trecho do vídeo. Para o reconhecimento das faces, é utilizada a técnica de análise das componentes principais para obter os descritores, combinada com uma função de distância. O objetivo deste trabalho foi aprimorar essa estratégia utilizando-se técnicas avançadas de reconhecimento de padrões, como classificação e modelos markovianos, para o reconhecimento das faces dos usuários das aplicações Web.

1.1 Objetivos

Esta dissertação de mestrado tem como objetivo principal o estudo, a implementação e a comparação de diferentes métodos de reconhecimento de padrões, tais como classificação e modelos Markovianos, para promover o reconhecimento rápido e eficiente de faces obtidas de vídeos capturados por web câmeras, e, desse modo, possibilitar a identificação *online* e mais robusta de usuários de aplicações Web.

Outro objetivo é a incorporação de técnicas mais adequadas e eficazes ao sistema de autenticação biométrica de usuários de aplicações WEB proposto e desenvolvido Penteado e Marana

(2009).

1.2 Estrutura da Dissertação

Esta dissertação está estruturada da seguinte forma:

- No capítulo 1 são apresentadas as motivações, a importância e os objetivos da realização deste trabalho.
- No capítulo 2 são apresentados conceitos sobre identificação de pessoas, diferenciando-se a identificação tradicional da biométrica.
- No capítulo 3 são apresentadas sucintamente as principais técnicas de classificação de padrões encontradas na literatura.
- No capítulo 4 são descritas técnicas para o reconhecimento facial em vídeo por meio de duas abordagens: a que faz uso de imagens estáticas e a que faz uso de vídeo, sendo que a abordagem que faz uso de vídeo pode considerar a informação temporal ou não.
- No capítulo 5 é apresentada a estratégia de autenticação biométrica de usuários em aplicações WEB proposta por Penteado e Marana.
- No capítulo 6 é apresentada, de forma sucinta a proposta deste trabalho.
- No capítulo 7 são apresentados o material utilizado neste trabalho e a metodologia proposta para a identificação de pessoas a partir de faces obtidas em vídeos.
- No capítulo 8 são descritos os experimentos realizados e apresentados os resultados obtidos.
- No capítulo 9 são feitas discussões sobre os resultados e apresentadas as conclusões deste trabalho.

Capítulo 2

Identificação de Pessoas

A identificação pessoal pode ser definida como o processo de associar uma identidade a uma pessoa (BOLLE; PANKANTI, 1998). Por meio de tal processo, é possível permitir ou negar o acesso a determinado recurso, tal como informações, documentos, websites, áreas restritas, entre outros. Desta forma, somente as pessoas para as quais o acesso foi liberado poderão acessar os recursos restritos.

A identificação de pessoas é imprescindível para a interação de seres humanos em comunidade. Os humanos usam, de forma inconsciente, características como face, voz, forma de andar, dentre outras, para reconhecer outros humanos no dia-a-dia.

Na história da civilização, existem vários relatos do uso de características biométricas como forma de identificação de pessoas. Na Nova Escócia, Canadá, foram encontradas pinturas com o desenho de mãos com diferentes padrões de minúcias, cujas evidências levam a crer que as mesmas foram feitas séculos antes de Cristo. Também existem evidências do uso de identificação biométrica (impressões digitais) na Babilônia, aproximadamente a 500 anos A.C.. Na história do Egito antigo, comerciantes eram identificados pelos seus descritores físicos, tornando possível diferenciar comerciantes confiáveis e com reputação conhecida, dos comerciantes novos no mercado. Mais recentemente, o explorador e escritor espanhol João de Barros relatou que os mercadores chineses utilizavam a impressão digital para fechar negócios no século XIV (NSTC, 2006).

A tarefa de identificação de pessoas, embora possa parecer simples, tornou-se cada vez mais desafiadora à medida que as populações cresceram e os meios de transporte e comunicação evoluíram. Diante da dificuldade no reconhecimento de pessoas a partir apenas de habilidades humanas é que se iniciou a busca de métodos mais robustos para a realização desta tarefa.

Vários mecanismos foram propostos. Dentre estes mecanismos tradicionais presentes no nosso dia-a-dia, estão os documentos de identidade, os cartões bancários, os registros governamentais, entre outros.

Esses mecanismos utilizados para identificação de pessoas apresentam alguns problemas, tais como: possibilidade de perda, deterioração, facilidade de falsificação, empréstimo ou cópia de documento ou cartão de identificação, dificuldade para memorização de códigos de acesso, grande quantidade de códigos e cartões e assim por diante.

Em sistemas computacionais, a identificação tradicional de usuários é baseada principalmente em cartões (posse) ou senhas (conhecimento) que podem facilmente ser perdidos ou esquecidas. Esses problemas podem ser minimizados pelo uso de autenticação biométrica (Biometria), que faz com que a interação homem-máquina para autenticação de usuários seja mais natural, conveniente e segura.

2.1 Identificação Biométrica

Biometria pode ser definida como o estudo das medidas e de estruturas e órgãos de seres vivos, bem como da importância funcional dessas medidas (HOUAISS, 2009). A palavra tem origem no grego *bios* (vida) e *metron* (mensuração). A Biometria se baseia no fato de que certas características físicas ou comportamentais possibilitam a diferenciação entre todos os seres que as apresentam. A Biometria também pode ser definida como sendo o "reconhecimento pessoal baseado em características comportamentais ou fisiológicas de um indivíduo"(PRABHAKAR et al., 2003).

Como exemplos de características físicas (anatômicas ou fisiológicas), destacam-se as impressões digitais, a aparência facial, o padrão da íris, a geometria das mãos, o DNA, o padrão de veias das mãos, a retina e a geometria das orelhas. Na Figura 2.1 são apresentados exemplos das características físicas mais comumente utilizadas na identificação de pessoas.

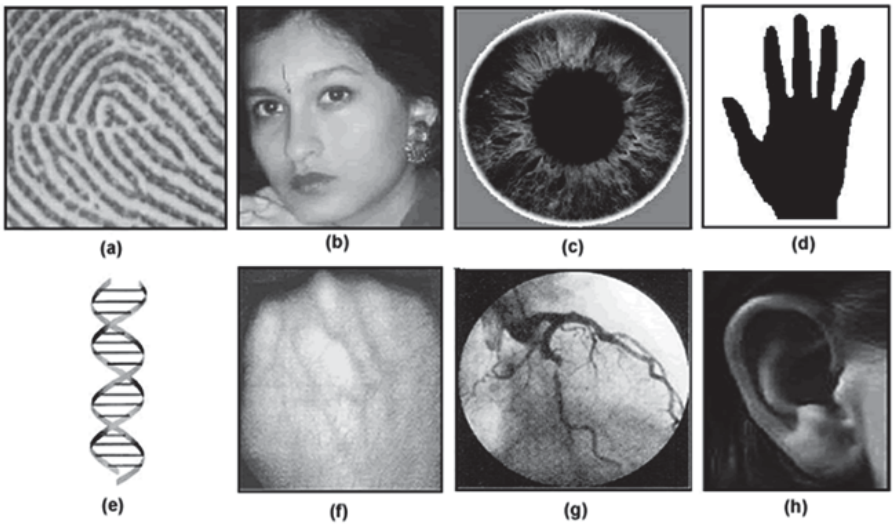

Figura 2.1: Características físicas: (a) Impressão Digital; (b) Aparência Facial; (c) Íris; (d) Geometria das Mãos; (e) DNA; (F) Padrão das Veias das Mãos; (g) Retina; (h) Geometria das Orelhas (JAIN; MALTONI, 2003).

As características comportamentais, diferentemente das características físicas, podem ser aprendidas ou treinadas ao longo do tempo. Como exemplos de características comportamentais, destacam-se o modo de andar, a assinatura, o padrão de voz, a dinâmica de digitação, os movimentos faciais e os movimentos labiais. Uma desvantagem dessas características é que elas podem variar muito ao longo do tempo. Na Figura 2.2 são apresentados exemplos de características comportamentais mais comumente utilizadas na identificação de pessoas.

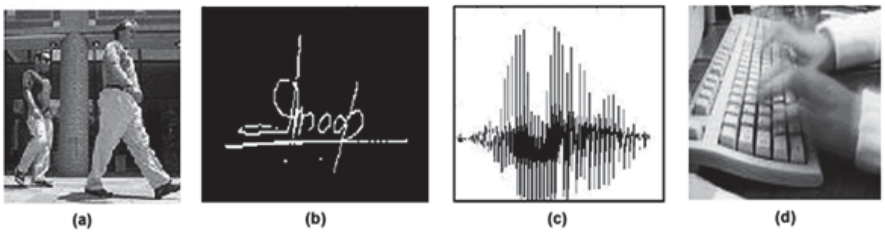

Figura 2.2: Características comportamentais: (a) Modo de Andar; (b) Assinatura; (c) Padrão de Voz; (d) Dinâmica de Digitação (JAIN; MALTONI, 2003).

Segundo Jain et al. (2004), as seguintes propriedades são extremamente desejáveis para que uma determinada característica física ou comportamental possa ser utilizada para a identificação de pessoas:

Universalidade : todas as pessoas devem possuir a característica;

Unicidade : a característica deve ser única para cada pessoa;

Permanência : a característica deve permanecer constante ao longo do tempo;

Coletabilidade : a característica deve ser passível de ser coletada;

Desempenho : a precisão e os recursos exigidos devem respeitar restrições da aplicação;

Aceitabilidade : os indivíduos a serem identificados devem aceitar fornecer suas características ao sistema;

Grau de Impostura : a característica deve ser difícil de ser forjada.

Na Tabela 2.1 são apresentadas algumas características biométricas com uma avaliação realizada por Jain et al. (2004) das suas respectivas propriedades biométricas:

Tabela 2.1: Comparação entre as características biométricas (JAIN et al., 2004).

Biometria	Universalidade	Unicidade	Permanência	Coletabilidade	Desempenho	Aceitabilidade	Impostura
Face	Alta	Baixa	Média	Alta	Baixa	Alta	Baixa
Impressão digital	Média	Alta	Alta	Média	Alta	Média	Média
Geometria das mãos	Média	Média	Média	Alta	Média	Média	Média
Íris	Alta	Alta	Alta	Média	Alta	Baixa	Alta
Veias das mãos	Média	Média	Média	Média	Média	Média	Alta
Orelha	Média	Média	Alta	Média	Média	Alta	Média
Digitação	Média	Média	Baixa	Média	Baixa	Média	Média
Odor	Alta	Alta	Alta	Baixa	Baixa	Média	Alta
DNA	Alta	Alta	Alta	Baixa	Alta	Baixa	Alta
Termograma facial	Alta	Alta	Baixa	Alta	Média	Alta	Alta
Retina	Alta	Alta	Média	Baixa	Alta	Baixa	Alta
Assinatura	Baixa	Baixa	Baixa	Alta	Baixa	Alta	Baixa
Voz	Média	Baixa	Baixa	Média	Baixa	Alta	Baixa
Modo de andar	Média	Baixa	Baixa	Alta	Baixa	Alta	Média

Nenhuma característica biométrica possui todas estas propriedades com o maior grau de aproveitamento. Portanto, nenhuma característica biométrica pode ser considerada ótima, de forma que elas devem ser escolhidas e combinadas de acordo com os requisitos de suas aplicações.

2.2 Sistemas Biométricos

Um sistema biométrico é um sistema de reconhecimento de padrões que realiza o reconhecimento de uma pessoa por meio de características tanto físicas quanto comportamentais (JAIN; MALTONI, 2003).

Dependendo do contexto, um sistema biométrico pode ser classificado como sistema de verificação/autenticação ou sistema de identificação. O primeiro tem como objetivo verificar se uma pessoa é realmente quem ela afirma ser enquanto que o último objetiva descobrir quem é determinada pessoa.

Independentemente do sistema biométrico ser de verificação ou identificação, em ambos existe a fase de cadastro. Tal fase consiste na obtenção das características dos indivíduos que são processadas para se obter uma representação reduzida e expressiva do indivíduo, chamada *template*. Para se garantir que uma amostra é confiável e passível de uso para o sistema é feita uma checagem de qualidade. De acordo com a aplicação, os *templates* podem ser armazenados em uma base de dados centralizada ou, então, em cartões fornecidos aos indivíduos.

No modo de verificação, o usuário a ser reconhecido deve, inicialmente, fornecer sua identidade ao sistema. Após isso, o leitor biométrico captura as características do indivíduo e as converte para formato digital, utilizando as mesmas técnicas empregadas no processo de cadastro. Os descritores obtidos por meio do leitor são comparados pelo sistema apenas com o *template* daquele usuário, registrado na base de dados.

No modo de identificação, nenhuma informação em relação à identidade do indivíduo é fornecida ao sistema. Desta forma, após obter os descritores do indivíduo por meio do leitor biométrico, estes são comparados com todos os *templates* registrados na base de dados. Após essa comparação, o sistema fornece a identidade do indivíduo baseado na maior similaridade encontrada, ou, então, informa que o indivíduo não está cadastrado na base de dados, caso nenhum *template* seja considerado suficientemente similar aos descritores obtidos por meio do leitor biométrico.

Os diagramas apresentados na Figura 2.3 ilustram as fases de cadastro, verificação e iden-

tificação de um sistema biométrico.

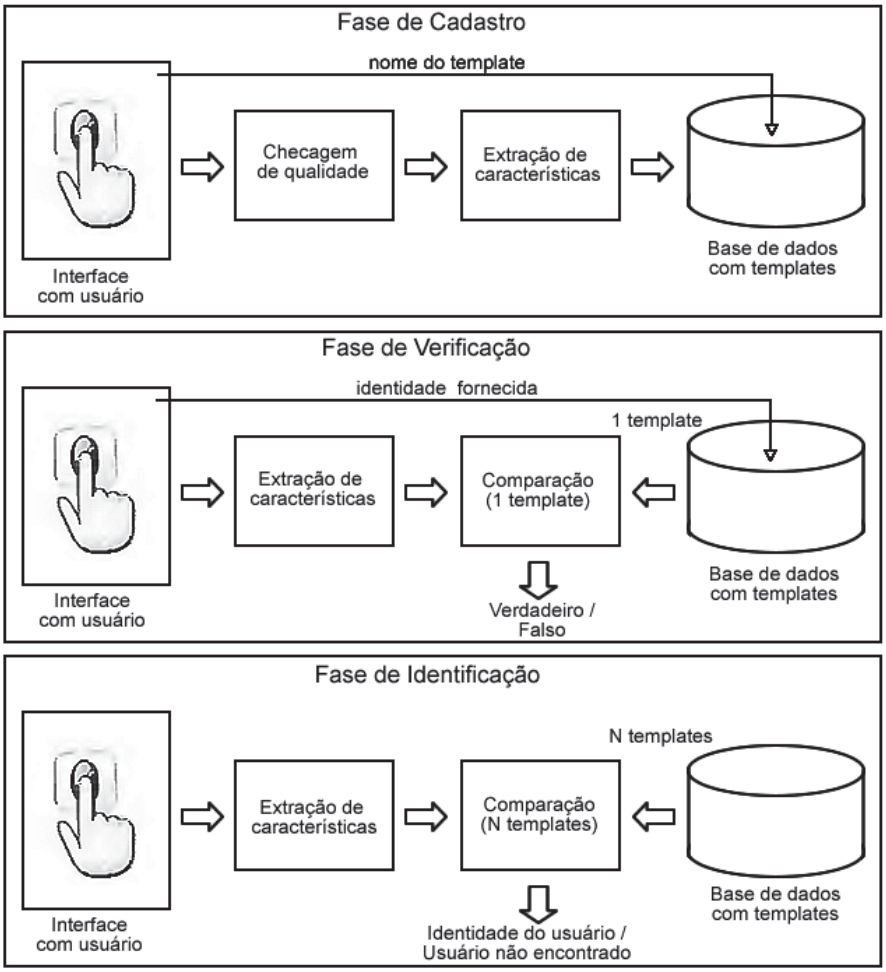

Figura 2.3: Fases dos sistemas biométricos (JAIN; MALTONI, 2003).

De acordo com Wayman (1999), os sistemas biométricos podem ser caracterizados de acordo com as seguintes propriedades:

- Cooperativo ou não-cooperativo: O usuário deseja ser identificado?
- Evidente ou sigiloso: O usuário sabe que está sendo identificado?
- Habituado ou não-habituado: O usuário submete-se frequentemente à identificação?
- Auxiliado ou não-auxiliado: O uso do dispositivo biométrico é observado e guiado por um operador do sistema?
- Ambiente controlado ou não controlado: O sistema irá operar em um ambiente controlado?

- Privado ou público: Os usuários do sistema são empregados (privado) ou clientes (público)?

- Aberto ou fechado: O sistema deverá usar métodos e *templates* padrões com o intuito de possibilitar alguma forma de interação com outros sistemas?

2.2.1 Avaliação dos Sistemas Biométricos

É praticamente impossível que um sistema biométrico forneça 100% de certeza que a identificação tenha sido realizada corretamente. Isto pode ser explicado pelo simples fato de que a característica biométrica de uma pessoa obtida em diferentes sessões não será exatamente a mesma devido às variações na captura da amostra biométrica, tais como diferenças no posicionamento do traço biométrico, mudanças ambientais, deformações, ruídos e má interação com o sensor (PRABHAKAR et al., 2003). É importante estar ciente de que um sistema biométrico fornece uma pontuação (*score*) que quantifica a similaridade apurada entre o descritor biométrico de consulta e um *template* da base de dados. Quanto mais alto for o *score*, maior é a certeza que as duas medições biométricas sejam provenientes da mesma pessoa.

Considerando que para o usuário final, o sistema deve fornecer a resposta da identificação baseada neste *score*, é necessário se estabelecer um limiar. Sendo assim, o sistema conclui que um par de descritores biométricos pertence à mesma pessoa se o *score* da comparação entre eles for maior que o limiar. Por outro lado, se o *score* for menor, o sistema irá concluir que os descritores em questão não pertencem à mesma pessoa.

A distribuição dos *scores* de comparações entre amostras das mesmas pessoas é chamada de distribuição genuína, enquanto que a distribuição de *scores* de comparações entre amostras de diferentes pessoas é chamada de distribuição impostora.

Um sistema biométrico pode cometer vários tipos de erros, dentre os quais destacam-se:

Falsa aceitação : O sistema apura que descritores biométricos de pessoas diferentes pertencem a uma mesma pessoa;

Falsa rejeição : O sistema apura que descritores biométricos de uma mesma pessoa pertencem a duas pessoas diferentes.

As taxas de erro de um sistema biométrico, FAR (FAR - *False Acceptance Rate*) e FRR (FRR - *False Rejection Rate*), são dependentes do limiar estabelecido para a operação desse sistema. Quando se aumenta o valor do limiar, a taxa de FRR tende a aumentar, enquanto que

a taxa de FAR tende a diminuir. Por outro lado, quando se diminui o valor do limiar, a taxa de FRR tende a diminuir, enquanto a taxa de FAR tende a aumentar.

Tal relacionamento entre o valor do limiar e as taxas FAR e FRR pode ser observado na Figura 2.4.

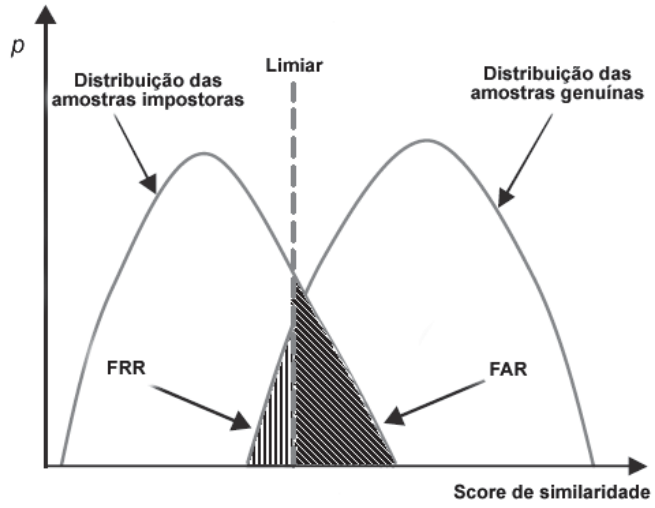

Figura 2.4: Relacionamento inverso entre as taxas de erros FAR e FRR em sistemas biométricos. Quanto maior o valor do limiar, maior tende a ser a taxa FRR e menor tende a ser a taxa FAR (e vice-versa) (WAYMAN, 1999).

Por meio do relacionamento inverso existente entre as taxas de FAR e FRR, é possível observar que o valor do limiar deve ser escolhido para um sistema biométrico de acordo com o nível de segurança que este necessite. Na Figura 2.5 está representada uma curva ROC (*Receiver Operating Characteristic*), na qual é possível observar que as taxas FAR e FRR variam de forma inversa, sendo que o ponto onde as taxas se igualam é chamado de EER (*Equal Error Rate*).

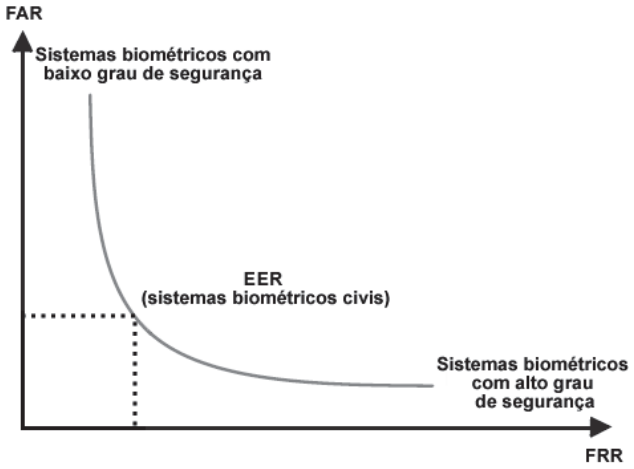

Figura 2.5: Curva ROC (WAYMAN, 1999).

Existem ainda duas outras taxas relacionadas à precisão dos sistemas biométricos: a FTC (*Failure To Capture*), que expressa a porcentagem de vezes que o sensor falha na captura automática da amostra biométrica, e FTE (*Failure To Enroll*), que denota a porcentagem de vezes que o usuário não consegue se registrar no sistema de reconhecimento.

2.3 Considerações Finais

Neste capítulo foram apresentados os conceitos sobre identificação de pessoas, destacando a identificação biométrica. Foram apresentadas as propriedades desejáveis para que uma característica biométrica possa ser utilizada na identificação de pessoas. Observou-se que nenhuma característica biométria pode ser considerada ótima e que a escolha de uma característica depende dos requisitos impostos ao sistema de identificação. Também foram apresentados os modos de operação de sistemas biométricos de identificação, suas propriedades e as taxas de erro mais comumente empregadas para avaliação de desempenho desses sistemas.

No próximo capítulo são apresentados métodos de classificação de padrões, que também foram utilizados neste trabalho para o reconhecimento de faces a partir de vídeo.

Capítulo 3

Métodos de Classificação

Reconhecimento de padrões é uma área do conhecimento onde um dos objetivos é a classificação de objetos (padrões) em categorias (classes) (THEODORIDIS; KOUTROUMBAS, 2006). Neste capítulo são apresentados de forma sucinta alguns dos principais métodos de classificação descritos da literatura e que são utilizados neste trabalho para o reconhecimento de faces a partir de vídeos.

3.1 Redes Neurais Artificiais

Os estudos sobre ANN (*Artificial Neural Network* - Redes Neurais Artificiais) baseiam-se no reconhecimento que o cérebro humano realiza, no entanto, quando essa tarefa é realizada computacionalmente, ela é feita de forma totalmente diferente. Uma rede neural é um processador distribuído fortemente paralelizado feito de unidades simples de processamento, denominadas neurônios, que têm uma tendência natural de armazenar conhecimento experimental e fazê-lo disponível para uso. Esta rede neural é semelhante ao cérebro em dois aspectos (HAYKIN, 1998):

- O conhecimento é adquirido pela rede por meio de um processo de aprendizagem;
- Os pesos de conexão entre neurônios, conhecidos como pesos sinápticos, são utilizados para armazenar o conhecimento adquirido.

O perceptron foi o primeira rede neural descrita de forma algorítmica, inventada por Rosenblatt (ROSENBLATT, 1958). Em geral, existem fundamentalmente três diferentes classes de arquitetura de rede dos perceptrons (HAYKIN, 1998):

Redes de uma camada sem retro alimentação: Esta é a forma mais simples de uma rede em camadas. Existe uma camada de entrada cujos nós são projetados em uma camada de neurônios de saída;

Redes de múltiplas camadas sem retro alimentação: Esta classe se distingue da primeira pela presença de uma ou mais camadas escondidas, cujos nós correspondentes são chamados neurônios escondidos ou unidades escondidas. A função desses neurônios escondidos é intervir entre as camadas de entrada e saída de uma maneira útil.

Redes recorrentes: Esta classe se distingue das outras por ter pelo menos um laço de retroalimentação. Ou seja, se existir pelo menos um neurônio na rede cuja saída seja direcionada para algum neurônio não pertencente à próxima camada, tal rede é denominada recorrente.

Sendo x um vetor de entrada, o um escalar de saída, w um vetor de pesos e θ um determinado limiar. O perceptron calcula $o = \sum_j w_j x_j$.

Caso $o \geq \theta$ quando se deseja $o < \theta$, então i está categorizado incorretamente e neste caso, altera-se os pesos e o limiar. Atribui-se $\theta \leftarrow \theta + 1$ para tornar menos provável que essa categorização incorreta ocorra novamente. Se $x_j = 0$ então nenhuma alteração é feita para w_j. Se $x_j = 1$, então atribui-se $w_j \leftarrow w_j - 1$ para diminuir a influência desse peso.

Caso $o < \theta$ quanto se deseja $o \geq \theta$, a atualização recíproca deve ser utilizada.

