

UNIVERSIDADE FEDERAL DO RIO DE JANEIRO  
INSTITUTO DE MATEMÁTICA  
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

ALEXANDRE TOMAN

**Reconhecimento de padrões ocultos em imagens via  
Otimização do Valor Ordenado**

Profa. Dra. Luziane Ferreira de Mendonça  
Orientadora

Rio de Janeiro, 2009.

T655 Toman, Alexandre

Reconhecimento de padrões ocultos em imagens  
via otimização do valor ordenado / Alexandre  
Toman - 2009.  
80 f.: il.

Dissertação (Mestrado em Informática) -  
Universidade Federal do Rio de Janeiro, Instituto de  
Matemática, Rio de Janeiro, 2009.

Orientador: Luziane Ferreira de Mendonça

I. de Mendonça, Luziane Ferreira (Orient.). II. Universidade  
Federal do Rio de Janeiro. Programa de Pós-Graduação em  
Informática. III. Reconhecimento de padrões ocultos em  
imagens via otimização do valor ordenado.

# Reconhecimento de padrões ocultos em imagens via Otimização do Valor Ordenado

Alexandre Toman

Dissertação de Mestrado submetida ao Corpo Docente do Departamento de Ciência da Computação do Instituto de Matemática, e Núcleo de Computação Eletrônica da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários para obtenção do título de Mestre em Informática.

Aprovada por:

---

Profa. Dra. Luziane Ferreira de Mendonça  
(Orientadora)  
PPGI - IM/UFRJ

---

Prof. Dr. Mauro Antonio Rincon  
PPGI - IM/UFRJ

---

Profa. Dra. Marcia Helena da Costa Fampa  
COPPE - UFRJ

Rio de Janeiro, agosto de 2009.

# Dedicatória

Aos alunos de baixa classe social e que apesar de todas as dificuldades, estudam com vontade de produção e competência, sabendo que os estudos, talvez, sejam sua única oportunidade sincera e honesta de inclusão social.

# Agradecimentos

Agradeço à orientadora Luziane Ferreira de Mendonça, pelas valiosas sugestões, correções e adaptações.

Agradeço também:

- Aos professores Haroldo Clark, Ivo Lopez, Marcello Goulart e Mauro Rincon pelas aulas que nos foram dadas durante o curso.
- Aos colegas de Mestrado: Cristiane, Mauro, Nilda, Rodolfo e Ronaldo.
- Aos grandes mestres da humanidade que de alguma forma contribuíram ou contribuem para o despertar da consciência do indivíduo.
- Aos livres pensadores de todo o mundo, que exercem suas atividades na construção de uma sociedade melhor.
- Aos meus pais e minha irmã, que apesar de todas as dificuldades que passamos durante a vida, nunca me disseram, por via direta ou subliminar, para abandonar os estudos.
- Aos funcionários da biblioteca do Instituto de Matemática da UFRJ por terem paciência comigo quando estava quase todos os dias devolvendo e renovando vários títulos que me eram emprestados.

# Resumo

TOMAN, Alexandre. **Reconhecimento de padrões ocultos em imagens via otimização do valor ordenado**. Rio de Janeiro, 2009. Dissertação (Mestrado em Informática)- Núcleo de Computação Eletrônica/Instituto de Matemática, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2009

Neste trabalho aplicamos a abordagem de otimização do valor ordenado (OVO) para o problema do reconhecimento facial. Essa nova abordagem permite que durante a estratégia de reconhecimento facial, o ajuste da imagem-teste (utilizando transformações geométricas de rotação e deslocamento) seja realizado levando em consideração o número de imagens de um mesmo indivíduo que estão presentes no banco. Foram utilizadas duas abordagens: na primeira usamos a abordagem OVO, e na segunda a abordagem OVO acoplado à técnica PCA (Análise dos Componentes Principais). O desempenho de ambas foi verificado por meio de testes numéricos para um banco de 24, 32 e 120 imagens.

Palavras-chave: Otimização do Valor Ordenado. Análise dos Componentes Principais. Padrões Ocultos.

# Abstract

TOMAN, Alexandre. **Reconhecimento de padrões ocultos em imagens via otimização do valor ordenado**. Rio de Janeiro, 2009. Dissertação (Mestrado em Informática)- Núcleo de Computação Eletrônica/Instituto de Matemática, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2009

In this work the Order Value Optimization (OVO) approach is applied for the face recognition problem. In this new approach, the image-test adjustments (by rotation and displacement transformations) is made taking into account the pictures number of the same person that are present in the image set. Two approaches are presented: OVO approach and OVO approach with PCA technique (Principal Component Analysis). The performance of both approaches was compared by numerical tests using a set of 24, 32, and 120 images.

Keywords: Order Value Optimization. Principal Component Analysis. Hidden Patterns.

# Sumário

Lista de Figuras	x
Lista de Tabelas	xi
<b>1 Reconhecimento Facial</b>	<b>12</b>
1.1 Histórico . . . . .	12
1.1.1 Descrição da área . . . . .	13
1.2 Sistemas de Reconhecimento . . . . .	15
1.2.1 Detecção da Faces . . . . .	16
1.2.2 Extração das Características e Reconhecimento . . . . .	19
1.3 Sobre a técnica PCA . . . . .	20
1.3.1 Um breve histórico sobre PCA . . . . .	20
1.4 A Abordagem deste trabalho . . . . .	22
<b>2 Abordagem Proposta</b>	<b>23</b>
2.1 Descrição . . . . .	23
2.2 Operadores de Ajuste . . . . .	24
2.2.1 Coordenadas Homogêneas . . . . .	24
2.2.2 Deslocamento . . . . .	26
2.2.3 Rotação . . . . .	27

2.2.4	Operador Geral . . . . .	27
2.3	Formulação do Problema . . . . .	28
2.4	Restrições de Caixa . . . . .	29
2.5	Aproximação por gradiente discreto . . . . .	30
2.5.1	Diferenças finitas . . . . .	30
2.5.2	Aproximação das derivadas parciais por diferenças finitas . . . . .	32
<b>3</b>	<b>Otimização do Valor Ordenado (OVO)</b>	<b>34</b>
3.1	Algoritmo local . . . . .	35
3.1.1	Algoritmo . . . . .	36
3.2	Resultados teóricos auxiliares . . . . .	38
3.3	Resultados de convergência e Aplicações . . . . .	41
<b>4</b>	<b>PCA - Análise dos Componentes Principais</b>	<b>44</b>
4.1	Definições básicas . . . . .	44
4.2	A Técnica PCA . . . . .	45
4.2.1	Componentes Principais . . . . .	48
4.2.2	SVD e Covariâncias . . . . .	50
4.3	A Distância de Mahalanobis . . . . .	52
4.3.1	Interpretação Geométrica . . . . .	54
4.4	Algoritmo PCA . . . . .	54
4.5	Resumo dos objetivos do PCA . . . . .	55
<b>5</b>	<b>Abordagem OVO acoplada à técnica PCA</b>	<b>56</b>
5.1	Motivação para unir o Algoritmo OVO à técnica PCA . . . . .	56
5.2	Detalhes sobre a abordagem OVO acoplado à rotina PCA . . . . .	57
5.3	Sobre os Experimentos Numéricos . . . . .	57

<b>6</b>	<b>Testes Numéricos</b>	<b>59</b>
6.1	Resultados com um banco de 24 imagens . . . . .	59
6.2	Resultados com um banco de 32 imagens . . . . .	62
6.3	Resultados com um banco de 120 imagens . . . . .	65
<b>7</b>	<b>Conclusões</b>	<b>70</b>
<b>A</b>	<b>O Método Potencial</b>	<b>72</b>
<b>B</b>	<b>O caso da Prosopagnosia</b>	<b>74</b>
	<b>Bibliografia</b>	<b>75</b>

# Lista de Figuras

1.1	Conjunto de características de não faces $\times$ faces. . . . .	14
2.1	Figura que esboça o novo sistema de coordenadas . . . . .	25
2.2	Figura que esboça a representação de pontos e vetores em coordenadas homogêneas. . . . .	25
2.3	Figura em sua posição original. . . . .	26
2.4	Figura após sofrer deslocamento para cima e para direita. . . . .	26
2.5	Rotação de 30 graus da figura . . . . .	27
2.6	Caixas bidimensional e tridimensional. . . . .	29
2.7	Esboço do gradiente projetado. . . . .	30
3.1	Exemplo de função OVO. Vermelho - $p=1$ ; Verde - $p=2$ . . . . .	34
3.2	Figuras que esboçam a não suavidade da função OVO. A função é não suave nos pontos indicados. Parte verde - $p=2$ ; Parte vermelha - $p=1$ . . . . .	35
3.3	Algoritmo OVO. . . . .	37
4.1	Exemplo em duas dimensões da utilização da técnica PCA para encontrar as direções principais de variação, a partir de amostras estatísticas. . . . .	48
4.2	A distribuição das medidas de altura e largura de um número de faces. . . . .	50
4.3	Pontos que têm a mesma distância euclidiana em relação ao centróide podem estar a uma distância relativa diferente do conjunto de treinamento. . . . .	52
4.4	Noção de distância ao conjunto de treinamento utilizando elipses. . . . .	54
5.1	Estrutura em caixa da rotina OVO acoplada à técnica PCA . . . . .	58

6.1	Imagem-teste: <i>teste1</i> . . . . .	59
6.2	Banco com 24 imagens - Imagens ordenadas por linha (de 1 a 24). . . . .	60
6.3	Imagem-teste: <i>teste2</i> . . . . .	62
6.4	Banco com 32 imagens - Imagens ordenadas por linha (de 1 a 32). . . . .	63
6.5	Imagens ordenadas por linha (de 1 a 120). . . . .	66
6.6	Imagem-teste: <i>teste3</i> . . . . .	67
B.1	Prosopagnosia. . . . .	74

# Lista de Tabelas

6.1	Rotina OVO - Imagem-teste: $teste1 - x_a = (1; 1; 0; NC/2; NL/2)$ - banco com 24 imagens - $p = 2$ . . . . .	61
6.2	Rotinas OVO e PCA acopladas - Imagem-teste: $teste1 - x_a = (1; 1; 0; NC/2; NL/2)$ - banco com 24 imagens - $p = 2$ . . . . .	61
6.3	Rotina OVO - Imagem-teste: $teste1 - x_b = (2; 5; -0.1; NC/2; NL/2)$ - banco com 24 imagens - $p = 2$ . . . . .	62
6.4	Rotina OVO acoplada ao PCA - Imagem-teste: $teste1 - x_b = (2; 5; -0.1; NC/2; NL/2)$ - banco com 24 imagens - $p = 2$ . . . . .	62
6.5	Rotina OVO - Imagem-teste: $teste2 - x_a = (1; 1; 0; NC/2; NL/2)$ - banco com 32 imagens - $p = 2$ . . . . .	64
6.6	Rotina OVO acoplada ao PCA - Imagem-teste: $teste2 - x_a = (1; 1; 0; NC/2; NL/2)$ - banco com 32 imagens - $p = 2$ . . . . .	64
6.7	Rotina OVO - Imagem-teste: $teste2 - x_b = (2; 5; -0.1; NC/2; NL/2)$ - banco com 32 imagens - $p = 2$ . . . . .	65
6.8	Rotina OVO acoplada ao PCA - Imagem-teste: $teste2 - x_b = (2; 5; -0.1; NC/2; NL/2)$ - banco com 32 imagens - $p = 2$ . . . . .	65
6.9	Rotina OVO - Imagem-teste: $teste3 - x_a = (1; 1; 0; NC/2; NL/2)$ - banco com 120 imagens - $p = 3$ . . . . .	67
6.10	Rotina OVO acoplada ao PCA - Imagem-teste: $teste3 - x_a = (1; 1; 0; NC/2; NL/2)$ - banco com 120 imagens - $p = 3$ . . . . .	68
6.11	Rotina OVO - Imagem-teste: $teste3 - x_b = (0.7; -0.4; 0.05; NC/2; NL/2)$ - banco com 120 imagens - $p = 3$ . . . . .	68
6.12	Rotina OVO acoplada ao PCA - Imagem-teste: $teste3 - x_b = (0.7; -0.4; 0.05; NC/2; NL/2)$ - banco com 120 imagens - $p = 3$ . . . . .	69

# Capítulo 1

## Reconhecimento Facial

### 1.1 Histórico

É notória a capacidade humana de realizar com sucesso o reconhecimento de uma face nas situações mais variadas<sup>1</sup>, mas a automatização desse processo é uma área em franca expansão. O reconhecimento facial é uma das várias vertentes da biometria, e esta tem a proposta de processar e reconhecer a identidade de uma pessoa a partir de suas características faciais. Nos últimos anos, têm sido desenvolvidos os mais importantes e bem sucedidos métodos de reconhecimento facial impulsionados pelo aperfeiçoamento dos programas e softwares ao longo dos últimos 30 anos, e pela demanda do comércio e de organizações de combate ao crime.

Aplicações novas e até futuristas já estão surgindo em diversas áreas, como por exemplo na área de educação, colaborando com o controle da frequência dos alunos e ajudando na diminuição de fraudes no ambiente escolar [57]. Como diversas secretarias de educação em várias partes do mundo pretendem implementar sistemas de reconhecimento facial nas escolas de sua rede de ensino (inclusive a Secretaria de Educação do Estado do Rio de Janeiro), os aperfeiçoamentos do sistema são essenciais para a segurança das pessoas, do ambiente escolar e conseqüentemente, na qualidade de ensino e aprendizado. Tais métodos também estão sendo aplicados no caso da educação à distância [19], facilitando o acesso de usuários e mantendo a segurança e a privacidade.

As pesquisas teóricas e computacionais sobre reconhecimento facial e temas correlatos estão presentes nas mais diversas áreas do conhecimento, tais como, reconhecimento de padrões, processamento de imagens, neurociência, psicologia, etc.

Os mais antigos trabalhos sobre reconhecimento facial foram elaborados nos anos 50 (em 1954 mais precisamente) na área de psicologia [18] e nos anos 60 na área de engenharia [15]. Entretanto, o “boom” sobre o assunto ocorreu nos anos 70 com a publicação dos artigos [37, 38], que versam, principalmente sobre identificação de pessoas através de computadores. Ainda nessa

---

<sup>1</sup>No Apêndice há a descrição de casos raros onde o cérebro humano não realiza o reconhecimento facial com sucesso.

década, algumas aplicações foram estudadas levando em consideração atributos mensuráveis e características (distância entre pontos) das faces ou caricaturas (biometria) [15].

