Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Identificação e localização de pessoas em Smartspaces

Danilo Ávila Monte Christo Ferreira
Tales Mundim Andrade Porto

Monografia apresentada como requisito parcial
para conclusão do Bacharelado em Ciência da Computação

Orientador
Prof. Dr. Carla Denise Castanho

Brasília
2011

Universidade de Brasília — UnB
Instituto de Ciências Exatas
Departamento de Ciência da Computação
Bacharelado em Ciência da Computação

Coordenador: Prof. Lamar

Banca examinadora composta por:

Prof. Dr. Carla Denise Castanho (Orientador) — CIC/UnB
Prof. Dr. Professor I — CIC/UnB
Prof. Dr. Professor II — CIC/UnB

CIP — Catalogação Internacional na Publicação

Ferreira, Danilo Ávila Monte Christo.

Identificação e localização de pessoas em Smartspaces / Danilo Ávila Monte Christo Ferreira, Tales Mundim Andrade Porto. Brasília : UnB, 2011.
65 p. : il. ; 29,5 cm.

Monografia (Graduação) — Universidade de Brasília, Brasília, 2011.

1. palvrachave1, 2. palvrachave2, 3. palvrachave3

CDU 004.4

Endereço: Universidade de Brasília
Campus Universitário Darcy Ribeiro — Asa Norte
CEP 70910-900
Brasília–DF — Brasil

Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Identificação e localização de pessoas em Smartspaces

Danilo Ávila Monte Christo Ferreira
Tales Mundim Andrade Porto

Monografia apresentada como requisito parcial
para conclusão do Bacharelado em Ciência da Computação

Prof. Dr. Carla Denise Castanho (Orientador)
CIC/UnB

Prof. Dr. Professor I
CIC/UnB

Prof. Dr. Professor II
CIC/UnB

Prof. Lamar
Coordenador do Bacharelado em Ciência da Computação

Brasília, 2 de maio de 2011

Dedicatória

Dedico a....

Agradecimentos

Agradeço a....

Resumo

A ciência...

Palavras-chave: palvrachave1, palvrachave2, palvrachave3

Abstract

The science...

Keywords: keyword1, keyword2, keyword3

Sumário

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introdução

A computação ubíqua a tempos vem sendo tema de diversas pesquisas ao redor do mundo. Mark Weiser diz que o computador do futuro deve ser algo invisível (16, 17) proporcionando ao usuário um melhor foco na tarefa e não na ferramenta. A computação ubíqua tenta atribuir essa invisibilidade aos computadores buscando cada vez mais a diminuição do tamanho, a especificidade da tarefa e se acoplando aos objetos do dia-a-dia.

Um ambiente onde a computação ubíqua acontece em sua totalidade é chamado de SmartSpace (5). Esse ambiente provê ao usuário uma melhor forma de interagir com os computadores usando diversas tecnologias que estimulam a interatividade natural. Tais tecnologias são capazes de fornecer inteligência, ao SmartSpace, necessária para concretizar a visão da ubicomp (3).

Para conseguir uma boa interação entre as diversas peças que compõem o SmartSpace é necessário que se tenha a disposição informações de contexto, como quem está no ambiente, onde está, o que está fazendo e outras que ajudam o sistema a definir o melhor ajuste dos equipamentos. Com uma base rica de informações de contexto, contendo os perfis dos usuários, garantimos uma maior acurácia na tomada de decisões. Informações de contexto como essas são complicadas de se obter devido a alta dinamicidade do ambiente, no qual usuários entram e saem a todo momento e interagem com diversos equipamentos.

A identificação de usuário em um SmartSpace é feita por meio de sistema de reconhecimento automático. Há alguns anos, um grande número de pesquisas vem sendo desenvolvidas para criação sistemas deste tipo (2). Um dos motivos clássicos é que os métodos baseados em cartões de identificação e senhas não são altamente confiáveis. Estes podem ser perdidos, extraviados e até fraudados (15).

Um ambiente ubíquo capaz de reconhecer seus usuários, pode prover uma personalização automática do ambiente de acordo com as prefrências de cada usuário e até mesmo prover um ambiente mais seguro com controle de acesso físico e prevenção de fraudes (2). Atualmente, os métodos de reconhecimento mais utilizados são baseados no uso de cartões magnéticos e senhas, que requerem sua utilização durante uma transação, mas que não verificam sua idoneidade (4).

Hoje em dia, várias técnicas de reconhecimento por meio de faces, íris, voz, entre outras, vêm sendo estudadas e utilizadas em sistemas de reconhecimento automático (15). O reconhecimento facial pode ser considerada como uma das principais funções do ser humano pois permite identificar uma grande quantidade de faces e aspectos psicológi-

cos demonstrados pela fisionomia. Pode ser considerada, também, como um problema clássico da visão artificial pela complexidade existente na detecção e reconhecimento de características e padrões (2).

O reconhecimento facial vem se desenvolvendo junto a "quarta geração" de computadores através de sua aplicação na nova geração de interfaces que consiste na detecção e reconhecimento de pessoas (2).

É proposta então uma solução para o problema de localização e identificação de perfis de usuários em um SmartSpace utilizando como base o middleware UbiquitOS (6) integrado com o Kinect.

1.1 Organização do trabalho

Explicar a estrutura da monografia.

Capítulo 2

Localização de usuários em um SmartSpace

Capítulo 3

Reconhecimento Facial

Neste capítulo será apresentada uma abordagem conceitual sobre biometria e reconhecimento facial, uma vez que esses tópicos consistem no alicerce de nosso trabalho.

Será apresentado conceitos gerais sobre biometria e sobre as características biométricas existentes.

Sobre reconhecimento facial, será apresentado os conceitos gerais e conceitos mais específicos das diferentes etapas do processo de reconhecimento: detecção de faces e reconhecimento das mesmas. Além desses conceitos, será apresentado alguns métodos utilizados atualmente em cada uma dessas etapas.

3.1 Biometria

As abordagens de identificação pessoal que utilizam "alguma coisa que você sabe", como Número de Identificação Pessoal (PIN - "Personal Identification Number"), ou "alguma coisa que você tenha", como um cartão de identificação, não são confiáveis o suficiente para satisfazer os requisitos de segurança de um sistema de transações eletrônicas porque não têm a capacidade de diferenciar um usuário legítimo de um impostor que adiquiriu de forma ilegal o privilégio de acesso (9). Esta fragilidade pode ser evitada se utilzarmos o nosso corpo como chave do sistema. Alguns traços fisícos ou comportamentais são muito mais complicados de serem forjados que uma cadeia de caracteres (8).

Biometria é uma tecnologia utilizada para identificação de um indivíduo baseado em suas características físicas ou comportamentais, baseia-se em "alguma coisa que você é ou faz" para realizar a identificação e, por isso, tem a capacidade de diferenciar entre um indivíduo legítimo de um impostor (9). As características físicas estão relacionadas a composição do corpo humano e seu formato e as comportamentais estão relacionadas ao comportamento das pessoas (8). A figura 3.1 contém alguns exemplos desses dois tipos diferentes de características biométricas.