As atualizações aos pesos e limiares são feitas da seguinte forma:

$$o = \sum_j w_j x_j \tag{3.1}$$

$$\Delta\theta = -(t_p - o_p) \tag{3.2}$$

$$\Delta w_i = (t_p - o_p) x_{pi} \tag{3.3}$$

onde: t_p representa o escalar de saída desejada, $\Delta\theta$ representa a alteração no limiar para o padrão p e Δw_i representa a alteração nos pesos do padrão p.

A equação 3.2 representa a alteração no limiar para o padrão p e a equação 3.3 representa a alteração nos pesos do padrão p. Um perceptron de apenas uma camada não é capaz de resolver problemas não-lineares. Para a solução destes problemas são necessários perceptrons

multi-camada, sendo que o mais comumente utilizado é o perceptron de retro-alimentação de três camadas (MURTAGH, 1990). Afirma-se que essas redes multi-camadas podem prover a solução ótima para um problema de classificação arbitrário (DUDA et al., 2000).

O procedimento de aprendizagem de uma rede neural consiste na modificação dos pesos sinápticos com o intuito de se atingir um determinado objetivo. Um dos métodos mais populares para treinamento de perceptrons multi-camadas é o algoritmo de *backpropagation*.

O aprendizado *backpropagation* consiste de duas fases entre as diferentes camadas da rede: a fase *forward* e a fase *backward*. Na fase *forward*, um vetor de entrada é aplicado aos neurônios de entrada da rede, e seu efeito é propagado pela rede, camada a camada, até o conjunto de saída ser gerado. Na fase *forward*, todos os pesos sinápticos dos neurônios da rede são fixados. Na fase *backward*, os pesos sinápticos são ajustados de acordo com a regra de correção de erros. A resposta da rede é subtraída da resposta esperada para se obter o sinal de erro, sendo que este é propagado da frente para trás da rede contra a direção das conexões sinápticas. Sendo assim, os pesos sinápticos são ajustados com o intuito de fazer com que a resposta obtida chegue o mais próximo possível da resposta desejada (HAYKIN, 1998).

As redes SOM (*Self Organizing Maps* - Mapas Auto Organizáveis) representam uma classe especial de redes neurais artificiais. Tais mapas são baseados em um aprendizado competitivo, no qual neurônios de saída da rede competem entre eles para serem ativados, sendo que apenas um neurônio entre todos ou entre um grupo é ativado em um determinado tempo. Este modelo é motivado pela característica do cérebro de mapear entradas sensoriais em diferentes áreas do cortex cerebral de uma maneira topologicamente ordenada (HAYKIN, 1998). Em um SOM, os neurônios são dispostos geralmente em uma ou duas dimensões. É possível também utilizar mapas de maior dimensão, porém não é comum.

O treinamento de uma rede neural SOM pode ser representado pelos seguintes passos:

1. Iniciar a rede, atribuindo os valores iniciais para os pesos dos neurônios w_i, $i = 1, 2, 3, ...n$ onde n é o numero de neurônios da rede.

2. Selecionar um padrão de entrada x aleatoriamente do conjunto de padrões;

3. Utilizar uma função de ativação para calcular o estado de cada neurônio em relação ao padrão selecionado x. Pode-se utilizar a distância Euclidiana nesse caso.

4. Escolher um neurônio vencedor que tenha a menor distância para o padrão selecionado x. No caso da distância Euclidiana, tem-se:

$$\|x - w_c\| = min_i \{\|x - w_i\|\} \tag{3.4}$$

5. Atualizar os pesos sinápticos do neurônio vencedor e dos que estão dentro de sua vizinhança de acordo com a equação:

$$w_i(t+1) = w_i(t) + h_c i(t) \cdot [x(t) - w_i(t)] \tag{3.5}$$

onde:

- t é a iteração atual;
- $x(t)$ é o padrão de treinamento geralmente escolhido de forma aleatória na iteração t;
- $h_{ci}(t)$ é o núcleo de vizinhança ao redor do neurônio vencedor na iteração t.

O núcleo de vizinhança $h_{ci}(t)$ é uma função decrescente com o tempo e com a distância do neurônio i ao neurônio vencedor c, representado pela fórmula:

$$h_{ci}(t) = \alpha(t) \cdot h(\|r_c - r_i\|, t) \tag{3.6}$$

onde:

- $\alpha(t)$ representa a taxa de aprendizado;
- $h(d,t)$ representa a função de vizinhança.

A classificação ocorre de forma semelhante, sendo que os padrões são classificados de acordo com o rótulo do neurônio vencedor.

3.2 Classificador Bayesiano

A teoria de decisão Bayesiana é uma abordagem estatística fundamental para o problema de classificação de padrões (DUDA et al., 2000). O classificador Bayesiano ingênuo, chamado assim por assumir que as características são independentes entre si, apresenta resultados bastante competitivos em relação aos outros classificadores (FRIEDMAN et al., 1997).

Considerando $p(\omega_i|x)$ como sendo a probabilidade de um dado padrão $x \in \mathbb{R}^n$ pertencer à classe $\omega_i, i = 1,2,...,c$, que pode ser definida pelo teorema de Bayes:

$$P(\omega_i|x) = \frac{p(x|\omega_i)P(\omega_i)}{p(x)} \tag{3.7}$$

A fórmula de Bayes indica que, ao observar os valores de x, é possível converter a probabilidade *a priori* $P(\omega_i)$ para uma probabilidade *a posteriori* $P(\omega_i|x)$ - probabilidade de uma amostra pertencer a classe ω_j dado o vetor de características x. $p(x|\omega_i)$ é a probabilidade de ω_i com respeito a x (um termo escolhido para indicar que, considerando o restante constante, a categoria ω_i para qual a probabilidade $p(x|\omega_i)$ for maior é a mais provável de ser a categoria correta). O produto da probabilidade e probabilidade *a priori* é o mais importante para se determinar a probabilidade *a posteriori*, uma vez que o fator $p(x)$ pode ser visto como apenas um fator para garantir que a soma das probabilidades *a posteriori* seja igual a 1 (DUDA et al., 2000).

Um classificador Bayesiano decide se uma amostra x pertence à uma classe ω_j se:

$$P(\omega_i|x) > P(\omega_j|x), i, j = 1, 2, ..., c, i \neq j, \tag{3.8}$$

que pode ser reescrita da seguinte forma:

$$p(x|\omega_i)P(\omega_i) > p(x|\omega_j)P(\omega_j), i, j = 1, 2, ..., x, i \neq j \tag{3.9}$$

Os valores da probabilidade de $P(\omega_i)$ podem ser facilmente obtidos através do cálculo do histograma de classes, por exemplo. No entanto, o problema mais complexo é calcular a função de densidade de probabilidade $p(x|\omega_i)$, uma vez que estão disponíveis apenas as informações dos conjuntos de padrões e seus respectivos rótulos. Uma função bastante utilizada para modelar tal problema é a Gaussiana ou normal. Dessa forma, assume-se que as funções de densidade de probabilidade são gaussianas e que é possível estimar seus parâmetros através das amostras da base de dados.

A densidade gaussiana de n dimensões do padrão da classe ω_i pode ser descrita como:

$$p(x|\omega_i) = \frac{1}{(2\pi)^{n/2}|C_i|^{1/2}} exp\left[-\frac{1}{2}(x-\mu_i)^T C_i^{-1}(x-\mu_i)\right] \tag{3.10}$$

onde μ_i e C_i representam, respectivamente, a média e a matriz de covariância da classe ω_i. Tais parâmetros são obtidos da seguinte forma:

$$\mu_i = \frac{1}{N_i} \sum_{x \in \omega_i} x \tag{3.11}$$

$$C_i = \frac{1}{N_i} \sum_{x \in \omega_i} (xx^T - \mu_i \mu_i^T) \tag{3.12}$$

onde N_i representa o número de amostras da classe ω_i.

3.3 K Vizinhos Mais Próximos

A regra do vizinho mais próximo (NN - *Nearest Neighbor*) apresenta simplicidade, tanto conceitual quanto computacional, e baseia-se na classificação de padrões por meio de uma função de distância, de tal forma que um dado padrão é classificado como sendo pertencente à classe do padrão mais próximo do mesmo dentre todos os padrões (DUDA et al., 2000). Uma função de distância utilizado pelo NN comumente é a distância Euclidiana.

Considera-se um conjunto de n pares $(x_1, \omega_2), ..., (x_n, \omega_n)$, onde os x_i contém valores em um espaço de características X no qual é definida uma função de distância d, e as classes ω_i recebem valores do conjunto $\{1, 2, ..., M\}$, sendo que M é o número de classes. Cada ω_i é considerado como sendo o índice da classe à qual o indivíduo x_i pertence. Dado um novo par (x, ω), cuja classe ω se deseja estimar, utilizando-se a informação contida em um conjunto de padrões previamente classificados, $x_n^{'} \in \{x_1, x_2, ..., x_n\}$ será o vizinho mais próximo de x se:

$$min\ d(x_i, x) = d(x_n^{'}, x) \qquad i = 1, 2, ..., n \tag{3.13}$$

Sendo assim, x será classificado como pertencente à classe $\omega_n^{'}$ do vizinho mais próximo $x_n^{'}$ (COVER; HART, 1967).

O classificador baseado nos k vizinhos mais próximos (KNN - *K Nearest Neighbors*) é uma extensão do NN, no qual são consideradas as k amostras mais próximas ao invés de apenas uma. Desta forma, a classe $\omega_n^{'}$ mais incidente entre os k padrões $x_n^{'}$ mais próximos de x será atribuída ao último.

Uma vantagem deste tipo de classificador é que pode facilmente ser adaptado para ter uma aprendizagem incremental, de tal forma que cada novo padrão apresentado ao classificador pode ser armazenado junto aos outros padrões.

Outra característica deste classificador é que a nenhum modelo de classificação é criado e sendo assim a cada novo padrão a ser classificado, o mesmo deve ser comparado a todos os padrões armazenados na fase de treinamento.

3.4 Máquinas de Vetores de Suporte

Um dos problemas fundamentais da teoria de aprendizagem é definido como: dados dois objetos conhecidos, atribua um deles a um novo objeto desconhecido. Desta forma, o objetivo em reconhecimento de padrões de duas classes é deduzir uma função (SCHöLKOPF; SMOLA, 2001):

$$f : X \to \pm 1 \tag{3.14}$$

considerando a entrada e saída do dado de treinamento.

Baseado no princípio da minimização do risco estrutural (VAPNIK, 1999), o processo de otimização SVM (*Support Vector Machines* - Máquinas de vetores de suporte) tem como objetivo estabelecer uma função de separação. Vapnik (VAPNIK, 1999) considerou uma classe de hiperplanos em algum espaço dotado de produto interno H,

$$\langle w, x \rangle + b = 0 \tag{3.15}$$

onde $w, x \in H, b \in R$, correspondentes à função de decisão:

$$f(x) = sgn\left(\langle w, x \rangle + b\right) \tag{3.16}$$

e com base em dois argumentos, Vapnik propôs o algoritmo de aprendizagem *Generalized Portrait* para problemas que são separáveis por hiperplanos:

1. Entre todos os hiperplanos que separam os dados, existe um único hiperplano ótimo distinguido pela margem máxima de separação entre qualquer ponto de treino e o hiperplano;
2. A capacidade de separação de classes do hiperplanos diminui com o aumento da margem.

Para ser possível usar o produto interno como medida de similaridade, os padrões precisam ser representados como vetores em um espaço dotado de produto interno H, para isso, usa-se o mapeamento:

$$\begin{gathered} \Phi : X \to H \\ \chi \to Hx = \Phi(\chi) \end{gathered} \tag{3.17}$$

A função de *kernel K* é aquela que corresponde ao produto interno em um espaço de características expandido. Neste contexto, várias funções podem ser utilizadas como *kernel*: RBF (*Radial Basis Function*), funções sigmoide, funções lineares, etc.

Para construir o hiperplano ótimo, é necessário resolver a equação:

$$\min_{w \in H, b \in R} \tau(w) = \frac{1}{2} \|w\|^2 \tag{3.18}$$

sujeito a

$$y_i (\langle w, x_i \rangle + b) \geq 1 \quad para \ \ todo \ \ i = 1, ..., m \tag{3.19}$$

com a restrição 3.19 fazendo com que $f(x_i)$ seja $+1$ para $y_i = +1$ e -1 para $y_i = -1$, e também fixando a escala para w.

A função τ em 3.18 é chamada de função objetivo, enquanto que 3.19 são constantes de desigualdade. Juntas, elas formam o problema de generalização restrita. A função de separação é um combinação ponderada dos elementos do conjunto de treino. Esses elementos são chamados de vetores de suporte e caracterizam a fronteira entre duas classes.

A substituição chamada de *kernel trick* (SCHöLKOPF; SMOLA, 2001) é utilizada para estender o conceito de classificadores de hiperplano para máquinas de vetores de suporte não lineares. No entanto, mesmo com a vantagem obtida pelo uso do *kernel* no problema, o hiperplano de separação ainda não irá existir. Para permitir que alguns exemplos possam violar a equação 3.19, as variáveis de folga $\varepsilon \geq 0$ são introduzidas, obtendo-se as seguintes restrições:

$$y_i (\langle w, x_i \rangle + b) \geq 1 - \varepsilon_i \quad para \ \ todo \ \ i = 1, ..., m \tag{3.20}$$

Um classificador com boa generalização então é obtido controlando a margem (por meio de $\|w\|$) e a soma das variáveis de folga $\sum_i \varepsilon_i$. Neste contexto, uma possível aplicação de tal classificador de margem suave é obtido ao minimizar a função objetivo

$$(w, \varepsilon) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{m} \varepsilon_i \tag{3.21}$$

sujeita à restrição 3.20, quando a constante $C > 0$ determina equilíbrio entre o *overfitting* e generalização.

3.5 Floresta de Caminhos Ótimos

O classificador baseado em floresta de caminhos ótimos (OPF - *Optimum Path Forest*) é rápido, simples, multiclasse, independente de parâmetro, não faz nenhuma suposição sobre o formato das classes e pode lidar com um certo nível de sobreposição entre classes (PAPA et al., 2009b).

O conjunto de treinamento do classificador é visto como um grafo completo, cujos vértices são as amostras e as arestas ligam todos os pares de vértices. Cada aresta desse grafo recebe o peso correspondente à distância entre os vetores de características dos vértices ligados por ela. Qualquer sequência de vértices distintos forma um caminho conectando os vértices das extremidades e a este caminho é dado um peso através de uma função de conectividade, que pode ser o maior peso de um vértice pelo caminho. Esse grafo é utilizado na classificação, de forma que são escolhidos protótipos para as classes e cada amostra a ser classificada por meio do grafo será atribuída à classe de seu protótipo mais fortemente conectado. Ou seja, aquele que oferece à amostra, um caminho de custo mínimo, considerando todos os possíveis caminhos dos protótipos.

Na Figura 3.1 são apresentados dois conjuntos de protótipos, S_1 e S_2, pertencentes respectivamente, às classes 1 e 2. A conexão de S_i à uma amostra t é representada por um caminho $\pi_t^{(i)}$ com término em t e raiz em algum protótipo de S_i, i = 1, 2. Em todos os casos, o caminho ótimo (para o qual o máximo do peso do vértice é mínimo) vem do protótipo de alguma classe de t. Esta abordagem pode lidar com os três casos apresentados na Figura 3.1, com a função de peso máximo de vértice e protótipos estimados como as amostras mais próximas de classes distintas. No caso de sobreposição entre classes (Figura 3.1c), esses protótipos funcionam como defensores de classes em regiões de sobreposição no espaço de características (PAPA et al., 2009b).

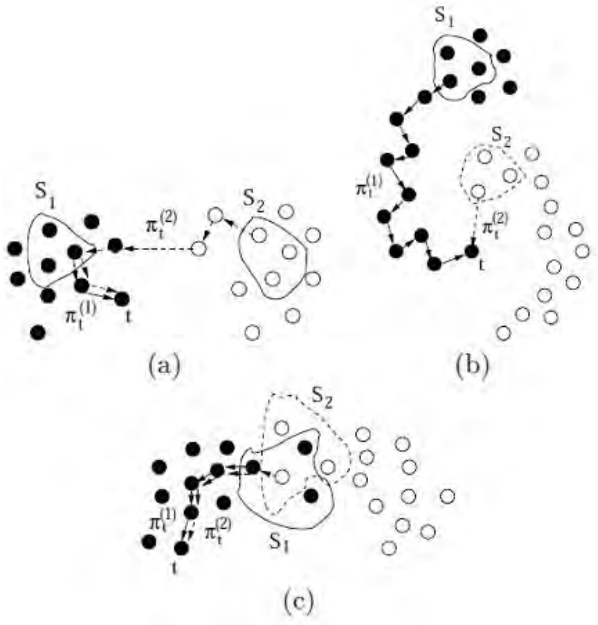

Figura 3.1: Exemplos de espaços de características bidimensionais utilizando duas classes: (a) linearmente separáveis (b) linearmente parcialmente separáveis (c) classes sobrepostas com formatos aleatórios. Protótipos podem ser identificados em cada classe, formando os conjuntos S_1 e S_2. Toda amostra t pode ser conectada a um protótipo S_i, $i = 1, 2$, por uma sequência $\pi_t^{(i)}$ de amostras distintas. A classificação é realizada baseando-se em conexões ótimas aos protótipos (PAPA et al., 2009b).

Os caminhos ótimos dos protótipos às outras amostras são computados pela IFT (*Image Foresting Transform*) (FALCÃO et al., 2004), uma ferramenta para a criação de operadores processadores de imagens baseado na conectividade, sendo que no contexto do OPF, a IFT é utilizada no domínio do espaço de características ao invés do domínio de imagem. O classificador OPF é uma floresta de caminhos ótimos com raiz nos protótipos, de forma que cada amostra de treino pertence à alguma árvore de caminho ótimo com raiz ao protótipo de conexão mais forte. Apesar de existirem outras funções de conectividade, a função geralmente considerada é a função de máximo peso das arestas.

O classificador OPF considera Z_1, Z_2, e Z_3 os conjuntos de treino, avaliação e testes com $|Z1|$, $|Z2|$ e $|Z3|$ amostras de uma dada base de dados. A divisão da base de dados é necessária para validar o classificador e avaliar a capacidade de aprendizagem com os erros. Z_1 é utilizada para projetar o classificador e Z_3 para medir a precisão, sendo que os rótulos de Z_3 não são utilizados. Um pseudo-teste com o conjunto Z_2 é usado para ajudar na aprendizagem do classificador por meio da troca de amostras de Z_1 com amostras classificadas incorretamente de Z_2. Após este processo de aprendizagem, espera-se uma classificação mais precisa de Z_3 (PAPA et

al., 2009b).

Seja $\lambda(s)$ a função que associa o rótulo correto i, $i = 1,2,...,c$, da classe i para qualquer amostra $s \in Z_1 \cup Z_2 \cup Z_3, S \subset Z_1$ um conjunto de protótipos de todas as classes, e v um algoritmo que extrai n características (cor, formato, propriedades de textura) de qualquer amostra $s \in Z_1 \cup Z_2 \cup Z_3$ e retorna um vetor $\vec{v}(s)$. A distância $d(s,t)$ entre duas amostras, s e t, é aquela entre os vetores de características $\vec{v}(s)$ e $\vec{v}(t)$. É possível utilizar qualquer função de distância para as amostras extraídas. A mais comum é a norma Euclidiana $\|\vec{v}(t) - \vec{v}(s)k\|$ (PAPA et al., 2009b).

Para descobrir o rótulo correto $\lambda(s)$ de uma amostra $s \in Z_3$, o classificador OPF cria uma partição do espaço de características de tal forma que a amostra $s \in Z_3$ possa ser classificada de acordo com sua partição. Essa partição é uma floresta de caminho ótimo (OPF) computada por Z_1 através do algoritmo (IFT) (PAPA et al., 2009b).

Seja (Z_1, A) um grafo completo cujos vértices são as amostras de treinamento e qualquer par de amostras define uma aresta em $A = Z1 \times Z1$ (Figura 3.2a). As arestas não necessitam ser armazenadas e, sendo assim, o grafo não precisa ser representado explicitamente. Um caminho é uma sequência de amostras distintas $\pi_t = \langle s_1, s_2, ..., t \rangle$ com terminação em uma amostra t. Um caminho é dito trivial se $\pi_t = \langle t \rangle$. É atribuído para cada caminho π_t um custo $f(\pi_t)$ dado pela função de conectividade f. Um caminho π_t é dito ótimo se $f(\pi_t) \leq f(\tau_t)$ para qualquer outro caminho τ_t. A concatenação de um caminho π_s e uma aresta (s,t) é denotada de $\pi_s \cdot \langle s,t \rangle$ (PAPA et al., 2009b).

O algoritmo OPF deve ser utilizado com qualquer função de conectividade *smooth* que pode agrupar amostras com propriedades similares. Uma função f é *smooth* em (Z_1, A) se, para qualquer amostra $t \in Z_1$, exista um caminho ótimo π_t que seja trivial ou tenha a forma $\pi_s \cdot \langle s,t \rangle$, onde:

- $f(\pi_s) \leq f(\pi_t)$,
- $f(\pi_s)$ é ótimo,
- para qualquer caminho ótimo $\tau_s, f(\tau_s \cdot \langle s,t \rangle) = f(\pi_t)$.

f_{max} pode ser expresso por:

$$\begin{aligned} f_{max}(\langle s \rangle) &= \begin{cases} 0 & \text{se } s \in S, \\ +\infty & \text{caso contrário} \end{cases} \\ f_{max}(\pi_s \cdot \langle s,t \rangle) &= \max\{f_{max}(\pi_s), d(s,t)\}, \end{aligned} \tag{3.22}$$

de forma que $f_{max}(\pi_s \cdot \langle s,t \rangle)$ computa a maior distância entre amostras adjacentes pelo caminho $\pi_s \cdot \langle s,t \rangle$ (PAPA et al., 2009b).

O algoritmo OPF minimiza f_{max}, armazenando os custos no mapa C. O classificador baseado em OPF associa um caminho $P^*(s)$ de S para todas as amostras $s \in Z_1$, resultando em uma OPF P, uma função acíclica que associa cada $s \in Z_1$ ao seu predecessor $P(s)$ em $P^*(s)$.

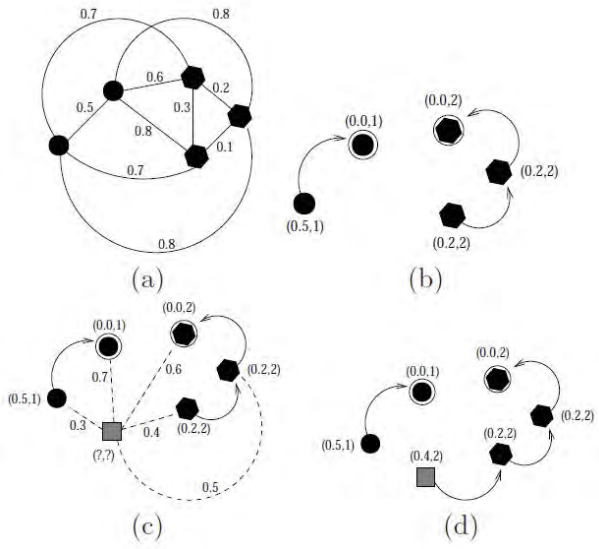

Figura 3.2: (a) Grafo completo e ponderado para um conjunto simples de treino (b) Floresta de caminho ótimo resultante para f_{max} e dois protótipos dados (vértices circulados). As entradas (x,y) sobre os vértices são, respectivamente, o custo e o rótulo das amostras. As arestas direcionadas indicam os vértices precedentes em um caminho ótimo. (c) Amostra de teste (quadrado cinza) e suas conexões (linha tracejada) com os vértices de treinamento. (d) O caminho ótimo do protótipo mais fortemente conectado, o rótulo 2 e o custo 0,4 são atribuídos à amostra de teste. Desta forma, a amostra é classificada como um hexágono, mesmo que a amostra de treino mais próxima pertença a classe de círculo (PAPA et al., 2009b).

3.5.1 Treinamento

S^* é um conjunto ótimo de protótipos obtido através da relação entre a MST (*Minimum Spanning Tree* - Árvore de Espalhamento Mínima) (CORMEN et al., 2001), que representa o subconjunto acíclico do grafo que conecta todos os vértices e cujo peso total é mínimo, com a árvore de caminho ótimo para f_{max}. O treinamento consiste em encontrar S^* e um classificador OPF com raiz em S^* (PAPA et al., 2009b).

Ao computar uma MST em um grafo completo (Z_1, A), um grafo conectado acíclico é obtido, cujos vértices são todas amostras de Z_1 e as arestas são não direcionadas e pondera-

das pela distância d entre os vértices adjacentes. A árvore geradora (*spanning tree*) é ótima, considerando-se que a soma dos pesos de suas arestas é mínima quando comparada com qualquer outra árvore geradora no grafo completo. Na MST, todo par de amostras é conectada por um único caminho que é ótimo de acordo com f_{max}. Sendo assim, a MST contém a árvore de caminho ótimo para qualquer vértice raiz selecionado (PAPA et al., 2009b).

Os protótipos ótimos são os elementos mais próximos na MST pertencentes a diferentes classes em Z_1. Removendo a aresta entre esses elementos, obtém-se os protótipos em S. Uma classe pode ser representada por um ou mais protótipos.

Os caminhos ótimos entre diferentes classes tendem a passar por essas arestas escolhidas para remoção da MST. Essa escolha de protótipos é feita justamente para bloquear estas passagens, reduzindo as chances de amostras de alguma classe serem alcançadas por caminhos ótimos de protótipos de outras classes (PAPA et al., 2009b).