Ao longo dos últimos 30 anos tal área prosseguiu em franca expansão e as técnicas desenvolvidas para a resolução do problema do reconhecimento facial se expandiram, devido as aplicações que cada vez mais eram visíveis. São inumeros os trabalhos sobre a determinação da identidade [8]; expressões faciais, movimento e sinais sociais [21]; fatores de estímulo [14]; interação de estímulo e fatores subjetivos [24]; debates sobre representação facial [13]; familiaridade e reconhecimento facial sobre inconsistências fotométricas [16]; etc.

Esses trabalhos geraram métodos que fazem o reconhecimento automaticamente para problemas como:

- localizar uma face<sup>2</sup> em meio a uma imagem dada ou em um video clip<sup>3</sup>.
- extração de características particulares (olhos, boca, orelhas, etc.)

### 1.1.1 Descrição da área

Um dos fatores de maior contribuição para o interesse nesta área foi o aumento de atividades criminais. A necessidade de se identificar pessoas de forma segura e eficaz é crescente. Uma das maneiras de se determinar essa identidade é o uso de suas características fisiológicas e comportamentais, as quais caracterizam a biometria (que é única de pessoa para pessoa e muito difícil de ser reproduzida). Geralmente, os sistemas biométricos incluem:

- reconhecimento facial;
- identificação por impressão digital;
- reconhecimento da voz;
- análise da íris;
- identificação pela geometria das mãos.

O diferencial do reconhecimento facial em relação aos demais é o fato de não haver necessidade da cooperação da pessoa para que sua face seja capturada e haja a tentativa de identificação (como o caso de uma criança desaparecida, por exemplo). Isto não é possível com os outros sistemas biométricos, como a análise da íris [51].

---

<sup>2</sup>A detecção de faces não deve ser confundida com o reconhecimento facial, cujo trabalho é descobrir a identidade de uma face, embora frequentemente ambos os sistemas sejam encontrados juntos.

<sup>3</sup>Um exemplo muito difundido dessa técnica nos dias atuais é o recurso denominado *face detection*, presente nas câmeras fotográficas digitais.

Entretanto realizar tal reconhecimento se torna um problema difícil graças às condições em que as faces são apresentadas (como o posicionamento de uma pessoa ou em uma imagem em que há pouco iluminação na região da face).

Ao se observar o conjunto de características no espaço da imagem, verifica-se que este é não linear e não convexo [12].

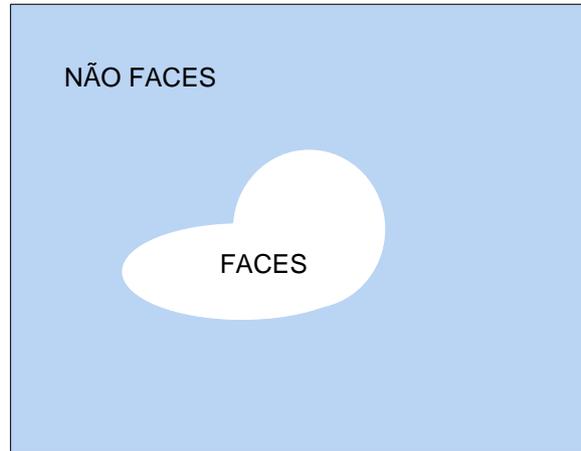


Figura 1.1: Conjunto de características de não faces  $\times$  faces.

Alguns estudos sobre reconhecimento facial indicam que a performance de vários métodos é extremamente prejudicada quando se muda a luminosidade do ambiente, a postura, e outros fatores [20]. Os itens a seguir descrevem os principais problemas encontrados:

- **Larga variabilidade na aparência facial**- A aparência da face está sempre sujeita a uma grande número de fatores, incluindo a pose da face (ou equivalentemente, o ponto de vista da câmera), iluminação, expressão facial, etc. Todos esses fatores são confundidos no banco de dados das imagens. Esta variabilidade traz dificuldades na extração intrínseca de informações.
- **A não linearidade das variedades são altamente complexas** - Em geral, são utilizados métodos que projetam os dados de um espaço de alta dimensão (o espaço de imagens) para um subespaço linear de menor dimensão. Estes métodos não preservam, em geral, as variações não convexas das variedades faciais necessárias para diferenciar indivíduos. Em um subespaço linear, a distância euclidiana e mais especificamente, a distância de Mahalanobis (distância a ser definida no Capítulo 4) [54], que são as normalmente utilizadas para a modelagem, não funcionam bem entre variedades de faces e não-faces e entre variedades de indivíduos. Este fato crucial limita o poder dos métodos lineares para atingir alta exatidão na detecção facial e reconhecimento.

- **Alta dimensionalidade e pequeno tamanho da amostragem** - Outro desafio é a habilidade para generalizar. Uma imagem canônica de  $112 \times 92$  reside em um espaço de características de dimensão 10304. Todavia, o número de exemplos por pessoa (tipicamente inferior a 10, até mesmo uma) avaliável para a apreensão da variedade é usualmente muito menor que a dimensão do espaço da imagem.

Existem duas estratégias para lidar com as dificuldades citadas: extração de características e classificação de padrões. A primeira tem o propósito de construir um bom espaço de características para que o espaço de variedades venha a ficar mais simples, isto é, menos não linear e não convexo. Isto inclui dois níveis do processo:

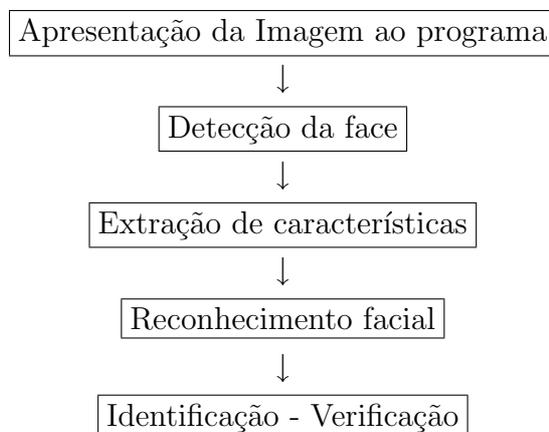
- Normalizar imagens faciais geometricamente e fotometricamente;
- Extrair características nas imagens normalizadas de forma estável com respeito às suas variações.

A segunda estratégia consiste em construir classificações hábeis para resolver as dificuldades em problemas de classificação e regressão não lineares no espaço de características criado. Cabe ressaltar que embora uma boa normalização e extração de características reduza a não linearidade e não convexidade, elas não resolvem os problemas completamente de forma a produzir classificações hábeis para lidar com tais dificuldades. Um algoritmo bem sucedido, em geral, combina ambas as estratégias.

## 1.2 Sistemas de Reconhecimento

O sistema de reconhecimento facial consiste de uma série de componentes, incluindo detecção facial, rastreamento, alinhamento, extração de características, e reconhecimento.

O esquema abaixo mostra as etapas do processo de reconhecimento facial:



A primeira parte do diagrama trata apenas da apresentação de uma imagem qualquer ao programa, podendo esta imagem ser de uma face ou não. A segunda parte do diagrama (detecção facial) é a primeira etapa significativa no processo de reconhecimento facial, responsável por selecionar a seção da imagem que contem a face. A terceira fase é a extração de características. Muitos sistemas de reconhecimento facial precisam da adição das características para uma análise holística, como sugerem vários estudos da psicologia. Como já se sabe, por exemplo, *eigenfaces*<sup>4</sup> [55] e *Fisherfaces* [10], precisam de várias características (olhos, nariz, boca) para uma detecção normal. A quarta fase se refere ao reconhecimento em si. A quinta fase é apenas a verificação e associação a alguma imagem pré-existente no banco. Isto é feito, em geral, por meio da utilização de uma função específica de distância entre a imagem-teste e as imagens do banco.

### 1.2.1 Detecção da Faces

Existem vários trabalhos sobre detecção facial, usando diversas técnicas desde a simples detecção de cores de pele até métodos neurais de procura por padrões na imagem. Há dois tipos básicos de detecção de faces, os métodos de detecção facial por características, onde existe uma preocupação em encontrar formas, cores ou detalhes próximos de uma face humana, estudadas *a priori*, e os métodos baseados na imagem propriamente dita, que faz uso de técnicas de reconhecimento de padrões como as redes neurais ou métodos probabilísticos.

#### Métodos baseados em características faciais

Métodos de detecção baseados em características de imagem foram os primeiros métodos de detecção facial. Estes tipos de métodos necessitam de informações sobre a face, normalmente conseguidas por observação e estudo das características típicas da face humana como cor, forma, proporção, posição ou movimento. Estão incluídos nestes métodos a detecção por bordas, por características colorimétricas, por movimento ou modelos de face.

- Detecção por bordas

Foi usada pela primeira vez na detecção facial, em um dos mais antigos trabalhos da área [59], o qual utiliza operadores de extração de bordas como o *Sobel* ou *Laplacian* [36], e busca comparar o resultado com padrões previamente conhecidos de face. Um dos padrões usados para validação é a clássica razão áurea [22], fixando a proporção entre a altura e a largura da face.

Entre os trabalhos que usam detecção facial por bordas estão [27, 31], ambos comparando as bordas retiradas da imagem original com padrões antropométricos, ou seja, padrões comuns da face humana estudados na medicina, como os mostrados em [22].

---

<sup>4</sup>*Eigenfaces* é uma técnica para redução de dimensionalidade e generalização. Esta normaliza e seleciona uma região da face em várias imagens e trata cada sub-imagem da face como um vetor de intensidade e encontra momentos dos vetores em uma base de dados.

Uma outra possibilidade é uma procura por densidade de bordas, onde são verificadas as regiões que possuem mais pontos de bordas. A região dos olhos normalmente é cercada por várias áreas que possuem muitas variações dos valores de luminância. Áreas mais escuras como sombrancelhas, cílios e pupilas, estão relativamente próximas de áreas mais claras como a esclerótica, a íris e as pálpebras. Estas variações de luminância podem ser detectadas como bordas, constituindo um modelo simples para encontrar a região da face.

- Detecção por características colorimétricas

Usar a informação de cores para detectar a face também é bastante comum como mostrado em [39].

Muitos sistemas, como o descrito em [40], executam a detecção facial por características colorimétricas de forma rápida e eficiente. Claramente, considerando que não há restrição de fundo ou iluminação nas figuras utilizadas, a detecção da pele poderá encontrar algumas informações falsas, que no caso de [40] são retirados por uma análise de formas feita para assegurar que apenas as regiões com forma elíptica sejam aproveitadas.

Ressalta-se que as informações de cor de uma imagem não são normalmente utilizadas como único método de detecção facial, mas sim como apoio a outros métodos, como forma de reduzir a região de procura pela face. Esta mesma idéia, ou seja, diminuir o espaço de busca, também é aplicada na detecção por bordas.

- Detecção usando movimento

Detectar a face pelo movimento pode ser possível se arquivos de vídeo estão disponíveis. Em imagens frontais, movimentos da boca podem ser usados para encontrar regiões candidatas a face.

Também em imagens frontais, a constante repetição do piscar dos olhos pode ser usada para encontrar regiões candidatas a face.

Assim como a detecção por características colorimétricas a detecção usando movimentos pode também ser usada para encontrar regiões candidatas a face, e evitar assim que o sistema precise procurar faces em todas as regiões dos novos *frames* capturados por uma câmera. Técnicas de detecção e movimento podem ser encontrados em [36].

- Detecção por modelos faciais

Características antropométricas da face são estudadas em diversos campos da medicina. Algumas destas características (olhos, orelhas, boca, queixo, espaço entre as sombrancelhas, etc.) podem ser usadas para simplificar a detecção facial.

Exemplos de detecção por modelos faciais podem ser encontrados em [9]. Estes utilizam a posição dos olhos e boca como referência de medida para encontrar a posição da face, principalmente em figuras que possuem apenas uma pessoa. Em [63] os olhos são inicialmente detectados usando detecção por densidade de bordas, e a seguir um padrão de face é ajustado comparativamente usando uma função de custo proporcional ao número de pontos de borda entre o modelo e a figura original.

## Métodos baseados na imagem

Métodos de detecção baseados na imagem são mais recentes que os baseados em características, e buscam encontrar padrões de face com técnicas probabilísticas, redes neurais e outros. Tendem a apresentar melhores resultados mas são computacionalmente mais complexos. A maioria dos trabalhos atuais se encaixa neste tipo de detecção.

- Métodos de detecção facial usando grafos

A detecção de face usando grafos foi originalmente proposta em [65], e esta consiste em criar um grafo elástico cujos nós estão em pontos que apresentam particularidades da face como o contorno da boca, as bordas dos olhos, o centro do queixo, etc.

Em [65], propõe-se que os vértices do grafo sejam encontrados em uma nova face através do cálculo de uma similaridade destes com um banco de dados previamente montado com outras figuras de faces.

Inicialmente, o grafo padronizado é ajustado como um todo, ou seja, todos os seus nós são movidos ao mesmo tempo, buscando o maior somatório. Este ajuste pode ser feito usando um algoritmo de busca como o gradiente descendente.

Em um segundo momento, os grafos podem ter seus nós movidos independentemente, possibilitando um ajuste mais fino, buscando aumentar a similaridade. Ao final do processo, a face estará detectada em conjunto com a localização de diversas particularidades, que podem fornecer informações para um algoritmo de reconhecimento de identidade.

Entretando o algoritmo descrito em [65] pressupõe que o fundo não contém outros objetos que poderiam causar perturbações no posicionamento dos grafos apresentados. Além disso, não se adapta para o caso onde possam existir  $N$  faces na figura, sendo melhor indicado para captura de imagens com fundo controlado, para aplicação direta de sistemas de reconhecimento de identidade.

- Métodos de detecção facial usando redes neurais

Redes neurais artificiais constituem um conhecido método de reconhecimento de padrões, sendo sua aplicação em detecção de faces avaliada principalmente em [58].

O método de detecção usando redes neurais tem por base um banco de dados de pequenas imagens com faces recortadas e centralizadas cuidadosamente para que a posição dos olhos, nariz e boca variem muito pouco entre as imagens.