Teoricamente, qualquer característica física/comportamental pode ser utilizada para identificação caso siga alguns dos seguintes requisitos (11):

1. **universidade**: qualquer pessoa comum pode ser avaliada sobre essa característica;
2. **singularidade**: dada duas pessoas distinas, elas não podem ter a mesma característica dentro de uma proporção satisfatória;

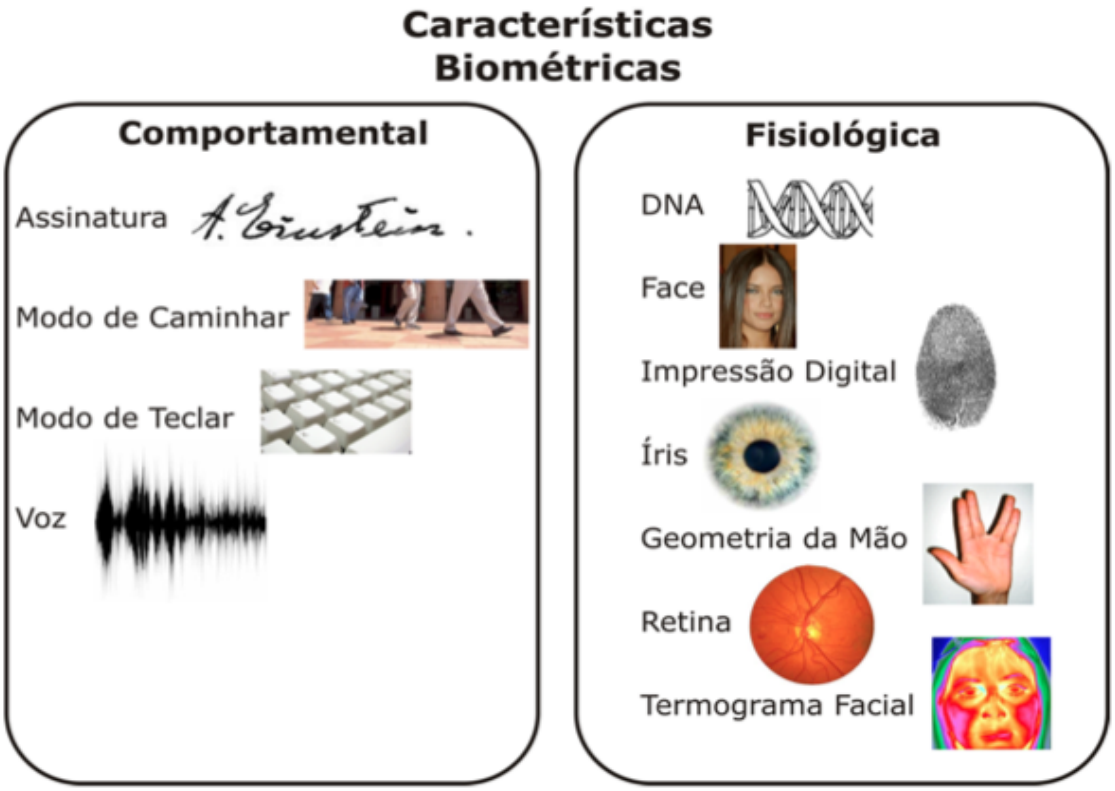

Figura 3.1: Exemplos de algumas características biométricas (8).

3. **permanência**: a característica não pode mudar significativamente de acordo com o tempo;
4. **exigibilidade**: pode ser mensurada quantitativamente;

Porém, na prática também são considerados outros requisitos (11):

1. **desempenho**: o processo de identificação deve apresentar um resultado aceitável;
2. **aceitação**: indica em que ponto as pessoas estão dispostas a aceitar o sistema biométrico;
3. **evasão**: refere a facilidade de ser adulterado;

São várias as vantangens que os sistemas biométricos têm em relação aos sistemas convencionais. Listamos as vantagens principais (8):

- características biométricas não podem ser perdidas ou esquecidas;
- características biométricas são difíceis de serem copiadas, compartilhadas e distribuídas;
- os sistemas biométricos necessitam que a pessoa esteja presente no local da autenticação;

Na prática um sistema biométrico deve ser capaz de (9):

1. atingir uma acurácia aceitável e com uma velocidade razoável;
2. não ser prejudiciável aos indivíduos e ser aceito pela população alvo;
3. ser suficientemente robusto para métodos fraudulentos;

Novas técnicas de reconhecimento por meio de face, íris, retina e voz, entre outras, têm sido abordadas para aplicações em sistemas de reconhecimento automático (2, 15). Das nove características utilizadas atualmente a face é uma das mais populares (11). Nas tabelas 3.1 e 3.2 são mostradadas as noves características e seus respectivos comportamentos baseados nos requisitos mencionados acima.

Tabela 3.1: Requisitos teóricos para algoritmos de reconhecimento facial (11).

Biometria	**Universidade**	**Singularidade**	**Permanência**	**Exigibilidade**
Face	Alta	Baixa	Média	Alta
Digital	Média	Alta	Alta	Média
Geometria da Mão	Média	Média	Média	Alta
"Hand Vein"	Média	Média	Média	Média
Iris	Alta	Alta	Alta	Média
"Retina Scan"	Alta	Alta	Média	Baixa
Assinatura	Baixa	Baixa	Baixa	Alta
Voz	Média	Baixa	Baixa	Média
Termograma	Alta	Alta	Baixa	Alta

Tabela 3.2: Requisitos práticos para algoritmos de reconhecimento facial (11).

Biometria	**Desempenho**	**Aceitação**	**Evasão**
Face	Baixa	Alta	Baixa
Digital	Alta	Média	Alta
Geometria da Mão	Média	Média	Média
"Hand Vein"	Média	Média	Alta
Iris	Média	Média	Alta
"Retina Scan"	Alta	Baixa	Alta
Assinatura	Baixa	Alta	Baixa
Voz	Baixa	Alta	Baixa
Termograma	Média	Alta	Alta

Os sistemas biométricos podem ser classificados em sistemas de verificação ou identificação. Sistemas de verificação são aqueles que autenticam a identidade dos usuários comparando-os com os próprios templates. Eles conduzem uma comparação "um para um" e determinam se o usuário é quem realmente diz ser. O maior desafio para esse tipo

de sistema é a acurácia. Geralmente, não é muito difícil satisfazer o requisito de tempo de resposta pois somente uma comparação "um para um" é feita (9).

Sistemas de identificação reconhecem um indivíduo pesquisando em todo o banco de dados procurando por uma correspondência. Eles conduzem uma comparação "um para muitos" para estabelecer a identidade do indivíduo. Ao contrário dos sistemas de verificação, nesse tipo de sistema tanto a acurácia quanto o tempo são os grandes desafios, por causa da necessidade de explorar todo o banco de dados. Geralmente, sistemas de identificação são mais complexos que sistemas de verificação (9).

Em um sistema biométrico exitem duas possíveis respostas, um usuário é ou não é quem afirma ser, sendo assim o sistema pode classificar o usuário como cliente ou impostor. Nessa tomada de decisão pode ocorrer dois tipos de erros: uma falsa aceitação, ao aceitar um impostor, (*False Acceptance* - FA) ou uma falsa rejeição (*False Rejection* - FR), ao rejeitar um cliente. Baseado nesses erros, duas taxas são utilizadas para avaliar sistemas biométricos: taxa de falsa aceitação (*False Acceptance Rate* - FAR) e taxa de falsa rejeição (*False Rejection Rate* - FRR) (8).

A FAR é a probabilidade de um sistema biométrico aceitar um impostor como cliente. Essa probabilidade é calculada pela equação (3.1) em que Nfa é o número de falsas aceitações e Ni é o número de impostores que tentaram acessar o sistema. A variação da taxa é representada pelo intervalo fechado $[0, 1]$, onde o valor 1 significa que todos os impostores foram falsamente aceitos e o valor 0 significa que todos impostores foram rejeitados. Logo quanto menor o FAR mais seguro o sistema é (8).

$$FAR = \frac{Nfa}{Ni}(8) \tag{3.1}$$

A FRR é a probabilidade de um sistema biométrico rejeitar um cliente e classifica-lo como impostor. Essa probabilidade é calculada pela equação (3.2) em que Nfr é o número de falsas rejeições e Nc é o número de clientes que tentaram acessar o sistema. A variação da taxa é representada pelo intervalo fechado $[0, 1]$, onde o valor 1 significa que todos os clientes foram falsamente rejeitados e o valor 0 significa que todo os cliente foram aceitos corretamente. Em sistemas cuja performance tem maior grau de prioridade que a segurança, deve-se reduzir a FRR para minimizar a ocorrência de falsas rejeições (8).

$$FRR = \frac{Nfr}{Nc}(8) \tag{3.2}$$

A partir dessas taxas de erro, pode-se obter outras medidas como a *Equal Error Rate* (ERR). Esta corresponde a taxa de erro na qual tanto a FAR quanto a FRR possuem o mesmo valor. Como muitos sistemas têm comportamentos diferentes, a ERR normalmente é utilizada para uma comparação mais rigorosa entre o sistemas. Quanto menor for a ERR, mas presciso é considerado o sistema (8).