3.5.2 Classificação

Para qualquer amostra $t \in Z_3$, são consideradas todas as arestas conectando t às amostras $s \in Z_1$, como se t fizesse parte do grafo (Figure 3.2c). Considerando todos os caminhos possíveis de S^* para t, é possível encontrar o caminho ótimo $P^*(t)$ de S^* e o rótulo t com classe $\lambda(R(t))$ de seus protótipos mais fortemente conectados $R(t) \in S^*$. Este caminho pode ser identificado de forma incremental, obtendo o custo ótimo $C(t)$ através da equação 3.23:

$$C(t) = min\{max\{C(s), d(s,t)\}\}, \forall s \in Z_1. \tag{3.23}$$

Após a avaliação feita com o conjunto Z_2, as amostras classificadas de forma errada podem ser utilizadas para aprender a distribuição de classes no espaço de características e melhorar a classificação em Z_3 (PAPA et al., 2009b).

3.6 Considerações Finais

Neste capítulo foram apresentados alguns dos principais métodos de classificação encontrados na literatura, que foram avaliados neste trabalho para o reconhecimeto de faces. Foi dado maior destaque para o classificador OPF por se tratar de um classificador recente que tem se mostrado superior em muitas aplicações. Além disso, esse classificador é rápido, simples, multiclasse, independente de parâmetro, não faz nenhuma suposição sobre o formato das classes e

pode lidar com um certo nível de sobreposição entre classes.

Um dos objetivos deste trabalho é substituir o módulo de reconhecimento baseado em distâncias do sistema proposto por Penteado e Marana (2009) para sistemas de autenticação biométrica de usuários em ambientes de e-Learning, por um módulo de reconhecimento baseado em classificação de padrões. Daí a necessidade de se analisar o desempenho dos principais métodos de classificação de padrões encontrados na literatura, tanto em relação à acurácia, quanto ao tempo de execução, uma vez que um dos resquisitos para a identificação de usuários em aplicações Web é a capacidade de decisão instantânea.

No próximo capítulo são descritas técnicas para o reconhecimento de faces, uma das mais populares características biométricas utilizadas para a identificação de pessoas.

Capítulo 4

Reconhecimento de Faces

Uma das formas mais básicas e naturais utilizadas pelos seres humanos para a identificação de pessoas é o reconhecimento facial.

O reconhecimento de faces, como uma das principais técnicas biométricas, está se tornando cada vez mais importante devido aos rápidos avanços tecnológicos como o desenvolvimento de câmeras digitais de alta resolução e de dispositivos móveis, que já trazem embutidas essas câmeras. Existe também uma demanda crescente por segurança e agilidade nos processos de identificação.

Como característica biométrica, a face tem várias vantagens sobre as outras tecnologias (JAIN; LI, 2005). Ela é natural, não intrusiva e fácil de utilizar. Dentre os seis atributos biométricos considerados no estudo realizado por Hietmeyer (2000), que são a assinatura, o dedo, a face, a mão, os olhos e a voz, as características da face são as que apresentam maior compatibilidade com os sistemas MRTD (*Machine Readable Travel Documents*), baseados em fatores de avaliação como treinamento, renovação, requisitos de máquinas e percepção pública. Na Figura 4.1, é mostrada a classificação de alguns atributos biométricos segundo Hietmeyer (2000).

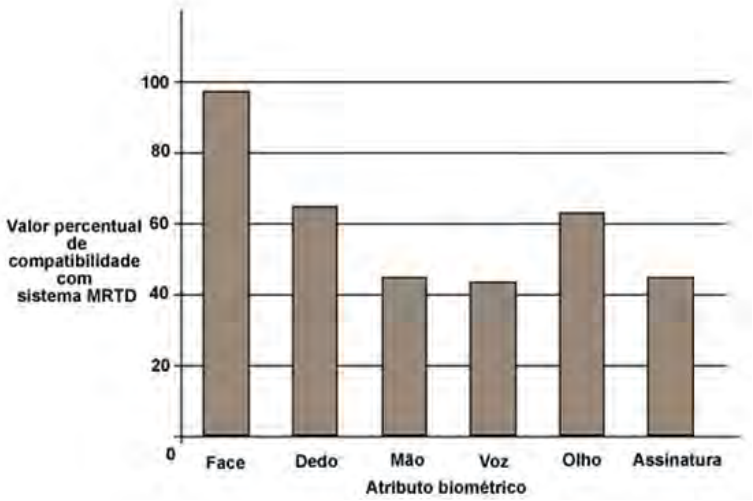

Figura 4.1: Classificação das características biométricas de acordo com a compatibilidade com sistemas MRTD (JAIN; LI, 2005).

A definição do problema de reconhecimento facial pode ser formulada como a identificação de uma ou mais pessoas em uma cena, dadas imagens estáticas ou dinâmicas, utilizando as faces armazenadas em uma base de dados (CHELLAPPA et al., 1995).

4.1 Reconhecimento de Faces a Partir de Imagens Estáticas

Estratégias de reconhecimento facial baseadas em imagens exploram apenas as informações estáticas da face, mais precisamente, as informações representadas pelos pixels que formam as imagens. A maioria dos algoritmos de reconhecimento facial é proposta para lidar com imagens individuais, onde os conjuntos de treinamento e teste consistem de imagens faciais.

4.1.1 *Eigenfaces*

Eigenfaces é, sem dúvida, a técnica mais conhecida e utilizada para o reconhecimento de pessoas por meio da aparência facial (TURK; PENTLAND, 1991). Ela tem sido amplamente estudada e largamente aplicada para imagens e vídeos. A técnica *eigenface* é baseada na redução de dimensionalidade, por meio da escolha de características que apresentem a maior representatividade dos dados. Um método para reduzir o espaço da imagem para um espaço de menor dimensão consiste na aplicação da análise das componentes principais (PCA - *Principal Component Analisys*) (DUDA et al., 2000).

PCA é uma técnica amplamente utilizada em visão computacional para redução de dimen-

sionalidade. Esta técnica escolhe uma projeção linear redutora de dimensionalidade que maximiza o espalhamento de todos os exemplos projetados (BELHUMEUR et al., 1997). Considerando-se N exemplos de imagens $x_1, x_2, ...x_N$ com valores em um espaço de imagem de n dimensões e que cada imagem pertença a uma das classes $X_1, X_2, ..., X_c$, o PCA faz uso de um mapeamento de transformação linear do espaço de imagem de n dimensões em um espaço de características de m dimensões, onde $m < n$. Os novos vetores de características $y_k \in R^m$ são definidos pela seguinte transformação linear:

$$y_k = W^T x_k, \quad k = 1, 2, ..., N \tag{4.1}$$

onde $W \in R^{n \times m}$ é projeção linear composta por uma matriz com colunas ortonormais.

A matriz de espalhamento total S_T é dada por:

$$S_T = \sum_{k=1}^{N} (x_k - \mu)(x_k - \mu)^T \tag{4.2}$$

onde N é o número de imagens de exemplo e $\mu \in R^n$ é a imagem média de todos os exemplos. Ao aplicar a transformação linear W^T, o espalhamento dos vetores de características transformados $\{y_1, y_2, ..., y_N\}$ é $W^T S_T W$.

No PCA, a projeção W_{opt} é escolhida para maximizar o determinante da matriz total de espalhamento dos exemplos projetados. W_{opt} é dada por:

$$W_{opt} = \arg\max_W |W^T S_T W| = [w_1 \ w_2 \ ... \ w_m] \tag{4.3}$$

onde $\{w_i | i = 1, 2, ..., m\}$ é o conjunto de autovetores de n dimensões de S_T correspondendo aos m maiores autovalores.

A escolha do número de autovetores depende da representatividade desejada. A representatividade é obtida por meio da soma dos autovalores em ordem decrescente, sendo que para se formar o novo espaço de características deve-se utilizar os autovetores equivalentes.

Na Figura 4.2 é possível observar a projeção dos vetores em W, lembrando que W foi escolhida de forma a maximizar o espalhamento dos vetores.

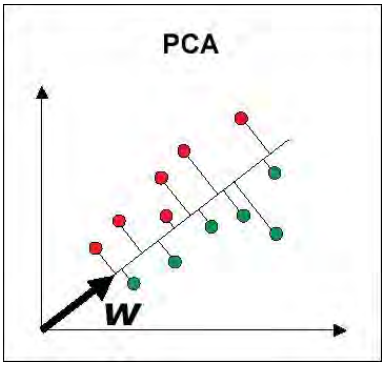

Figura 4.2: Projeção das amostras no espaço gerado pelo PCA.

Após reduzir a dimensionalidade de uma imagem de face, é necessário definir uma métrica que informe quão diferente uma face é de outra. Uma métrica possível é a distância Euclidiana (DUDA et al., 2000), dada por $D(A,B) = (\sum_{k=1}^{d} (a_k - b_k)^2)^{1/2}$ onde D é a distância Euclidiana, A e B são os vetores entre os quais quer se medir a distância e d é o número de dimensões dos vetores a e b.

4.1.2 *Fisherfaces*

Fisherface é outra técnica para reconhecimento de pessoas por meio do uso da aparência facial. De forma similar ao *eigenface*, a técnica *fisherface* também se baseia na redução de dimensionalidade do espaço de características. A projeção ótima neste caso é obtida por meio da análise discriminante linear de Fisher(FLD - *Fisher's Linear Discriminant*). O FLD é um exemplo de método específico de classe, no sentido que tenta "modelar"o espalhamento dos dados de forma que a classificação se torne mais confiável. Esse método seleciona a transformação linear W de forma que a razão entre o espalhamento inter-classes e o intra-classes seja maximizado (BELHUMEUR et al., 1997). O espalhamento inter-classes é definido como:

$$S_B = \sum_{i=1}^{c} N_i (\mu_i - \mu)(\mu_i - \mu)^T \tag{4.4}$$

e o espalhamento intra-classes é definido como:

$$S_W = \sum_{i=1}^{c} \sum_{x_k \in X_i} (x_k - \mu_i)(x_k - \mu_i)^T \tag{4.5}$$

onde μ é a imagem média entre todas as classes, μ_i é a imagem média da classe X_i e N_i é o número de exemplos da classe X_i.

Se S_W for uma matriz não singular, a projeção ótima W_{opt} é escolhida como a matriz com colunas ortonormais que maximiza a taxa do determinante da matriz de espalhamento entre classes dos exemplos projetados e também que minimiza a taxa do determinante da matriz de espalhamento intra-classe dos exemplos projetados.

$$W_{opt} = \arg\max_{W} \frac{|W^T S_B W|}{|W^T S_W W|} = [w_1 \; w_2 \; ... \; w_m] \tag{4.6}$$

Na Figura 4.3, é possível observar a projeção dos vetores em W, lembrando que W foi escolhida de forma a maximizar o espalhamento dos vetores de classes diferentes e minimizar o espalhamento entre os vetores de mesma classe.

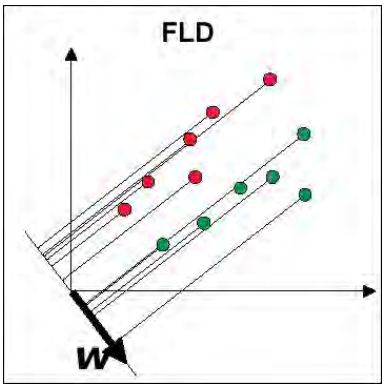

Figura 4.3: Projeção das amostras no espaço gerado pelo FLD.

4.1.3 Modelo de Aparência Ativa

O Modelo de Aparência Ativa (AAM - *Active Appearance Model*) é um modelo estatístico baseado no formato dos objetos. O AAM é composto por um modelo estatístico do formato e da aparência em tons de cinza do objeto de interesse que pode generalizar quase qualquer objeto válido. Casar uma imagem implica em encontrar os parâmetros do modelo que minimizem a diferença entre a imagem e o modelo sintetizado, projetado na imagem (COOTES, 1998).

Edwards et al. (1999) usam AAM para o reconhecimento de faces a partir de imagens estáticas. Duas faces F_1 e F_2 podem ser comparadas ao realizar o casamento de ambas com o modelo AAM e extrair os parâmetros de identidade d_1 e d_2. O FLD utilizado para se construir o subespaço de identidade é normalizado de forma que a distância Euclidiana entre d_1 e d_2 forneça uma medida direta de probabilidade das duas faces pertencerem à mesma pessoa. O desempenho da classificação depende do limiar do casamento, T. Considerando d_{dt} como a distância entre um par de imagens I_d e I_t (uma imagem da base de dados e outra de teste), a

regra para identificação da imagem de teste é a apresentada na condição 4.7.

$$se \begin{cases} d_{dt} < T, \text{pessoa é a mesma} \\ d_{dt} > T, \text{ pessoa não é a mesma} \end{cases} \tag{4.7}$$

4.2 Reconhecimento de Faces a Partir de Vídeos

O reconhecimento facial a partir de imagens apresenta o desempenho bastante prejudicado por variações como pose, iluminação e mudanças de expressões (ZHAO et al., 2003). Desta forma, os pesquisadores desta área buscam o desenvolvimento de métodos que utilizem o reconhecimento facial baseado em vídeo, onde os conjuntos de treinamento e de teste são sequências de vídeos das faces representando os usuários do sistema.

O reconhecimento de faces em vídeos pode ser visto como uma extensão do reconhecimento de faces em imagens. Porém, no primeiro, considera-se uma sequência de imagens ao invés de uma única imagem.

O reconhecimento de pessoas com o uso de informações de vídeos apresenta várias vantagens em relação ao reconhecimento baseado em imagens estáticas, dentre as quais destacam-se:

- Disponibilidade de uma quantidade enorme de dados quando comparados às imagens estáticas;
- Possibilidade de se ter representações mais eficientes dos indivíduos;
- Disponibilidade de informações temporais obtidas a partir da sequência de imagens.

Ainda existem vários desafios para o reconhecimento de faces baseado em vídeo. Dentre estes desafios, destacam-se (ZHAO et al., 2003):

Baixa qualidade do vídeo capturado: as amostras de vídeo são adquiridas em ambientes com más condições para captura e sem cooperação dos usuários, além de muitas vezes serem utilizadas câmeras com baixa resolução;

Imagens das faces pequenas: devido às condições de aquisição, o tamanho das imagens pode ser bem menor do que os das faces cadastradas na base de dados, o que pode impactar negativamente o desempenho do reconhecimento e da detecção de pontos ou características usadas para a representação da face;

Variações significativas de pose, expressão facial e iluminação : em uma mesma sequência de vídeo, um indivíduo pode ter variações significativas da pose e de expressões faciais, dependendo do tipo de atividade que o mesmo desempenha. Além disso, as câmeras que capturam os vídeos podem estar localizadas em ambientes internos, sujeitos à variação de iluminação bem como em ambientes externos com fatores que alteram fortemente a iluminação, como sol, farol de veículos, lanternas, entre outros;

Poses não-frontais: geralmente, as câmeras de vídeo são posicionadas de tal modo que os indivíduos não notem sua presença, para que o reconhecimento seja realizado sem seu conhecimento ou para que a câmera não atrapalhe a utilização do sistema. Assim, dificilmente a imagem capturada da face terá qualidade adequada;

Incerteza na detecção: devido ao movimento das pessoas em ambientes onde câmeras de vídeo estão instaladas, a detecção e o rastreamento das faces presentes na cena tornam-se um problema de estimativa dos movimentos, não tão eficientes quanto para imagens estáticas ou para aplicações que contam com colaboração dos usuários;

O número de algoritmos com a finalidade de realizar o reconhecimento facial por meio de vídeos ainda é pequeno quando comparado ao número de algoritmos que utilizam imagens para a mesma finalidade. Uma das razões para tal fato é o alto custo de câmeras de vídeo de alta qualidade. Outra razão é a maior complexidade algorítmica. Enquanto a extração de características de imagens estáticas já é algo complexo, tal processo realizado em sequências dinâmicas se torna ainda mais complexo (ZHANG; MARTÍNEZ, 2006).

A percepção humana não usa somente a estrutura facial para reconhecer faces, mas também outros traços como cor, movimento facial, conhecimento contextual, entre outros. Estudos neuropsicológicos demonstram que o movimento facial ajuda no processo de reconhecimento das faces especialmente em ambientes degradados.

De acordo com os estudos realizados sobre a percepção humana no que diz respeito ao reconhecimento de faces, as seguintes afirmações são apresentadas (KNIGHT; JOHNSTON, 1997) e (O'TOOLE et al., 2002):

- Tanto as informações estáticas da face quanto as dinâmicas são úteis para o processo de reconhecimento;
- As pessoas se baseiam principalmente em informação estática, pois a informação dinâmica fornece informação menos precisa do que a estrutura facial estática;

- A informação dinâmica contribui mais para o reconhecimento em condições desfavoráveis de visualização, como baixa iluminação, baixa resolução da imagem, reconhecimento a distância, etc;
- O movimento facial contribui para o reconhecimento ao facilitar a percepção da estrutura tridimensional da face;
- O movimento facial é aprendido mais lentamente do que a estrutura facial estática;
- O reconhecimento de faces familiares é mais efetivo quando são mostradas como uma sequência animada do que como um conjunto de múltiplos quadros sem animação;
- Para faces não familiares, a sequência animada não fornece mais informação útil do que múltiplas imagens estáticas.

Baseados nestas descobertas, pesquisadores têm tentado explorar a dinâmica facial para melhorar o desempenho dos sistemas biométricos no processo de reconhecimento de faces.

Diante dessas considerações, uma classificação plausível para as abordagens do reconhecimento fácial a partir de vídeo é a seguinte:

- Abordagem não temporal, na qual usa-se inúmeras imagens estáticas capturadas dos vídeos;
- Abordagem temporal, na qual usa-se as informações temporais de forma adicional às informações estáticas.

4.2.1 Abordagem Não Temporal

São apresentadas a seguir algumas técnicas de reconhecimento facial em vídeo nas quais a informação temporal é ignorada, levando-se em consideração apenas a informação estática.

Eigenfaces

Satoh (2000) faz uso da abordagem padrão das *eigenfaces* aplicadas aos *frames* do vídeo, sendo que a medida de dissimilaridade entre as faces é calculada pela distância Euclidiana. Para fazer o casamento de sequências de faces, ele utiliza um método que consiste em uma medida de similaridade entre dois vídeos, na qual é necessário comparar cada *frame* de um vídeo com cada *frame* do outro vídeo. Visto isso, o par de *frames* que obtiver a menor medida

de dissimilaridade por meio da distância Euclidiana irá representar a medida de dissimilaridade entre os dois vídeos.

Torres e Vila (2001) propõem o conceito de *self-eigenface*. Este método é baseado na criação de um conjunto de *eigenfaces* para cada indivíduo na fase de treinamento, diferenciando do método padrão que consiste em utilizar um conjunto de *eigenfaces* criado a partir de imagens de todos os indivíduos. Este conjunto de *eigenfaces* individual é criado a partir de diferentes poses de cada integrante do sistema. Além deste conjunto de *eigenfaces* armazenado para cada indivíduo, guarda-se todas as imagens utilizadas na geração do mesmo. Na fase de reconhecimento, cada amostra é reconhecida se a maior similaridade entre a mesma e todas armazenadas for maior que um limiar determinado.

Uma outra estratégia baseada em *eigenface* empregada no reconhecimento facial em vídeo, é o MSM (Mutual Subspace Method) (YAMAGUCHI et al., 1998). Nesta abordagem utiliza-se o conceito de subespaço também na detecção de face, onde guarda-se o subespaço facial e o não facial com o intuito de distinguir entre as classes faciais das não faciais. Como a boca é uma característica bastante instável, a mesma é retirada durante o pré-processamento da face extraída. Para se obter o subespaço de cada indivíduo, captura-se um determinado número fixo de observações do mesmo e o subespaço é computado da matriz de correlação dos vetores de observação. Então o método *eigenspace* é utilizado para encontrar os autovetores. Tendo obtido os subespaços individuais, usa-se o MSM para se calcular a similaridade entre estes e os de referência armazenado em uma base de dados. Para a tarefa de reconhecimento, a amostra é classificada como pertencente à classe que forneçe a maior similaridade.

Existe ainda uma abordagem na qual utiliza-se a técnica *eigenfaces* em conjunto com a regra da maioria dos votos utilizada por Huang e Trivedi (2002). Nesta técnica, é necessário realizar a identificação do indivíduo presente em cada *frame* utilizando para isso o template cuja projeção no *eigenspace* apresentar menor distância Euclidiana para a projeção da face encontrada no *frame* em questão. Após esse passo, chega-se a uma decisão final utilizando a regra da maioria dos votos, na qual escolhe-se a identidade final de acordo com a identidade com maior frequência durante todos os *frames* do vídeo.

Fisherfaces

Uma abordagem para o reconhecimento facial baseado no método *fisherface* foi proposta por Satoh (2000). Nesse trabalho, a medida de similaridade entre dois vídeos consiste na menor distância entre pares de *frames* (sendo um de cada vídeo) em um espaço reduzido de características. Esta mesma medida de similaridade foi utilizada tanto no subespaço gerado pelo

eigenspace quanto no gerado pelo *fisherface*, sendo que o que utilizou *fisherface* apresentou um resultado mais satisfatório com um tempo de classificação também menor.

Matching de Grafos Elásticos

Matching de grafos elásticos (EGM - *Elastic Graph Matching*) é uma técnica que consiste em representar formas por meio de grafos rotulados, nos quais os vértices são rotulados com características que descrevem a distribuição local de tons de cinza da imagem da forma e as arestas são rotuladas com as medidas de posição relativa entre vértices (WISKOTT et al., 1997).

No trabalho de Wiskott et al. (1997), os indivíduos são representados por um FBG (*Face Bunch Graph*) que combina um conjunto representativo de modelos de grafos individuais em uma estrutura em forma de pilha. Os nós deste modelo são rotulados pelos *bunches*, que são conjuntos de *jets*. Um *jet* descreve um pequeno trecho de valores de tons de cinza ao redor de um *pixel* em uma imagem. As arestas são rotuladas com as médias de vetores de distância. Durante a comparação entre imagens, o *jet* com maior semelhança de cada *bunch*, destacado pela cor cinza na Figura 4.4, é selecionado independentemente. Na figura 4.4 está representado um FBG. Por meio do uso deste modelo, é possível se ter uma representação geral ao invés de se ter modelos de faces individuais.

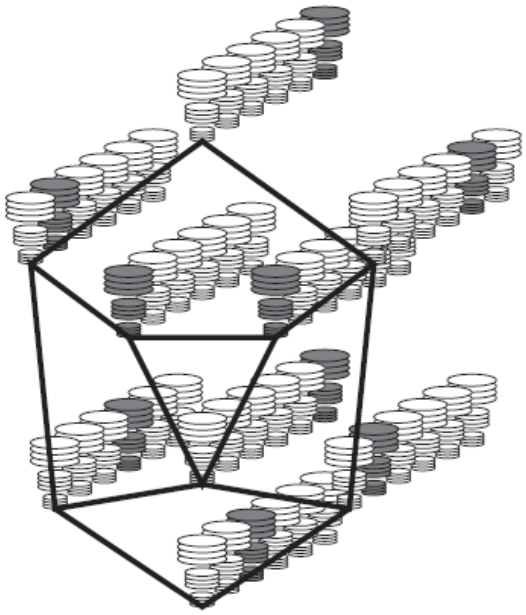

Figura 4.4: Face Bunch Graph (WISKOTT et al., 1997).

Para se encontrar os pontos principais em uma imagem de teste e extrair da mesma um grafo que maximize a similaridade com o FBG, é utilizada a técnica EBGM (*Elastic Bunch Graph*

Matching).

Após extrair os modelos de grafos das imagens de treinamento e os grafos de imagens das imagens de testes, o reconhecimento é realizado comparando-se um grafo de imagem com todos os modelos de grafos e escolhendo-se o modelo de grafo que apresentar maior similaridade. A função de similaridade utilizada para se comparar os grafos é uma média sobre as similaridades entre pares de vértices correspondentes.

Agrupamento não Supervisionado aos Pares

Métodos de agrupamento tentam particionar os dados disponíveis em agrupamentos de acordo com as categorias naturais presentes nestes dados, na falta de informações explícitas de categorias (RAYTCHEV; MURASE, 2003). Os métodos de agrupamento podem ser classificados como centrais ou aos pares.

No agrupamento central, os padrões podem ser representados como amostras em um espaço n-dimensional e assume-se que cada agrupamento pode ser parametrizado por um centro ao redor do qual são distribuídas as amostras de acordo com alguma função de distribuição.

No agrupamento aos pares, o objetivo é particionar os dados baseando-se apenas nas relações aos pares, que podem ser formuladas matematicamente como um problema de otimização combinatória.

No caso de reconhecimento facial em vídeo, existem vários fatores que favorecem o uso de agrupamento aos pares em relação ao agrupamento central. Quando, por exemplo, as pessoas se movem em cenas dinâmicas, expondo visões diferentes de suas faces para as câmeras, a sequência resultante forma *manifolds* complexos e não-lineares no espaço de imagem de face. *Manifolds* são superfícies de baixa dimensão em um espaço de alta dimensão utilizada para a representação de dados em uma forma reduzida (ZHANG et al., 2004). Mesmo que seja possível tratar cada face como um exemplo separado representado pelas suas coordenadas em um certo espaço de faces, centróides serão sem sentido ou difíceis de definir e manusear. Métodos de agrupamento central também têm a desvantagem de exigir que se saiba a priori qual o número de agrupamentos enquanto que os métodos de agrupamento aos pares não necessitam tal conhecimento a priori.

Agrupamento aos pares pode ser visualizada como um grafo, onde cada nó representa um padrão e as arestas correspondem às ligações entre valores próximos. Na Figura 4.5 representa-se um agrupamento aos pares aplicada ao reconhecimento de pessoas em vídeos. Os números nas arestas indicam as distâncias, as letras nos vértices indicam o indivíduo e os números nos

vértices indicam o índice da sequência de tal indivíduo. As arestas consistentes conectam nós da mesma classe enquanto que arestas inconsistentes conectam nós de classes distintas.