Dessa forma, uma rede neural artificial com dois neurônios de saída recebe as imagens, e é treinada para permitir ativação de um dos neurônios quando figuras de face são apresentadas, ou seja, figuras que tem a disposição típica de face, com olhos nariz e boca com posições adequadas. Um banco de dados composto de ruído e figuras geométricas aleatórias que não representam faces é apresentado à rede de forma a ativar o outro neurônio, indicando que não se trata de uma face.

## 1.2.2 Extração das Características e Reconhecimento

Uma vez detectada a face, o processo de reconhecimento facial segue para o próximo passo, utilizando técnicas para extrair as características da imagem para posterior reconhecimento.

Na abordagem geométrica usada recentemente em [17], características faciais como olhos, nariz, boca e queixo são detectados. Propriedades e relações (isto é, áreas, distâncias, ângulos) entre as características são usadas como parâmetro para o reconhecimento facial. As vantagens desta abordagem incluem economia e eficiência quando se realiza a redução de dados e a intensidade de variação na iluminação e posicionamento. Entretanto, a detecção das características faciais e as técnicas de mensuração desenvolvidas até agora não são muito confiáveis para o reconhecimento das características baseado em um modelo geométrico, e tais propriedades geométricas, quando aplicadas isoladamente, são inadequadas para o reconhecimento imediato pois o excesso de informações contido na textura ou na aparência facial é rejeitado.

A abordagem por via da estatística coleta os dados de um determinado conjunto (as imagens ou um trecho delas) e extrai as principais características para o processo de detecção. Durante a abordagem, o conhecimento prévio sobre as faces e as variações consideradas no conjunto de dados em questão são tomados em consideração.

Os métodos baseados na aparência, tal como o PCA (Principal Component Analysis) [64] e LDA (Linear Discriminant Analysis) [10], deram avanços significativos nas técnicas de reconhecimento facial. Usando PCA, um subespaço da face é construído para representar “otimamente” apenas o objeto facial; usando LDA, o subespaço discriminante é construído para diferenciar as “faces ótimas” das diferentes pessoas. Tais métodos tem se mostrado muito eficazes no problema do reconhecimento facial e são de simples aplicação e desenvolvimento.

Embora estes métodos ajudem bastante, eles não são capazes de “trabalhar” no campo de variedades original do espaço de imagens. Isso ocorre pelo fato de não lidar bem com a não linearidade do reconhecimento facial.

O PCA e o LDA podem ser estendidos usando as técnicas de Kernel ([61] para o PCA e [46] para o LDA) para lidar com a não linearidade no reconhecimento facial [29]. Existe uma técnica de projeção não linear (redução de dimensão) do espaço de imagens para um espaço de características; as variedades que surgem no espaço de características são simples, ainda com as sutilezas preservadas.

Outra aproximação que lida com a não linearidade é construir uma base de aparências locais no espaço de características, usando filtros de imagens apropriados, então as distribuições das faces são menos afetadas pelas várias mudanças. A análise local de características (LFA)[49], os filtros de Gabor [41] e o padrão local binário (LBP)[6] têm sido usados com este propósito. Alguns desses algoritmos podem ser considerados como uma combinação geométrica (ou estrutural) de detecção e da extração de características da aparência local, para acrescentar certa estabilidade ao reconhecimento quanto às variações em pontos de vista, iluminação, e expressão. Tais métodos são bastante complexos de serem aplicados e combinados.

## 1.3 Sobre a técnica PCA

A técnica PCA será utilizada como um melhoramento da abordagem proposta. Este método será acoplado à abordagem OVO na tentativa de aperfeiçoamento e melhoria do processo. No Capítulo 4 deste trabalho, essa importante técnica será melhor detalhada.

A idéia principal da técnica PCA (Principal Component Analysis)[23] é reduzir a dimensão de um conjunto grande de dados interrelacionados, enquanto se conserva ao máximo possível a variação dos dados dentro deste conjunto. Este procedimento é realizado por meio da transformação do conjunto original de dados em um novo conjunto de variáveis, as componentes principais (PC's, ou em imagens, *eigenfaces*), que tem baixa correlação e são ordenadas em ordem de preferência, segundo as amostras que apresentam maior significância variacional levando em consideração o conjunto de amostras original.

As técnicas de análise de subespaços para reconhecimento facial são baseadas no fato que uma classe de padrões de interesse, tal como são as faces, “residem” em um subespaço do conjunto de imagens total. Por exemplo, um pequena imagem de  $64 \times 64$  possui 4096 pixels que podem expressar um grande número de classe de padrões, tal como árvores, casas e faces. De todo modo, entre os  $256^{4096} > 10^{9864}$  possíveis “configurações”, apenas algumas correspondem a faces. Então, a representação da imagem original é amplamente reduzida, e a dimensionalidade desta representação pode também ser bastante reduzida, pois apenas o padrão da face é o que interessa.

Na abordagem [55], um pequeno número (igual a 40 ou menos) de *eigenfaces* derivam de um conjunto da imagem das faces usando a técnica PCA. Uma imagem é representada eficientemente como um vetor de características (um vetor de “pesos”) de pequena dimensão. As características em cada subespaço provém mais aparentes e ricas de informação para o reconhecimento que a imagem pura. O uso de técnicas da modelagem do subespaço possui avanço significativo na tecnologia de reconhecimento facial.

A técnica PCA tem grande aplicabilidade e encontra-se em franco estudo. Isto está claramente exemplificado pelo fato de que o *site* de busca de trabalhos científicos *Web of Science* identifica cerca de 2000 artigos publicados apenas em dois anos (2007-2008) que incluem a expressão “Principal Component Analysis” em seus títulos, resumos ou palavras-chave. Livros ou artigos que utilizam o PCA estão sendo publicados nas mais diversas áreas: agricultura, demografia, ecologia, economia, genética, geologia, metereologia, oceanografia, psicologia e controle de qualidade.

### 1.3.1 Um breve histórico sobre PCA

As mais remotas descrições da técnica conhecida hoje como PCA foram dadas por Pearson (1901) [48] e Hotteling (1933) [32].

Os artigos de Pearson e Hotelling citados anteriormente, adotam abordagens distintas. O artigo de Hotelling deu mais ênfase à parte algébrica. Já no artigo de Pearson encontra-se uma

abordagem de ordem um pouco mais geométrica, que consiste em encontrar planos e retas que mais se aproximem de um conjunto de pontos, em um espaço de dimensão  $p$ .

Os comentários de Pearson levam em consideração aspectos computacionais e colocam o fato de seu método ser facilmente aplicado em problemas numéricos, e embora ele tenha dito que os cálculos sejam volumosos para 4 ou mais variáveis, ele sugere que tais cálculos são bastante viáveis. Nos 32 anos que separam as publicações de Pearson e Hotelling, vários materiais sobre o assunto foram publicados.

A motivação de Hotelling é que é o menor conjunto de variáveis independentes que determina o “valor” das  $p$  variáveis originais. Ele observa que variáveis semelhantes tem sido denominadas “fatores” na literatura de psicologia, mas introduz o termo alternativo “componentes” na tentativa de evitar a confusão com outros usos da palavra “fatores” dentro da matemática.

Hotelling escolhe estas “componentes” com o intuito de maximizar as sucessivas contribuições para o total de variância das variáveis originais, e denomina as componentes que apareceram ao longo deste desenvolvimento como “componentes principais” (PC’s).

O rumo que os PC’s desenvolvidos por Hotelling tomam é similar àquele em que se precisa utilizar multiplicadores de Lagrange, e é finalizado com o problema de autovetores/autovalores, mas difere deste em três aspectos. Como o primeiro, ele trabalha com uma correlação, tanto quanto com covariância; o segundo é que ele analisa o conjunto de variáveis originais expressas como funções lineares das componentes tal como as componentes expressas em termos das variáveis originais; e a terceira, ele não usa a notação matricial.

Hotelling segue mostrando como encontrar as componentes usando o método potencial (ver Apêndice (A)). A técnica PCA não é apresentada em sua forma padrão, mas segundo a abordagem de “análise de fatores”.

Em [25], Girshick realizou grandes avanços sobre o método da potência para encontrar PC’s. Ao surgir o interesse e a expansão da técnica PCA (fomentado pelo desenvolvimento dos computadores), alguns trabalhos se destacaram. Estes são:

- **Anderson (1963) [1]** - Este é um trabalho teórico que discute sobre a distribuição assintótica de amostras dos coeficientes e sobre discordâncias das amostras de PC’s.
- **Rao (1964) [56]** - Este apresenta novas idéias e aplicações, interpretações e extensões do PCA.
- **Gower (1966) [28]** - Faz paralelos do PCA com outras técnicas de estatística, o que trouxe para o desenvolvimento da teoria importantes *insights* do ponto de vista geométrico.
- **Jeffers (1967) [35]** - Este artigo promoveu um grande avanço na parte prática e numérica do PCA.

## 1.4 A Abordagem deste trabalho

O objetivo desta dissertação é propor uma nova abordagem para o processo de extração das características e reconhecimento de uma imagem facial. Para tanto, serão determinados via processo de otimização os parâmetros ótimos que permitem o melhor ajuste da imagem-teste em relação à cada imagem do banco.

O ajuste da imagem-teste deve ser realizado levando em consideração que existe um número  $p$  de imagens mais importantes (aquelas referentes ao mesmo indivíduo) do que o banco de imagens completo. Assim, definida uma função que mede a diferença entre duas imagens, pode-se levar em consideração (tomar como base para ajuste) as  $p$  imagens mais semelhantes à imagem-teste.

Para realizar tal ajuste seguindo essas idéias, faremos uso de um método bastante específico de minimização que resolve a abordagem de Otimização do Valor Ordenado (OVO - Order Value Optimization). A função a ser minimizada será a função diferença no nível  $p$ , após a extração das características principais do banco de imagens via PCA.

No Capítulo 2, será apresentada uma descrição detalhada do problema abordado (bem como de sua formulação). Em seguida, o Capítulo 3 explana sobre a Otimização do Valor Ordenado, o método numérico para a sua resolução e os seus resultados teóricos de convergência. No Capítulo 4 é apresentada a técnica PCA e seus principais resultados. No Capítulo 5 apresenta-se o esquema OVO acoplado ao PCA. O Capítulo 6 possui os testes numéricos e a análise comparativa das duas abordagens propostas. Por fim, no Capítulo 7 apresentamos as conclusões.

# Capítulo 2

## Abordagem Proposta

Neste capítulo descrevemos a abordagem proposta para o problema de reconhecimento facial. Como veremos a seguir, tal abordagem é otimizada por métodos de programação não linear, recebendo o nome de abordagem OVO.

O uso da abordagem OVO já foi estudado no caso da localização de padrões ocultos em uma nuvem de pontos [60], mas não há conhecimento de sua aplicação em reconhecimento facial na literatura, o qual consiste em determinar se uma imagem é compatível com alguma outra previamente armazenada em um banco, utilizando para ajuste deslocamentos e rotações.

### 2.1 Descrição

O problema que estamos interessados em resolver consiste em: dada uma imagem-teste (por exemplo, a foto de um funcionário), deveremos identificar se esta pertence a algum indivíduo cuja foto está em um banco de imagens ou não. Para tanto, é necessário definir como será realizado ajuste entre a imagem-teste e as imagens do banco.

Para auxiliar na comparação entre as imagens, serão permitidos pequenos ajustes para correção da posição da imagem-teste em relação às imagens do banco, que podem ter sido capturadas em posições diferentes. Será usada a composição de dois tipos de funções:

- rotação - que irá depender de um ângulo  $-\frac{\pi}{2} \leq \theta \leq \frac{\pi}{2}$ , e de um ponto  $P_0$  em torno do qual a rotação será realizada;
- deslocamento - que pode ser relativo ao eixo  $x$ , representado por  $d_x$  (positivo se o movimento é para a direita, e negativo se o movimento é para a esquerda), e/ou ao eixo  $y$ , representado por  $d_y$  (positivo se o movimento é para cima, e negativo se o movimento é para baixo).

Após o procedimento de leitura, cada imagem é armazenada em formato matricial, onde as

componentes da matriz serão denotadas por *pixels*<sup>1</sup>. As imagens utilizadas em nossos testes estão em preto e branco e, portanto, cada *pixel* representa a intensidade da cor em escala de cinza na posição indicada, mensurada por números no intervalo [0 256].

## 2.2 Operadores de Ajuste

Os ajustes realizados na imagem-teste são realizados por meio do deslocamento e rotação da imagem, representado em coordenadas homogêneas.

### 2.2.1 Coordenadas Homogêneas

Ao representarmos um ponto e um vetor em coordenadas tridimensionais, teremos uma confusão entre os dois, já que não existe diferenciação em termos de notação.

O uso das coordenadas homogêneas [11] auxiliará na criação do operador geral. Em coordenadas homogêneas, pontos de um espaço afim de dimensão 3 e de vetores do espaço vetorial são representados univocamente por elementos do  $\mathbb{R}^4$ .

Em um espaço afim, um marco é definido por um ponto, chamado origem e pelos vetores de uma base do espaço vetorial associado. No caso de dimensão 2, o marco é representado por:  $(v_1, v_2, P_0)^T$ , onde  $P_0 = 1$  e  $v_1$  e  $v_2$  são as componentes. Caso estejamos trabalhando com um vetor,  $P_0 = 0$ . Assim podemos diferenciar facilmente pontos de vetores.

Como utilizaremos *pixels*, faremos uma adaptação da representação em forma de marco, inserindo a intensidade da cor do *pixel* como a quarta componente, alterando assim sua representação para o  $\mathbb{R}^4$ . A idéia de coordenadas homogêneas para pontos é semelhante a esta que será adaptada para *pixels*, pois tal adaptação será feita apenas para que as transformações geométricas não apresentem dificuldades computacionais (permite que o operador geral seja escrito como uma multiplicação de matrizes. Ver Seções (2.2.2) e (2.2.3)).