3.2 Reconhecimento Facial

O reconhecimento facial é uma das atividades mais comuns realizadas diariamente por seres vivos dotados de certa inteligência. Essa simples atividade vem despertando o interesse de pesquisadores que trabalham com Visão Computacional e Inteligência Artificial.

O objetivo desses pesquisadores é de construir sistemas artificiais capazes de realizar o reconhecimento de faces humanas e a partir desta capacidade construir os mais diferentes tipos de aplicações: sistemas de vigilância, controles de acesso, definções automáticas de perfis, entre outros (13).

No anos 70, os estudos do reconhecimento facial eram baseados sobre atributos faciais mensuráveis como olhos, nariz, sobrancelhas, bocas, entre outros. Porém, os recursos computacionais eram escassos e os algoritmos de extração de características eram ineficiêntes. Nos anos 90, as pesquisas na área ressurgiram, inovando os métodos existentes (2, 9) e disseminando a técnica.

Um dos motivos que incentivou os diversos estudos sobre reconhecimento facial são as vantagens que o mesmo possui em relação a impressão digital e a íris. No reconhecimento por impressão digital, a desvantagem consiste no fato que nem todas as pessoas possuem uma impressão digital com "qualidade" suficiente para ser reconhecida por um sistema. Já o reconhecimento por íris apresenta uma alta confiabilidade e larga variação, sendo estável pela vida toda. Porém, a desvantagem está relacionada ao modo de captura da íris que necessita de um alinhamento entre a câmera e os olhos da pessoa (2).

Basicamente existem duas particularidades que fazem da face uma característica biométrica bastante atrativa (8):

1. A aquisição da face é feita de forma fácil e não-intrusiva;
2. Possui uma baixa privacidade de informação: como a face é exposta constantemente, caso uma base de faces seja roubada, essas informações não representam algum risco e não possibilitam um uso impróprio;

Umas das maiores dificuldades dos sistemas de reconhecimento é tratar a complexidade dos padrões visuais. Mesmo sabendo que todas as faces possuem padrões reconhecidos, como boca, olhos e nariz, elas também possuem variações únicas que devem ser utilizadas para determinar as características relevantes. Outra dificuldade encontrada em relação a essas características é que elas possuem uma larga variação estatística para serem consideradas únicas para cada indivíduo. O ideal seria que a variância inter-classe seja grande e a intra-classe pequena, pois assim imagens de diferentes faces geram os códigos mais diferentes possíveis, enquanto imagens de uma mesma face geram os códigos mais similares possíveis. Portanto, estabelecer uma representação que capture as características ideias é um difícil problema (2).

Do ponto de vista geral, o recohecimento facial continua sendo um problema aberto por causa de várias dificuldades que aumentam a variância intra-classe (9). Entre estas, destacamos as mais comuns (2):

- iluminação;
- ângulos e poses;
- expressões;
- comésticos e acessórios;
- extração da face do contexto ou do fundo;

No contexto de identificação, o reconhecimento facial se resume no reconhecimento de um "retrato" frontal, estático e controlado. Estático pois os "retratos" utilizados nada mais são que imagens, podendo ser tanto de intensidade quanto de profunidade e controlado pois a iluminação, o fundo, a resolução dos dispositivos de aquisição e a distância entre eles e as faces são essencialmente fixas durante o processo de aquisição da imagem (9).

Basicamente, o processo de reconhecimento facial pode ser divido em duas tarefas principais (9):

1. Detecção de faces em imagens;
2. Reconhecimento das faces encontradas;

Falaremos dessas duas tarefas separadamente nas próximas subseções.

3.2.1 Detecção de Faces em imagens

A primeira etapa para o reconhecimento de faces é a detecção de um rosto, e a partir daí a comparação do mesmo com modelos conhecidos pelo sistema (9, 13). Em um sistema de reconhecimento facial, tanto o tempo de resposta quanto a confiabilidade desta etapa influênciam diretamente no desempenho e o emprego deste sistema (13).

A detecção de faces é definida como o processo que determina a existência ou não de faces em uma imagem e uma vez encontrada alguma face, sua localização deve ser apontada através de um enquadramento ou através de suas coordenadas dentro da imagem (13). A Figura 3.2 representa um exemplo da detecção de uma face em uma imagem.

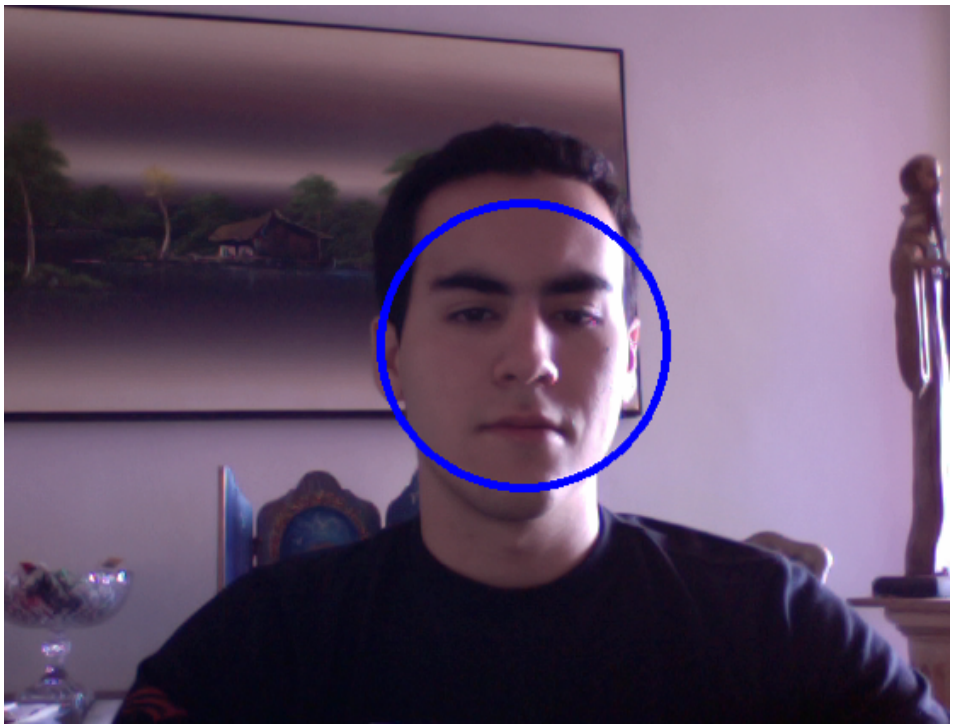

Figura 3.2: Exemplo de um processo de detecção de uma face em uma imagem.

O processo de detecção de faces geralmente é dificultado pelas seguintes razões mostradas a seguir:

1. **Pose**: as imagens de uma face podem variar de acordo com a posição relativa entre a camêra e a face (frontal, 45 graus, perfil, "de cabeça para baixo"), e com isso algumas características da face, como olhos e nariz, podem ficar parcialmente ou totalmente ocultadas (12).
2. **Presença de acessórios**: características faciais básicas importantes para o processo de detecção podem ficar ocultadas pela presença de acessórios, como óculos, bigode, barba, entre outros (12, 13).
3. **Expressões faciais**: embora a maioria das faces apresente estruturas semelhantes (olhos, bocas, nariz, entre outros) e são dispostas aproximadamente na mesma configuração de espaço, pode haver um grande número de componentes não rigídos e texturas diferentes entre as faces. Um exemplo são as flexibilizações causadas pelas expressões faciais (12, 13);
4. **Obstrução**: faces podem ser obstruídas por outros objetos. Em uma imagem com várias faces, uma face pode obstruir outra (12).
5. **Condições da imagem**: a não previsibilidade das condições da imagem em ambientes sem restrições de ilimuniação, cores e objetos de fundo (12, 13).