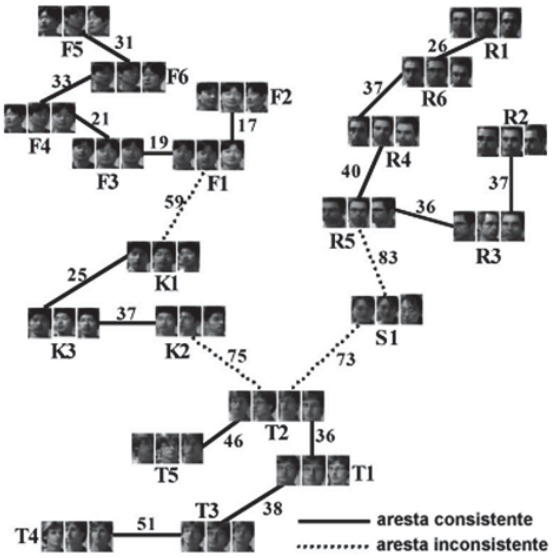

Figura 4.5: Representação de um agrupamento aos pares (RAYTCHEV; MURASE, 2003).

Na abordagem utilizada em Raytchev e Murase (2003), foram propostos dois métodos para agrupamento, chamados de CAR1 (*Clustering by Attratction and Repulsion*) e CAR2 (*Clustering by Attraction and Repulsion global Optimization*). Tais métodos particionam os dados de entrada em agrupamentos de identidade realizando otimizações combinatórias guiadas por dois tipos de forças (atração e repulsão) impostas pelas matrizes de proximidade. Tais métodos são utilizados tanto no agrupamento quanto na classificação.

4.2.2 Abordagem Temporal

São apresentadas a seguir algumas técnicas de reconhecimento facial em vídeo nas quais a informação temporal é levada em consideração em conjunto com a informação espacial.

Análise Discriminante no Fluxo Óptico Facial

O movimento facial pode ser representado por um vetor de características de alta dimensão que é construído concatenando-se uma sequência de movimentações. Cada indivíduo pode ser representado por um vetor de características que reúne informações espaciais e temporais da face ao mesmo tempo.

Chen et al. (2001) introduziram um novo esquema de representação de pessoas, que adota

características do movimento facial. Tais características são extraídas de uma sequência de imagens de faces cujos campos de fluxo são computados baseados na aproximação por wavelets. Uma vez que wavelets podem representar informações de forma eficiente, os campos de fluxo estimados das imagens das faces podem detectar até pequenos movimentos, como o movimento da pele facial. A equação 4.8 representa a estimação do fluxo:

$$E = \int\int (I_x u + I_y v + I_t)^2 + \alpha(|\nabla u|^2 + |\nabla v|^2) dxdy \tag{4.8}$$

onde $I = I(x,y;t)$ é a função de brilho da imagem no tempo t; $[u,v] = [u(x,y), v(x,y)]$ é o vetor de fluxo; ∇ é o operador gradiente; I_x, I_y e I_t são as derivadas parciais de $I = I(x,y;t)$ referentes às coordenadas x, y e t respectivamente.

O vetor de alta dimensionalidade é utilizado nesta abordagem para representar uma sequência de fluxos de campos de uma face, sendo que este vetor contém informação espaço-temporal da mesma. A Figura 4.6 ilustra os passos para a obtenção deste vetor de alta dimensionalidade, onde I representa o brilho da imagem e U e V são os vetores de fluxo.

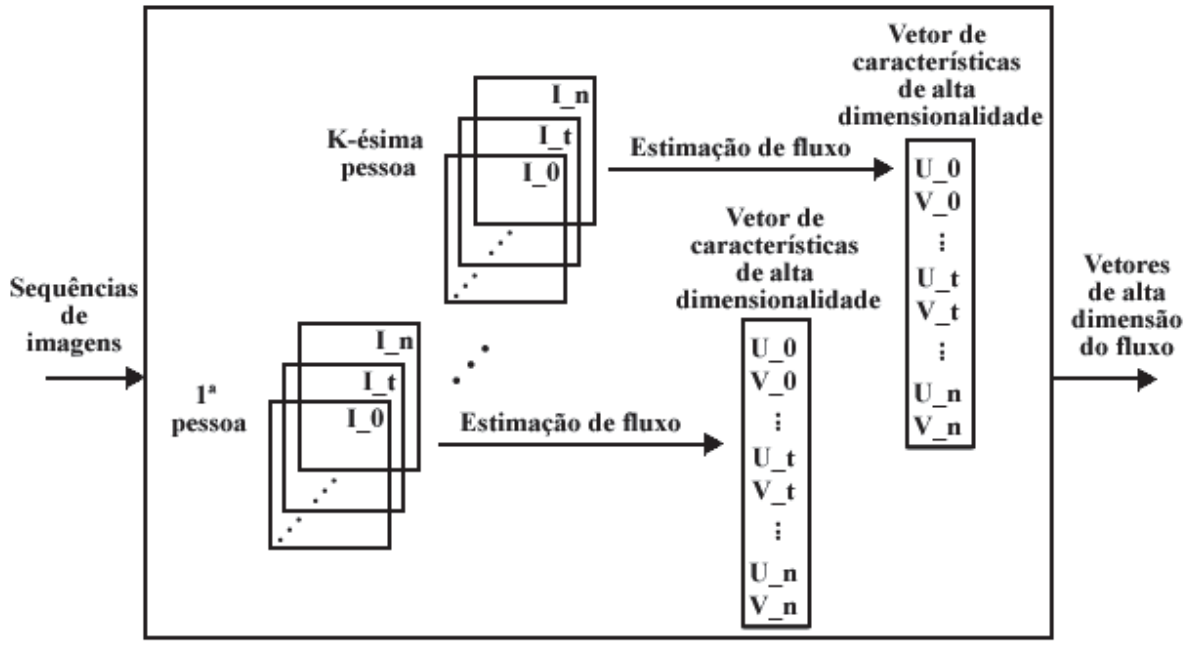

Figura 4.6: Estimação do movimento facial (CHEN et al., 2001).

Tanto na fase de treinamento quanto na fase de testes, os vetores são obtidos da mesma forma. Após obter estes vetores na fase de treinamento, é utilizado o PCA para se encontrar o conjunto ótimo de eixos de projeção e se reduzir a dimensionalidade destes vetores. Os vetores

são então projetados neste conjunto de eixos para a construção da base de dados. Na fase de testes, após realizar o mesmo processo de obtenção do vetor de características e sua projeção no conjunto de eixos, é calculada a distância deste vetor projetado àqueles armazenados na base de dados e a partir disso é possível checar a identidade.

Rastreamento e Reconhecimento Estocástico por meio do Filtro de Partículas

O filtro de partículas é uma técnica de inferência para estimar um estado desconhecido de movimentação θ_t de uma coleção de observações ruidosas obtidas de forma sequencial (DOUCET et al., 2001).

No trabalho desenvolvido por Zhou et al. (2004), é apresentada uma abordagem que incorpora modelos de aparência adaptativa no filtro de partículas para realizar rastreamento visual e reconhecimento robustos. Para realizar o rastreamento é necessário modelar o movimento entre *frames* e as mudanças de aparência enquanto que para modelar o reconhecimento é necessário modelar as mudanças de aparência entre os *frames* e a base de dados com *templates* dos indivíduos.

Com o objetivo de deixar o rastreamento mais robusto e estável, além de se utilizar um número adaptável de partículas, propõe-se o uso dos seguintes modelos:

- Modelo de observação decorrente de um modelo de aparência adaptativo;
- Modelo de velocidade adaptativa obtido utilizando-se um preditor linear de primeira ordem baseado na diferença de aparência entre a observação de entrada e a configuração anterior da partícula.

O rastreamento é realizado de forma simultânea ao reconhecimento ao embutir ambos em um filtro de partículas. As alterações nas aparências entre *frames* e *templates* da base de dados são modeladas construindo-se espaços intra e extra-pessoais.

Rastreamento e Reconhecimento Utilizando *Manifolds* Probabilísticos de Aparência

Lee et al. (2005) apresentaram uma abordagem para rastreamento e reconhecimento em vídeos, utilizando *manifolds* probabilísticos de aparência. Nesta abordagem, cada pessoa registrada é representada por um *manifold* de aparência de baixa dimensionalidade. Para construir tal representação, são extraídos *frames* de vídeos e tais *frames* são agrupados por meio de um algoritmo k-médias. Cada agrupamento é representado por um plano computado via PCA. A

conexão entre os *manifolds* de pose armazena a probabilidade de transição entre as imagens em cada um dos *manifolds* e é aprendida por meio de uma sequência de vídeo de treinamento que caracteriza a probabilidade de se mover de uma pose para outra entre dois *frames* consecutivos. Ou seja, a dinâmica entre os *manifolds* de pose é obtida por meio dos vídeos de treinamento como pode ser observado na Figura 4.7, onde:

- C_{Ki} são *manifolds* de pose da pessoa K aproximados por um plano computado pelo PCA;
- $P(C_{Ki}|C_{Kj})$ representa a transição do manifold C_{Ki} para o C_{Kj};
- M_K representa o manifold de aparência.

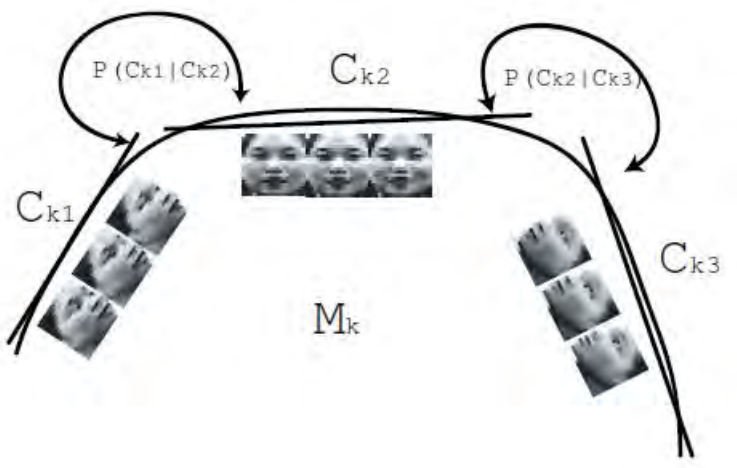

Figura 4.7: Dinâmica entre *manifolds* de pose (LEE et al., 2005).

Além disso, este trabalho também utiliza *frames* consecutivos para se definir uma máscara cujos elementos representam a probabilidade de um pixel corresponder à uma oclusão. Essa máscara é atualizada iterativamente por meio da diferença entre a imagem observada em cada *frame* e a reconstrução da imagem prevista do *frame* anterior.

Modelagem em Mistura Gaussiana no Movimento da Face

Saeed et al. (2006) apresentaram um sistema de reconhecimento de pessoas que explora tanto a informação comportamental quanto a espacial. A informação comportamental baseia-se nas características estatísticas obtidas por meio dos sinais de deslocamento facial enquanto que a informação espacial consiste em uma extensão à abordagem tradicional *eigenface*.

Tanto as características das movimentações faciais quanto as variações pessoais no espaço de faces são modeladas por um GMM (*Gaussian Mixture Model*), sendo que a tarefa da classificação é realizada como um problema de tomada de decisão Bayesiano.

Este sistema proposto pode ser organizado em três diferentes módulos: reconhecedor estático, reconhecedor temporal e módulo de fusão. O reconhecedor estático calcula o PCA sobre um conjunto geral de imagens de faces com o objetivo de se obter um conjunto ortogonal de vetores (espaço de faces), no qual as faces são projetadas para se obter seus respectivos componentes *eigenface*. Neste ponto, a tarefa de identificação/verificação é realizada por um *framework* Bayesiano, sendo que cada indivíduo tem sua distribuição de imagens no espaço de imagens modelado por um GMM. No módulo de reconhecimento temporal é analisado o movimento facial inicialmente por meio do deslocamento dos olhos, nariz e boca em cada *frame*. Depois disso, estes sinais são transformados e normalizados para se obter vetores de características independentes do vídeo. A distribuição dos deslocamentos representada por estes vetores é modelada ao longo do tempo treinando-se um GMM e a classificação também é obtida por meio de um classificador Bayesiano. Por fim, o módulo de fusão integra as duas medidas de similaridade e computa as taxas de identificação/verificação do sistema multimodal.

Modelos Ocultos de Markov (HMM)

Liu e Chen (2003) propuseram uma técnica que utiliza o HMM (*Hidden Markov Model*) adaptativo para realizar o reconhecimento de faces baseado em vídeos.

O modelo de Markov é um modelo estatístico onde a informação futura depende apenas da informação atual. Ele é considerado estocástico, pois todas as transições de estado são probabilísticas. Este modelo é composto basicamente pelos estados e pelas probabilidades de transições entre os estados.

Na Figura 4.8 pode ser observada a representação de um modelo de Markov onde X_i representa os estados e a_{ij} representa a transição do estado X_i para o estado X_j.

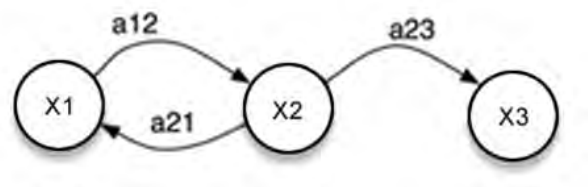

Figura 4.8: Exemplo de modelo de Markov.

De acordo com a suposição de Markov, os próximos estados dependem apenas do estado

atual, ou seja:

$$P(X_n = x | X_{n-1} = y) \tag{4.9}$$

onde x e y representam estados individuais e X_n representa o estado na iteração n do modelo de Markov.

O modelo oculto de Markov (HMM) é um processo duplamente estocástico, com um processo estocástico não observável (daí o nome oculto), mas que pode ser inferido por meio de outro processo estocástico que produz a sequência de observações (OLIVEIRA; MORITA, 1999). Os processos ocultos consistem em um conjunto de estados conectados por transições com probabilidades (autômato finito), enquanto que os processos observáveis (não ocultos) consistem de um conjunto de saídas ou observações que podem ser produzidas por cada um dos estados não observáveis de acordo a função de densidade de probabilidade.

Na Figura 4.9 pode ser observada a representação de um modelo oculto de Markov onde X_i representa os estados não observáveis, Y_i representa os estados observáveis, b_{ij} representa a probabilidade de saída (probabilidade de um estado observável Y_j ter sido gerado por um determinado estado não observável X_i) e a_{ij} representa a probabilidade de transição entre os estados não observáveis X_i e X_j.

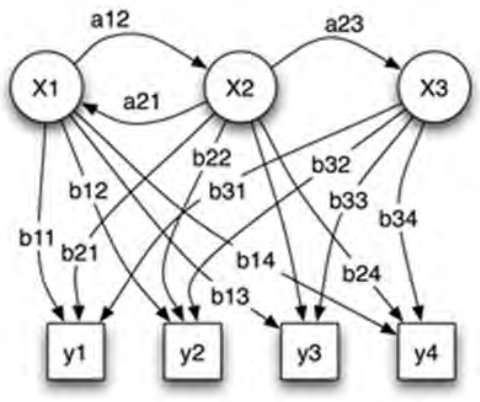

Figura 4.9: Exemplo de modelo oculto de Markov.

De acordo com a teoria do HMM, as seguintes suposições são feitas (WARAKAGODA, 1996):

A suposição de Markov : O próximo estado depende apenas do estado atual. Isto define na realidade, um HMM de primeira ordem. Existe a possibilidade de se levar em considera-

ção mais de um estado, de forma que ao se levar em consideração os N últimos estados, tem-se um HMM de ordem N.

A suposição estacionária : As probabilidades das transições entre os estados são independentes do tempo em que as transições ocorrem. Ou seja, as probabilidades de transição de estado levam em consideração apenas o estado atual e não o momento no tempo em que se encontra o estado atual.

Independência de saída : A observação atual é estatisticamente independente de observações anteriores.

Durante o processo de reconhecimento de faces, as características temporais do vídeo de teste são analisadas ao longo do tempo pelos HMMs referentes a cada indivíduo. Além disso, durante o processo de reconhecimento também existe uma adaptação dos HMMs dos indivíduos com as sequências de teste.

Um HMM contínuo pode ser modelado pela seguinte tripla:

$$\lambda = (A, B, \pi) \tag{4.10}$$

onde:

- A é a matriz de probabilidade de transição de estado. Essa matriz contém as probabilidades de transição de cada um dos estados não observáveis para todos os outros também não observáveis.

$$a_{i,j} = P(q_t = S_j | q_{t-1} = S_i), \quad 1 \leq i, j \leq N \tag{4.11}$$

- B são funções de densidade de probabilidade. Essas funções contêm as probabilidades de cada estado gerar todas as observações. No caso em que as observações são continuas, essas funções são dadas como misturas de Gaussianas.

$$\sum_{k=1}^{M} c_{ik} G(O; \mu_{ik}, U_{ik}), \quad 1 \leq i \leq N \tag{4.12}$$

- π é a distribuição do estado inicial. Este elemento do HMM é formado pelas possibilidades do primeiro estado do modelo pertencer a cada um dos possíveis estados não observáveis.

$$\pi = P(q_1 = S_i), \quad 1 \leq i \leq N \tag{4.13}$$

Onde:

S = estado;

q_t = estado no tempo t;

$c_{i,k}$ = coeficiente de mistura do k-ésimo componente da mistura Gaussiana;

O = vetor de observações;

μ = vetor médio;

U = matriz de covariância;

$G(O; \mu_{ik}, U_{ik})$ = função Gaussiana definida pelo vetor médio μ_{ik} e a matriz de covariância U_{ik}.

No algoritmo proposto por Liu e Chen (2003), cada *frame* da sequência de vídeo é considerado como uma observação. Para realizar a redução de dimensão das imagens, utiliza-se o PCA. Desta forma, cada imagem é reduzida para um vetor de características com dimensão menor.

A equação 4.14 representa uma base F de imagens de face com L indivíduos, cada um com uma sequência de vídeo contendo T *frames*:

$$F = \left\{ f_{l,1}, f_{l,2}, ..., f_{l,T} \right\}, \quad 1 \leq l \leq L \tag{4.14}$$

Cada imagem da base de dados contém apenas a parte da face. Realizando a transformada PCA para esses $L * T$ exemplos, obtém-se um *eigenspace* formado pelos autovetores $V_1, V_2, ..., V_d$. Os vetores de características obtidos a partir destas imagens com o uso do PCA são utilizados como os vetores de observação no treinamento do HMM.

Treinamento

Cada indivíduo da base de dados é modelado por um HMM totalmente conectado de N estados. Na Figura 4.10 é ilustrado o processo de treinamento, onde as estatísticas da sequência de treinamento e suas dinâmicas temporais são aprendidas por um HMM.

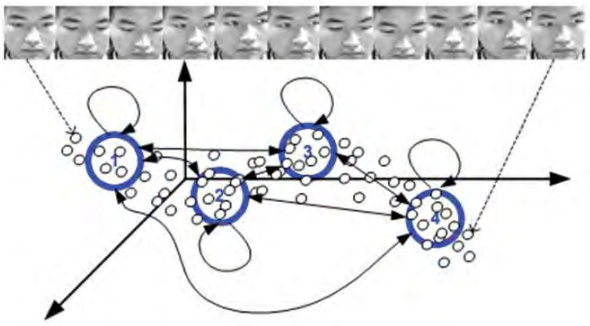

Figura 4.10: Processo de treinamento do HMM (LIU; CHEN, 2003).

O treinamento para cada indivíduo ocorre da seguinte forma:

1. O vetor HMM $\lambda = (A, B, \pi)$ é iniciado. A quantização de vetor é utilizada para separar os vetores de observação em N classes, sendo que os vetores associados com cada classe são utilizados para gerar as estimativas iniciais para B.

2. Utiliza-se o algoritmo EM (*Expectation Maximization*) para reestimar os parâmetros do modelo com o intuito de maximizar a probabilidade $P(O|\lambda)$, que representa a probabilidade de se gerar o conjunto de observações O a partir do modelo λ. Até que a probabilidade $P(O|\lambda)$ convirja, os coeficientes devem ser reestimados, de acordo com o EM (GAUVAIN; LEE, 1994), da seguinte forma:

$$\pi_i = \frac{P(O, q_i = i|\lambda)}{P(O|\lambda)} \tag{4.15}$$

$$a_{ij} = \frac{\sum_{t=1}^{T} P(O, q_{t-1} = i, q_t = j|\lambda)}{\sum_{t=1}^{T} P(O, q_{t-1} = i|\lambda)} \tag{4.16}$$

$$c_{ik} = \frac{\sum_{t=1}^{T} P(q_t = i, m_{q_t t} = k|O, \lambda)}{\sum_{t=1}^{T} \sum_{k=1}^{M} P(q_t = i, m_{q_t t} = k|O, \lambda)} \tag{4.17}$$

$$\mu_{ik} = \frac{\sum_{t=1}^{T} O_t P(q_t = i, m_{q_t,t} = k|O, \lambda)}{\sum_{t=1}^{T} P(q_t = i, m_{q_t t} = k|O, \lambda)} \tag{4.18}$$

$$U_{ik} = (1-\alpha)C_e + \alpha \frac{\sum_{t=1}^{T} (O_t - \mu_{ik})(O_t - \mu_{ik})^T P(q_t = i, m_{q_t t} = k|O, \lambda)}{\sum_{t=1}^{T} P(q_t = i, m_{q_t t} = k|O, \lambda)} \tag{4.19}$$

onde $m_{q_t t}$ indica o componente de mistura para o estado q_t e tempo t. A equação 4.19 é utilizada para adaptar a estimação de variância de C_e, que representa um modelo geral para a variância de todos os indivíduos. O parâmetro α é um fator de ponderação.

Reconhecimento

Durante o processo de reconhecimento, todas as faces são projetadas no *eigenspace* obtido na fase de treinamento, formando os vetores de características que são utilizados como vetores de observação para cada HMM. A sequência é reconhecida como um indivíduo k se:

$$P(O|\lambda_k) = \max P(O|\lambda) \tag{4.20}$$

Ou seja, o indivíduo será reconhecido se o seu modelo HMM λ tiver a maior probabilidade de ter gerado o conjunto de observações O de entrada.

Adaptação

Além do reconhecimento, o trabalho de Liu e Chen (2003) ainda propõe a adaptação do modelo das pessoas na fase de teste. Tal abordagem se baseia na idéia dos sistemas de reconhecimento de fala onde se obtém um desempenho melhor nos que são dependentes do locutor em relação aos que são independentes de locutor. Ou seja, os sistemas dependentes de locutor necessitam de uma grande quantidade de dados de treinamento de cada locutor do sistema fazendo com que o resultado seja melhor.

De forma análoga à abordagem do reconhecimento de falas, para se ter um resultado mais satisfatório, durante o processo de reconhecimento de face (depois de reconhecer a sequência de teste como um sujeito) é possível usar tal sequência para atualizar o modelo do indivíduo.

Tal atualização não é feita incondicionalmente a partir do momento do reconhecimento. Antes de realizar a mesma, é necessário se medir o quão confiável é o resultado do reconhecimento para a sequência atual baseado em algum critério. O critério utilizado por Liu e Chen (2003) é a diferença entre o maior *score* e o segundo maior *score*. A razão para o uso deste critério é que a diferença de *scores* para reconhecimentos corretos tende a ser maior do que para reconhecimentos incorretos. Desta forma, compara-se esta diferença entre os dois melhores *scores*. Caso seja maior que um determinado limiar, o modelo é atualizado e, caso contrário, mantém-se o modelo inalterado.

Para fazer a atualização do HMM, utiliza-se a técnica de adaptação MAP (*Maximum a Posteriori*) (GAUVAIN; LEE, 1994). Tal técnica consiste em estimar um novo $\lambda = (A, B, \pi)$ dados os vetores de observação de uma sequência de teste. Tal estimativa é feita da seguinte forma:

1. Usa-se o λ_{old} como os parâmetros iniciais de λ. Usa-se o algoritmo EM para re-estimar λ , exceto a estimação da média.

2. A estimação da média é calculada da seguinte forma:

$$\mu = (1-\beta)\mu_{ik}^{old} + \beta \frac{\sum_{t=1}^{T} O_t P(q_t = i, m_{q,t} = k|O,\lambda)}{\sum_{t=1}^{T} P(q_t = i, m_{q,t} = k|O,\lambda)} \tag{4.21}$$

onde μ_{ik}^{old} é o vetor médio do HMM λ_{old} e β é o fator de peso que fornece o viés entre a estimativa anterior e o dado atual.

4.3 Considerações Finais

Neste capítulo foram apresentados alguns métodos de reconhecimento de faces descritos na literatura. Os métodos foram divididos em duas categorias: os baseados em imagens estáticas e os baseados em vídeos. Os métodos baseados em vídeos, por sua vez, foram divididos em duas sub-categorias: métodos que não utilizam informações temporais e métodos que utilizam informações temporais. Os métodos baseados nas autofaces (*eigenfaces*) e nos modelos ocultos de Markov(HMM) foram apresentados de forma mais detalhada, pois foram utilizados neste trabalho para descrição e reconhecimento de faces a partir dos *frames* dos vídeos.

No próximo capítulo é apresentada a arquitetura proposto por Penteado e Marana (2009) para sistemas de autenticação biométrica de usuários em ambientes de e-Learning, baseada em reconhecimento de faces a partir de vídeo.

Capítulo 5

Reconhecimento de Faces a Partir de Vídeo para Sistemas de *E-Learning*

Como foi mencionado, um dos objetivos deste trabalho é analisar técnicas avançadas de reconhecimento de padrões visando a substituição do módulo de reconhecimento de faces da arquitetura proposta por Penteado e Marana (2009) para sistemas de autenticação biométrica de usuários em ambientes *e-Learning*, baseada em reconhecimento de faces a partir de vídeo. Este capítulo apresenta detalhes dessa arquitetura.

5.1 Arquitetura Proposta por Penteado e Marana

Para a autenticação biométrica de usuários em ambientes *e-Learning*, Penteado e Marana (2009) propuseram uma arquitetura cliente/servidor de modo a encapsular o acesso ao ambiente de *e-Learning*, reforçando seu processo de autenticação.