Será denotado por  $N$  o número de *pixels* na vertical e  $M$  o número de *pixels* na horizontal. Para cada *pixel* situado na linha  $i$  e coluna  $j$  da matriz, com intensidade  $v_{ij}$ , será associado um vetor do  $\mathbb{R}^4$  a fim de que o mesmo seja convertido em um vetor com coordenadas homogêneas (a origem dos eixos coordenados estará localizada no vértice superior esquerdo da imagem-teste). Tal vetor conterà:

- primeira componente:  $j$ ;
- segunda componente:  $i$ ;

---

<sup>1</sup>Um *pixel* é o menor ponto que forma uma imagem digital, sendo que o conjunto de milhares de *pixels* formam a imagem inteira.

- terceira componente: 1 (notação em coordenadas homogêneas);
- quarta componente:  $v_{ij}$  (intensidade da cor do *pixel*).

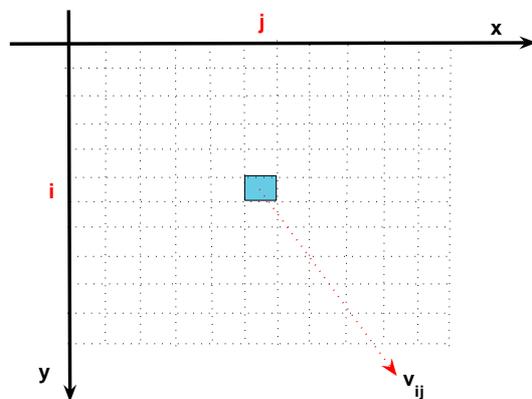


Figura 2.1: Figura que esboça o novo sistema de coordenadas

Na figura abaixo,  $p_1$  e  $p_2$  são representações dos pontos  $(1, 1)$  e  $(11, 11)$ , enquanto  $P_1$  e  $P_2$  são as representações dos vetores  $(1, 1)$  e  $(11, 11)$ .

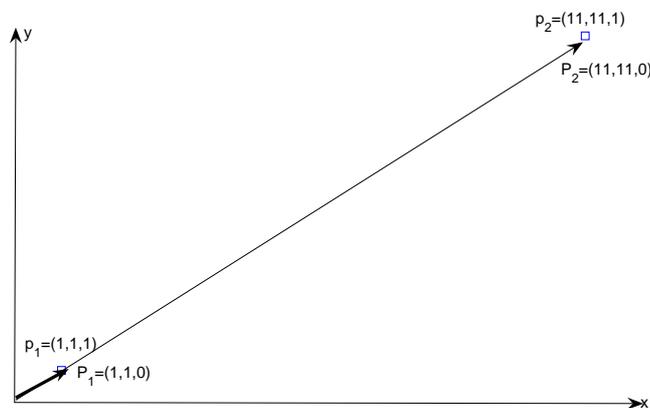


Figura 2.2: Figura que esboça a representação de pontos e vetores em coordenadas homogêneas.

Portanto, o *pixel* que se situa na linha  $i$  e coluna  $j$  da matriz original será representado por:

$$P_{(i-1)M+j} = \begin{pmatrix} j \\ i \\ 1 \\ v_{ij} \end{pmatrix}.$$

Uma vez determinada essa notação, cada imagem pode ser armazenada como uma sequência ordenada de 4 componentes, que podem ser agrupados em uma matriz  $4 \times NM$ . Para realizarmos ajustes à imagem-teste, aplicaremos as funções de ajuste a cada *pixel*, simultaneamente através de uma multiplicação de uma matriz, que representa o operador utilizado, com a matriz  $P$ :

$$P = [P_1 | P_2 | \dots | P_{NM}].$$

### 2.2.2 Deslocamento

O deslocamento de cada *pixel*  $P_i$  será realizado por meio de uma multiplicação à esquerda por uma matriz  $4 \times 4$  que denotaremos  $D$ . Sendo  $d_x$  o deslocamento de tal *pixel* em relação ao eixo  $x$  e  $d_y$  o deslocamento da imagem em relação ao eixo  $y$ , a matriz  $D$  será representada da seguinte forma:

$$D = \begin{pmatrix} 1 & 0 & d_x & 0 \\ 0 & 1 & d_y & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

O deslocamento inverso pode ser obtido fazendo o produto da matriz inversa de  $D$  por cada *pixel*:

$$D^{-1} = \begin{pmatrix} 1 & 0 & -d_x & 0 \\ 0 & 1 & -d_y & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

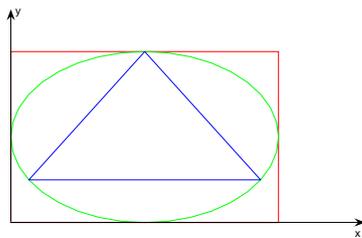


Figura 2.3: Figura em sua posição original.

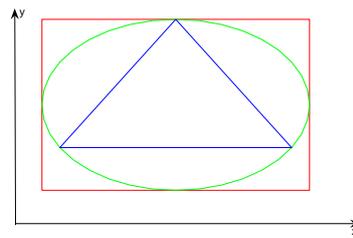


Figura 2.4: Figura após sofrer deslocamento para cima e para direita.

### 2.2.3 Rotação

Utilizaremos para a rotação de um *pixel*  $P_i$  uma matriz  $4 \times 4$  que denotaremos por  $R_\theta$  (rotação de um ângulo  $\theta$ , onde  $-\frac{\pi}{2} \leq \theta \leq \frac{\pi}{2}$ ). Tal rotação de um ângulo  $\theta$  permite que o rosto da pessoa que foi fotografada (imagem teste), pode estar rotacionado até  $\frac{\pi}{2}rad$  em sentido horário ou anti-horário.

A matriz  $R_\theta$  é uma extensão da matriz de Givens que representa a rotação em torno da origem tem a seguinte forma:

$$R_\theta = (r_{ij})_{4 \times 4} = \begin{pmatrix} \cos(\theta) & -\sin(\theta) & 0 & 0 \\ \sin(\theta) & \cos(\theta) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Para uma rotação em torno de um *pixel* qualquer  $P_{x_0, y_0} = (x_0, y_0, 1, v_{x_0, y_0})^T$  da imagem-teste, deve-se primeiro realizar o deslocamento de todos os pontos, de tal forma que  $P_{x_0, y_0}$  seja transportado para a origem; após esse deslocamento, pode-se aplicar a rotação utilizando a matriz  $R_\theta$ , e por fim realiza-se o deslocamento inverso, devolvendo  $P_{x_0, y_0}$  a sua posição original.

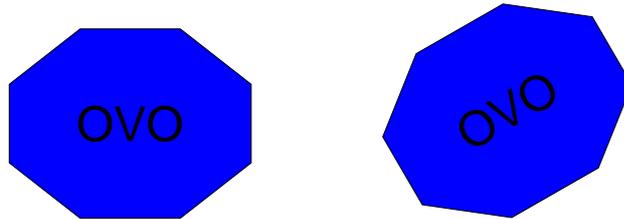


Figura 2.5: Rotação de 30 graus da figura

Esse procedimento será matematicamente realizado diretamente pela multiplicação:

$$D_0^{-1} R_\theta D_0 [P_1 | P_2 | \dots | P_{NM}],$$

onde  $D_0$  é o operador deslocamento para o *pixel*  $P_{x_0, y_0}$ .

### 2.2.4 Operador Geral

Então o operador geral  $F$  que opera sobre cada *pixel*, dependerá de 5 variáveis:  $d_x$ ,  $d_y$ ,  $\theta$ ,  $x_0$  e  $y_0$  e será a composição do operador deslocamento com o operador rotação:

$$F(d_x, d_y, \theta, x_0, y_0) = D * (D_0^{-1} * R_\theta * D_0).$$

Fazendo os cálculos, encontramos para tal operador a seguinte matriz  $4 \times 4$  :

$$F(d_x, d_y, \theta, x_0, y_0) = \begin{pmatrix} \cos(\theta) & \sin(\theta) & -x_0 \cos(\theta) + y_0 \sin(\theta) + x_0 + d_x & 0 \\ \sin(\theta) & \cos(\theta) & -x_0 \sin(\theta) - y_0 \cos(\theta) + y_0 + d_y & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

## 2.3 Formulação do Problema

O problema a ser resolvido será um problema de minimização com restrições da seguinte forma: minimizar a distância<sup>2</sup> da imagem-teste ao nível  $p^3$ , onde  $p$  é o número de fotos de uma mesma pessoa que compõe o banco de imagens (por exemplo, se existem 4 fotos do funcionário “João”,  $p = 4$ ). Formalizando:

$$\begin{aligned} \min \quad & f(x) = f_{i_p(x)}(x) \\ \text{s.a.} \quad & \left\{ \begin{array}{l} -\frac{M}{4} \leq x(1) \leq \frac{M}{4} \\ -\frac{N}{4} \leq x(2) \leq \frac{N}{4} \\ -\frac{\pi}{6} \leq x(3) \leq \frac{\pi}{6} \\ -\frac{M}{4} \leq x(4) \leq \frac{M}{4} \\ -\frac{N}{4} \leq x(5) \leq \frac{N}{4} \end{array} \right. \end{aligned} \quad (2.1)$$

A função objetivo tem o formato de uma função do tipo OVO, que possui algoritmo específico, e esta será melhor explicada no próximo capítulo.

As variáveis que compõem o vetor  $x = (x(1), x(2), x(3), x(4), x(5))^T \in \mathbb{R}^5$  são

$$(d_x, d_y, \theta, x_0, y_0).$$

<sup>2</sup>É necessário aferir a distância da imagem-teste a cada uma das imagens do banco; após calculadas, as  $p$  menores distâncias estão associadas às  $p$  imagens mais semelhantes à imagem-teste.

<sup>3</sup>Dado um conjunto de  $m$  funções e  $p \leq m$ , tendo fixado um ponto do domínio  $z_0$ , deseja-se obter a  $p$ -ésima maior função aplicada neste  $z_0$  fixo. Utilizaremos a notação  $f(z_0) = f_{i_p(z_0)}(z_0)$  para tal procedimento.

As restrições sobre as variáveis são todas lineares, tratando-se de um problema de programação com restrições de caixa. Esse tipo de restrição será melhor comentado na próxima seção. Tais limitantes para as variáveis foram determinados de forma a permitir uma variação máxima para o ângulo de rotação da face ( $\pi/6$ ), assim como dos deslocamentos horizontais e verticais de  $1/4$  da largura e da altura da imagem.

## 2.4 Restrições de Caixa

Nesta seção comentaremos as propriedades que as restrições de caixa possuem, uma vez que o Problema (2.1) a ser estudado possui apenas esse tipo de restrição, e portanto o método utilizado em sua resolução deverá ser adaptado para esse tipo de restrição.

Vamos considerar o problema:

$$\begin{aligned} \min f(x) \\ \text{s.a. } l \leq x \leq u \end{aligned}$$

onde  $f : \mathbb{R}^n \rightarrow \mathbb{R}$ . O conjunto  $\Omega \in \mathbb{R}^n$  formado pelos pontos que satisfazem essas restrições é denominado uma caixa (ou hipercaixa) do  $\mathbb{R}^n$ .

Os vetores  $l$  e  $u$  são os limites superiores e inferiores, respectivamente, das variáveis. Para as variáveis  $x_i$  que são ilimitadas, admitimos os valores  $-\infty$  para  $l_i$  e  $+\infty$  para  $u_i$ .

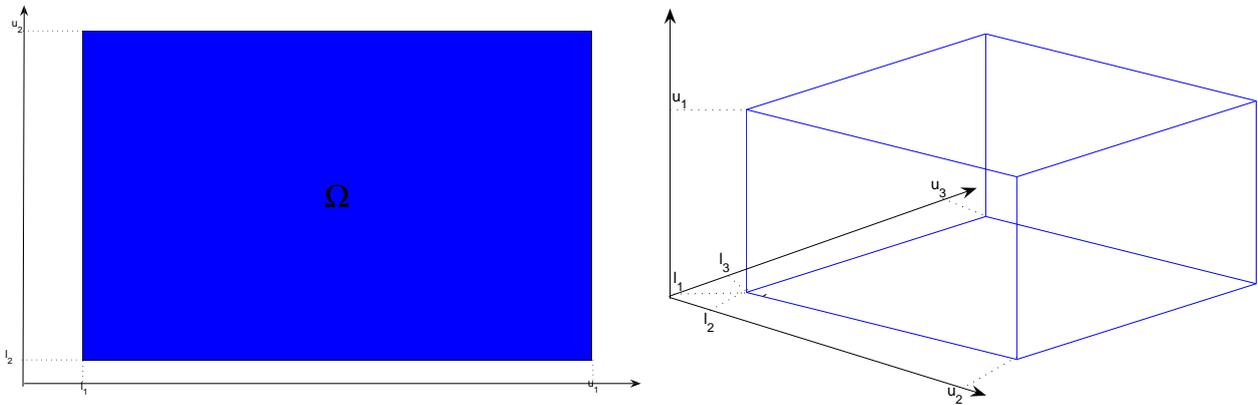


Figura 2.6: Caixas bidimensional e tridimensional.

Durante a implementação do algoritmo, a estrutura de caixa será preservada utilizando a norma infinito para os vetores. Para tanto, utilizaremos um adaptação do método do gradiente projetado. No método do gradiente projetado, usamos a caixa em todas as etapas de um determinado algoritmo, assegurando que os novos pontos gerados também estarão dentro da caixa.

Teremos certeza que os pontos  $x_k$  não sairão da caixa se o vetor que representa o passo a ser dado está dentro da caixa. Caso ele não estiver, devemos projetá-lo na caixa.

Durante o desenvolvimento do algoritmo e o critério de parada do algoritmo, é utilizado o vetor gradiente projetado. Por exemplo, sendo  $g_j$  a notação para o gradiente projetado,  $g$  o gradiente e tendo uma caixa no  $R^2$ ; se as coordenadas dos vértices dessa caixa são:  $(u_1, u_2), (u_1, l_2), (l_1, u_2), (l_1, l_2)$ ; ou seja,  $u = (u_1, u_2)$ ,  $l = (l_1, l_2)$  e  $g = (g_1, g_2)$  podemos deduzir facilmente que:

$$g_j = \begin{pmatrix} \min \{u_1, \max \{g_1, l_1\}\} \\ \min \{u_2, \max \{g_2, l_2\}\} \end{pmatrix}.$$

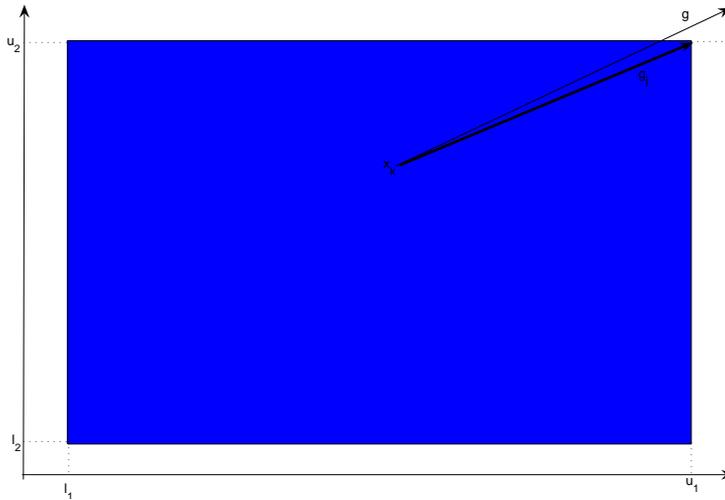


Figura 2.7: Esboço do gradiente projetado.