Atualmente, já existem diferentes métodos/técnicas de detecção de faces. Tais métodos podem ser baseados em imagens de intensidade e de cor ou em imanges 3D. Focaremos nos principais métodos de imagens de cor e de intensidade que serão utilizados neste trabalho. Estes podem ser divididos em 4 categorias:

1. **Métodos baseados em Conhecimento:** métodos, desenvolvidos principalmente para localização facial, beaseados em regras derivadas do conhecimento dos pesquisadores do que constitui uma típica face humana. Normalmente, captura as relações existentes entre as características faciais. É fácil econtrar regras que descrevem as caracterísicas faciais. Por exemplo, uma face sempre é constituída por dois olhos simétricos, um nariz e uma boca. As relações entre essas características podem ser representadas pelas distâncias relativas e posições. Este método possui desvantagens em relação a construção do conjunto de regras. Se estas são muito gerais, corre-se o risco de que o sistema que as utiliza apresentar uma alta taxa de falsos positivos. Se são muito específicas podem ser ineficazes ao tentar detectar faces por não satisfizerem todas as regras, diminuindo muito a precisão da detecção (10, 12);
2. **Métodos baseados em Características Invariantes:** esses algoritmos tem como objetivo principal encontrar as características estruturais que existem mesmo quando a postura, "ponto de vista", condições de iluminação variam. E por meio dessas características localizar a face. São desenvolvidos principalmente para localização facial (12). A principal desvantagem desse método é que tais características invariantes podem ser corrompidas devido a condições de ilumincação ou algum tipo de ruído, comprometendo a eficiência. A cor da pele e a textura da face são as principais características invariantes que podem ser utilizadas para separar a face de outros objetos (10);

3. **Métodos baseados em Templates:** vários padrões comuns de um rosto são armazenados tanto para descrever o rosto como um todo quanto para descrever as características faciais separadamente. As correlações entre as imagens de entrada e os padrões armazenados são comuptados para detecção. Esses métodos são desenvolvidos para serem utlizados como localização e detecção facial (12);

4. **Métodos baseados em Aparência:** recebem este nome devido ao fato de não utilizarem nenhum conhecimento a priori sobre o objeto ou características a serem detectadas. Em contraste com os métodos baseado em templates, os modelos são retirados de um conjunto de imagens de treinamento que devem capturar a variabilidade da face. Esses modelos retirados são utilizados para detecção. Nesta classe de algoritmos surge os conceitos de aprendizado e treinamento, uma vez que as informações necessárias para realizar a tarefa de detecção são retiradas do próprio conjunto de imagens sem intervenção externa. São métodos desenvolvidos principalmente para detecção de faces (10, 12);

Um problema relacionado e muito importante é como avaliar a performance dos métodos de detecção de faces propostos. Para isso, muitas métricas foram adotadas como tempo de aprendizagem, número de amostras necessárias no treinamento e a proporção entre taxas de detecção e "falso alarme". Esta última é dificultada pelas diferentes definições para as taxas de detecção e falso alarme adotadas pelos pesquisadores (12).

O método que será utilizado nesse trabalho será o método *Viola-Jones*. Este é um método para detecção de objetos que minimiza o tempo de computação e possui uma alta acurácia permindo detecção de faces em tempo real. Esse método pode ser utilizado para construir uma abordagem de detecção facial rápida e eficaz (14). É o método utilizado na biblioteca *OpenCV* (Open Source Computer Vision) e muito utilizado autalmente.

O Método *Viola-Jones* para detecção facial utiliza quatro conceitos chaves (1, 14):

1. ***Haar features*:** simples características retangulares;

2. ***Integral Image*:** uma nova representação da imagem que permite uma rápida avaliação de recursos e caracterísitcas;

3. **O método *AdaBoost***: um método de aprendizagem de máquina utilizado para construir um classificador, selecionando um pequeno número de características importantes usando *AdaBoost*;

4. **Classificador em Cascata**: classificador que combina muitas características de maneira eficiente;

A detecção facial em imagens é baseado em simples caracteríticas retangulares (*Haar features*), exemplificada na Figura 3.3. Existem vários motivos para se usar essas características ao invés de usar diretamente os pixels da imagem. Uma das principais é que sistemas baseados em características são muito mais rápidos que sistemas baseados em pixels (14).

Tais simples características são remanecentes das funções de base *Haar*. O método utiliza três tipos de características, exemplificadas na Figura 3.4: características com dois, três ou quatro retângulos (14). A presença de uma característica em uma imagem é determinada pela subtração da média dos valores dos pixels da região escura pela média

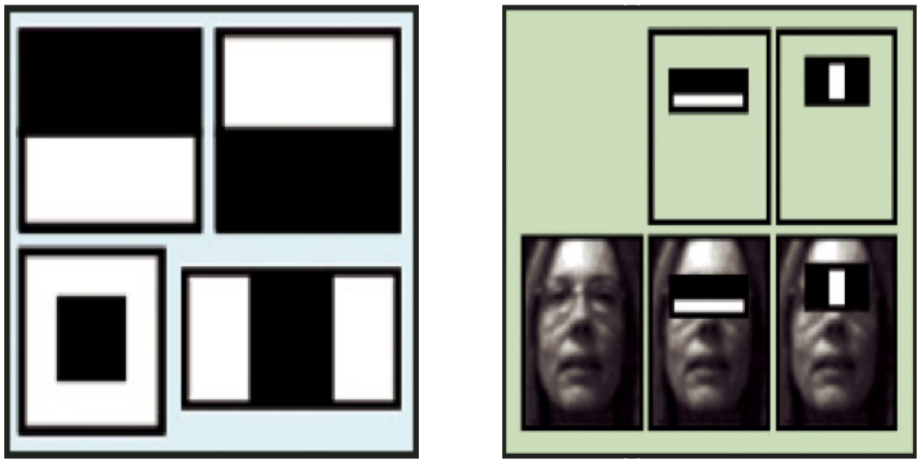

Figura 3.3: Exemplo de simples características retangulares (Haar Features). Adaptada de (1).

dos valores dos pixels da região clara. Caso o valor seja maior que um limiar, então essa característica é tida como presente (1).

O Método *Viola-Jones* não trabalha diretamente com as intensidades da imagem. Para determinar a presença ou ausência de centenas de *Haar features* em cada posição de imagem e em várias escalas de forma eficiente, utiliza uma técnica chamada *Integral Image*. Basicamente, o método consiste em acrescentar pequenas unidades juntas. Neste caso, pequenas unidades são valores de pixels. O valor "integral" para cada pixel é a soma de todos os pixels acima e a esquerda. Começando pelo canto superior esquerdo da imagem e atravessando para direita e para baixo, toda a imagem pode ser "integrada" com poucas operações por pixels (1, 14). Com a nova representação de imagem criada, qualquer *Haar Feature* pode ser computada para qualquer escala e localização em um tempo constante (14).

Na Figura 3.5, após a "integração", o pixel (x, y) contém a soma de todos os valores de pixels dentro da região retangular sombreada no canto superior esquerdo de (x, y). Para calcular a média dos valores de pixel neste retângulo, basta dividir o valor em (x, y) pela área do retângulo (1).

A Figura 3.6 mostra como calcular os valores de um retângulo que não está localizado no canto superior esquerdo da imagem. Suponha que se deseja obter a soma dos valores do retângulo em D. De maneira intuitiva, pode-se realizar esse cálculo somando os valores dos pixels no retângulo formado por $A + B + C + D$, menos a soma nos retângulos $A + B$ e $A + C$, mais a soma dos valores de pixel em A. Em outras palavra, $D = A + B + C + D - (A + B) - (A + C) + A$ (1, 14).