Para frequentar o curso a distância, via Web, o aluno deve executar uma aplicação desktop em seu computador. Então, a aplicação requisita a página do curso para o servidor Web. Junto com a requisição, uma consulta é também enviada para verificar se a página Web sendo requisitada exige autenticação biométrica. Este poderia ser o caso para uma avaliação ou um conteúdo protegido, por exemplo. Se não for, a página é enviada de volta ao cliente. Caso contrário, a aplicação cliente começa a capturar o vídeo por meio da *webcam* conectada ao computador do usuário. A aplicação processa o vídeo, detecta e pré-processa as faces presentes nos quadros amostrados, concatena os vetores de características extraídos e envia-os para o servidor. O servidor consulta a base de dados pelo traço biométrico e retorna a resposta para o cliente desktop. A aplicação desktop então bloqueia ou retorna a página Web. A Figura 5.1 ilustra o fluxo de

informações entre os módulos da arquitetura proposta.

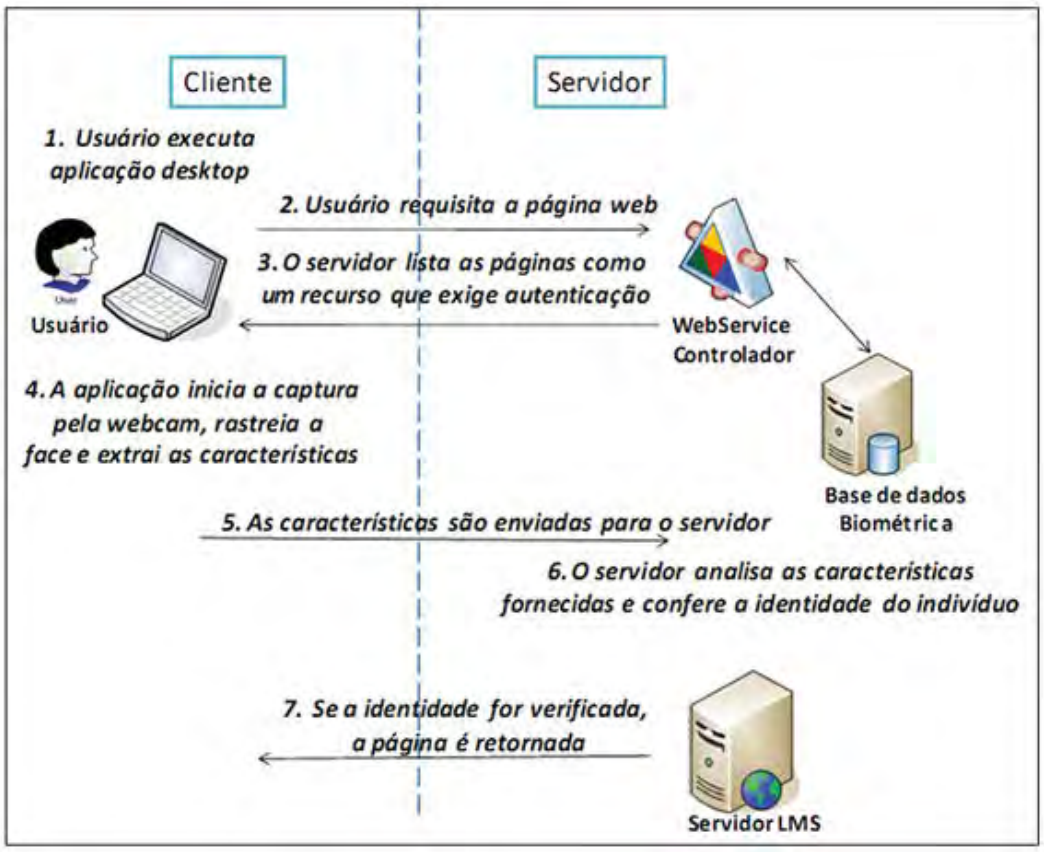

Figura 5.1: Arquitetura proposta para a autenticação biométrica de usuários em ambientes *e-Learning*, baseada em reconhecimento de faces a partir de vídeo. Passos 1 a 7: interação entre os módulos localizados no cliente e no servidor (PENTEADO, 2009).

5.2 Detecção das Faces no Vídeo

O algoritmo utilizado para detectar as faces no vídeo é o de Viola-Jones (VIOLA; JONES, 2001). Este algoritmo procura características que codificam alguma informação do padrão a ser detectado em uma imagem. Neste contexto, as características utilizadas são as de Haar (VIOLA; JONES, 2001), onde são explorados os contrastes naturais da face, respeitando seus relacionamentos espaciais.

Cada característica de Haar é representada por quadriláteros (quadrados e retângulos) com partes pretas e brancas. Para calcular o valor de tal característica na imagem, deve-se posicionar o quadrilátero sobre a imagem e calcular o somatório de todos os pixels que são abrangidos pela parte branca e, após, subtrair o somatório dos pixels abrangidos pela parte preta do quadrilátero. Na Figura 5.2 são apresentadas algumas características de Haar utilizadas para a detecção de

faces.

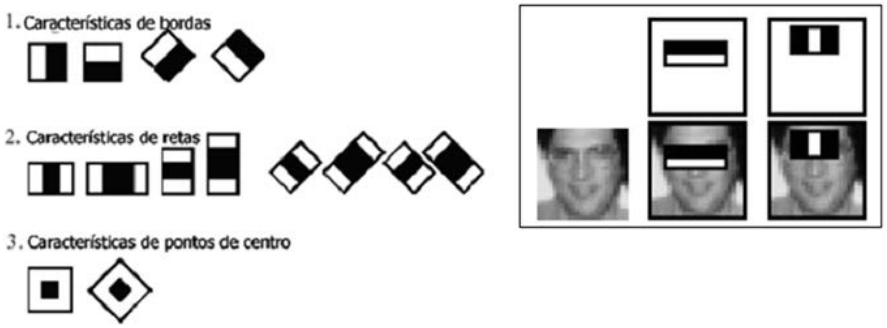

Figura 5.2: Características de Haar para detecção de faces (VIOLA; JONES, 2001).

Para facilitar os cálculos das características, utiliza-se uma imagem integral, onde o pixel $ii(x,y)$ armazena a soma de todos os valores dos pixels posicionados acima e à esquerda dele. Depois de calculada a imagem integral, para o cálculo das características de Haar basta utilizar os valores das extremidades dos quadriláteros (que já representam somatórios). Na Figura 5.3 é apresentada uma representação da imagem integral.

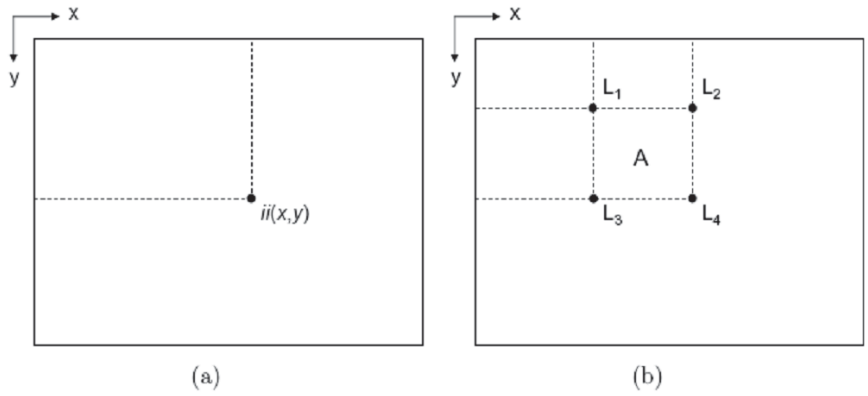

Figura 5.3: Representação da imagem integral: (a) o valor do ponto $ii(x,y)$ é obtido por meio da soma dos valores de todos os pixels acima e à esquerda; (b) a região A pode ser calculada em termos dos valores $L1$, $L2$, $L3$ e $L4$ como sendo igual a $L4+L1-(L2+L3)$ (VIOLA; JONES, 2001).

5.3 Pré-Processamento

Na etapa de pré-processamento, as imagens das faces são extraídas e convertidas para escala de cinza, redimensionadas para um tamanho padrão (64x64 pixels, escolhido empirica-

mente), por interpolação bilinear, e submetidas à equalização de histograma para ajuste nos seus contrastes. A Figura 5.4 ilustra o reflexo destas operações em uma imagem de face.

Figura 5.4: Resultados (esquerda para a direita) das operações de pré-processamento realizadas em uma imagem de face (PENTEADO, 2009).

5.4 Extração das Características

Para a extração de características, foi utilizado o PCA como redutor de dimensão das imagens de face, como explicado na seção 4.2.1.

Inicialmente, um conjunto de imagens representando o domínio da aplicação é selecionado para o treinamento, ou seja, para a criação do espaço de faces. Em um segundo momento, as imagens de suas faces são coletadas e projetadas neste espaço, gerando os *templates*, ou seja, os vetores de características que representam a face de cada indivíduo. A seguir é criada a base de dados contendo os *templates* de todos os usuários.

5.5 Reconhecimento da Face

Uma vez gerado o espaço de faces, as imagens de face a serem reconhecidas são projetadas nele. Após a projeção, tem-se a representação da face em termos da combinação linear da base dos autovetores determinados na fase de treinamento. Os coeficientes obtidos são então usados como vetores de características representando as faces. Para medir a dissimilaridade entre os vetores de características armazenados na base de dados e os obtidos nos *frames*, faz-se o uso da distância Euclidiana.

Dado um *frame* do vídeo, a distância Euclidiana é calculada entre o vetor de característica desse *frame* e cada vetor de características (template) armazenado na base de dados, de forma que a menor distância obtida determina a identidade da pessoa naquele *frame*.

5.6 Identificação do Usuário

Para determinar a identidade do usuário (aluno) que está acessando o ambiente de e-Learning, é usada a regra de maioria de votos. Nesta abordagem, a cada *frame i* do vídeo, identifica-se a face detectada, evento representado como uma função binária 5.1:

$$d_{i,m} = \begin{cases} 1, \text{se o resultado do i-ésimo } \textit{frame} \text{ for a classe m} \\ 0, \text{ caso contrário} \end{cases} \tag{5.1}$$

A identidade do aluno será atribuída à pessoa cadastrada no banco de dados para a qual se obtém a maioria de votos, conforme a equação 5.2:

$$ID = \arg \max_{m=1...M} \left(\sum_{i=1}^{N} d_{i,m} \right) \tag{5.2}$$

onde N é o número de *frames* do vídeo e M é o número de identidades cadastradas no banco de dados do sistema.

5.7 Considerações Finais

Neste capítulo foi apresentado sucintamente a arquitetura proposta por Penteado e Marana (2009) para sistemas de autenticação biométrica de usuários em ambientes *e-Learning*, baseada em reconhecimento de faces a partir de vídeo. Pela natureza da arquitetura cliente/servidor dos sistemas Web de e-Learning, eles projetaram uma arquitetura eficiente, de modo a se distribuir a carga computacional e o tráfego de informações entre as estações cliente e servidor, independentemente do sistema de gerenciamento de aprendizado utilizado.

Resultados experimentais indicaram que o sistema biométrico de identificação proposto apresentou um bom desempenho, respeitadas certas restrições, como a padronização da iluminação. O algoritmo de detecção de faces de Viola e Jones mostrou-se eficiente tanto na taxa de acertos quanto no tempo de processamento para processamento *online*. Embora os resultados também tenham sido considerados bons com relação à determinação da identificação dos alunos a partir dos *frames* dos vídeos, foi recomendada uma investigação mais aprofundada com relação ao uso de técnicas mais avançadas de reconhecimento de padrões e que pudessem considerar as informações temporais, intrínsicas dos vídeos (PENTEADO, 2009; PENTEADO; MARANA, 2009). Tais recomendações motivaram a proposição deste trabalho, apresentada no próximo capítulo.

Capítulo 6

Proposição

Esta dissertação de mestrado tem como objetivo principal o estudo, a implementação e a comparação de diferentes métodos de reconhecimento de padrões para promover o reconhecimento rápido e eficiente de faces obtidas de vídeos capturados por web câmeras, e, desse modo, possibilitar a identificação *online* e mais robusta de usuários de aplicações Web, estendendo, desse modo o trabalho proposto por Penteado e Marana (2009), com a incorporação de técnicas mais adequadas e eficazes ao sistema de autenticação biométrica de usuários de aplicações WEB proposto por esses pesquisadores.

Propõe-se avaliar os seguinte classificadores:

- Bayesiano;
- Redes Neurais Artificiais Multicamadas (ANN MLP - *Artificial Neural Network Multilayer Perceptron*);
- Redes Neurais Artificiais baseadas em Mapas Autoorganizáveis de Kohonen (ANN SOM - *Artificial Neural Network Self-Organizing Maps*);
- K Vizinhos mais Próximos (KNN - *K Nearest Neighbors*);
- Máquinas de Vetores de Suporte (SVM - *Suport Vector Machines*), com as variantes sem *Kernel*, *Kernel* linear, *Kernel* RBF (*Radial Basis Function*) e *Kernel* sigmóide;
- Floresta de Caminhos Ótimos (OPF - *Optimum Path Forest*).

Como método de reconhecimento de padrões faciais considerando-se as informações temporais, propõe-se a avaliação da técnica baseada em Modelos Ocultos de Markov (HMM - *Hidden Markov Model*).

Capítulo 7

Material e Métodos

Neste capítulo são apresentados o material e a metodologia utilizados para o desenvolvimento deste trabalho.

7.1 Material

Para comparar o desempenho das técnicas avançadas de reconhecimento de padrões (classificadores e HMM) com os resultados obtidos anteriormente com o uso do vizinho mais próximo utilizando a medida de distância Euclidiana (PENTEADO, 2009; PENTEADO; MARANA, 2009), foram utilizadas duas bases de dados: *Honda/UCSD Video Database* e *Recogna Video Database*, descritas nas subseções 7.1.1 e 7.1.2, respectivamente. A configuração dos computadores utilizados nos experimentos é descrita na subseção 7.1.3.

7.1.1 Banco de Dados *Honda/UCSD Video Database*

O banco de dados *Honda/UCSD Video Database*, distribuído pela Universidade da Califorinia, San Diego, Estados Unidos[1], foi criado para permitir a avaliação de algoritmos de reconhecimento e rastreamento de faces em vídeos. Esta base de dados de vídeos é composta por dois conjuntos. O primeiro conjunto de dados foi gravado com uma câmera SONY EVI-D30 no instituto de pesquisa da Honda, California, San Diego, Estados Unidos em 2002. Os vídeos deste conjunto são divididos em 3 categorias diferentes: treinamento, teste, teste de oclusão, com 20, 42, 13 vídeos respectivamente de 20 seres humanos (LEE et al., 2005) com duração entre 15 a 57 segundos. O segundo conjunto de dados foi gravado com uma câmera SONY DFW-V500 no

[1] http://vision.ucsd.edu/ leekc/HondaUCSDVideoDatabase/HondaUCSD.html.

laboratório de Visão Computacional da Universidade da California, San Diego, Estados Unidos em 2004. Este conjunto é composto de 30 vídeos de 15 pessoas com duração entre 31 e 125 segundos, divididos em 2 categorias diferentes: treinamento e teste (LEE; KRIEGMAN, 2005).

Nessa base de dados, os vídeos foram gravados em um ambiente fechado, com uma taxa de 15 *frames* por segundo e uma resolução de 640x480 pixels. Existem pelo menos dois vídeos para cada pessoa, gravados em sessões distintas.

Como a variação das poses é um dos grandes desafios do reconhecimento de faces, durante a captura dos vídeos as pessoas foram orientadas a rotacionar e movimentar suas cabeças com velocidades variadas e em várias direções. Com isso, existem nos vídeos desse banco de dados imagens da mesma face com grande variabilidade nas poses, tanto no plano 2D como no espaço 3D.

Neste trabalho, foram utilizados os vídeos das categorias de treinamento e testes do primeiro conjunto de vídeos da base de dados *Honda/UCSD Video Database*. Na Figura 7.1 são apresentados alguns exemplos de faces extraídas de vídeos da base de dados Honda/UCSD, por meio do algoritmo Viola-Jones.

Figura 7.1: Imagens de faces extraídas dos vídeos da base Honda/UCSD por meio do algoritmo Viola-Jones (LEE et al., 2003).

7.1.2 Banco de Dados *Recogna Video Database*

O banco de dados *Recogna Video Database* é uma base de dados de vídeos criada especialmente para os experimentos deste trabalho. Como a base de dados Honda/UCSD tem algumas limitações, como número reduzido de pessoas (20 pessoas), os vídeos são curtos e as poses não são muito naturais para um ambiente de *e-Learning*, foi decidido criar outra base de dados para permitir uma melhor avaliação dos métodos de reconhecimento de padrões das faces.

A base de dados *Recogna Video Database* é dividida em dois subconjuntos: o conjunto de treinamento (gravado na primeira sessão) e o conjunto de testes (gravado na segunda sessão), ambos contendo 50 vídeos (um para cada um dos 50 indivíduos). O intervalo de tempo entre as sessões foi de 10 dias.

Os vídeos foram gravados em duas sessões distintas, em um ambiente fechado, com uma taxa de 30 *frames* por segundo e uma resolução de 640x480 pixels, com duração fixa de 5 minutos. Para a gravação dos vídeos, as pessoas foram instruídas a agirem naturalmente ao operar um *netbook*, tanto na sessão de treino quanto na sessão de teste. O *netbook* utilizado nos experimentos é do modelo Magalhaes Novadata, com a seguinte configuração: sistema operacional Windows 7, processador Intel Atom N270 (512K de cache, 1.60 GHz de clock), memória RAM de 1GB, HD de 160GB e câmera de 1,3MP com resolução máxima de 640x480 pixels a 30fps.

Na Figura 7.2, são apresentados alguns exemplos de faces extraídas de vídeos da base de dados *Recogna Video Database*, por meio do algoritmo Viola-Jones.

Figura 7.2: Imagens de faces extraídas dos vídeos da base de dados *Recogna Video Database* por meio do algoritmo Viola-Jones.

7.1.3 Hardware

Praticamente todos os experimentos foram realizados em um computador pessoal com a seguinte configuração: sistema operacional Windows XP com SP3, processador Intel Core 2 Duo E6420 (4M de cache, 2.13 GHz de clock), memória RAM de 2GB, HD de 240GB. Os únicos experimentos que foram realizados em outra máquina (pelo fato de necessitarem um tempo de processamento significativamente maior) foram aqueles nos quais foram utilizados classificadores de padrões treinados com todos os descritores de faces obtidos nos *frames* de todos os vídeos de treinamento e de testes da base de dados de vídeos *Recogna Video Database*. Para esses experimentos, utilizou-se o computador servidor com a seguinte configuração: sistema operacional Windows XP SP3, processador Intel Xeon E5540 (8M de cache, 2.53 GHz de clock), memória RAM de 3GB (máximo reconhecido pelo sistema operacional Windows XP SP3), 3 HDs de 1 TB cada.

7.2 Métodos de Reconhecimento de Faces

Neste trabalho foram avaliados métodos de reconhecimento de padrões baseados em classificação e em modelos de Markov, para o reconhecimento de faces a partir de vídeos.

7.2.1 Classificadores

A primeira proposta baseia-se na substituição do classificador de padrão vizinho mais próximo com a distância Euclidiana como medida de dissimilaridade no reconhecimento facial por classificadores de padrões mais robustos. Neste contexto, os padrões são os descritores das faces que devem ser submetidos aos classificadores para se determinar a qual classe (indivíduo) pertencem. Foram avaliados os seguintes métodos de classificação: Redes Neurais ANN MLP e ANN SOM (seção 3.1), Classificador Bayesiano (seção 3.2), K Vizinhos mais Próximos (seção 3.3), Máquinas de Vetores de Suporte (SVM) (seção 3.4) e Floresta de Caminhos Ótimos (OPF) (seção 3.5).

Os descritores das faces, utilizados como padrões para os classificadores, são obtidos pelo método *eigenfaces*, sendo a projeção da imagem da face, detectada por meio do algoritmo Viola-Jones, no espaço de faces. Sendo assim, os descritores das faces obtidas nos vídeos do conjunto de treinamento são rotulados e utilizados no processo de treinamento supervisionado dos classificadores.

Os passos de detecção, segmentação, pré-processamento, extração das características e identificação por maioria dos votos são semelhantes aos passos propostos por Penteado e Marana (2009) e descritos no Capítulo 5, de forma que os *frames* nos quais não são encontradas faces são ignorados.

As diferenças são as seguintes:

- Na implementação utilizada por Penteado e Marana (2009), a base de dados é formada a partir dos *templates* dos indivíduos pertencentes ao sistema, enquanto que ao usar classificadores, todos os descritores de faces obtidos no conjunto de vídeos de treinamento devem ser rotulados e utilizados para o treinamento do classificador, dispensando o armazenamento desses *templates* na base de dados;

- Na fase de identificação do indivíduo presente em cada *frame*, os classificadores são utilizados com o intuito de se determinar o indivíduo que está presente em um dado *frame*, enquanto que na implementação de Penteado e Marana (2009), a classificação de uma amostra individual era feita de acordo com a menor distância (Euclidiana) apresentada entre uma amostra e os *templates* de todos os indivíduos da base de dados.

Conforme proposto por Penteado e Marana (2009), utiliza-se a função cosseno de Mahalanobis para se escolher 3 descritores de faces que menos variam dentre todos os descritores

obtidos ao longo do vídeo de treinamento de cada indivíduo. Essas 3 amostras de cada indivíduo são armazenadas como *templates* na base de dados.

A função cosseno de Mahalanobis (RAMANATHAN, 1999) é dada por:

$$D_{MahCosine(u,v)} = cos(\theta_{mn}) = \frac{mn}{|m|\,|n|} \tag{7.1}$$

onde m e n são os dois vetores de características (descritores de faces) a serem comparados.

Com a base de dados já criada, a fase de testes consiste em analisar todos os vídeos da sessão de testes. De cada um dos vídeos, são extraídas as faces de todos os seus *frames* por meio do algoritmo Viola-Jones, sendo que após a extração, esta é projetada no espaço de faces e o vetor com os coeficientes obtidos é utilizado como descritor da face. Para todos os experimentos, usou-se normalização desses descritores, da seguinte forma (HUNT, 2007):

$$Z = \frac{X - \mu}{\sigma} \tag{7.2}$$

onde Z representa o descritor normalizado, X representa o descritor sem normalização, μ representa a média entre todos os descritores de todos os vídeos de uma base de dados de vídeos e σ representa o desvio padrão entre todos os descritores de uma base de dados de vídeos. Os valores de μ e σ são calculados para cada base de dados de vídeos utilizada.

Utilizando-se a técnica da maioria dos votos, a identidade do indivíduo presente em cada vídeo é determinada como sendo a identidade cadastrada que acumular mais votos ao longo da análise de todos os *frames* do vídeo.

O diagrama apresentado na Figura 7.3 ilustra as etapas que compõem o experimento que utiliza os classificadores de padrões para o reconhecimento de faces a partir de vídeos.

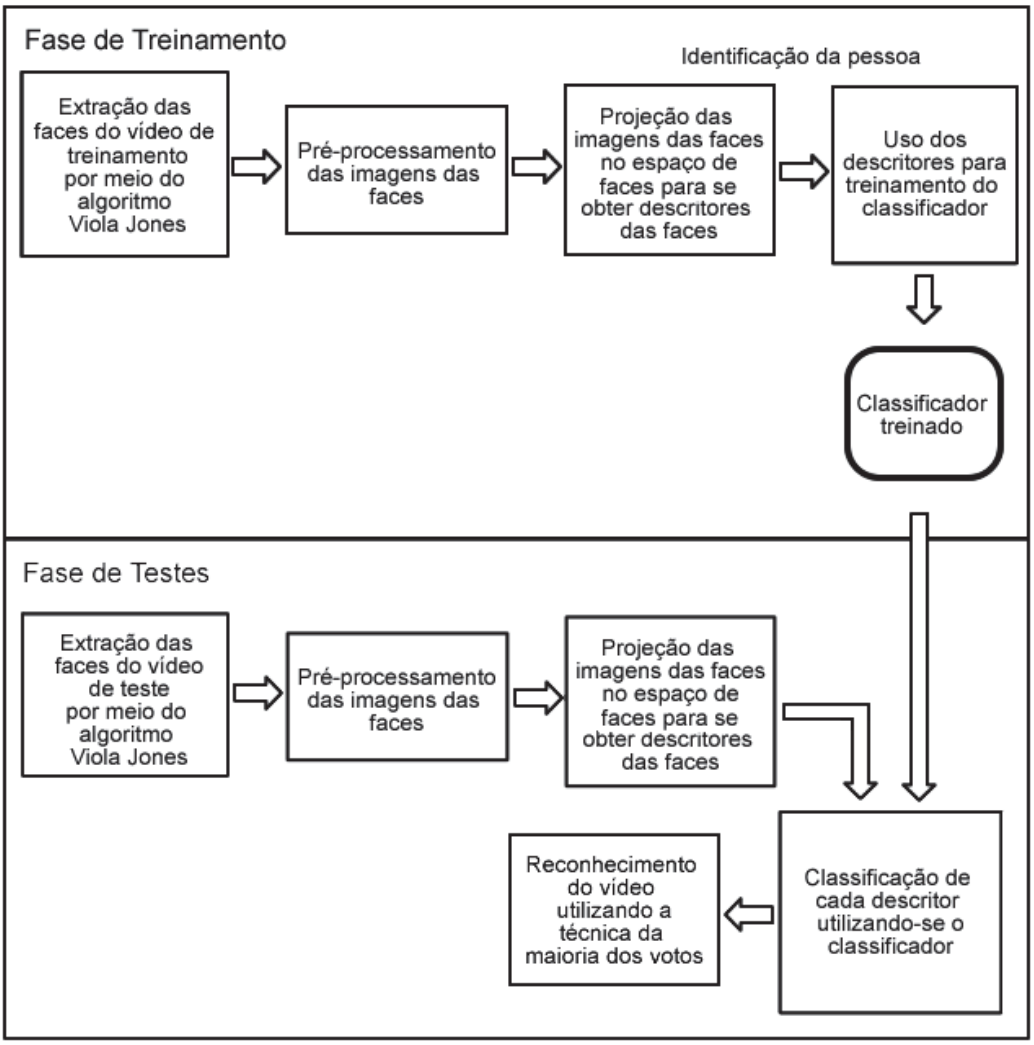

Figura 7.3: Etapas do experimento com uso dos classificadores para a realização do reconhecimento facial a partir de vídeo.