Tal idéia pode ser estendida para o caso de  $n$  mais elevado, inclusive em  $R^n$  com  $n = 5$ .

## 2.5 Aproximação por gradiente discreto

Durante nossas implementações numéricas, utilizaremos a aproximação por diferenças finitas para estimar as derivadas parciais da função objetivo.

### 2.5.1 Diferenças finitas

A idéia geral do método de diferenças finitas é a discretização do domínio e a substituição das derivadas por aproximações que envolvem apenas valores numéricos da função. Na prática, substituímos as derivadas pela razão incremental que converge para o valor da derivada quando o incremento tende a zero. Dizemos então que o problema foi discretizado. Quando o domínio

tem mais de uma variável, o método é semelhante, bastando aplicar a mesma idéia para cada uma das variáveis separadamente.

Por exemplo, considere uma função  $u : \mathbb{R}^2 \rightarrow \mathbb{R}$ . Seja  $x_0$  um número real e  $h$ , um número positivo. Definimos malha na direção  $x$  (mantendo  $y$  fixo) de passo  $h$  associada a  $x_0$  como o conjunto de pontos:

$$x_i = x_0 \pm ih, i = 1, 2, \dots, N.$$

Nos pontos desta malha serão calculadas aproximações de uma função  $u(x, y)$  e suas derivadas em relação a  $x$ .

A ferramenta matemática básica no cálculo de aproximações para as derivadas é a fórmula de Taylor. Então a série de Taylor para  $u(x, y_0)$  com respeito ao ponto  $(x_0, y_0)$  pode ser escrita como:

$$u(x, y_0) = u(x_0, y_0) + (x - x_0)u_x(x_0, y_0) + \frac{(x - x_0)^2 u_{xx}(\epsilon, y_0)}{2},$$

em que  $\epsilon$  se encontra compreendido entre  $x$  e  $x_0$ . Se  $x = x_0 + h$ , obtém-se depois do desenvolvimento algébrico necessário,

$$u_x(x_0, y_0) - \frac{u(x_0 + h, y_0) - u(x_0, y_0)}{h} = \frac{-h}{2} u_{xx}(\epsilon, y_0).$$

Em outras palavras, se aproxima  $u_x$  pelo quociente

$$u_x(x_0, y_0) = \frac{u(x_0 + h, y_0) - u(x_0, y_0)}{h} \tag{2.2}$$

e o erro de truncamento é dado por

$$E_T = \frac{-h}{2} u_{xx}(\epsilon, y_0).$$

A Equação (2.2) é obtida substituindo  $x = x_0 + h$  na série de Taylor. O resultado se chama *diferença adiantada*. Podemos obter uma aproximação diferente, chamada *diferença atrasada*, fazendo  $x = x_0 - h$ . O resultado será aproximadamente:

$$u_x(x_0, y_0) = \frac{u(x_0, y_0) - u(x_0 - h, y_0)}{h} \tag{2.3}$$

e seu erro de truncamento é igual a

$$E_T = \frac{h}{2} u_{xx}(\epsilon, y_0),$$

onde  $\epsilon$  também se encontra compreendido entre  $x$  e  $x_0$ .

Existe também a *diferença centrada* [62] e este será aproximado por

$$u_x(x_0, y_0) = \frac{u(x_0 + h, y_0) - u(x_0 - h, y_0)}{2h} \quad (2.4)$$

e seu erro de truncamento é igual a

$$\frac{h^2}{3!} u_{xxx}(\epsilon, y_0),$$

onde  $\epsilon \in (x_0 - h, x_0 + h)$ .

Procedimento similar pode ser feito para a obtenção das derivadas de  $u$  em relação a  $y$ , por meio do cálculo de  $u$  em pontos da malha para  $y$ .

## 2.5.2 Aproximação das derivadas parciais por diferenças finitas

Nesta subseção, será utilizada a técnica de diferenças finitas para aproximar o gradiente da função de ajuste na implementação do método numérico.

A função objetivo do problema principal deste trabalho tem o formato

$$f_j(\theta, x_0, y_0, d_x, d_y) = \sum_{s=1}^N \sum_{r=1}^M (I_t(x_r, y_s) - I_j(r, s))^2, \text{ com } 1 \leq x_r \leq M \text{ e } 1 \leq y_r \leq N$$

onde  $I_t$  representa a função de intensidade da cor do *pixel* da imagem-teste,  $I_j$  é a intensidade de coloração do *pixel* da  $j$ -ésima imagem do banco, na posição  $(r, s)$ , e

$$x_r(r, s) = x_r = r \cos \theta + s \sin \theta - x_0 \cos \theta + y_0 \sin \theta + x_0 + d_x,$$

$$y_s(r, s) = y_s = r \sin \theta + s \cos \theta - x_0 \sin \theta - y_0 \cos \theta + y_0 + d_y.$$

Então para alguma variável  $w$  das 5 variáveis possíveis, tem-se, pela regra da cadeia:

$$\frac{\partial f_j}{\partial w} = 2 \sum_{s=1}^N \sum_{r=1}^M \left[ (I_t(x_r, y_s) - I_j(r, s)) \left( \frac{\partial I_t}{\partial x_r} \cdot \frac{\partial x_r}{\partial w} + \frac{\partial I_t}{\partial y_s} \cdot \frac{\partial y_s}{\partial w} \right) \right],$$

onde

- $\frac{\partial x_j}{\partial \theta} = -j \sin \theta + i \cos \theta + x_0 \sin \theta + y_0 \cos \theta$ ;
- $\frac{\partial x_j}{\partial x_0} = -\cos \theta + 1$ ;

- $\frac{\partial x_j}{\partial y_0} = \sin \theta;$
- $\frac{\partial x_j}{\partial d_x} = 1;$
- $\frac{\partial x_j}{\partial d_y} = 0;$
- $\frac{\partial y_i}{\partial \theta} = j \cos \theta - i \sin \theta - x_0 \cos \theta + y_0 \sin \theta;$
- $\frac{\partial y_i}{\partial x_0} = -\sin \theta;$
- $\frac{\partial y_i}{\partial y_0} = -\cos \theta + 1;$
- $\frac{\partial y_i}{\partial d_x} = 0;$
- $\frac{\partial y_i}{\partial d_y} = 1.$

O espaçamento em nosso problema pode ser considerado é igual a 1 ( $h = 1$ ), uma vez que trata-se do reposicionamento de pixels. Assim as aproximações das derivadas parciais por diferenças finitas serão:

- $\frac{\partial I_t}{\partial x_r} \simeq \frac{I_t(x_r+1, y_s) - I_t(x_r-1, y_s)}{2},$  se  $2 \leq x_r \leq M - 1;$
- $\frac{\partial I_t}{\partial x_r} \simeq I_t(x_r + 1, y_s) - I_t(x_r, y_s),$  se  $x_r = 1;$
- $\frac{\partial I_t}{\partial x_r} \simeq I_t(x_r, y_s) - I_t(x_r - 1, y_s),$  se  $x_r = M;$
- $\frac{\partial I_t}{\partial y_s} \simeq \frac{I_t(x_r, y_s+1) - I_t(x_r, y_s-1)}{2},$  se  $2 \leq y_s \leq N - 1;$
- $\frac{\partial I_t}{\partial y_s} \simeq I_t(x_r, y_s + 1) - I_t(x_r, y_s),$  se  $y_s = 1;$
- $\frac{\partial I_t}{\partial y_s} \simeq I_t(x_r, y_s) - I_t(x_r, y_s - 1),$  se  $y_s = N.$

É utilizada a diferença centrada (devido ao menor erro), exceto nos contornos onde é utilizada diferenças atrasada (contorno superior) e adiantada (contorno inferior).

O Problema (2.1) deve ser minimizado pela abordagem OVO, pois primeiro ordenaremos as funções para depois minimizá-las. No próximo capítulo apresentaremos os principais resultados desta abordagem para aplicarmos ao problema do reconhecimento facial.

# Capítulo 3

## Otimização do Valor Ordenado (OVO)

No capítulo anterior vimos que o Problema (2.1) a ser resolvido corresponde à minimização do valor da função  $f$ , que se situa a  $p$ -ésima posição após ordenação, ou seja, dado  $x_0$  pertencente ao domínio, deseja-se obter  $f(x_0) = f_{i_p(x_0)}(x_0)$ .

A abordagem OVO possui diversas aplicações, mas ainda não foi aplicada para o problema de reconhecimento facial. Tanto a teoria quanto as aplicações da abordagem OVO são bem recentes (seu surgimento data de 2003).

**Definição 3.0.1.** Dado  $m$  funções contínuas  $f_1, \dots, f_m$ , definidas em  $\Omega \subset \mathbb{R}^n$  e um inteiro  $p \in \{1, \dots, m\}$ , a função  $f$  do Valor Ordenado de ordem  $p$  é dada por

$$f(x) = f_{i_p(x)}(x),$$

para todo  $x \in \Omega$ , onde

$$f_{i_1(x)}(x) \leq \dots \leq f_{i_m(x)}(x).$$

Se  $p = 1$ , temos que  $f(x) = \min\{f_1(x), \dots, f_m(x)\}$ , enquanto que para  $p = m$  tem-se  $f(x) = \max\{f_1(x), \dots, f_m(x)\}$ .

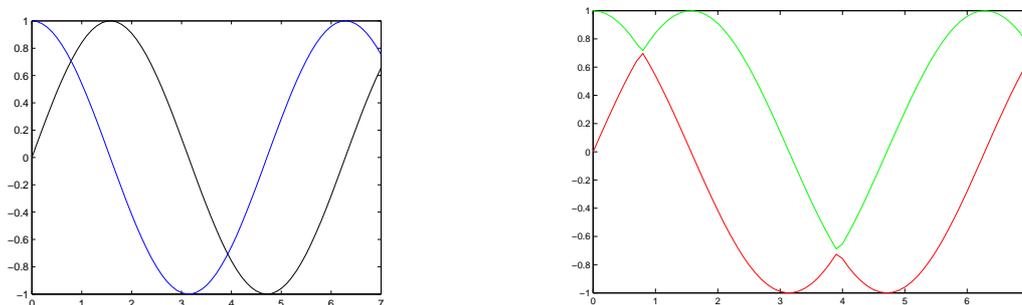


Figura 3.1: Exemplo de função OVO. Vermelho -  $p=1$ ; Verde -  $p=2$ .

Independentemente do valor de  $p$ , a função OVO é contínua. Entretanto, mesmo se as funções  $f_i$  são diferenciáveis, a função OVO é não suave, e portanto sua minimização requer algoritmos especialmente elaborados para esse fim.

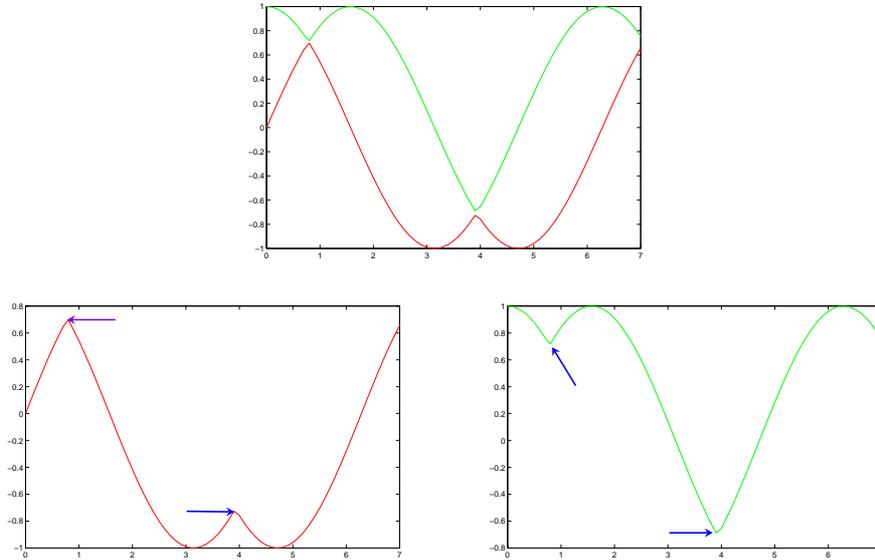


Figura 3.2: Figuras que esboçam a não suavidade da função OVO. A função é não suave nos pontos indicados. Parte verde -  $p=2$ ; Parte vermelha -  $p=1$ .

### 3.1 Algoritmo local

O único algoritmo proposto para resolução da abordagem OVO, até o presente momento, foi proposto por Roberto Andreani, Cibele Dunder e José Mario Martínez [2]. Em seu trabalho, os autores descrevem um método do tipo Cauchy, o qual resolve o problema no sentido em que todo ponto limite satisfaz uma condição de otimalidade adequada. Neste mesmo artigo observou-se que a cada iteração do método de Cauchy, as funções  $f_i$  são aproximadas por funções lineares.