Porém, usando *Integral Image*, $A+B+C+D$ é o valor do ponto 4, $A+B$ é o valor do ponto 2, $A+C$ é o valor do ponto 3, e A é o valor do ponto 1. Então, com *Integral Image*, você pode obter a soma dos valores de pixel de qualquer retângulo na imagem original usando somente três operações (1, 14):

$$(x_4, y_4) - (x_2, y_2) - (x_3, y_3) + (x_1, y_1) \tag{3.3}$$

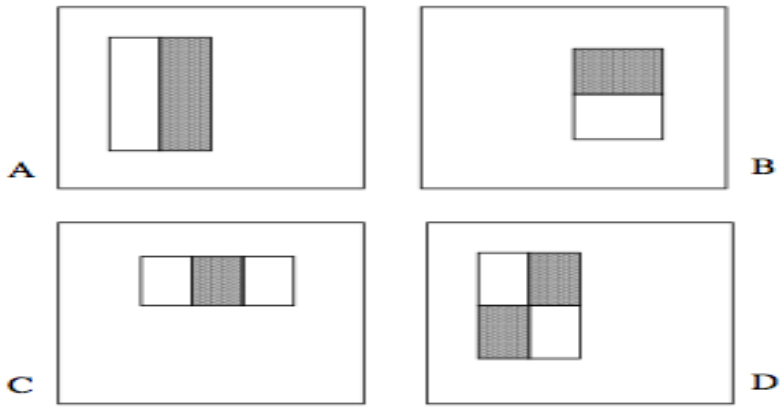

Figura 3.4: Haar Features com dois, três e quatro retângulos (14).

Para selecionar os *Haar features* que serão utilizados e para definir os limiares, o método *Viola-Jones* utiliza um método de aprendizagem de máquina chamado *AdaBoost*. Este combina vários classificardores "fracos" para criar um classificador "forte".

Um classificador fraco é aquele que só obtém a resposta correta um pouco mais frequente que um "palpite aleatório". A combinação desses classficadores "fracos", onde cada um "empurra" a resposta final um pouco na direção certa, pode ser considerado como um classificador "forte". O método AdaBoost seleciona um conjunto de classificadores "fracos" para combinar e atribui pesos a cada um. Essa combinação ponderada resulta em um classificador "forte" (1).

Em qualquer região de uma imagem, o número total de *Haar features* é muito grande, muito maior que o número de pixels. Para assegurar uma classficação rápida, o processo de aprendizagem deve excluir o maior número de características disponíveis, e focar nas que são críticas. A seleção dessas características é alcançada através de uma simples modificação no método *AdaBoost*: o mecanismo de aprendizagem é construído de forma que cada classificador "fraco" retornado dependa de somente uma única característica. Como resultado, cada estágio do processo seleciona um novo classificador "fraco" o que pode ser visto como um processo de seleção de características. *AdaBoost* fornece um algoritmo de aprendizagem eficaz (14).

O método *Viola-Jones* combina uma série de classificadores *AdaBoost* na forma de uma cadeia de filtros, como na Figura 3.7, que recebe o nome de Classificadores em Cascata. Cada filtro em si é um classificador *AdaBoost* com um número relativamente pequeno de classificadores "fracos" (1).

O limiar de aceitação em cada nível é definido "baixo" o suficiente para que passe por todos, ou quase todos, os exemplos de face do conjunto de treinamento (um grande banco de imagens contendo faces). Os filtros em cada nível são treinados para classificar imagens de treinamento que passaram por todas fases anteriores (1).

Durante a utilização, se alguma região de uma imagem falhar em passar em um desses filtros, esta é imediatamente classificada como "não face". Quando uma região de uma imagem passa por um filtro, ela vai para o próximo filtro na cadeia. As regiões das imagens

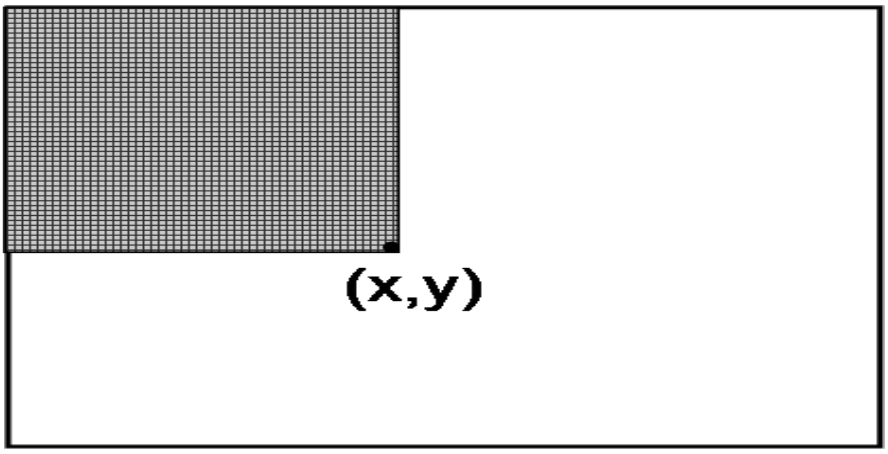

Figura 3.5: Exemplo da aplicação da técnica Integral Image. Após a "integração" o pixel (x, y) contém a soma dos valores dos pixels do retângulo sombreado. Adaptada de (1).

que passarem por todos os filtros na cadeia são classificadas como "faces" (1).

O algoritmo utilizado para construção dos Classificadores em Cascata alcança um ótimo desempenho e, ao mesmo tempo, reduz drasticamente o tempo de computação. O aspecto chave é que os menores classificadores (filtros), e por isso mais eficientes, podem ser utilizados para rejeitar a maioria das regiões das imagens que não são faces antes que os classifcadores mais complexos sejam utilizados (14). A ordem dos filtros no classificador é baseado nos pesos que o método *AdaBoost* define para cada filtro. Os filtros com maior peso são colocados no início, para eliminar as regiões das imagens que não são faces o mais rápido possível (1).

O método *Viola-Jones* é adequado para utilização em sistemas de detecção de faces em tempo real. Agora, o próximo passo para o processo de Reconhecimento Facial é comparar as faces encontradas com modelos conhecidos com o sistema para realizar a identificação, o que será mostrado na próxima subseção.

3.2.2 Reconhecimento das Faces encontradas

Na etapa de reconhecimento, as faces detectadas e processadas, serão comparadas com um banco de dados de faces conhecidas. Essa comparação tem uma acurácia média de 30-90% entre as diversas técnicas. Esse é um forte campo de pesquisa desde a década de 90 e as técnicas se invovam ano ápos ano.

Existem duas variações principais entre as técnicas, as que usam como entrada dados imagens 2D (imagens de intensidade e cor) e as que usam como entradas dados imagens 3D (*depth images*). As principais técnicas 2D são:

1. ***Eigenfaces***
2. **Redes Neurais**
3. ***Fisher Faces***

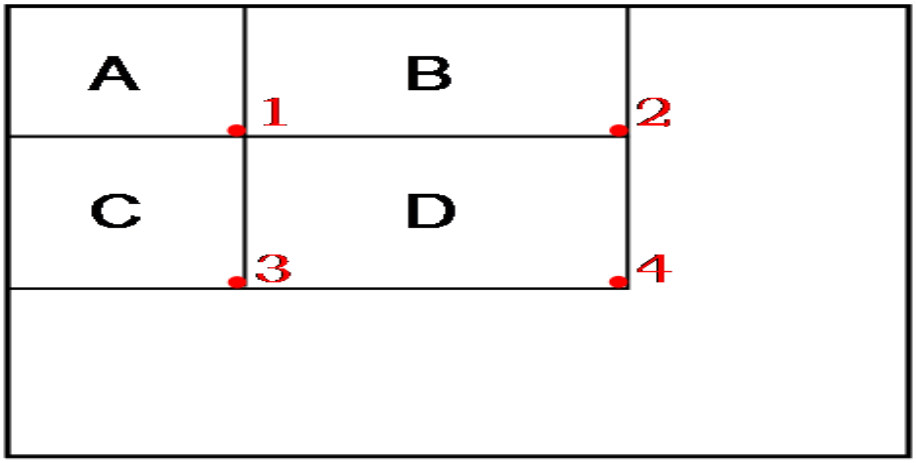

Figura 3.6: Exemplo utlizado para mostrar como calcular a soma dos valores de pixel de um retângulo que não está localizado no canto superior esquerdo da imagem utilizando *Integral Image*. Adaptado de (1).