Os passos descritos desse experimento foram realizados tanto para a base de dados de vídeos *Honda/UCSD Video Database* quanto para a *Recogna Video Database*.

Os experimentos que envolvem classificadores foram realizados de três formas diferentes, em relação ao número de *frames* utilizados:

- Todos os *frames* são considerados, e consequentemente todos os descritores de faces encontrados nesses *frames*;
- Números fixos de *frames* para treinamento em conjuntos separados (50, 100 e 200), de forma que o classificador treinado com cada conjunto desses é utilizado na classificação de todos os descritores de faces encontrados nos *frames* do conjunto de testes;

- Intervalos de 1 segundo para a captura de *frames*. Os vídeos foram divididos em intervalos de 1 segundo para se verificar o desempenho semelhante ao de um sistema que realiza a autenticação após coletar o vídeo por 1 segundo.

Pelo motivo da base *Recogna Video Database* apresentar uma maior quantidade de vídeos e os vídeos serem mais longos, para esta base foram realizados experimentos com subconjuntos de diferentes tamanhos. Foram extraídos subconjuntos de 50, 100 e 200 descritores de cada vídeo dos conjuntos de treinamento e testes. Esses descritores foram selecionados entre todos os descritores obtidos dos vídeos de teste, de forma que os subconjuntos de descritores eram compostos da seguinte forma:

$$d_1, d_{\lfloor \frac{N}{M} \rfloor}, d_{2\times \lfloor \frac{N}{M} \rfloor}, , d_{3\times \lfloor \frac{N}{M} \rfloor} ... d_N \tag{7.3}$$

onde d representa o descritor de determinado índice, N representa o conjunto total de descritores obtidos de um indivíduo e M representa o número desejado de amostras para o conjunto (50, 100 ou 200).

Para realizar a comparação entre tais classificadores, várias bibliotecas foram utilizadas. Todas as bibliotecas têm o código fonte em C/C++ e foram compiladas no Windows, utilizando-se a IDE Visual Studio 2005. A implementação do classificador baseado em OPF foi obtido por meio do *framework* LibOPF Papa et al. (2009a). As implementações do classificador SVM foram obtidas por meio de duas bibliotecas, a LibSVM (CHANG; LIN, 2001) e a SVM Torch (COLLOBERT; BENGIO, 2001). A implementação das redes neurais artificiais foi obtida por meio da biblioteca FANN (NISSEN, 2003). Além desses classificadores, foram utilizados também o SOM, KNN e Bayes, todos implementados pelo grupo de pesquisa em Biometria e Reconhecimento de Padrões, RECOGNA (UNESP-Bauru).

Para o classificador SVM da biblioteca SVM Torch foi utilizado o *kernel* RBF e para o classificador SVM da biblioteca LibSVM foram utilizados os *kernels* RBF, sigmóide, linear ou sem *kernel*.

A configuração para a ANN MLP utilizada para a base de dados de vídeos *Honda/UCSD Video Database* foi i : h1 : h2 : o, onde i = 50 (tamanho do descritor), h1 = h2 = 32 e o = 50 (número de indivíduos) são os números de neurônios nas camadas de entrada, escondida e saída, respectivamente. A configuração para a ANN MLP utilizada para a base de dados de vídeos *Recogna Video Database* foi i : h1 : h2 : o, onde i = 50 (tamanho do descritor), h1 = h2 = 128 e o = 50 (número de indivíduos). O classificador ANN MLP foi treinado com o algoritmo *backpropagation*, com uma arquitetura escolhida empiricamente.

O tamanho do mapa utilizado no classificador SOM foi 50x50 neurônios e foram utilizadas 10 iterações no treinamento.

A fase de treinamento do KNN compreendeu testes alternando-se o valor de k para se chegar a um k que possibilitasse a maior precisão de classificação.

Todos os experimentos foram repetidos 3 vezes com os conjuntos de treinamento e testes fixos para computar os tempos médios de execução e as taxas de reconhecimento.

7.2.2 Modelo de Markov

A segunda proposta baseia-se nos Modelos Ocultos de Markov (HMM) para se levar em consideração a informação dinâmica do vídeo no reconhecimento de faces. Os algoritmos do HMM utilizados neste trabalho foram obtidos na *toolkit* HTK (*Hidden Markov Model Toolkit*) 3.4 (YOUNG et al., 2006).

Nesta proposta, a base de dados é composta pelos modelos de HMM individuais que devem ser treinados com vídeos individuais do conjunto de treinamento. As fases de detecção das faces no vídeo, segmentação e pré-processamento da imagem, extração das características são as mesmas propostas por Penteado e Marana (2009). No entanto, a fase de reconhecimento consiste em obter, para cada um dos modelos individuais, a probabilidade de uma dada sequência de *frames* (observações) ser gerada pelo modelo. A pessoa cujo modelo apresentar maior probabilidade de gerar aquela sequência, será a pessoa reconhecida.

Esta técnica se assemelha à apresentada na seção 4.2.2, com a ressalva de ser utilizado o algoritmo Baum Welch (YOUNG et al., 2006) no treinamento dos modelos. As alterações do método proposto são focadas basicamente nas fases de treinamento e reconhecimento:

- Fase de treinamento: Para realizar o treinamento do modelo HMM, deve ser gerado o arquivo de observações no formato HTK com as imagens pré processadas após a extração do vídeo. Com as observações já no formato HTK, deve-se treinar o modelo HMM utilizando-se a ferramenta HInit do HTK. Esta ferramenta utiliza o algoritmo Baum Welch (RABINER, 1989) para treinar um HMM.

- Fase de reconhecimento: Para a realização do reconhecimento, é necessário também gerar os arquivos com as observações no formato HTK para o vídeo do qual se deseja descobrir o indivíduo presente. Após gerar este arquivo é necessária a utilização de algum algoritmo para se obter a probabilidade do conjunto de observações a ser gerado por cada um dos modelos. O algoritmo escolhido para desempenhar tal função neste trabalho é

o Forward (RABINER, 1989), responsável por retornar a probabilidade $P(O|\lambda)$, ou seja a probabilidade do conjunto de observações O ser gerado a partir do modelo λ. Para reconhecer o sujeito presente em um vídeo, é necessário repetir este passo para todos os modelos, sendo que o modelo do indivíduo que retornar a maior probabilidade de gerar o conjunto de observação em questão, indicará o sujeito reconhecido.

O diagrama apresentado na Figura 7.4 ilustra as etapas que compõem o experimento que utiliza o HMM para o reconhecimento de faces a partir de vídeos.

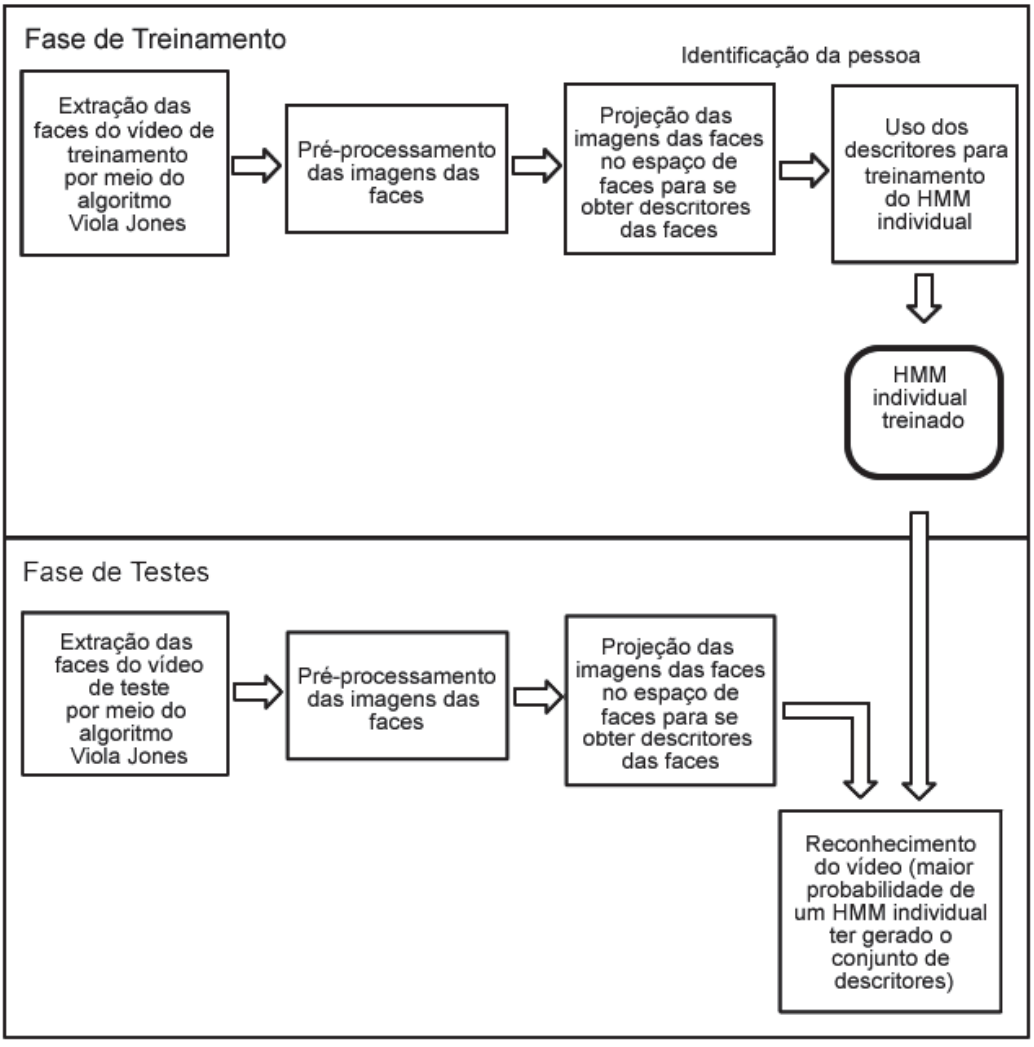

Figura 7.4: Etapas do experimento com uso do HMM para a realização do reconhecimento facial a partir de vídeo.

Os passos descritos desse experimento foram realizados tanto para a base de dados de vídeos *Honda/UCSD Video Database* quanto para a *Recogna Video Database*.

7.3 Acurácia dos Classificadores de Padrões

Uma forma de se medir a exatidão de um classificador de padrões na base de dados na qual ele foi aplicado é se utilizar a medida de acurácia proposta por Papa et al. (2009b). Essa forma de cálculo da acurácia não leva em consideração apenas o número de classificações corretas no total, mas sim o número de falsos positivos e falsos negativos para cada classe.

Seja Z as amostras do conjunto de testes, $|Z|$ o número total de amostras do conjunto de testes e $NZ(i), i = 1,...,c$ o número de amostras do conjunto de testes Z da classe i. Define-se $e_{i,1}$ e $e_{i,2}$ para cada classe i da seguinte forma:

$$e_{i,1} = \frac{FP(i)}{|Z| - NZ(i)} \quad \text{e} \quad e_{i,2} = \frac{FN(i)}{NZ(i)}, i = 1,...,c \tag{7.4}$$

onde $FP(i)$ e $FN(i)$ representam os falsos positivos e falsos negativos da classe i, respectivamente. Os erros $e_{i,1}$ e $e_{i,2}$ são utilizados para definir a soma parcial do erro de cada classe i da seguinte forma:

$$E(i) = e_{i,1} + e_{i,2} \tag{7.5}$$

A fórmula da acurácia é dada por:

$$Acc = \frac{2c - \sum_{i=1}^{c} E(i)}{2c} = 1 - \frac{\sum_{i=1}^{c} E(i)}{2c} \tag{7.6}$$

7.4 Considerações Finais

Neste capítulo foi apresentada a metodologia utilizada neste trabalho para abordar o problema do reconhecimento facial a partir de vídeos, além das bases de dados de vídeos nas quais as técnicas foram aplicadas. Por último, foi apresentada uma forma de se medir a precisão dos classificadores no que se refere à classificação correta de padrões (descritores de faces).

No próximo capítulo são descritos os resultados obtidos a partir da aplicação da metodologia apresentada no capítulo atual. São apresentados resultados do uso das medidas de distância, classificadores de padrões e HMM para o reconhecimento facial a partir de vídeo, em termos de precisão e tempo de execução.

Capítulo 8

Resultados

Neste capítulo são apresentados os experimentos realizados assim como os resultados obtidos no reconhecimento de faces a partir de vídeos. Nesses experimentos, foram utilizadas as bases de dados *Honda/UCSD Video Database* (LEE et al., 2003) e a *Recogna Video Database*, descritas nas seções 7.1.1 e 7.1.2, respectivamente.

Os experimentos foram realizados com o objetivo de comparar os vários métodos de classificação de padrões e o método baseado no modelo oculto de Markov, com a abordagem que utiliza o classificador vizinho mais próximo com as funções de distância Euclidiana e Mahalanobis para o reconhecimento de faces.

No primeiro experimento, o reconhecimento foi realizado usando-se o classificador vizinho mais próximo (baseado na menor medida de dissimilaridade entre os descritores de faces obtida por meio da distância Euclidiana), como é realizado no sistema desenvolvido por Penteado e Marana (2009), e também por meio da distância de Mahalanobis. No segundo experimento, foram utilizados diferentes classificadores de padrões (Bayesiano, Redes Neurais Artificiais Multicamadas, Redes Neurais Artificiais baseadas em Mapas Autoorganizáveis de Kohonen, K Vizinhos mais Próximos, Máquinas de Vetores de Suporte, com as variantes sem *Kernel*, *Kernel* linear, *Kernel* RBF (*Radial Basis Function*) e *Kernel* sigmóide, e OPF (Floresta de Caminhos Ótimos), com ou sem o uso do conjunto Z_2 para avaliação, como explicado na subseção 3.5.2. No terceiro experimento, foi utilizado o HMM para se considerar a informação temporal presente nos vídeos.

8.1 Reconhecimento de Faces Baseado em Distâncias

Neste experimento, gerou-se inicialmente o espaço de faces como descrito na subseção 4.2.1, utilizando-se os autovetores correspondentes aos 50 maiores autovalores (equivalente a uma representatividade de 71,1% no caso da base de dados *Recogna Video Database* e 87,5% na base de dados *Honda/UCSD Video Database*) e depois os descritores das faces foram obtidos por meio das projeções das imagens obtidas dos vídeos neste espaço. Portanto, o descritor de cada face é um vetor de características de dimensão 50.

Na Tabela 8.1 são mostrados os valores da taxa de reconhecimento correto dos *frames*, taxa de reconhecimento correto dos vídeos e o tempo médio de reconhecimento dos vídeos, aplicando-se as medidas de dissimilaridade nos descritores de faces obtidos nos vídeos do conjunto de teste, da base de dados *Honda/UCSD Video Database*.

Tabela 8.1: Comparação entre os resultados obtidos com as medidas de distância para a base de vídeos *Honda/UCSD Video Database*.

Medida de Distância	Taxa de Reconhecimento Correto dos *Frames* (%)	Taxa de Reconhecimento Correto dos Vídeos (%)	Tempo Médio de Reconhecimento por Vídeo (s)
Euclidiana	57,21	94,87	0,687
Mahalanobis	49,99	92,30	0,683

Na Tabela 8.2 são mostrados os valores da taxa de reconhecimento correto dos *frames*, taxa de reconhecimento correto dos vídeos e o tempo médio de reconhecimento dos vídeos, aplicando-se as medidas de dissimilaridade nos descritores de faces obtidos nos vídeos do conjunto de teste, da base de dados *Recogna Video Database* com o conjunto de descritores testes de diferentes tamanhos (50, 100, 200 e completo) escolhidos de acordo com a regra 7.3

Tabela 8.2: Comparação entre os resultados obtidos com as medidas de distância para a base de vídeos *Recogna Video Database*, com melhores resultados em azul.

Medida de Distância	Taxa de Reconhecimento Correto dos *Frames* (%)	Taxa de Reconhecimento Correto de Vídeos (%)	Tempo Médio de Reconhecimento por Vídeo (s)
Euclidiana (50)	46,440	58,0	0,429
Mahalanobis (50)	42,440	58,0	0,467
Euclidiana (100)	46,060	58,0	0,870
Mahalanobis (100)	42,780	58,0	0,899
Euclidiana (200)	46,590	56,0	1,758
Mahalanobis (200)	43,890	**60,0**	1,845
Euclidiana (completo)	48,006	**60,0**	49,335
Mahalanobis (completo)	45,129	56,0	51,548

No contexto de aplicações Web que exigem autenticação dos usuários, esta deve ser feita em um tempo que não torne a interação do usuário com o sistema cansativa. O uso de longos trechos de vídeos capturados pelas web câmeras seria inviável na prática, em aplicações Web. O uso de trechos mais longos de vídeos pode ser aceitável se o sistema biométrico de identificação estiver operando no modo de reconhecimento, pois neste caso, a identificação do usuário pode ser realizada simultaneamente, enquanto o usuário utiliza a aplicação Web. Tendo isso em vista, foram feitos experimentos considerando-se intervalos de 1 segundo em cada vídeo. Ou seja, cada vídeo foi dividido em intervalos de 1 segundo e cada intervalo foi considerado individualmente de forma análoga a uma sessão de reconhecimento em uma aplicação de autenticação. Os intervalos podem ter diferentes números de *frames* pelo motivo da compactação de vídeo MPEG4 omitir *frames* consecutivos semelhantes ou ainda pelo fato do algoritmo Viola-Jones não conseguir detectar uma face em um determinado *frame*. No caso da base de dados *Honda/UCSD Video Database*), tem-se 15 *frames* por segundo e no caso da base de dados *Recogna Video Database* tem-se, em média, 20 *frames* por segundo.

Na Tabela 8.3 são mostrados os valores de tempo médio de teste para cada intervalo de 1 segundo e a taxa de reconhecimento correto de intervalos de 1 segundo para a base de dados de vídeos *Honda/UCSD Video Database*.

Tabela 8.3: Comparação entre os resultados obtidos com as medidas de distância para a base de dados *Honda/UCSD Video Database* com os vídeos do conjunto de testes divididos em intervalos de 1 segundo.

Medida de Distância	Tempo Médio de Testes para cada Intervalo (s)	Taxa de Reconhecimento Correto de Intervalos (%)
Euclidiana	0,035	57,569
Mahalanobis	0,036	47,706

Na Tabela 8.4 são mostrados os valores de tempo médio de teste para cada intervalo de 1 segundo e taxa de reconhecimento correto de intervalos de 1 segundo para a base de dados de vídeos *Recogna Video Database*.

Tabela 8.4: Comparação entre os resultados obtidos com as medidas de distância para a base de vídeos *Recogna Video Database* com os vídeos do conjunto de testes divididos em intervalos de 1 segundo.

Medida de Distância	Tempo Médio de Testes para Cada Intervalo (s)	Taxa de Reconhecimento Correto de Intervalos (%)
Euclidiana	0,162	50,142
Mahalanobis	0,165	48,296

8.2 Reconhecimento de Faces Baseado em Classificadores

Na Tabela 8.5 são apresentados os resultados obtidos com os classificadores de padrões sobre a base de dados *Honda/UCSD Video Database*. A Tabela contém valores relativos ao tempo de treinamento, tempo médio necessário para a identificação do indivíduo por vídeo, acurácia (definida na equação 7.6), taxa de classificação correta dos *frames* e taxa de identificação correta dos indivíduos nos vídeos, para todos os classificadores.

Tabela 8.5: Comparação entre os resultados obtidos nos testes com diferentes classificadores na base *Honda/UCSD Video Database*, com melhores resultados em azul e piores em vermelho.

Classificador	Tempo de Treinamento (s)	Tempo Médio de Teste por Vídeo(s)	Acurácia (%)	Taxa de Classificação Correta de *Frames* (%)	Taxa de Reconhecimento Correto dos Vídeos
ANN MLP	2491,001	**0,000001**	63,536	53,654	71,79
Bayes	9,000	8,179	91,334	81,210	**100,00**
KNN	16187,000	0,308	91,334	81,210	**100,00**
OPF	9,000	0,333	91,221	80,856	**100,00**
OPF (com avaliação)	41,667	0,308	89,614	77,774	**100,00**
Kohonen	1053,000	0,128	90,646	79,534	**100,00**
LibSVM (Linear)	**96333,000**	0,103	88,488	74,339	**100,00**
LibSVM (Sem Kernel)	224,000	0,026	88,479	76,567	**100,00**
LibSVM (RBF)	5339,000	0,179	92,670	82,980	**100,00**
LibSVM (Sigmóide)	4155,000	0,179	88,317	73,860	97,44
SVM Torch (RBF)	9,000	0,179	**92,719**	**83,479**	**100,00**

Na Tabela 8.6 são apresentados os resultados do uso dos classificadores de padrões na base de dados *Honda/UCSD Video Database*, com os vídeos do conjunto de testes separados em intervalos de 1 segundo. A coluna Tempo Médio de Testes refere-se ao tempo médio necessário para a classificação de um conjunto de *frames* no intervalo de 1 segundo e a coluna Taxa de Reconhecimento Correto de Intervalos refere-se à relação entre o número de intervalos de 1 segundo nos quais o reconhecimento acertou o resultado em relação ao número total de intervalos de 1 segundo em cada vídeo nos quais foi encontrada pelo menos uma face pelo algoritmo Viola-Jones.

Tabela 8.6: Comparação entre os resultados obtidos nos testes com diferentes classificadores na base *Honda/UCSD Video Database* para vídeos do conjunto de testes divididos em intervalos de 1 segundo, com melhores resultados em azul e piores em vermelho.

Classificadores	Tempo Médio de Testes (s)	Taxa de Reconhecimento Correto de Intervalos (%)
ANN MLP	**0,000001**	**57,683**
Bayes	0,260	82,913
KNN	0,011	82,913
OPF	0,013	82,683
OPF (com avaliação)	0,007	79,893
Kohonen	0,007	80,734
LibSVM (Linear)	0,209	77,638
LibSVM (Sem Kernel)	0,032	79,817
LibSVM (RBF)	0,290	84,060
LibSVM (Sigmóide)	0,203	77,179
SVM Torch (RBF)	0,281	**84,633**

Na Tabela 8.7 são apresentados os resultados do uso dos classificadores de padrões na base de dados *Recogna Video Database*, de forma que tais classificadores foram treinados com 50 descritores do conjunto de treinamento selecionados de acordo com a regra 7.3.

Tabela 8.7: Comparação entre os resultados obtidos nos testes com diferentes classificadores na base *Recogna Video Database* com o conjunto de treinamento composto por 50 descritores, com melhores resultados em azul e piores em vermelho.

Classificador	Tempo de Treinamento (s)	Tempo Médio de Teste por Vídeo(s)	Acurácia (%)	Taxa de Classificação Correta de *Frames* (%)	Taxa de Reconhecimento Correto de Vídeos
ANN MLP	30,000	**0,267**	61,566	55,502	75,333
Bayes	1,667	**171,120**	84,642	69,789	**90,000**
KNN	942,000	3,560	84,642	69,789	**90,000**
OPF	1,667	3,233	84,347	69,496	**90,000**
OPF (com avaliação)	3,667	2,053	83,506	65,336	81,333
Kohonen	455,000	3,287	82,852	67,128	88,000
LibSVM (Linear)	1429,667	2,347	85,297	71,051	84,000
LibSVM (Sem Kernel)	28,667	0,420	83,745	69,644	**90,000**
LibSVM (RBF)	**2053,667**	2,747	85,737	72,886	84,000
LibSVM (Sigmóide)	1869,666	3,413	85,310	71,244	82,000
SVM Torch (RBF)	2,667	2,607	**86,729**	**74,379**	84,000

Na Tabela 8.8 são apresentados os resultados obtidos com os classificadores de padrões na base de dados *Recogna Video Database*, de forma que tais classificadores foram treinados com 100 descritores do conjunto de treinamento selecionados de acordo com a regra 7.3.

Tabela 8.8: Comparação entre os resultados obtidos nos testes com diferentes classificadores na base *Recogna Video Database* com o conjunto de treinamento composto por 100 descritores, com melhores resultados em azul e piores em vermelho.

Classificador	Tempo de Treinamento (s)	Tempo Médio de Teste por Vídeo(s)	Acurácia (%)	Taxa de Classificação Correta de *Frames* (%)	Taxa de Reconhecimento Correto de Vídeos
ANN MLP	66,000	**0,233**	76,154	57,774	70,667
Bayes	6,000	**341,807**	85,510	71,829	**88,000**
KNN	**7875,000**	7,060	85,510	71,829	**88,000**
OPF	7,333	6,740	85,509	71,703	**88,000**
OPF (com avaliação)	13,000	4,020	84,648	68,004	83,333
Kohonen	933,333	3,307	84,743	70,198	**88,000**
LibSVM (Linear)	1829,333	3,167	86,071	72,470	86,000
LibSVM (Sem Kernel)	58,000	0,413	82,640	70,093	84,000
LibSVM (RBF)	**4496,667**	3,787	86,651	73,606	84,000
LibSVM (Sigmóide)	3685,000	4,727	86,069	72,468	86,000
SVM Torch (RBF)	4,000	3,073	**86,922**	**75,210**	84,000

Na Tabela 8.9 são apresentados os resultados do uso dos classificadores de padrões na base *Recogna Video Database*, de forma que tais classificadores foram treinados com 200 descritores do conjunto de treinamento selecionados de acordo com a regra 7.3.