Suponha que  $\Omega \subset \mathbb{R}^n$  é fechado e convexo, e  $f_1, \dots, f_m$  têm derivadas parciais contínuas em um conjunto aberto que contém  $\Omega$ . Denotaremos  $g_j = \nabla f_j$  a partir de agora,  $\Omega$  é o conjunto de pontos viáveis. Para todo  $x, y \in \Omega$ ,  $j = 1, \dots, m$  suponhamos que existam  $c, L \in \mathbb{R}$  tais que

$$\|g_j(x)\|_\infty \leq c$$

e

$$\|g_j(y) - g_j(x)\|_\infty \leq L\|y - x\|_\infty.$$

Consequentemente, para todo  $x, y \in \Omega$ ,  $j = 1, \dots, m$ , pelas observações da próxima seção

$$|f_j(y) - f_j(x)| \leq c\|y - x\|_\infty$$

e

$$f_j(y) \leq f_j(x) + g_j(x)^T(y - x) + \frac{L}{2}\|y - x\|_\infty^2.$$

Antes de escrever o algoritmo, define-se, para todo  $\varepsilon \geq 0$ ,  $x \in \Omega$  :

$$I_\varepsilon(x) = \{j \in \{1, \dots, m\} | f(x) - \varepsilon \leq f_j(x) \leq f(x) + \varepsilon\}.$$

$I_\varepsilon(x)$  é o conjunto de índices  $\{j \in \{1, \dots, m\}\}$ , nos quais a função OVO está a uma distância de no máximo  $\varepsilon$  da  $j$ -ésima função aplicada no mesmo ponto.

### 3.1.1 Algoritmo

Dado  $\epsilon > 0$ , seja  $I_\epsilon(x) = \{j \in \{1, \dots, m\} | f(x) - \epsilon \leq f_j(x) \leq f(x) + \epsilon\}$ . Sendo  $g_j(x_k) = \nabla f_j(x_k)$  e  $x_0 \in \Omega$  um ponto inicial arbitrário. Sejam  $\Theta \in (0, 1)$ ,  $\Delta > 0$ ,  $\varepsilon > 0$ ,  $0 < \sigma_{\min} < \sigma_{\max} < 1$ ,  $\eta \in (0, 1]$ . Seja  $M_k(d_k) = \max_{j \in I_\varepsilon(x_k)} \{g_j(x_k)^T d\}$ . Dado  $x_k \in \Omega$ , os passos da  $k$ -ésima iteração são:

**Algoritmo 3.1.1.** .

**Passo 1** - (*Resolvendo o subproblema*)

*Resolva o subproblema*

$$\min M_k(d) \quad \text{s.a. } x_k + d \in \Omega, \|d_k\| \leq \Delta \quad (3.1)$$

*ou o problema de otimização convexo equivalente*

$$\begin{aligned} \min \quad & w \\ \text{s.a.} \quad & g_j(x_k)^T d \leq w, \quad \forall j \in I_\varepsilon(x_k) \\ & x_k + d \in \Omega, \end{aligned} \quad (3.2)$$

e denote por  $\bar{d}_k$  uma solução de (3.1).

Obtenha  $d_k$  tal que  $x_k + d_k \in \Omega$ ,  $\|d_k\| \leq \Delta$  e

$$M_k(d_k) \leq \eta M_k(\bar{d}_k).$$

Se  $M_k(d_k) = 0$  pare.

**Passo 2** - (Calculo do tamanho do passo)

Tome  $\alpha \leftarrow 1$ . Se

$$f(x_k + \alpha d_k) \leq f(x_k) + \theta \alpha M_k(d_k) \quad (3.3)$$

tome  $\alpha_k = \alpha$ ,  $x_{k+1} = x_k + \alpha_k d_k$  e acabe a iteração. Caso contrário, escolha  $\alpha_{new} \in [\sigma_{\min} \alpha, \sigma_{\max} \alpha]$ , faça  $\alpha \leftarrow \alpha_{new}$  e repita o Teste (3.3).

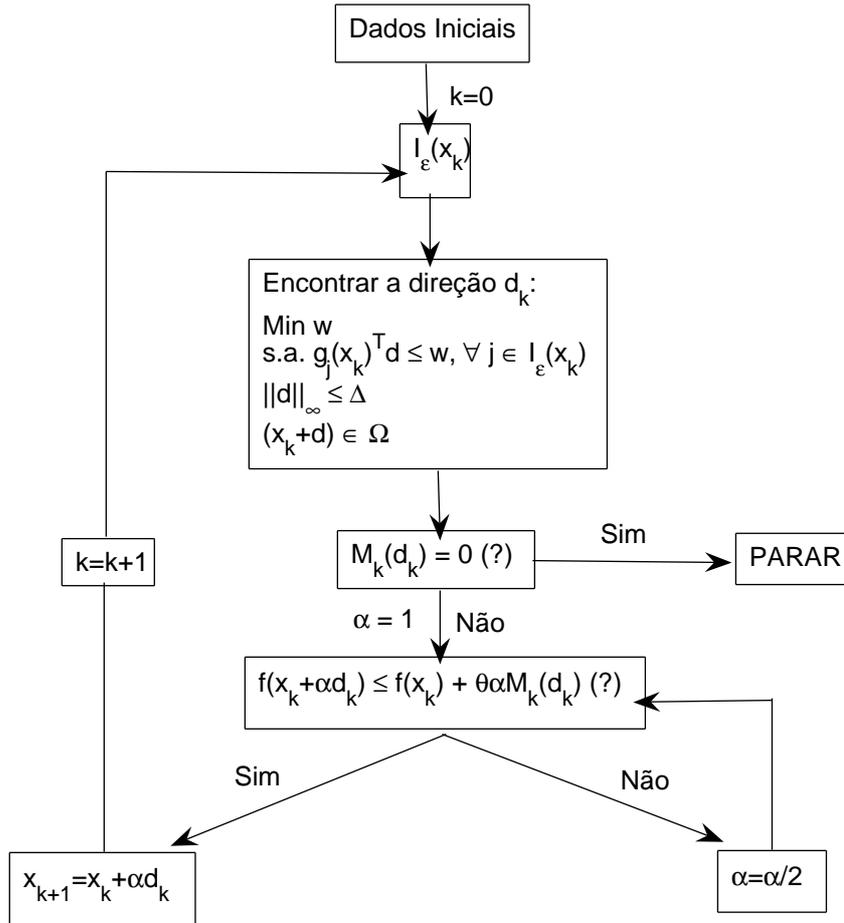


Figura 3.3: Algoritmo OVO.

**Observação 3.1.1.** Para todo  $d_k$ , temos que  $M_k(d_k) \leq 0$ .

Suponha por absurdo que a afirmação seja falsa e seja  $M_k(d'_k)$  é o primeiro dos termos desse algoritmo com a propriedade

$$M_k(d'_k) > 0$$

e lembrando que  $0 < \eta \leq 1$ , e  $M_k(\bar{d}_k) \leq 0$ , tem-se que  $\eta M_k(\bar{d}_k) \leq 0$ . Logo

$$0 < M_k(d'_k) \leq \eta M_k(\bar{d}_k) \leq 0$$

o que é absurdo. Logo, a afirmação é verdadeira.

**Observação 3.1.2.** Se  $M_k(d_k)$  for negativo, a direção continua sendo de descida, fazendo com que o algoritmo prossiga. Então, para que o algoritmo pare, deve-se ter

$$M_k(d_k) \geq 0.$$

Como  $M_k(d_k) \leq 0$ , o algoritmo deve parar exatamente quando  $M_k(d_k) = 0$ .

**Observação 3.1.3.** O Passo 2 do Algoritmo acima é uma busca inspirada na regra de Armijo. A diferença entre estas duas é que no lado direito da desigualdade (3.3), temos a parcela  $\theta\alpha M_k(d_k)$ , ao invés de  $\theta\alpha g_j(x_k)^T d$ , já que devemos levar em consideração todas as funções  $f_j$  na vizinhança do ponto.

## 3.2 Resultados teóricos auxiliares

Os resultados apresentados nesta seção servem de base para a demonstração do teorema de convergência do Algoritmo 3.1.1.

**Teorema 3.2.1.** Seja  $U \subset \mathbb{R}^n$  aberto. Se  $f : U \rightarrow \mathbb{R}$  é tal que  $\nabla f(x)$  é limitado por  $c$  real, então  $f$  é contínua.

*Demonstração.* Da hipótese em questão, tem-se que  $c \geq \left| \frac{\partial f}{\partial x_i}(x) \right|$  para todo  $x \in U$  e todo  $i = 1, 2, \dots, n$ . Dados  $x, x + v \in U$ , sendo  $v = (\alpha_1, \dots, \alpha_n)$ , define-se  $v_0, v_1, \dots, v_n \in \mathbb{R}^n$  colocando  $v_0 = 0$  e  $v_i = v_{i-1} + \alpha_i e_i$  para  $i = 1, \dots, n$ , de modo que  $v_n = v$ . Escrevendo  $f(x + v) - f(x)$  como uma soma telescópica, tem-se:

$$f(x + v) - f(x) = \sum_{i=1}^n f(x + v_i) - f(x + v_{i-1}).$$

Pelo Teorema do Valor Médio unidimensional, e tomando a soma

$$\left| \sum_{i=1}^n f(x + v_i) - f(x + v_{i-1}) \right|_S \leq \sum_{i=1}^n |f(x + v_i) - f(x + v_{i-1})|_S = \sum_{i=1}^n \left| \frac{\partial f}{\partial x_i}(z) \alpha_i e_i \right|_S,$$

onde

$$z \in [v_{i-1}, v_i].$$

Então

$$|f(x + v) - f(x)|_S \leq c \sum_{i=1}^n |\alpha_i e_i|_S = c \sum_{i=1}^n |\alpha_i|_S = c |v|_S$$

e assim  $f$  é lipschitziana, e conseqüentemente, contínua. ■

**Lema 3.2.1.** *Seja  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  uma função com derivadas parciais contínuas em um conjunto convexo aberto  $U \subset \mathbb{R}^n$ . Então, para  $x \in U$  e uma perturbação não nula  $\rho \in \mathbb{R}^n$ , a derivada direcional de  $f$  em  $x$  na direção de  $\rho$ , definida por*

$$\frac{\partial f}{\partial \rho} \equiv \lim_{\epsilon \rightarrow 0} \frac{f(x + \epsilon \rho) - f(x)}{\epsilon},$$

*existe e é igual a  $\nabla f(x)^T \rho$ . Para qualquer  $x, x + \rho \in U$ ,*

$$f(x + \rho) = f(x) + \int_0^1 \nabla f(x + t\rho)^T \rho dt.$$

*Demonstração.* Parametriza-se  $f$  pela reta que liga  $x$  e  $(x + \rho)$  como uma função de uma variável

$$g : \mathbb{R} \rightarrow \mathbb{R}, g(t) = f(x + t\rho).$$

Defina  $x(t) = x + t\rho$ . Então pela regra da cadeia, para  $0 \leq \alpha \leq 1$ ,

$$\begin{aligned} \frac{dg}{dt}(\alpha) &= \sum_{i=1}^n \frac{\partial f(x(t))}{\partial x(t)_i}(x(\alpha)) \frac{dx(t)_i}{dt}(\alpha) \\ &= \sum_{i=1}^n \frac{\partial f}{\partial x_i}(x(\alpha)) \cdot \rho_i \\ &= \nabla f(x + \alpha\rho)^T \rho. \end{aligned} \tag{3.4}$$

Substituindo  $\alpha = 0$  em (3.4) se reduz a

$$\frac{dg}{dt}(0) = \frac{\partial f}{\partial \rho}(x) = \nabla f(x)^T \rho.$$

Pelo Teorema Fundamental do Cálculo,

$$g(1) = g(0) + \int_0^1 g'(t) dt$$

que, pela definição de  $g$  e (3.4) é equivalente a

$$f(x + \rho) = f(x) + \int_0^1 \nabla f(x + t\rho)^T \rho dt.$$

■

**Teorema 3.2.2.** *Seja  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  uma função com derivadas parciais contínuas em um aberto convexo  $U \subset \mathbb{R}^n$ ,  $x \in U$ , e seja  $\|\nabla f(y) - \nabla f(x)\| \leq L\|y - x\|$  em uma vizinhança de  $U$ . Então, para qualquer  $x + \rho \in U$ ,*

$$\|f(x + \rho) - f(x) - \nabla f(x)^T \rho\| \leq \frac{L}{2} \|\rho\|^2. \quad (3.5)$$

*Demonstração.* Pelo lema anterior,

$$\begin{aligned} f(x + \rho) - f(x) - \nabla f(x)^T \rho &= \left[ \int_0^1 \nabla f(x + t\rho)^T \rho dt \right] - \nabla f(x)^T \rho \\ &= \int_0^1 \left[ \nabla f(x + t\rho)^T - \nabla f(x)^T \right] \rho dt. \end{aligned}$$

Tomando a norma

$$\begin{aligned} \|f(x + \rho) - f(x) - \nabla f(x)^T \rho\| &\leq \int_0^1 \|\nabla f(x + t\rho)^T - \nabla f(x)^T\| \cdot \|\rho\| dt \leq \\ &\leq \int_0^1 L \|t\rho\| \cdot \|\rho\| dt = L \|\rho\|^2 \int_0^1 t dt = \frac{L}{2} \|\rho\|^2. \end{aligned}$$

■

**Observação 3.2.1.** *Para  $x + \rho = y$ , temos especificamente que que*

$$\|f(y) - f(x) - \nabla f(x)^T (y - x)\| \leq \frac{L}{2} \|y - x\|^2,$$

*ou seja,*

$$-\frac{L}{2} \|y - x\|^2 \leq f(y) - f(x) - \nabla f(x)^T (y - x) \leq \frac{L}{2} \|y - x\|^2$$

*ou simplesmente*

$$f(y) \leq f(x) + \nabla f(x)^T (y - x) + \frac{L}{2} \|y - x\|^2.$$

**Teorema 3.2.3.** *A função OVO de ordem  $p$  é contínua.*

*Demonstração.* Ver [2].

■

### 3.3 Resultados de convergência e Aplicações

A abordagem OVO surgiu em 2003, com a publicação de um artigo oriundo das pesquisas de tese de Doutorado de C. Dunder [2]. Em seus trabalhos foram apresentados a abordagem OVO, a sua função e também foram determinadas algumas de suas propriedades (um algoritmo e propriedades para a convergência deste). Tais resultados estão apresentados a seguir.

**Definição 3.3.1.** Dizemos que  $x$  é  $\epsilon$ -ótimo se

$$A \equiv \{d \in \mathbb{R}^n | x + d \in \Omega \text{ e } g_j(x)^T d < 0, \forall j \in I_\epsilon(x)\} = \emptyset.$$

Isto significa que um  $x$  é  $\epsilon$ -ótimo quando não existe direção de descida para função alguma  $f_j \in I_\epsilon(x)$ .