De todos os métodos apresentados o *Eigenfaces* se mostrou de melhor custo benefício, devido ao seu baixo custo computacional e a sua, relativamente alta, confiabilidade.

Este é um algoritmo simples e fácil de implementar e os passos utilizados por ele também são utilizados em muitos métodos avançados. Os princípios básicos por trás dele, como PCA (*Principal Component Analisys* - Analíse de componente principal) e *Distance-Based Matching* (Correspondência Baseada na Distância) aparecem cada vez mais na computação visual e em diversas aplicações de inteligencia artificial (7). O *Eigenfaces* trabalha de forma simples, dada uma imagem de um rosto desconhecido e imagens do rosto das pessoas conhecidas executa as seguintes ações (7):

1. Computa a distância entre a nova imagem e cada uma das imagens já conhecidas.
2. Seleciona a imagem mais proxima do novo rosto.
3. Se a distância da nova imagem para a imagem já catalogada for menor que o limite predefinido, "reconhece" a imagem caso contrário classifica como "desconhecida".

A distância entre as imagens é medida ponto a ponto. Esta é também chamado de distância euclidiana. A distância euclidiana em duas dimensões entre os pontos P_1 e P_2 é dada pela fórmula $d_{12} = \sqrt{(d_{x2} + d_{y2})}$, onde $d_x = x_2 - x_1$ e $d_y = y_2 - y_1$ e representada na Figura 3.8 (7).

Imagens possuem "ruídos" e vamos definir ruído como qualquer coisa que atrapalhe na identificação, como por exemplo as diferenças na luminosidade. Cada pixel possui uma intensidade de ruído diferente. Com cada pixel contribuindo para o ruído total, este dificulta o processo de encontrar a imagem correta referente a imagem testada. Uma solução é diminuir a dimensionalidade da imagem tornando assim o ruído menor e sendo possível extrair da imagem as informações importantes (7).

Um dos métodos existentes para reduzir a dimesionalidade da imagem é o "PCA - Principal Components Analysis" (7).

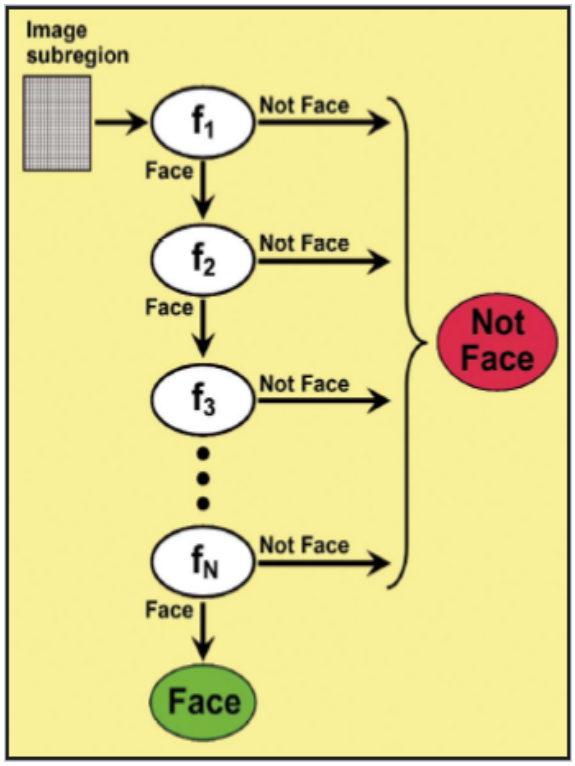

Figura 3.7: Ilustração de uma classificador em cascata composto com uma cadeia de filtros (1).

Para se ter uma idéia do que é o PCA, vejamos um caso especial chamado de *least squares line fit*. O lado esquerdo da Figura 3.9 mostra um exemplo de uma linha média entre três pontos, que são, no mapa em 2D, Los Angeles, Chicago e Nova York. Estes três pontos do mapa são quase, mas não completamente, uma única linha. A linha tem apenas uma dimensão. Por isso, se podemos substituir localizações dos pontos de 2D com localizações ao longo de uma única linha 1D, vamos ter reduzido a sua dimensionalidade (7).

Como os pontos, nos quais se baseia o nosso exemplo, estão praticamente alinhados, uma linha pode ser traçada através deles com pouco erro. O erro no ajuste da linha é medido pela soma do quadrado da distância de cada ponto da linha. A linha de melhor ajuste é aquela que possui o menor erro (7).

Embora a linha encontrada acima seja um objeto 1D, esta é localizada dentro de um espaço maior, 2D, e tem como orientação sua inclinação. A inclinação da linha expressa algo importante sobre os três pontos. Ele indica a direção em que eles estão mais espalhados (7).

Se posicionarmos a origem do nosso plano cartesiano em algum lugar dessa linha, podemos escrever a equação da linha como uma simples $y = mx$, onde m é a inclinação da linha: dy/dx (7).

Quando ela é descrito desta maneira, a linha é um subespaço do espaço 2D definido pelo sistema de coordenadas. Esta descrição enfatiza o aspecto dos dados que estamos interessados, ou seja, a direção que mantém esses pontos mais separados um do outro (7).

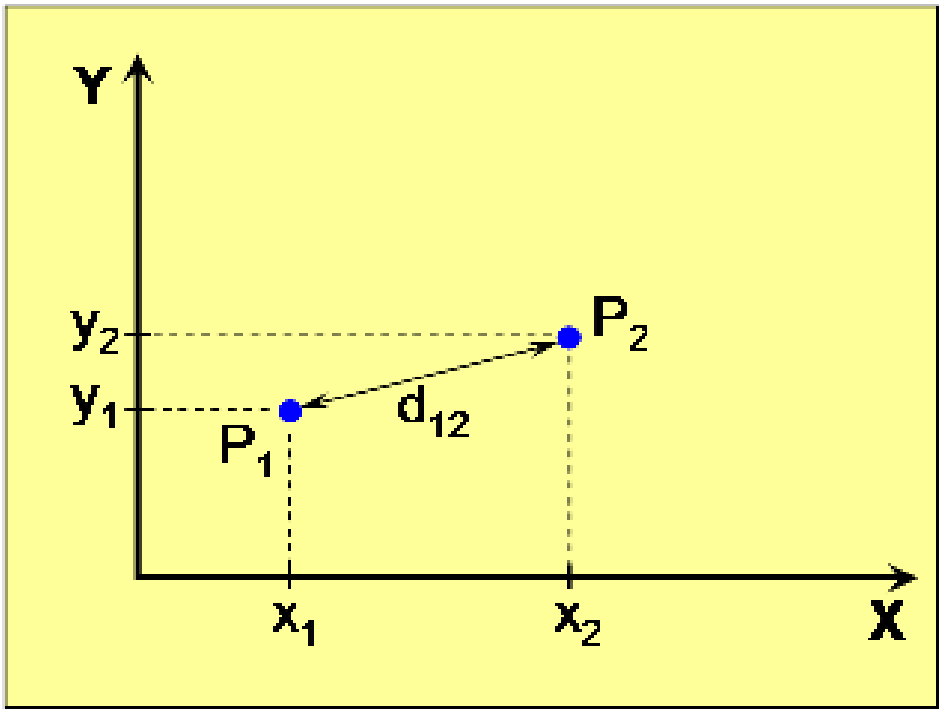

Figura 3.8: Distância Euclidiana entre dois pontos em duas dimensões (7).

Esta direção da separação máxima é chamada de primeira componente principal de um conjunto de dados. A próxima direção com máxima separação é a perpendicular a esta. Essa é a segunda componente principal. Em um conjunto de dados 2D, podemos ter no máximo duas componentes principais (7).

No entanto, o número de componentes principais que podemos encontrar também é limitada pelo número de pontos de dados. Para ver o porque disto, podemos pensar em um conjunto de dados que consiste de apenas um ponto. Não há sentido da separação máxima para esse conjunto de dados, porque não há nada para separar. Agora, considere um conjunto de dados com apenas dois pontos. A linha que conecta esses dois pontos é a primeira componente principal. Mas não há uma segunda componente principal, porque não há nada mais para separar, pois os dois pontos estão totalmente na linha (7).