Tabela 8.9: Comparação entre os resultados obtidos nos testes com diferentes classificadores na base *Recogna Video Database* com o conjunto de treinamento composto por 200 descritores, com melhores resultados em azul e piores em vermelho.

Classificador	Tempo de Treinamento (s)	Tempo Médio de Teste por Vídeo(s)	Acurácia (%)	Taxa de Classificação Correta de *Frames* (%)	Taxa de Reconhecimento Correto de Vídeos
ANN MLP	164,666	**0,293**	78,671	61,088	78,000
Bayes	24,000	**684,933**	86,204	73,168	88,000
KNN	**151825,000**	14,140	86,204	73,168	88,000
OPF	28,667	14,253	86,197	73,134	88,000
OPF (com avaliação)	55,000	8,167	85,589	70,342	84,000
Kohonen	1888,000	3,300	85,064	70,993	**90,000**
LibSVM (Linear)	3155,333	4,787	86,221	72,654	86,000
LibSVM (Sem Kernel)	172,000	0,393	83,582	71,332	88,000
LibSVM (RBF)	13883,666	6,080	86,923	73,884	82,000
LibSVM (Sigmóide)	10436,000	6,673	86,350	72,728	86,000
SVM Torch (RBF)	10,000	3,927	**87,838**	**76,557**	86,000

Na Tabela 8.10 são apresentados os resultados do uso dos classificadores de padrões na base *Recogna Video Database*, de forma que tais classificadores foram treinados utilizando-se todos os descritores obtidos nos vídeos do conjunto de treinamento.

Tabela 8.10: Comparação entre os resultados obtidos nos testes com diferentes classificadores na base *Recogna Video Database* com o conjunto de treinamento completo, com melhores resultados em azul e piores em vermelho.

Classificador	Tempo de Treinamento (s)	Tempo Médio de Teste por Vídeo(s)	Taxa de Classificação Correta de *Frames* (%)	Taxa de Reconhecimento Correto de Vídeos
ANN MLP	9139,667	**0,293**	67,778	80,667
Bayes	14454,999	**15929,280**	77,180	**90,000**
OPF	19444,667	597,740	77,167	**90,000**
Kohonen	41260,000	2,693	71,544	88,000
LibSVM (Linear)	**295619,000**	14,820	73,071	84,000
LibSVM (Sem Kernel)	13773,667	0,620	73,366	86,000
SVM Torch (RBF)	960,000	40,393	**79,087**	86,000

Na Tabela 8.11 são apresentados os resultados do uso dos classificadores de padrões na base *Recogna Video Database*. Para estes resultados. foram utilizados os classificadores já treinados com os diferentes conjuntos de treinamento, porém o conjunto de testes foi dividido em intervalos de 1 segundo.

Tabela 8.11: Comparação entre os resultados obtidos nos testes com diferentes classificadores na base *Recogna Video Database* com o conjunto de testes dividido em intervalos de 1 segundo, com melhores resultados em azul e piores em vermelho.

Classificadores	50 descritores		100 descritores		200 descritores	
	Tempo médio de testes (s)	Taxa de Reconhecimento Correto de Intervalos (%)	Tempo Médio de Testes (s)	Taxa de Reconhecimento Correto de Intervalos (%)	Tempo Médio de Testes (s)	Taxa de Reconhecimento Correto de Intervalos (%)
ANN MLP	**0,001**	56,733	**0,001**	60,472	**0,001**	64,433
Bayes	**0,579**	73,682	**1,156**	75,142	**2,312**	76,068
KNN	0,013	73,682	0,023	75,142	0,048	76,068
OPF	0,011	73,472	0,023	74,993	0,051	76,055
OPF (com avaliação)	0,011	67,976	0,024	71,120	0,045	73,114
Kohonen	0,011	70,836	0,009	73,702	0,012	74,439
LibSVM (Linear)	0,313	72,938	0,230	74,290	0,294	74,594
LibSVM (Sem Kernel)	0,117	73,465	0,039	72,742	0,038	74,094
LibSVM (RBF)	0,253	74,669	0,251	75,020	0,335	75,602
LibSVM (Sigmóide)	0,281	72,952	0,249	74,304	0,299	74,831
SVM Torch (RBF)	0,327	**76,190**	0,343	**76,859**	0,338	**78,061**

Para os experimentos com intervalos de 1 segundo na base *Honda/UCSD Video Database*, a distância Euclidiana apresentou uma taxa de reconhecimento correto de intervalos de 1 segundo de 57,569% e o classificador SVM Torch (melhor entre os classificadores) apresentou uma taxa 84,633%, todos com tempo bem abaixo de 1 segundo (o que seria aceitável em uma aplicação

real). Para os experimentos com intervalos de 1 segundo na base *Recogna Video Database*, a distância Euclidiana apresentou, uma taxa de reconhecimento correto de intervalos de 1 segundo de 50,142% e o classificador SVM Torch (melhor entre os classificadores) apresentou uma taxa 78,061%, todos com tempo bem abaixo de 1 segundo.

Tendo em vista que a implementação SVM Torch do classificador SVM, com Kernel RBF, apresentou o melhor compromisso entre acurácia e tempo de treinamento e teste, este classificador foi utilizado em experimentos adicionais realizados para avaliar outras características do reconhecimento de faces.

No gráfico da Figura 8.1 são comparadas as taxas de reconhecimento para todos os vídeos da base de dados *Honda Video Database* entre a distância Euclidiana e o classificador SVM Torch treinado com 200 amostras. Ambos métodos foram escolhidos para a comparação por apresentarem os resultados mais expressivos em suas respectivas categorias.

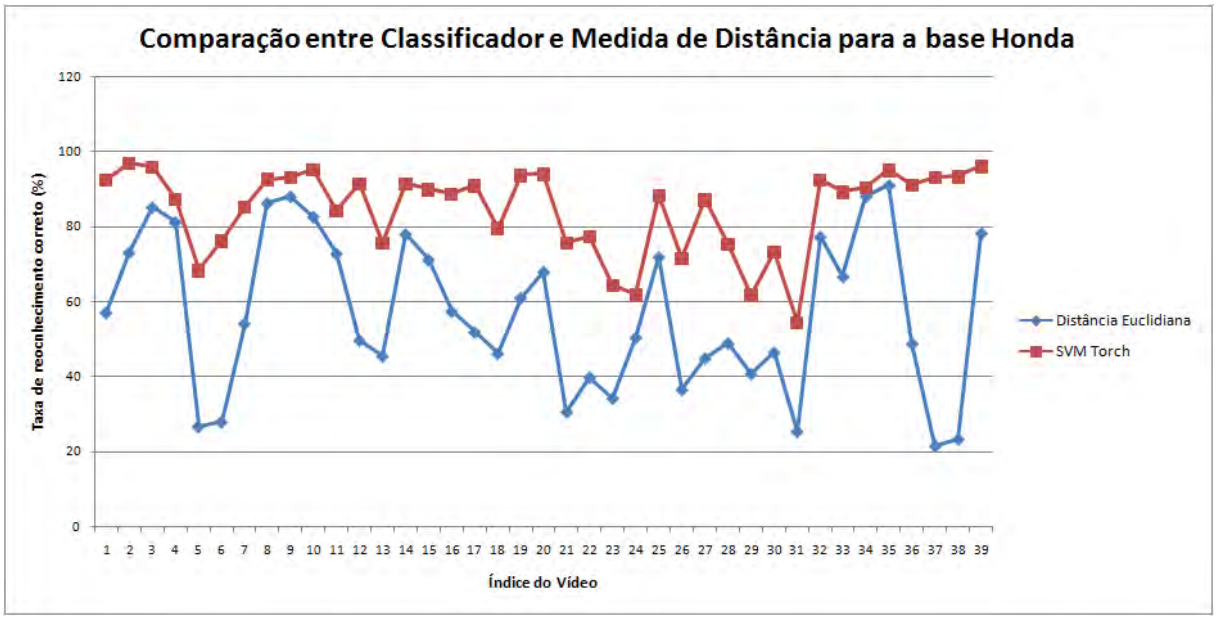

Figura 8.1: Comparação entre resultados do classificador SVM Torch com a distância Euclidiana para a base de dados de vídeos *Honda Video Database*.

No gráfico da Figura 8.2 são comparadas as taxas de reconhecimento para todos os vídeos da base de dados *Recogna Video Database* entre a distância Euclidiana e o classificador SVM Torch.

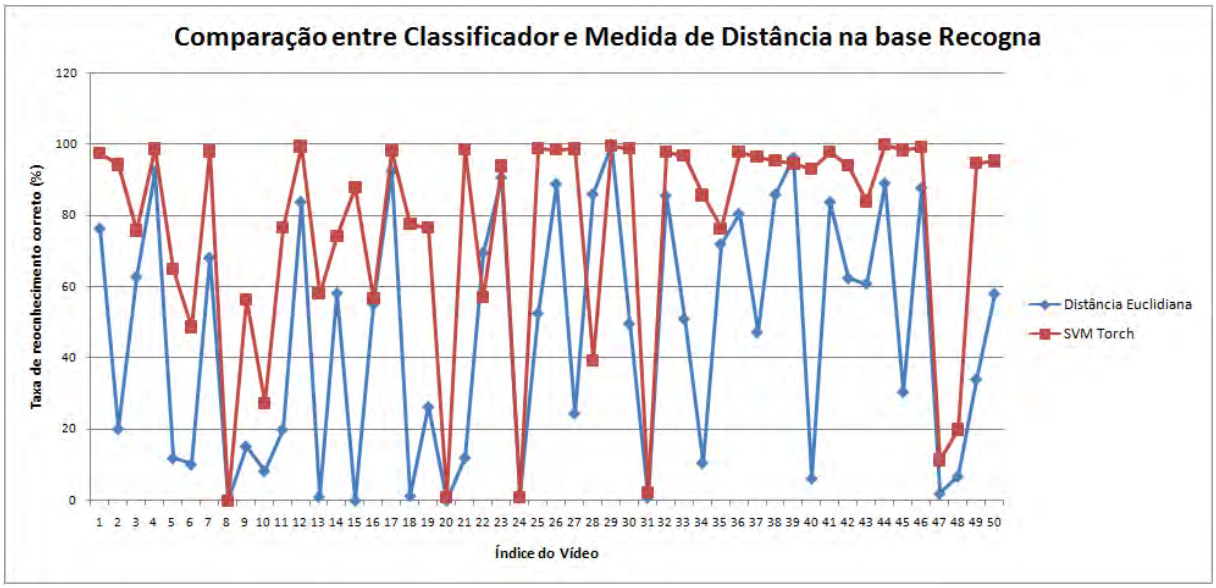

Figura 8.2: HMM na base de dados de vídeos *Recogna Video Database*.

No gráfico da Figura 8.3 são mostradas as taxas de classificação correta em cada um dos vídeos da base de dados de vídeos *Recogna Video Database* para os 4 classificadores de padrões (o pior e 3 entre os melhores em relação à taxa de classificação correta de *frames* no vídeo) treinados com o conjunto de 200 amostras, como pode ser visto na tabela 8.9.

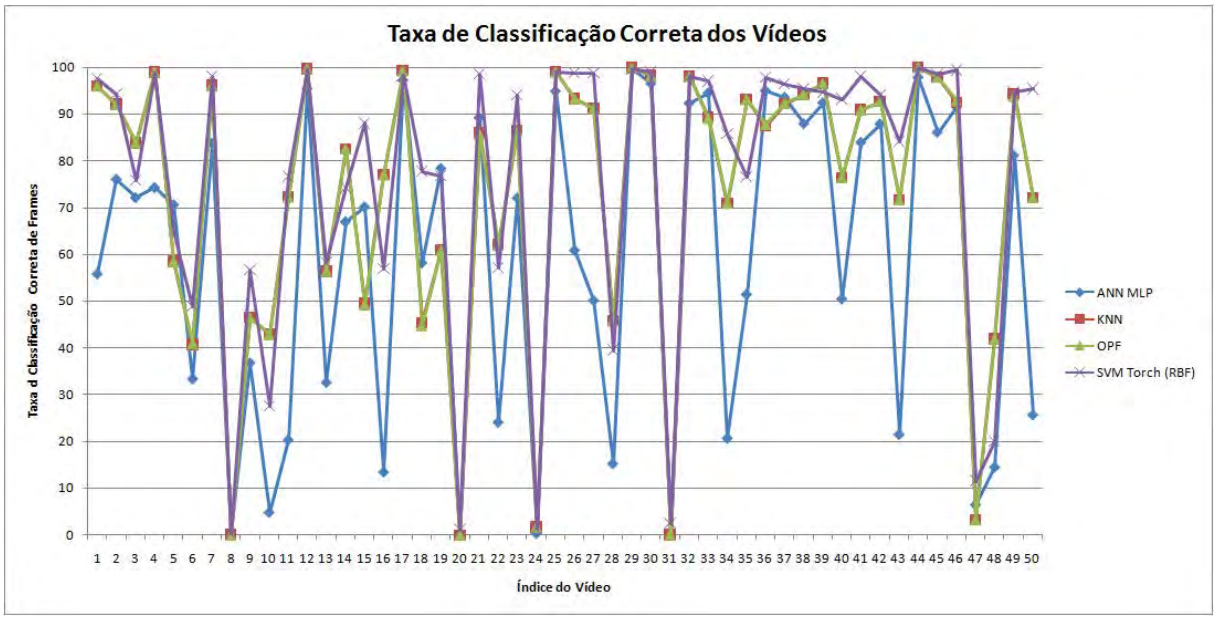

Figura 8.3: Comparação entre resultados dos classificadores SVM Torch, OPF, KNN e ANN-MLP para cada um dos vídeos da base de dados de vídeos *Recogna Video Database*.

A partir do gráfico da figura 8.3 é possível notar que, mesmo com diferentes taxas de reconhecimento correto de *frames*, os classificadores apresentam suas curvas em formato semelhante e os vídeos com uma baixa taxa de reconhecimento de *frames* apresentam o resultado bem próximo para todos os classificadores.

Os experimentos realizados mostraram que o uso de todos os *frames* dos vídeos durante o treinamento dos classificadores e o reconhecimento dos indivíduos nos vídeos tornam esses processos extremamente lentos, inviabilizando o uso de classificadores em sistemas *online* de autenticação. Por exemplo, o classificador Bayesiano treinado com todos os descritores obtidos nos *frames* do conjunto de treinamnto da base de dados de vídeos *Recogna Video Database* levou em média quase 4,5 horas para classificar todos os *frames* de apenas um vídeo de teste, em um computador com a seguinte configuração: sistema operacional Windows XP SP3, processador Intel Xeon E5540 (8M de cache, 2.53 GHz de clock), memória RAM de 3GB (máximo reconhecido pelo sistema operacional Windows XP SP3), 3 HDs de 1 TB cada. Diante desse fato, foram realizados experimentos adicionais com conjuntos reduzidos de 50, 100 e 200 *frames* para treinamento, selecionados de acordo com a regra 7.3.

No gráfico da Figura 8.4 são mostradas as taxas de classificação do classificador SVM Torch para os conjuntos de 50, 100 e 200 amostras para testes.

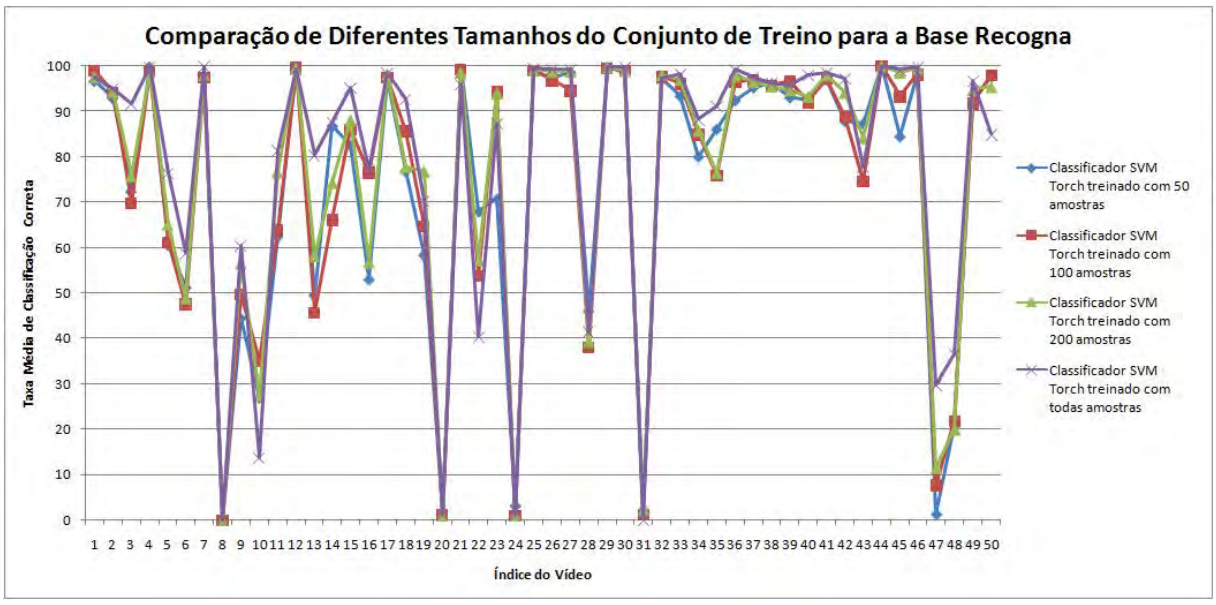

Figura 8.4: Comparação do classificador SVM Torch treinado com diferentes tamanhos de conjunto de amostras.

Observando-se o gráfico comparativo entre o uso de diferentes taxas de amostragem para

o classificador SVM Torch, apresentado na Figura 8.4, é possível concluir que a redução das taxas de amostragem dos *frames* dos vídeos, embora tenha influenciado nos resultados, não diminuiu significativamente as taxas de acerto. Portanto, para a utilização dos classificadores de padrões em sistemas reais, para os quais deseja-se que a identificação dos usuários seja *online*, é recomendável o uso de taxas reduzidas de amostras de *frames* dos vídeos, tanto na fase de treinamento, quanto na fase de reconhecimento.

8.3 Reconhecimento de Faces Baseado em Modelos de Markov

Na Tabela 8.12 são apresentados os valores obtidos nos testes com a aplicação do HMM na base de dados de vídeos *Honda/UCSD Video Database*, onde no rótulo horizontal (E) estão os diferentes números de estados utilizados e no rótulo vertical (M) estão os diferentes números de componentes na mistura gaussiana de cada estado. Para estes experimentos, foram medidos o tempo de treinamento, teste e taxa de reconhecimento correto variando-se o número de estados e número de componentes da mistura gaussiana de cada estado.

Tabela 8.12: Comparação entre as taxas de reconhecimento correto de vídeos (com diferentes valores de estados e de componentes nas misturas gaussianas destes) por meio do HMM na base de dados *Honda/UCSD Video Database*, com melhores resultados em azul.

Tempo médio de Treinamento por Vídeo (s)									
E / M	1	2	3	4	5	6	7	8	16
1	0,054	0,051	0,059	0,051	0,055	0,041	0,074	0,058	0,052
2	0,041	0,071	0,049	0,072	0,062	0,055	0,049	0,088	0,057
3	0,039	0,058	0,069	0,076	0,055	0,064	0,058	0,078	0,055
4	0,074	0,063	0,060	0,052	0,066	0,064	0,068	0,080	0,052
Tempo médio de Teste por Vídeo (s)									
E / M	1	2	3	4	5	6	7	8	16
1	1,620	1,715	1,496	1,444	1,658	1,482	1,774	1,659	2,012
2	1,478	1,532	1,550	1,644	1,985	1,649	1,699	1,710	2,461
3	1,604	1,532	1,655	1,632	1,909	1,650	2,136	1,848	1,634
4	1,570	1,816	1,569	1,725	1,670	2,016	1,933	2,350	2,549
Taxa de Reconhecimento Correto de Vídeos									
E / M	1	2	3	4	5	6	7	8	16
1	97,436	97,436	79,487	43,590	2,564	28,205	17,949	35,897	0,000
2	97,436	97,436	**100,000**	61,538	20,513	10,256	20,513	0,000	15,385
3	97,436	**100,000**	92,308	61,538	28,205	20,513	25,641	17,949	10,256
4	94,872	97,436	87,179	69,231	38,462	30,769	10,256	10,256	10,256

No gráfico da Figura 8.5 são comparadas as Taxas de Reconhecimento Correto de Vídeos para diferentes números de estados e de componentes na mistura gaussiana de cada estado para a base de dados de vídeos *Honda/UCSD Video Database*.

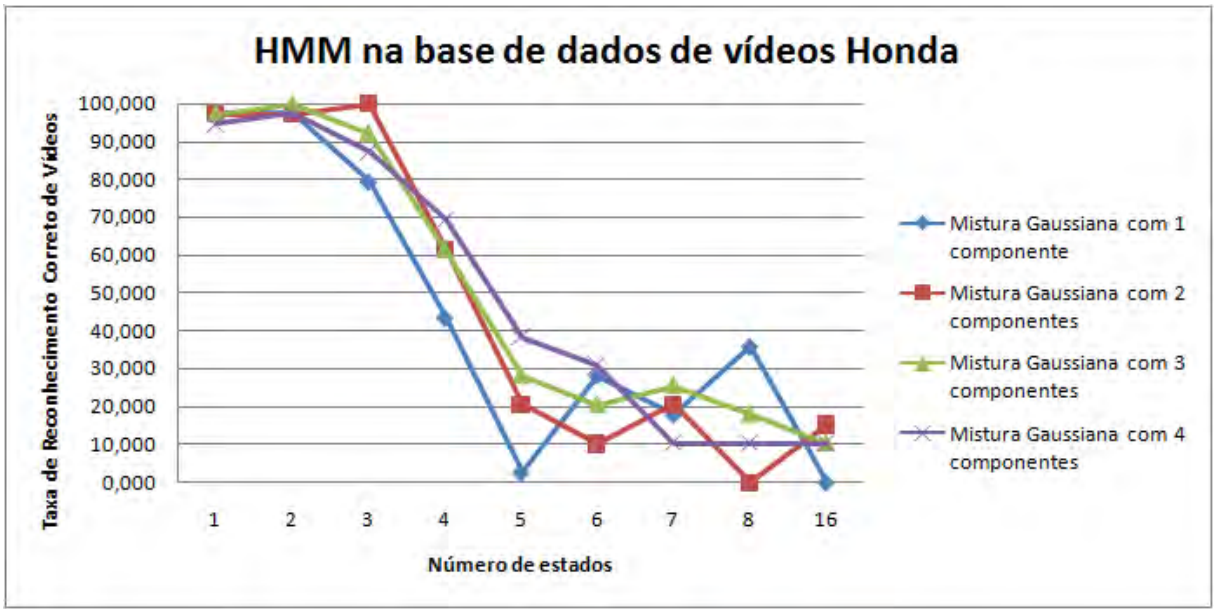

Figura 8.5: HMM na base de dados de vídeos *Honda/UCSD Video Database*.

Na Tabela 8.13 são mostrados os valores obtidos nos testes com a aplicação do HMM na base de dados de vídeos *Recogna Video Database*, onde no rótulo horizontal (E) estão os diferentes números de estados utilizados e no rótulo vertical (M) estão os diferentes números de componentes na mistura gaussiana de cada estado. Para estes experimentos, foram medidos o tempo de treinamento, teste e taxa de reconhecimento correto variando-se o número de estados e número de componentes da mistura gaussiana de cada estado.

Tabela 8.13: Comparação entre as taxas de reconhecimento correto de vídeos (com diferentes valores de estados e de componentes nas misturas gaussianas destes) por meio do HMM na base de dados *Recogna Video Database*, com melhores resultados em azul.

Tempo médio de Treinamento por Vídeo (s)									
E / M	1	2	3	4	5	6	7	8	16
1	0,309	0,792	1,241	1,479	1,720	1,736	2,233	2,305	4,247
2	1,436	2,036	2,456	2,653	3,151	3,353	3,565	4,183	5,850
3	1,170	2,416	2,910	3,441	4,081	4,595	5,010	5,218	10,963
4	1,931	3,182	3,696	4,590	5,082	5,229	5,459	6,106	8,571
Tempo médio de Teste por Vídeo (s)									
E / M	1	2	3	4	5	6	7	8	16
1	11,005	7,531	7,544	7,838	9,569	10,086	10,338	12,035	28,490
2	8,814	7,906	7,961	9,156	10,251	11,462	12,166	14,168	29,228
3	7,095	9,030	8,552	9,848	11,226	12,234	14,091	15,957	40,531
4	7,379	8,498	9,318	10,895	13,047	13,619	16,115	19,046	40,858
Taxa de Reconhecimento Correto de Vídeos									
E / M	1	2	3	4	5	6	7	8	16
1	64,000	**68,000**	62,000	56,000	36,000	44,000	32,000	18,000	2,000
2	**68,000**	64,000	66,000	62,000	48,000	52,000	40,000	36,000	2,000
3	64,000	66,000	60,000	62,000	54,000	46,000	40,000	28,000	8,000
4	60,000	64,000	62,000	56,000	50,000	46,000	34,000	36,000	2,000

No gráfico da Figura 8.6 são comparados as Taxas de Reconhecimento Correto de Vídeos para diferentes números de estados e de componentes na mistura gaussiana de cada estado para a base de dados de vídeos *Recogna Video Database*.