**Teorema 3.3.1.** Suponha que  $x_k \in \Omega$  é o  $k$ -ésimo iterado do Algoritmo 3.1.1. Então:

- (a) O algoritmo pára em  $x_k$  se, e somente se,  $x_k$  é  $\epsilon$ -ótimo.  
 (b) Se o algoritmo não pára em  $x_k$ , então a iteração está bem definida e

$$\alpha_k \geq \min \left\{ \frac{2\sigma_{\min}\gamma_k(1-\theta)}{L\Delta^2}, \frac{\epsilon\sigma_{\min}}{3c\Delta} \right\}$$

onde

$$\gamma_k = - \max_{j \in I_\epsilon(x_k)} \{g_j(x_k)^T d\}.$$

*Demonstração.* Ver [2]. ■

**Teorema 3.3.2.** Suponha que  $x_* \in \Omega$  é um ponto limite de uma sequência gerada pelo Algoritmo 3.1.1. Então  $x_*$  é  $\epsilon$ -ótimo.

*Demonstração.* Ver [2]. ■

Em outro artigo dos mesmo autores [3], foi dada atenção particular para a relação entre minimizadores locais e pontos estacionários de ambas as abordagens (a abordagem OVO e um problema de programação matemática com restrições que é elaborado no *paper*) e também sobre as condições de otimalidade da abordagem OVO. Tais resultados seguem abaixo.

**Definição 3.3.2.** Seja  $\Omega \subset \mathbb{R}^n$  e  $x \in \Omega$ . A abordagem OVO considerada é

$$\text{Min } f(x) \quad \text{s.a. } x \in \Omega \tag{3.6}$$

**Lema 3.3.1.** Assuma que  $z_1, \dots, z_m$  são números reais tais que

$$z_1 \leq z_2 \leq \dots \leq z_m.$$

Então, para todo  $p \in \{1, \dots, m\}$  existem  $r', u', w' \in \mathbb{R}^m$  tais que  $(r', u', w', z_p)$  é solução do seguinte problema

$$\begin{aligned} & \min z \\ & \text{s.a.} \left\{ \begin{array}{l} \sum_{i=1}^m r_i w_i = 0 \\ \sum_{i=1}^m (1 - r_i) u_i = 0 \\ \sum_{i=1}^m r_i = p \\ u_i - z + z_i - w_i = 0, \quad i = 1, \dots, m \\ u \geq 0, 0 \leq r \leq e, w \geq 0. \end{array} \right. \end{aligned} \quad (3.7)$$

*Demonstração.* Ver [3]. ■

**Teorema 3.3.3.** *O ponto  $x \in \Omega$  é uma solução do problema OVO (3.6) se, e somente se, existem  $r', u', w' \in \mathbb{R}^m$  e  $z' \in \mathbb{R}$  tal que  $(r', u', w', z')$  é uma solução de*

$$\begin{aligned} & \min z \\ & \text{s.a.} \left\{ \begin{array}{l} \sum_{i=1}^m r_i w_i = 0 \\ \sum_{i=1}^m (1 - r_i) u_i = 0 \\ \sum_{i=1}^m r_i = p \\ u_i - z + f_i(x) - w_i = 0, \quad i = 1, \dots, m \\ u \geq 0, 0 \leq r \leq e, w \geq 0, x \in \Omega. \end{array} \right. \end{aligned} \quad (3.8)$$

Neste caso,  $z' = f(x)$ .

*Demonstração.* Ver [3]. ■

**Teorema 3.3.4.** *Assuma que  $\Omega = \mathbb{R}^n$  e  $(r, u, w, z_*)$  é um ponto KKT<sup>1</sup> de (3.7). Então  $z_* = f(x^*)$ .*

*Demonstração.* Ver [3]. ■

Nas referências [2, 3, 4, 5] foram desenvolvidas algumas aplicações do problema OVO. Em sua tese de doutorado [60], Salvatierra apresentou algumas aplicações específicas da abordagem OVO:

- A primeira teve como objetivo modelar o problema de padrões ocultos com a função OVO. Dado um conjunto de dados no plano  $(t_1, y_1), \dots, (t_m, y_m)$ , com alguns deles representando um padrão (uma função). O objetivo era encontrar uma expressão para este padrão. Por exemplo, poderia ser o caso de existir uma função  $\phi$  tal que para alguns valores de  $i$  tem-se  $y_i = \phi(t_i, x)$ . Neste caso a abordagem OVO será definir funções erro como sendo  $f_i(x) = (\phi(t_i, x) - y_i)^2$ , e minimizar a função OVO de ordem  $p$ , para vários valores de  $p$ .
- O segundo foi uma generalização do algoritmo dado em [2] para resolver a abordagem OVO, na qual se conseguiu uma convergência local, e aplicou-se ao modelo de Sistema Bancário Ideal (SBI), no qual foi identificado a função OVO com a função Valor-no-Risco (Var). Neste foi introduzido um algoritmo novo do tipo Quasi-Newton para problemas de Otimização do Valor Ordenado com restrições convexas. Este novo algoritmo pode ser usado para a otimização do Valor-no-Risco, especialmente quando as funções de perda são não lineares e as restrições são lineares.

No próximo capítulo apresentaremos os resultados necessários para o desenvolvimento da rotina PCA, para que depois, no Capítulo 5, desenvolvamos a acoplagem da abordagem OVO com a técnica PCA.

---

<sup>1</sup>Para ver a definição de Ponto KKT deve-se consultar [34].

# Capítulo 4

## PCA - Análise dos Componentes Principais

A técnica PCA é conhecida como Transformada de Karhunen-Loève, e é uma técnica que provém da estatística (área de análise multivariada) que é bastante utilizada para detectar, em um conjunto de amostras multidimensionais, os subespaços lineares mais significativos. Quando as direções principais são identificadas, projetam-se as amostras em um espaço de dimensão menor, ignorando as direções de menor significância, realizando portanto uma redução de dimensionalidade. No Capítulo 5, a técnica PCA será utilizada para melhorar o desempenho da abordagem proposta, por um pré-processamento das imagens. Resultados numéricos serão comparados com aqueles obtidos pela rotina implementada sem o PCA. A seguir, apresentaremos alguns conceitos básicos para o desenvolvimento dessa técnica.

### 4.1 Definições básicas

Considere um número de amostras imersas em um espaço vetorial real de dimensão  $n$  (número natural fixo), onde cada amostra  $\Gamma \in \mathbb{R}^n$ , pode ser representado por um vetor

$$\Gamma = (x_1, x_2, \dots, x_n)$$

sendo que cada coordenada  $x_i$  pertence a um subespaço de dimensão 1 do  $\mathbb{R}^n$ . No caso em questão, tais subespaços são os espaços gerados pelos vetores canônicos

$$e_1 = (1, 0, \dots, 0)$$

$$e_2 = (0, 1, \dots, 0)$$

$$\vdots$$

$$e_n = (0, 0, \dots, 1)$$

que formam uma base para todo o espaço. Tal base é denominada Espaço Amostral.

A **Covariância** é uma medida clássica da estatística e, também é, uma extensão do conceito de variância [45]. A covariância mostra a relação de interdependência entre duas dimensões de uma amostra multidimensional.

Seja  $\Omega = \{w_1, w_2, \dots, w_m\}$  uma amostra discreta, e seja  $X$  uma variável aleatória que associa cada amostra  $w_i$  a um real  $X_i$ . Representa-se a média dos  $X_i$  por  $\bar{X}$

$$\bar{X} = \frac{1}{m} \sum_{i=1}^m X_i$$

e a Variância de  $X$  por  $var(X)$

$$var(X) = \frac{1}{m} \sum_{i=1}^m (X_i - \bar{X})^2.$$

Analogamente, associando duas variáveis aleatórias duas a cada amostra,  $X$  e  $Y$ , é possível definir a covariância entre  $X$  e  $Y$

$$cov(X, Y) = \frac{1}{m} \sum_{i=1}^m (X_i - \bar{X})(Y_i - \bar{Y}),$$

a qual satisfaz as seguintes propriedades:

$$cov(X, Y) = cov(Y, X)$$

e

$$cov(X, X) = var(X).$$

A matriz de covariância  $C$  é definida abaixo, onde  $o$  é a média dos vetores  $x_k$ :

$$C = \frac{1}{M-1} \sum_{k=1}^M (x_k - o)(x_k - o)^T$$

## 4.2 A Técnica PCA

Para se detectar padrões, deve-se ter um modelo estatístico com o propósito de identificar um subconjunto do espaço amostral  $\mathbb{R}^n$  que representa a presença do padrão procurado. Tal subconjunto será denominado **evento amostral** e será denominado um evento  $E$  como a presença do padrão que se procura num ponto amostral.

Em um problema de reconhecimento de padrões ocultos, dado um vetor qualquer do espaço em questão, deseja-se saber se existe ou não um padrão. Tem-se então, um subconjunto amostral  $\Omega \in E$  para representar o evento. O conjunto de amostras  $\Omega$  será chamado de **Conjunto de Treinamento**.

Inicialmente não se sabe nada sobre as propriedades e nem a forma geométrica de  $E$ , então pode-se *a priori* estimar a presença do padrão a partir das informações obtidas do conjunto  $\Omega$ . É possível obter um método de classificação onde, dado uma amostra  $\Gamma \in \mathbb{R}^n$ , a probabilidade (condicional) que ela pertença a  $E$  é dada por

$$P\{E|\Gamma\} \geq \theta,$$

onde  $\theta$  é um limiar que determina a sensibilidade do classificador.

Suponha que cada amostra pode ser representada por um vetor aleatório  $\Gamma$  como vetor do  $\mathbb{R}^n$

$$\Gamma = (x_1, x_2, \dots, x_n).$$

Digamos que se tenha  $M$  amostras  $\Gamma_i$ , sendo  $1 \leq i \leq M$  em um conjunto de treinamento  $\Omega$ . Se  $\Omega$  é uma boa representação do conjunto  $E$ , o comportamento das amostras implica em uma estimativa de boa qualidade deste evento.

Define-se  $\Psi$  o *vetor médio* das amostras como sendo a média das amostras, ou seja,

$$\Psi = \frac{1}{M} \sum_{i=1}^M \Gamma_i = (\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n)$$

e levando em consideração as amostras  $\Phi_i$  como sendo

$$\Phi_i = \Gamma_i - \Psi,$$

obtemos novo formato de  $\Phi$  dado por

$$\Phi = (y_1, y_2, \dots, y_n) = (x_1 - \bar{x}_1, x_2 - \bar{x}_2, \dots, x_n - \bar{x}_n) = \Gamma - \Psi.$$

Obviamente o conjunto  $\Omega_\Psi$ , que é formado pelos  $\Phi_i$  é o mesmo conjunto de treinamento  $\Omega$ , transladado pelo vetor  $-\Psi$ , isto é, para recuperar qualquer vetor no novo espaço, basta somar o vetor  $\Psi$  e tem-se novamente um vetor em  $\Omega$ . Efetuar esta translação garante o fato de que o conjunto  $\Omega_\Psi$  tem média zero, ou seja, que a média de cada coordenada  $y_j$  de cada vetor amostral  $\Phi_i$  é zero. Logo, para calcular a covariância<sup>1</sup> entre quaisquer duas dimensões  $y_j$  e  $y_k$  basta calcular

<sup>1</sup>De fato há uma leve diferença entre o fator multiplicativo  $\frac{1}{M-1}$  e o fator  $\frac{1}{M}$  definido anteriormente na fórmula de covariância. Isto ocorre porque possuímos apenas uma amostra do evento  $E$ , logo o primeiro fator dá uma estimativa melhor do valor real de covariância. Este valor também é chamado de covariância amostral [45].

$$cov(y_i, y_j) = \frac{1}{M-1} \sum_{i=1}^M \Phi_{i,j} \Phi_{i,k}$$

onde  $\Phi_{i,j}$  é a  $j$ -ésima coordenada do vetor  $\Phi_{i,j}$  e  $\Phi_{i,k}$  é a  $k$ -ésima coordenada do vetor  $\Phi_i$  [45].

Assim, considerando a matriz  $A$  como sendo a matriz dos vetores  $\Phi_i$  dispostos em coluna

$$A = (\Phi_1, \Phi_2, \dots, \Phi_M) = \begin{pmatrix} \Phi_{1,1} & \Phi_{2,1} & \cdots & \Phi_{M,1} \\ \Phi_{1,2} & \Phi_{2,2} & \cdots & \Phi_{M,2} \\ \vdots & \vdots & \ddots & \vdots \\ \Phi_{1,n} & \Phi_{2,n} & \cdots & \Phi_{M,n} \end{pmatrix}$$

pode-se definir a matriz  $C \in \mathbb{R}^{n \times n}$  como

$$\begin{aligned} C &= \frac{1}{M-1} AA^T \\ &= \frac{1}{M-1} \begin{pmatrix} \Phi_{1,1} & \Phi_{2,1} & \cdots & \Phi_{M,1} \\ \Phi_{1,2} & \Phi_{2,2} & \cdots & \Phi_{M,2} \\ \vdots & \vdots & \ddots & \vdots \\ \Phi_{1,n} & \Phi_{2,n} & \cdots & \Phi_{M,n} \end{pmatrix} \begin{pmatrix} \Phi_{1,1} & \Phi_{1,2} & \cdots & \Phi_{1,n} \\ \Phi_{2,1} & \Phi_{2,2} & \cdots & \Phi_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ \Phi_{M,1} & \Phi_{M,2} & \cdots & \Phi_{M,n} \end{pmatrix} \\ &= \frac{1}{M-1} \begin{pmatrix} \sum_{i=1}^M \Phi_{i,1} \Phi_{i,1} & \sum_{i=1}^M \Phi_{i,1} \Phi_{i,2} & \cdots & \sum_{i=1}^M \Phi_{i,1} \Phi_{i,n} \\ \sum_{i=1}^M \Phi_{i,2} \Phi_{i,1} & \sum_{i=1}^M \Phi_{i,2} \Phi_{i,2} & \cdots & \sum_{i=1}^M \Phi_{i,2} \Phi_{i,n} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^M \Phi_{i,n} \Phi_{i,1} & \sum_{i=1}^M \Phi_{i,n} \Phi_{i,2} & \cdots & \sum_{i=1}^M \Phi_{i,n} \Phi_{i,n} \end{pmatrix} \\ &= \begin{pmatrix} cov(y_1, y_1) & cov(y_1, y_2) & \cdots & cov(y_1, y_n) \\ cov(y_2, y_1) & cov(y_2, y_2) & \cdots & cov(y_2, y_n) \\ \vdots & \vdots & \ddots & \vdots \\ cov(y_n, y_1) & cov(y_n, y_2) & \cdots & cov(y_n, y_n) \end{pmatrix} \end{aligned}$$

$$\text{pois } cov(y_i, y_j) = \frac{1}{M-1} \sum_{i=1}^M \Phi_{i,j} \Phi_{i,k}.$$

A matriz acima, como dito anteriormente, é denominada *matriz de covariância* de  $\Omega$  no ponto  $\Psi$ .