Em *Eigenfaces*, cada imagem da face, de tamanho 50x50, é tratada como um ponto com 2500 dimenções. Portanto, o número de componentes principais que podemos encontrar nunca será maior que o número de imagens de faces menos um (7).

Embora seja importante ter um entendimento conceitual do que as componentes principais são, você não precisa saber os detalhes de como encontrá-las para implementar o *Eigenface*. Essa parte já foi implementada em bibliotecas de processamento de imagens *open source*, como por exemplo a bliblioteca *OpenCV* (7).

O processo de encontrar um ponto correspondente num subespaço de menor dimensão se chama projeção. Há uma função no *OpenCV* para projetar os pontos sobre um subespaço, então, novamente, você só precisa de um entendimento conceitual. Você pode deixar os detalhes algorítmicos para a biblioteca (7).

As marcas azuis na Figura 3.9 mostram as localizações no subespaço das três cidades

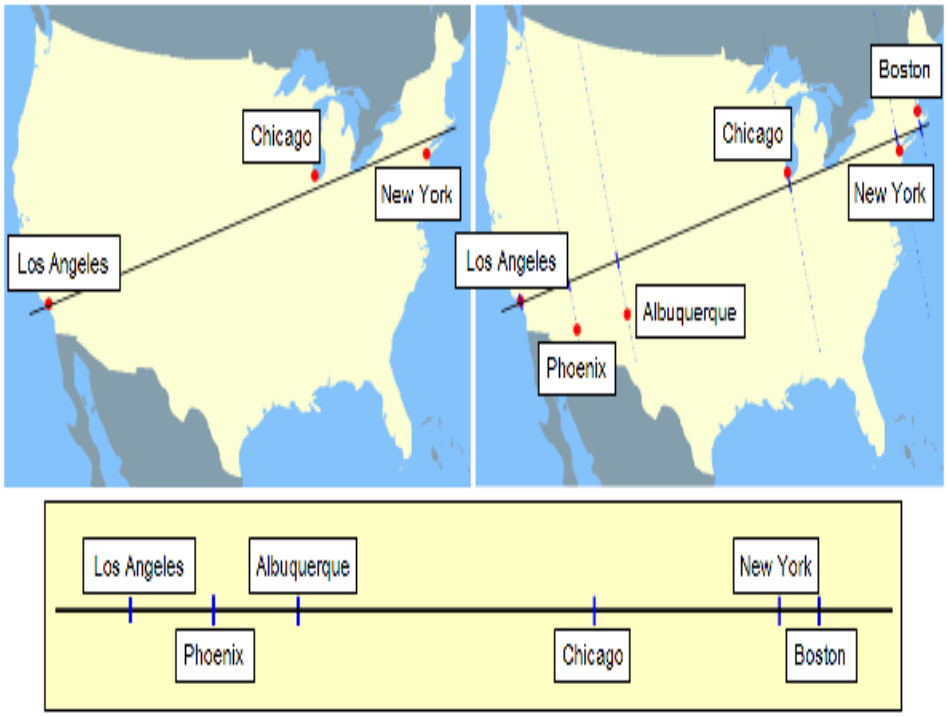

Figura 3.9: Mapa exemplo para redução de dimensionalidade (7).

que definiram a linha. Outros pontos 2D também pode ser projetado para esta linha. O lado direito da Figura 3.9 mostra a localização prevista para Phoenix, Albuquerque, Boston (7).

Em *Eigenface*, a distância entre duas imagens é a distância euclidiana entre os pontos projetados em um subespaço, ao invés da distância no espaço original da imagem de 2500 dimenções. A distância entre as faces neste subespaço de menor dimensão é a técnica utilizada para melhorar a relação sinal / ruído (7).

Muitas técnicas avançadas de reconhecimento de face são extensões deste conceito básico. A principal diferença entre *Eigenface* e estas técnicas avançadas é o processo de definição do subespaço. Em vez de usar PCA, o subespaço pode ser baseada em Análise de Componentes Independentes (ICA) ou em Análise Discriminante Linear (LDA), e assim por diante (7).

Em nossa definição de uma linha como um subespaço 1D, usamos x e y coordenadas para definir m, que é sua inclinação em 2D. Quando m é um componente principal de um conjunto de pontos, ela é chamada de autovetor ou *eingenvector*, daí o nome *Eigenface* (7).

Para o reconhecimento facial em imagens de 50x50, cada autovetor representa a inclinação de uma linha em um espaço de 2.500 dimenções. Como no caso 2D, precisamos de todas as 2.500 dimensões para definir a inclinação de cada linha. Embora seja impossível visualizar uma linha em muitas dimensões, podemos visualizar os autovetores de uma maneira diferente. Podemos converter as suas 2.500 dimenções em uma simples imagem usando a sua "inclinação" para colocar cada pixel valor em seu local correspondente. Quando fazemos isso, obtemos imagens *facelike* chamadas de *Eigenface* (7).

Eigenfaces é um método interessante para dar-nos alguma intuição sobre os componentes principais para o nosso conjunto de dados. O lado esquerdo da Figura 3.10 mostra as imagens das faces de dez pessoas. Estas imagens foram encontradas no *Yale Face Database B* (referências 4 e 5). Ele contém imagens de rostos com uma variedade de condições de iluminação. Foram usadas sete imagens de cada uma dessas dez pessoas para criar um subespaço PCA (7).

O lado direito da Figura 3.10 mostra os seis primeiros componentes principais deste conjunto de dados, apresentados como eigenfaces. O *Eigenfaces* muitas vezes têm um olhar fantasmagórico, porque combinam elementos de várias faces. As regiões de pixels mais brilhantes e as regiões mais escuras em cada imagem são as que mais contribuem para as componentes principais (7).

Figura 3.10: Direita: imagens de rosto para dez pessoas. Esquerda: os seis primeiros componentes principais, visto como *Eigenfaces* (7).

As componentes principais encontradas pelo PCA apontam para a maior variação de dados. Uma das premissas do *Eigenfaces* é que a variabilidade das imagens subjacentes

correspondem à diferença entre as faces. Esta suposição nem sempre é válida. A Figura 3.11 mostra as faces de dois indivíduos apresentadas em quatro diferentes condições de iluminação (7).

Na verdade, elas são imagens de faces de duas das dez pessoas mostradas na Figura 3.10. Quando a iluminação é muito variável esse algoritmo não é muito efetivo (7).

Figura 3.11: Imagens das faces de dois indivíduos. A face de cada indivíduo é apresentada em quatro diferentes cosndições de iluminação. A variabilidade devido à iluminação aqui é maior do que a variabilidade entre os indivíduos. Eigenfaces tende a confundir as pessoas quando os efeitos de iluminação são muito fortes (7).

Outros fatores que podem aumentar a variabilidade da imagem em direções que tendem a diluir a identidade no espaço PCA incluem mudanças na expressão, ângulo da câmera e posição da cabeça (7).

A Figura 3.12 mostra como a distribuição de dados afeta o desempenho do Eigenfaces.

Quando os pontos referentes as imagens de cada indivíduo ficam aglutinadas e satisfatoriamente separadas das imagens do conjunto de imagens de outros individuos temos o melhor caso para o funcionamento do Eigenfaces.

Caso os pontos referentes as imagens dos indivíduos tenham uma variabilidade muito grande, a probabilidade de choque de imagens de dois indivíduos num mesmo ponto do subespaço PCA se torna muito grande tornando extremamente difícil separar os dois indivíduos (7).

Na prática, a projeção de determinadas imagens de uma pessoa no subespaço PCA provavelmente colidirá com projeções de imagens de outras pessoas. Como os autovetores (eigenvector) são determinados pela variabilidade dos dados, ficamos limitados a quão grande deve ser essa. Podemos tomar medidas para limitar, ou para gerir de outra forma, as condições ambientais que podem confundi-lo. Por exemplo, colocar a câmera na altura do rosto irá reduzir a variabilidade no ângulo da câmera (7).