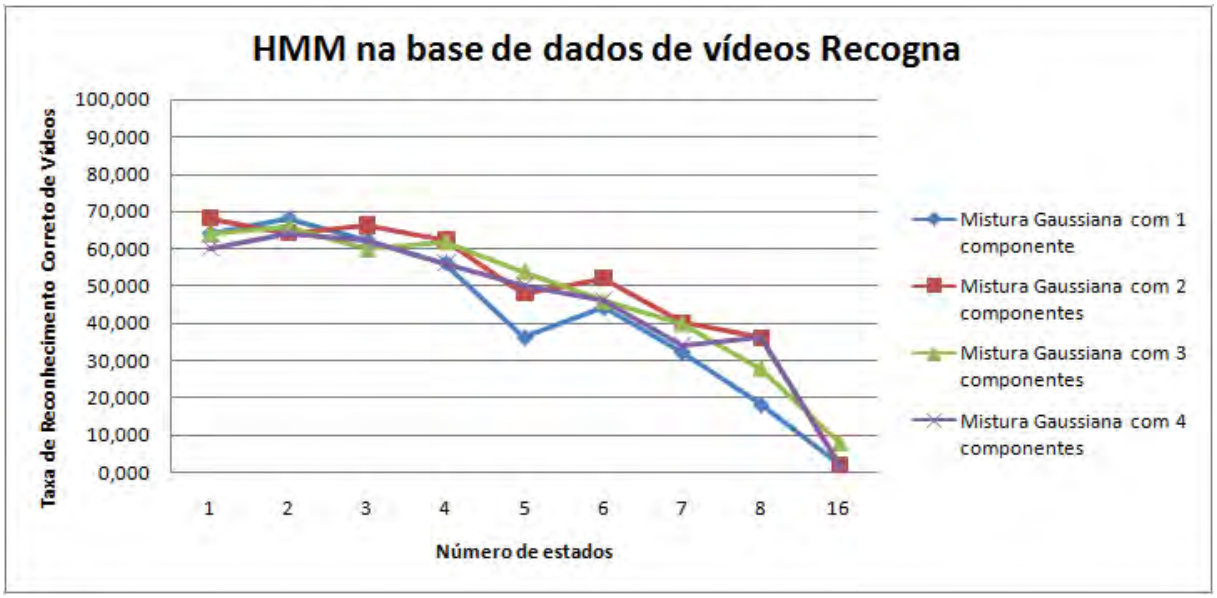

Figura 8.6: HMM na base de dados de vídeos *Recogna Video Database*.

Com relação ao reconhecimento de faces utilizando-se o método HMM, é possível notar nas Figuras 8.5 e 8.6 que os resultados obtidos com o HMM pioram à medida em que o número de estados do modelo aumenta, tanto para as base de dados *Honda/UCSD Video Database* quanto para a base de dados *Recogna Video Database*. Ainda que a taxa de identificação correta dos indivíduos nos vídeos com o uso do HMM (para 1, 2 e 3 estados) tenha sido melhor que a taxa de identifacação utilizando-se o vizinho mais próximo com as medidas de distância Euclidiana e Mahalanobis, a utilização de informação temporal não se mostrou tão importante para as bases de dados utilizadas no experimentos. Tal conclusão é possível uma vez que, ao aumentar o número de estados, a taxa de identificação correta dos indivíduos se deteriora. Quando se usa um modelo com apenas um estado, obviamente não existem transições entre estados e o algoritmo *Forward* retorna apenas a multiplicação entre as probabilidades de cada observação ser gerada pela distribuição Gaussiana do único estado, ignorando totalmente a informação dinâmica do vídeo. Durante este trabalho foram realizados estudos e experimentos sobre o uso da técnica de *cross-validation* com algoritmos de agrupamento para se obter um número de conjuntos para cada modelo, sendo que tal informação seria utilizada para se determinar um número diferente de estados para cada modelo individual. No entanto, notou-se que quanto menor o número de estados, maior é a probabilidade obtida pelo algoritmo *Forward* de um modelo gerar um dado conjunto de observações, ou seja, um modelo com menos estados sempre teria vantagem no momento do reconhecimento. Observado tal fato, optou-se por manter o número de estados constante para os modelos de todos os indivíduos.

8.4 Variação do Tamanho do Descritor das Faces

Como foi dito anteriormente, o uso dos 50 primeiros autovetores da matriz de espalhamento na criação do espaço de faces na base de dados de vídeos *Recogna Video Database* apresentou um representatividade de 71,1%. Para se ter uma representatividade de 85% (semelhante à obtida na base de dados *Honda/UCSD Video Database*), foi necessário o uso de 120 autovetores. Visando a comparação entre os dois tamanhos de descritores, foram feitos experimentos utilizando descritores com os dois tamanhos diferentes, de forma que os classificadores foram treinados com 200 descritores de cada vídeo de treinamento (escolhidos como explicado na seção 7.3), e foram testados em todos os vídeos do conjunto de testes. Na Tabela 8.14 é apresentada uma comparação entre os dois tamanhos diferentes do vetor descritor (50 e 120), por meio de dois classificadores, em relação às seguintes medidas: Tempo de Treinamento, Tempo médio de Teste por Vídeo, Acurácia (como explicado na seção 7.3), Taxa de Classificação Correta de *Frames* e Taxa de Reconhecimento Correto de Vídeos para todos os classificadores.

Tabela 8.14: Comparação entre resultados para base de dados de vídeos *Recogna Video Database* entre dois tamanhos para os vetores de características.

Classificadores		Tempo de Treinamento (s)	Tempo Médio de Teste por Vídeo (s)	Acurácia	Taxa de Classificação Correta de *Frames* (%)	Taxa de Reconhecimento Correto de Vídeos (%)
Descritor com 50 posições	OPF	28,667	14,253	86,197	73,134	88,000
	SVM Torch (RBF)	10,000	3,927	87,838	76,557	86,000
Descritor com 120 posições	OPF	55,001	31,720	87,972	76,968	90,000
	SVM Torch (RBF)	34,666	12,120	89,809	81,071	88,000

Na comparação entre os resultados para os diferentes tamanhos de descritores presentes na Tabela 8.14, é possível notar que o tempo de treinamento e testes aumenta significativamente nesses casos (dobrando ou até triplicando), porém a acurácia, a taxa de identificação correta dos indivíduosnos *frames* e a taxa de identificação correta dos indivíduos nos vídeos não apresenta melhoras tão significativas que justifiquem o aumento dos tempos de treinamentos e testes.

Esse é um resultado favorável, pois, segundo a arquitetura proposta (Figura 5.1) os vetores de características têm que ser enviados do cliente para o servidor e quanto menores forem tais vetores, menos impacto negativo haverá no tempo necessário para a autenticação dos usuários das aplicações Web.

8.5 Seleção Randômica dos Conjuntos de Treinamento e Teste

Diante do fato de que alguns indivíduos tiveram a taxa de reconhecimento próxima de 0% enquanto que outros tiveram a taxa próxima de 100%, foram feitos experimentos com duas formas diferentes de se dividir as amostras entre treinamento e teste. Na primeira forma, usou-se a forma convencional utilizada em todos os outros experimentos: os descritores da primeira sessão foram utilizados no treinamento do classificador, enquanto que os da segunda sessão foram utilizados nos testes. Na segunda forma, todos descritores são unidos em um único conjunto e depois são divididos em conjunto de teste e treinamento de forma randômica. Por meio desse experimento, procurou-se determinar se o fator que influenciou nesses resultados inesperados foi alguma deficiência dos classificadores ou alguma irregularidade no descritor utilizado, de forma que se o resultado se mantivesse nos dois casos, provavelmente o problema seria do classificador e caso contrário provavelmente o que prejudicou o resultado foi a grande diferença intra-classe do descritor. Para esses experimentos, foram selecionados conjuntos de 50, 100 e 200 descritores de face como na seção 7.3 de cada vídeo da primeira sessão vídeos e mais 50, 100 e 200 descritores de face de cada vídeo da segunda sessão de vídeos. Na Tabela 8.15 estão comparados os valores de Acurácia para diferentes números de amostras e classificadores para as duas formas de se obter os conjuntos de treinamento e teste.

Tabela 8.15: Comparação entre resultados para base de dados de vídeos Recogna entre duas formas de dividir o conjunto de treinamento e teste.

Classificadores		Acurácia para 50 Amostras	Acurácia para 100 Amostras	Acurácia para 200 Amostras
Conjunto de Teste e Treinamento Fixos	OPF	84,388	85,429	85,765
	SVM Torch	86,388	87,112	87,821
Conjunto de Teste e Treinamento Sortidos	OPF	97,755	98,245	98,949
	SVM Torch	98,878	99,235	99,510

Observando a Tabela 8.15, é possível notar que alterando a forma de se obter os conjuntos de faces para treinamento e testes para a forma randômica fez com que os resultados melhoras-

sem de forma bastante significativa, chegando-se a uma acurácia de 98% em média. Tal fato indica que o erro nas classificações parece estar relacionado com as diferenças entre os descritores de faces de uma mesma pessoa obtidas em sessões distintas, decorrentes de variações na iluminação, uso de acessórios e maquiagem, etc.

Capítulo 9

Discussão e Conclusões

Neste capítulo são discutidos os resultados e apresentadas as conclusões e as contribuições deste trabalho. Também são feitas considerações acerca do uso das técnicas de reconhecimento de padrões avaliadas no sistema proposto e desenvolvido por Penteado e Marana (2009).

Os resultados apresentados no capítulo 8 mostram que o uso dos classificadores de padrões e do HMM no reconhecimento de faces a partir de vídeos apresenta resultados significativamente melhores, tanto para a base de dados *Honda/UCSD Video Database* quanto para a base de dados *Recogna Video Database*. Considerando os melhores resultados de cada técnica de reconhecimento de vídeos para a base de dados *Honda/UCSD Video Database*, o vizinho mais próximo usando a distância Euclidiana apresentou uma taxa de 94,87% de reconhecimento correto dos indivíduos nos vídeos, enquanto que o HMM (com 2 estados) e a maioria dos classificadores utilizados apresentaram taxas de 100% de reconhecimento correto. Da mesma forma, para a base de dados *Recogna Video Database*, o vizinho mais próximo usando a distância Euclidiana apresentou uma taxa de 60% de reconhecimento correto dos indivíduos nos vídeos, enquanto que o HMM (com 2 estados) apresentou uma taxa de 68% e a melhor taxa entre os classificadores foi de 90% de reconhecimento correto.

Analisando-se os vídeos, foi possível observar uma oclusão bastante recorrente: a mão cobrindo parte da face. Essa oclusão interfere negativamente no desempenho das técnicas de reconhecimento facial. No caso do reconhecimento facial a partir de imagens estáticas, a chance do reconhecimento falhar é bastante grande. Nos experimentos com os vídeos da base de dados *Recogna Video Database*, foi possível notar que a pessoa que mais vezes apresentou esse tipo de oclusão ao longo do vídeo teve apenas 30% dos *frames* obtidos pelo algoritmo Viola-Jones com oclusão. Porém, esse vídeo apresentou uma taxa de acerto de *frames* de 87,48% com o classificador SVM Torch treinado com a base toda e 74,31% para o mesmo classificador

treinado com o subconjunto de treinamento com 200 descritores. Portanto, é possível concluir que ao longo de um vídeo, mesmo que ocorra oclusões, o resultado do reconhecimento ainda pode ser positivo. Na Figura 9.1, são apresentados alguns exemplos de faces com oclusão extraídas de vídeos da base de dados *Recogna Video Database*, por meio do algoritmo Viola-Jones.

Figura 9.1: Imagens de faces com oclusão extraídas dos vídeos da base de dados *Recogna Video Database* por meio do algoritmo Viola-Jones.

9.1 Contribuições

Ao final deste trabalho, as seguintes contribuições podem ser enumeradas:

1. Aplicação pioneira do classificador OPF para o reconhecimento de faces a partir de vídeos;

2. Indicação do classificador SVM, com o Kernel RBF, na implementação da biblioteca SVM Torch, como uma excelente técnica de reconhecimento de padrões para a aplicação em questão, tanto no aspecto tempo de processamento, quanto no aspecto acurácia;

3. Aperfeiçoamento do sistema biométrico de autenticação de usuários em ambientes de *E-Learning* proposto e desenvolvido por Penteado e Marana (2009), com a incorporação já concluída do classificadores OPF e SVM;

4. Criação de uma base de dados de vídeos, *Recogna Video Database*, composta por vídeos de duração maior do que os encontrados em bases de dados de vídeos disponibilizadas para acesso público na Internet e na literatura;

9.2 Trabalhos Futuros

Dentre as sugestões de trabalhos futuros, pode-se destacar:

- Utilizar técnicas para se determinar qual a expressão facial de uma pessoa em um dado *frame*, e definidas as possíveis expressões faciais de forma qualitativa, usar um Modelo de Markov para modelar a dinâmica que um indivíduo apresenta ao alterar sua expressão facial ao longo do tempo. Poderia ser feita uma fusão do resultado obtido pelo reconhecimento do descritor da face por um classificador de padrões com essa pontuação obtida no Modelo de Markov individual para se chegar à uma pontuação final.

- Aprimorar a técnica de extração da face para se ter um quadro com a face mais isolada e a menor porção de fundo possível. Técnicas que possibilitem encontrar os pontos principais da face, como olhos, nariz e boca poderiam ser utilizadas nesse contexto;

- Incorporar ao sistema projetado e desenvolvido por Penteado e Marana, um módulo no qual o sistema opere em modo de autenticação, tornando possível o uso de vídeos mais longos para a realização do reconhecimento facial, sem que seja necessário um tempo inviável de espera por parte do usuário;

- Verificar o uso do sistema projetado e desenvolvido por Penteado e Marana, utilizando os classificadores de padrões OPF e SVM, em ambientes reais para análise de aceitação de indivíduos, precisão do mesmo em diferentes ambientes, tempo de treinamento e testes em conjuntos maiores de indivíduos que os conjuntos das bases de dados de vídeos utilizadas, etc;

- Utilizar outros descritores de faces para comparar os resultados com o PCA, visto que para alguns casos específicos o descritor obtido pelo PCA não foi discriminativo o suficiente, a ponto de se ter taxas baixíssimas de reconhecimento facial.

Referências Bibliográficas

BELHUMEUR, P. N.; HESPANHA, J. P.; KRIEGMAN, D. J. *Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection*. 1997.

BOLLE, R.; PANKANTI, S. *Biometrics, Personal Identification in Networked Society: Personal Identification in Networked Society*. Norwell, MA, USA: Kluwer Academic Publishers, 1998. ISBN 0792383451.

CANTONI, V.; CELLARIO, M.; PORTA, M. Perspectives and challenges in e-learning: towards natural interaction paradigms. *Journal of Visual Languages and Computing*, v. 15, p. 333–345, 2004.

CHANG, C. C.; LIN, C. J. *LIBSVM: a library for support vector machines*. [S.l.], 2001. Software available at `http://bengio.abracadoudou.com/SVMTorch.html`.

CHELLAPPA, R.; WILSON, C. L.; SIROHEY, S. Human and machine recognition: A survey. v. 83, n. 5, p. 705–741, 1995.

CHEN, L.-F.; LIAO, H.-Y. M.; LIN, J.-C. Person identification using facial motion. In: *ICIP (2)*. [S.l.: s.n.], 2001. p. 677–680.

COLLOBERT, R.; BENGIO, S. *SVMTorch: Support Vector Machines for Large-Scale Regression Problems*. [S.l.], 2001. v. 1, 143-160 p. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

COOTES, T. *Active Appearance Models*. 1998. Acessado em agosto de 2010. Disponível em: <http://bagpuss.smb.man.ac.uk/~bim/Models/aam.html>.

CORMEN, T. H. et al. *Introduction to Algorithms*. 2nd revised edition. ed. [S.l.]: B&T, 2001. Taschenbuch. ISBN 0262531968.

COVER, T.; HART, P. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, v. 13, n. 1, p. 21–27, jan 1967. ISSN 0018-9448.

DOUCET, A.; FREITAS, N. D.; GORDON, N. (Ed.). *Sequential Monte Carlo methods in practice*. [S.l.: s.n.], 2001.

DUDA, R. O.; HART, P. E.; STORK, D. G. *Pattern Classification (2nd Edition)*. 2. ed. [S.l.]: Wiley-Interscience, 2000. ISBN 0471056693.

EDWARDS, G. J.; TAYLOR, C. J.; COOTES, T. F. Improving identification performance by integrating evidence from sequences. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, IEEE Computer Society, Los Alamitos, CA, USA, v. 1, p. 1486, 1999. ISSN 1063-6919.

FALCÃO, A. X.; STOLFI, J.; LOTUFO, R. A. The image foresting transform theory, algorithms, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 26, p. 19–29, 2004.

FRIEDMAN, N.; GEIGER, D.; GOLDSZMIDT, M. Bayesian network classifiers. In: *Machine Learning*. [S.l.: s.n.], 1997. p. 131–163.

GAUVAIN, J. luc; LEE, C. hui. Maximum a posteriori estimation for multivariate gaussian mixture observations of markov chains. *IEEE Transactions on Speech and Audio Processing*, v. 2, p. 291–298, 1994.

HAYKIN, S. *Neural Networks: A Comprehensive Foundation (2nd Edition)*. 2. ed. [S.l.]: Prentice Hall, 1998. Hardcover. ISBN 0132733501.

HIETMEYER, R. Biometric identification promises fast and secure processing of airline passengers. *The International Aviation Organization Journal*, v. 55, n. 9, p. 10–11, 2000.

HOUAISS, A. *Dicionário eletrônico da língua portuguesa 3.0*. [S.l.]: Ed. Objetiva Ltda., Junho 2009. Instituto Antônio Houaiss.

HUANG, K.; TRIVEDI, M. Streaming face recognition using multicamera video arrays. In: *ICPR '02: Proceedings of the 16 th International Conference on Pattern Recognition (ICPR'02) Volume 4*. [S.l.: s.n.], 2002. p. 213–216.

HUNT, E. *The mathematics of behavior*. [S.l.]: Cambridge University Press, 2007. ISBN 9780521615228.

JAIN, A. K.; LI, S. Z. *Handbook of Face Recognition*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2005. ISBN 038740595X.

JAIN, A. K.; MALTONI, D. *Handbook of Fingerprint Recognition*. Secaucus, NJ, USA: Springer-Verlag, 2003. ISBN 0387954317.

JAIN, A. K.; ROSS, A.; PRABHAKAR, S. An introduction to biometric recognition. *IEEE Trans. on Circuits and Systems for Video Technology*, v. 14, p. 4–20, 2004.

KNIGHT, B.; JOHNSTON, A. The role of movement in face recognition. *Visual Cognition*, Psychology Press, part of the Taylor & Francis Group, v. 4, n. 3, p. 265–273, Setembro 1997. ISSN 1350-6285.

LEE, K.-C. et al. Visual tracking and recognition using probabilistic appearance manifolds. *Comput. Vis. Image Underst.*, Elsevier Science Inc., New York, NY, USA, v. 99, n. 3, p. 303–331, 2005. ISSN 1077-3142.

LEE, K.-C. et al. Video-based face recognition using probabilistic appearance manifolds. In: *In Proc. IEEE Conference on Computer Vision and Pattern Recognition*. [S.l.: s.n.], 2003. p. 313–320.

LEE, K.-C.; KRIEGMAN, D. Online learning of probabilistic appearance manifolds for video-based recognition and tracking. In: *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01*. Washington, DC, USA: IEEE Computer Society, 2005. (CVPR '05), p. 852–859. ISBN 0-7695-2372-2. Disponível em: <http://dx.doi.org/10.1109/CVPR.2005.260>.

LIU, X.; CHEN, T. Video-based face recognition using adaptive hidden markov models. In: . [S.l.: s.n.], 2003. p. 340–345.

MURTAGH, F. Multilayer perceptrons for classification and regression. *Neurocomputing*, v. 2, n. 5, p. 183–197, 1990.

NISSEN, S. *Implementation of a Fast Artificial Neural Network Library (fann)*. [S.l.], 2003. Software version 2.0 available at `http://leenissen.dk/fann/wp/`.

NSTC. *Biometrics History*. 2006. 1–2 p. National Science and Technology Council - Comitee on Homeland and National Security - Subcomitee on Biometrics. Disponível em: <http://www.biometrics.gov/Documents/BioHistory.pdf>.

OLIVEIRA, L. E. S.; MORITA, M. E. *Introdução aos Modelos Escondidos de Markov (HMM)*. 1999. Pontifícia Universidade Católica do Paraná - PUC-Pr.

O'TOOLE, A. J.; ROARK, D. A.; ABDI, H. Recognizing moving faces: A psychological and neural synthesis. In: *Trends in Cognitive Sciences*. [S.l.: s.n.], 2002. p. 261–266.

PAPA, J. P.; SUZUKI, C. T. N.; FALCÃO, A. X. *LibOPF: A library for the design of optimum-path forest classifiers*. [S.l.], 2009. Software version 2.0 available at `http://www.ic.unicamp.br/~afalcao/LibOPF`.

PAPA, J. P.; SUZUKI, C. T. N.; FALCÃO, A. X. Supervised pattern classification based on optimum-path forest. *Int. J. Imaging Syst. Technol.*, John Wiley & Sons, Inc., New York, NY, USA, v. 19, n. 2, p. 120–131, 2009. ISSN 0899-9457.

PENTEADO, B. E. *Autenticação Biométrica de Usuários em Sistemas de E-Learning baseada em Reconhecimento de Faces a partir de vídeo*. Dissertação (Mestrado) — UNIVERSIDADE ESTADUAL PAULISTA "Júlio de Mesquita Filho", 2009.

PENTEADO, B. E.; MARANA, A. N. A video-based biometric authentication for e-learning web applications. In: *11th International Conference, ICEIS 2009*. [S.l.: s.n.], 2009. v. 24, p. 770–779.

PRABHAKAR, S.; PANKANTI, S.; JAIN, A. K. Biometric recognition: Security and privacy concerns. *IEEE Security and Privacy*, IEEE Computer Society, Los Alamitos, CA, USA, v. 1, p. 33–42, 2003. ISSN 1540-7993.

RABINER, L. R. A tutorial on hidden markov models and selected applications in speech recognition. In: *Proceedings of the IEEE*. [S.l.: s.n.], 1989. p. 257–286.

RABUZIN, K.; BACA, M.; SJAKO, M. E-learning: biometrics as a security factor. In: *Proceedings of the International Multi-Conference on Computing in the Global Information Technology*. [S.l.: s.n.], 2006. p. 64–74.

RAMANATHAN, N. Facial similarity across age,disguise,illumination and pose. In: *Proceedings of International Conference on Image Processing*. [S.l.: s.n.], 1999.

RAYTCHEV, B.; MURASE, H. Unsupervised recognition of multi-view face sequences based on pairwise clustering with attraction and repulsion. *Comput. Vis. Image Underst.*, Elsevier Science Inc., New York, NY, USA, v. 91, n. 1-2, p. 22–52, 2003. ISSN 1077-3142.

ROSENBLATT, F. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review*, v. 65, n. 6, p. 386–408, nov 1958.

SAEED, U.; MATTA, F.; DUGELAY, J.-L. Person recognition based on head and mouth dynamics. In: *MMSP 2006, IEEE International Workshop on Multimedia Signal Processing, October 3-6, 2006, Victoria, Canada*. [S.l.: s.n.], 2006.

SATOH, S. *Comparative Evaluation of Face Sequence Matching for Content-based Video Access*. 2000.

SCHöLKOPF, B.; SMOLA, A. J. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond (Adaptive Computation and Machine Learning)*. 1st. ed. [S.l.]: The MIT Press, 2001. Hardcover. ISBN 0262194759.

THEODORIDIS, S.; KOUTROUMBAS, K. *Pattern Recognition, Third Edition*. Orlando, FL, USA: Academic Press, Inc., 2006. ISBN 0123695317.

TORRES, L.; VILA, J. *Automatic Face Recognition For Video Indexing Applications*. 2001.

TURK, M.; PENTLAND, A. Eigenfaces for Recognition. *Journal of Cognitive Neuroscience*, v. 3, n. 1, p. 71–86, 1991.

VAPNIK, V. N. An overview of statistical learning theory. *Neural Networks, IEEE Transactions on*, v. 10, n. 5, p. 988–999, 1999.

VIOLA, P.; JONES, M. Robust real-time object detection. *International Journal of Computer Vision*, 2001.

WARAKAGODA, N. *Assumptions in the theory of HMMs*. 1996. Acessado em novembro de 2009. Disponível em: <http://jedlik.phy.bme.hu/~gerjanos/HMM/node5.html>.

WAYMAN, J. L. Technical testing and evaluation of biometric identification devices. In: *Biometrics: Personal Identification in a Networked Society*. [S.l.: s.n.], 1999. p. 345–368.

WISKOTT, L. et al. Face recognition by elastic bunch graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, IEEE Computer Society, Los Alamitos, CA, USA, v. 19, p. 775–779, 1997. ISSN 0162-8828.

YAMAGUCHI, O.; FUKUI, K.; MAEDA, K. Face recognition using temporal image sequence. *Automatic Face and Gesture Recognition, IEEE International Conference on*, IEEE Computer Society, Los Alamitos, CA, USA, v. 0, p. 318, 1998.

YOUNG, S. J. et al. *The HTK Book Version 3.4*. [S.l.]: Cambridge University Press, 2006.

ZHANG, J.; LI, S. Z.; WANG, J. Manifold learning and applications in recognition. In: *in Intelligent Multimedia Processing with Soft Computing*. [S.l.]: Springer-Verlag, 2004. p. 281–300.

ZHANG, Y.; MARTÍNEZ, A. M. A weighted probabilistic approach to face recognition from multiple images and video sequences. *Image and Vision Computing*, p. 626–638, 2006.

ZHAO, W. et al. Face recognition: A literature survey. *ACM Computing Surveys*, p. 399–458, 2003.

ZHOU, S.; CHELLAPPA, R.; MOGHADDAM, B. Visual tracking and recognition using appearance-adaptive models in particle filters. *IEEE Transactions on Image Processing*, v. 13, p. 1434–1456, 2004.