As condições de iluminação, tais como iluminação lateral vinda de uma janela, são mais difíceis de controlar. Mas você pode considerar o acréscimo de inteligência em cima de reconhecimento facial para compensar isso. Por exemplo, se sabemos onde ele está

localizado, e em que direção está olhando, ela pode comparar a imagem do rosto atual apenas com aquelas em situação semelhante (7).

Mesmo com sistemas altamente ajustados, sistemas de reconhecimento facial estão sujeitos a casos de identidade equivocada (7).

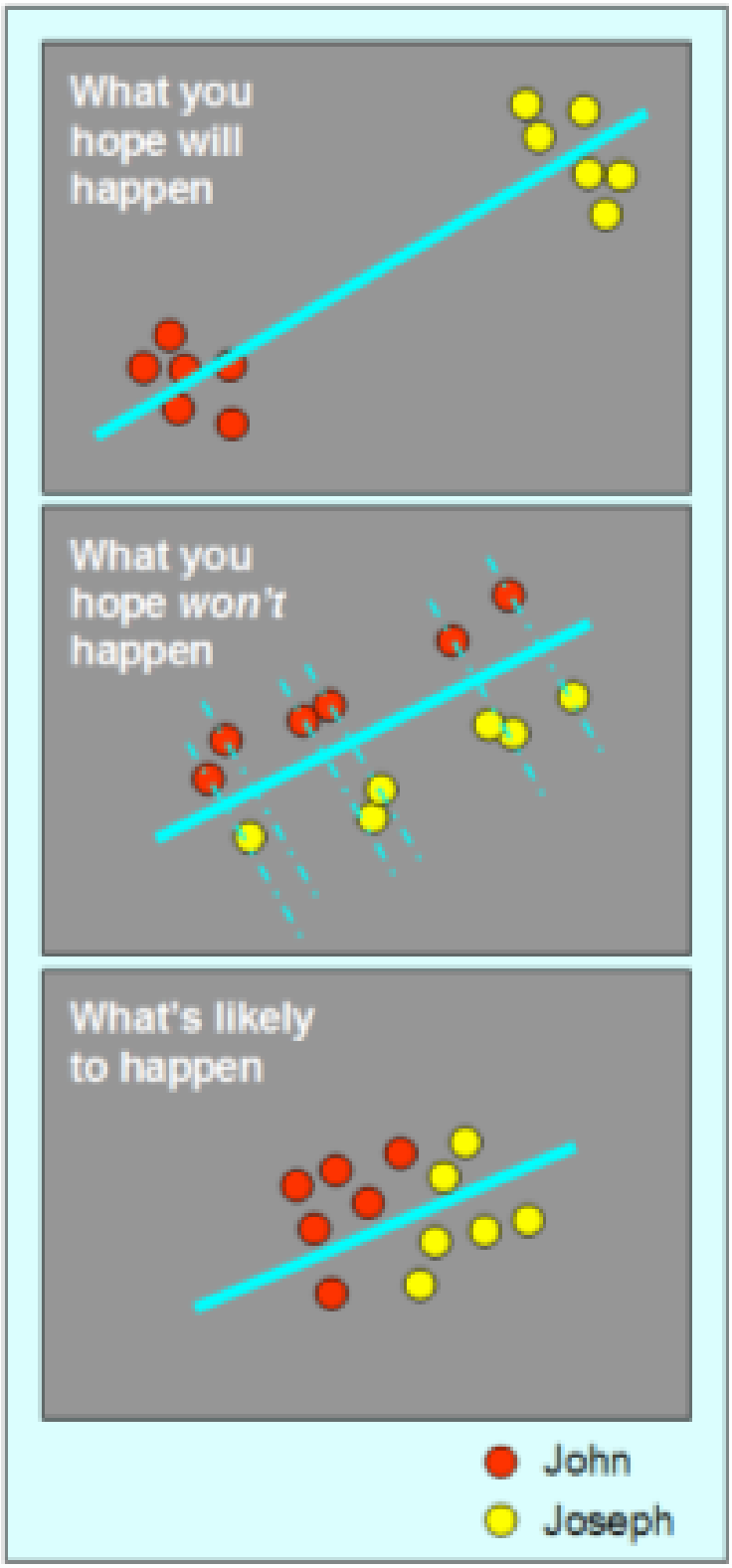

Figura 3.12: Como as distribuições de dados afetam o reconhecimento com Eigenfaces. Topo: O melhor cenário possível - pontos de dados para cada pessoa bem separados. Meio: O pior cenário - a variabilidade entre as imagens das faces de cada individuo é maior do que a variabilidade entre os indivíduos. Embaixo: um cenário realista - separação razoável, com alguma sobreposição (7).

Referências

[1] How face detection works. *SERVO Magazine*, fevereiro 2007. vi, 11, 12, 13, 14, 15, 16

[2] Â. R. Bianchini. Arquitetura de redes neurais para o reconhecimento facial baseado no neocognitron. Master's thesis, Universidade Federal de São Carlos, 2001. 1, 2, 6, 8

[3] F. N. Buzeto. Um conjunto de soluções para a construção de aplicativos de computação ubíqua. Master's thesis, Departamento de Ciência da Computação, Universidade de Brasília, http://monografias.cic.unb.br/dspace/handle/123456789/257, 2010. 1

[4] J. Daugman. Face and gesture recognition: Overview. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7), 1997. 1

[5] I. Essa G. Abowd, C. Atkeson. Ubiquitous smart spaces. *A white paper submitted to DARPA (in response to RFI)*, 1998. 1

[6] A. R. Gomes. Ubiquitos – uma proposta de arquitetura de middleware para a adaptabilidade de serviços em sistemas de computação ubíqua. Master's thesis, Departamento de Ciência da Computação, Universidade de Brasília, http://monografias.cic.unb.br/dspace/handle/123456789/110, 2007. 2

[7] R. Hewitt. Face recognition with eigenface. *SERVO Magazine*, 2007. vi, 15, 16, 17, 18, 19, 20, 21, 22

[8] S. A. D. Junior. Reconhecimento facial 3d utilizando o simulated annealing com as medidas surface interpenetration measure e m-estimator sample consensus. Master's thesis, Programa de Pós-Graduação em Informática, Setor de Ciências Exatas, Universidade Federal do Paraná, 2007. vi, 4, 5, 7, 8

[9] Hong L. and Jain A. Integrating faces and fingerprints for personal identification. *IEEE Transactions on Pattern and Machine Intelligence*, 20(12):1295–1307, dezembro 1998. 4, 6, 7, 8, 9

[10] E . C. Lopes. Detecção de faces e características faciais. Technical report, Pontifícia Universidade Católica do Rio Grande do Sul. 10, 11

[11] J. H. Saito M. Arantes, A. N. Ide. A system for fingerprint minutiae classification and recognition. In *Proceedings of the 9th International Conference on Neural Information Processing(ICONIP'O2)*, volume 5, pages 2474 – 2478. vii, 4, 5, 6

[12] N. Ahuja M. Yang, D. J. Kriegman. Detecting faces in images: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(1):34–58, janeiro 2002. 10, 11

[13] D. R. Oliveira. Reconhecimento de faces usando redes neurais e biometria. Master's thesis, São José dos Campos: Instituto Nacional de Pesquisas Espaciais (INPE), setembro 2003. 8, 9, 10

[14] M. Jones P. Viola. Robust real-time object detection. *Second International Workshop on Statistical and Computational Theories of Vision – Modeling, Learning, Computing, and Sampling*, julho 2001. vi, 11, 12, 13, 14

[15] A. Jain S. Pankanti, R. M. Bolle. Guest editors' introduction: Biometrics-the future of identification. *Computer*, 33:46–49, 2000. 1, 6

[16] M. Weiser. The world is not a desktop. *Interactions*, 1:7–8, Janeiro 1994. 1

[17] M. Weiser. The computer for the 21st century. *SIGMOBILE Mob. Comput. Commun. Rev.*, 3:3–11, Julho 1999. 1