Pode-se observar que  $C = C^T$ , ou seja, a Matriz  $C$  é simétrica, pelo fato de

$$cov(y_j, y_k) = cov(y_k, y_j).$$

**Afirmção 4.2.1.** *A matriz de covariância  $C$  é definida positiva.*

*Demonstração.* Consideremos a forma quadrática de uma matriz real e simétrica  $C$  com todos autovalores positivos:

$$d^2 = y^T C y,$$

onde  $y$  possui dimensão  $n$ .

Sem perda de generalidade, podemos assumir que os vetores  $y$  são originados de uma transformação ortonormal dos vetores  $x$ :  $y = Zx$ . Lembrando que toda matriz simétrica pode ser diagonalizada através de uma transformação de semelhança ortogonal, teremos:

$$d^2 = y^T C y = Zx^T C Zx = x^T Z^T C Zx = x^T \Lambda x = \sum_{i=1}^n \lambda_i x_i^2.$$

Isto mostra que  $d^2$  é positiva para todo  $x$  não nulo.

Como a matriz de covariância é simétrica, real e possui todos os autovalores positivos (que representam as variâncias após a transformação ortonormal), esta é definida positiva. ■

Usar-se-á estes resultados para garantir a existência das componentes principais.

### 4.2.1 Componentes Principais

A técnica PCA consiste em encontrar as direções que melhor representam a variação entre as dimensões, também conhecida como *Componentes Principais*. Então, a partir de um certo número de amostras bem representativas, pode-se estimar as direções de maior variação dos dados assim como mostra a figura abaixo.

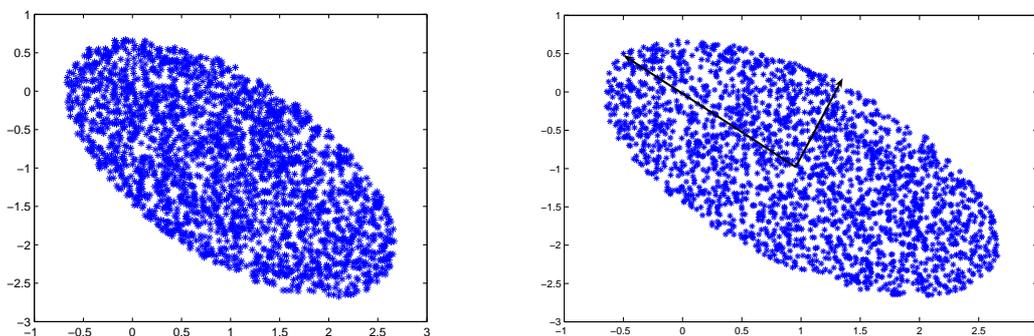


Figura 4.1: Exemplo em duas dimensões da utilização da técnica PCA para encontrar as direções principais de variação, a partir de amostras estatísticas.

Para se encontrar as componentes principais, deve-se primeiro extrair os autovetores da matriz de covariância  $C$ .

Como  $C$  é uma matriz simétrica, podemos afirmar que  $C$  possui  $n$  autovetores associados a  $n$  autovalores  $\lambda_i$  [30], onde

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n \geq 0.$$

Cada autovalor  $\lambda_i$  é uma solução da equação  $\det(C - I\lambda) = 0$ , e associado a ele existe um autovetor unitário  $u_k$ , que satisfaz a equação  $Cv = \lambda v$ . Os autovetores  $\{u_1, u_2, \dots, u_n\}$  são ortogonais, formando uma base ortonormal do  $\mathbb{R}^n$ .

Para se realizar a redução da dimensionalidade das amostras  $\Gamma_i$  para um subespaço de dimensão  $m < n$ , serão utilizados como amostragem os autovetores  $u_k$  que possuem os maiores autovalores associados. Tais vetores serão as componentes principais. Para realizar isto, é suficiente calcular as projeções dos vetores  $\Psi_i$  no subespaço gerado por  $\{u_1, u_2, \dots, u_m\}$ , isto é, dado um vetor amostral  $\Gamma$  qualquer, temos que

$$w_k = (u_k, \Gamma - \Psi) = (u_k, \Phi),$$

onde os coeficientes  $w_k$  são as coordenadas do novo vetor  $\vec{w}$

$$\vec{w} = \{w_1, w_2, \dots, w_m\}$$

em um novo espaço reduzido de apenas  $m$  dimensões.

Como tomou-se apenas alguns autovetores, desprezando outros, quando um vetor projetado é novamente reescrito no espaço amostral de dimensão  $n$ , através de uma transformação inversa, há uma “perda de informação” devido ao desprezo das componentes menos significativas. É possível, no entanto, mostrar que esta perda de informações não afeta o problema inicial [55]. Para se recuperar o vetor reconstruído  $\Phi_f$ , basta realizar uma combinação linear

$$\Phi_f = \sum_{k=i}^m w_k u_k,$$

onde  $i$  é o menor índice das componentes mais significativas.

Para estimar o erro, basta executar a maneira clássica, ou seja, calcular o comprimento do vetor  $\Phi - \Phi_f$

$$\varepsilon^2 = \|\Phi - \Phi_f\|^2;$$

se o erro em cada amostra for mínimo, isto significa que a redução dimensional foi aplicada com sucesso.

## 4.2.2 SVD e Covariâncias

Assumamos que dado  $n$  vetores  $x_j$ ,  $j = 1, \dots, n$  cada um de dimensão  $m$ , e média zero,  $\alpha = \frac{1}{n} \sum_{j=1}^n x_j$ . Se a média não é zero, este valor nulo pode ser obtido subtraindo a média  $\alpha$  de todos os vetores.

Antes, estamos interessados em encontrar uma base ortonormal para o subespaço gerado por estes vetores. Ao invés disso, vamos investigar a “correspondência” entre estes vetores. Geometricamente, buscamos a direção que melhor aproxima a distribuição dos vetores, ou seja, buscamos pelos vetores que representam a maior variabilidade dos dados. Se imaginarmos uma nuvem de pontos em uma região bidimensional (incluindo a origem, pois assumimos que a média vale zero), procuramos a direção de máxima variação. Para dados com um média não-nula, com a subtração da média, estamos observando a direção da máxima variação em relação à média ( $u_1$  na Figura 4.2),

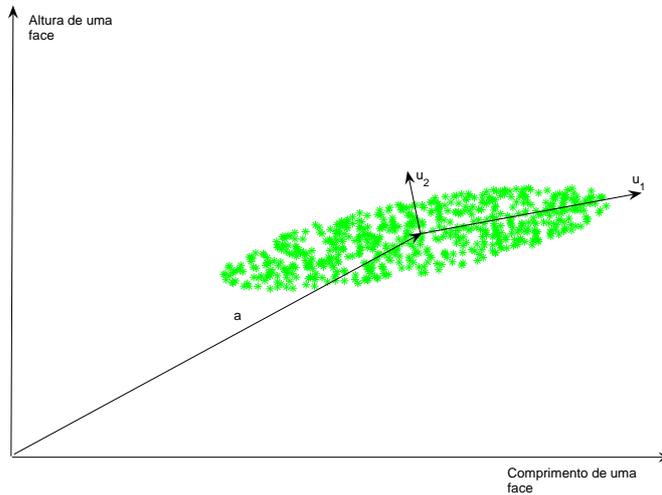


Figura 4.2: A distribuição das medidas de altura e largura de um número de faces.

Matematicamente estamos procurando pela direção  $u$  tal que

$$\mu = \max_{\|u\|_2=1} \frac{1}{n} \sum_{j=1}^n (u^T x_j)^2. \quad (4.1)$$

Com a Matriz de Covariância

$$C = \frac{1}{n} \sum_{j=1}^n x_j x_j^T, \quad (4.2)$$

é possível reescrever (4.1) como

$$\mu = \max_{\|u\|_2=1} (u^T C u). \quad (4.3)$$

Como  $C$  é real e simétrica, pode ser diagonalizada com uma matriz ortonormal  $U_C$ [30],

$$C = U_C \Lambda_C U_C^T. \quad (4.4)$$

Aqui  $\Lambda_C = \text{diag}(\lambda_1, \dots, \lambda_m)$ , onde  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_m$ . Logo,

$$\mu = \max_{\|u\|_2=1} (u^T U_C \Lambda_C U_C^T u) = \max_{\|y\|_2=1} (y^T \Lambda_C y). \quad (4.5)$$

Como  $U_C$  é ortonormal e  $y = U_C^T u$ , sabemos que  $\|y\|_2 = \|u\|_2 = 1$ . Devemos calcular

$$\mu = \max_{\|y\|_2=1} \sum_{j=1}^m \lambda_j |y_j|^2. \quad (4.6)$$

sujeito a

$$\sum_{j=1}^m |y_j|^2 = 1. \quad (4.7)$$

onde,  $\mu = \lambda_1$  e  $y = e_1 = [1 \ 0 \ 0 \ \dots \ 0]^T$ . Como  $u = U_C y = U_C e_1$ , a direção de variação máxima  $u$  é exatamente o autovetor  $u_1$  da matriz de covariância  $C$ , que pertence ao maior autovalor  $\lambda_1$ . Além disso,  $\lambda_1$  mede a variação na direção de  $u_1$ .

Procurando pela direção de máxima variação ortogonal à  $u_1$ , devemos calcular (4.6), este estando sujeito a (4.7) e  $e_1^T y = 0$ . Isto implica que a primeira componente de  $y$  é igual a 0. Segue

que (4.6) é calculado estando sujeito a  $\sum_{j=2}^m |y_j|^2 = 1$ . Isto é conseguido por  $y = e_2 = [0 \ 1 \ 0 \ \dots \ 0]^T$  ou  $u = U_C e_2 = u_2$ , o autovetor de  $C$  pertencente ao segundo maior autovalor.

Continuando este processo concluímos que: o primeiro autovetor da matriz de covariância  $C$  aponta na direção de máxima variação, e o correspondente autovalor mede a variação em sua direção, isto é, é a variância desta direção. Os autovetores subsequentes apontam na direção de máxima variação que sejam ortogonais às direções anteriores, e os seus correspondentes autovalores medem as variações.

Então, as direções de máxima variação são dadas pelas colunas de  $U$  e a variância (autovalores  $\lambda_j$  de  $C$ )  $\lambda_j$  pelo quadrado dos valores singulares, isto é,  $\lambda_j = \sigma_j^2$ .

### 4.3 A Distância de Mahalanobis

Dado um ponto amostral qualquer, a *Distância de Mahalanobis*<sup>2</sup> será utilizada para indicar se tal ponto está “próximo” ou “longe” do conjunto de treinamento.

Quando o PCA é bem utilizado, nota-se que entre as direções principais, algumas direções são mais significativas que outras. Isto é refletido diretamente na “forma” que o conjunto de treinamento toma. Por isso, para criar um método de classificação para determinar se um ponto do espaço amostral está de acordo com os dados de treinamento, deve-se utilizar uma métrica conveniente, ou seja, uma métrica que leve em consideração a forma do conjunto. A métrica euclidiana não é adequada para esse tipo de situação como mostra a figura abaixo.

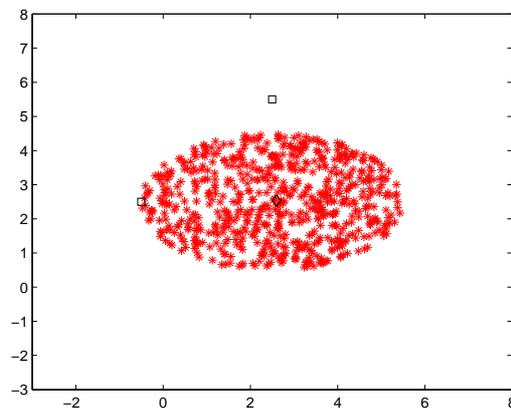


Figura 4.3: Pontos que têm a mesma distância euclidiana em relação ao centróide podem estar a uma distância relativa diferente do conjunto de treinamento.

<sup>2</sup>Medida de distância introduzida pelo matemático indiano Prasanta Chandra Mahalanobis em 1936.

**Definição 4.3.1.** *Defini-se o quadrado da Distância de Mahalanobis entre uma amostra  $\Gamma$  e o centróide  $\Psi$  como sendo*

$$M^2(\Gamma) = \|\Gamma - \Psi\|_M^2 = (\Gamma - \Psi)^T C^{-1} (\Gamma - \Psi),$$

onde  $C^{-1}$  é a inversa da matriz de covariância.

Quando  $C = I$ , a distância de Mahalanobis é a distância Euclidiana, ou seja,

$$(\Gamma - \Psi)^T (\Gamma - \Psi) = ((\Gamma - \Psi), (\Gamma - \Psi)) = \|\Gamma - \Psi\|^2.$$

Sabe-se que toda matriz simétrica  $A$  pode ser diagonalizada através de uma transformação de semelhança ortogonal. Além disso, a matriz  $Q$  que é usada para diagonalizar  $A$  tem nas suas colunas um conjunto de autovetores de  $A$  ortonormais. A matriz diagonal resultante tem nos seus elementos da diagonal os autovalores de  $A$  [30].

Sendo assim, considerando  $\nu$  como a matriz dos autovetores  $u_k$  dispostos em coluna, pode-se decompor a matriz de covariância da seguinte forma:

$$\Lambda = \nu^T C \nu,$$

onde  $\Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$  é a matriz dos autovalores. Como  $\nu$  é a matriz de mudança de base formada por vetores ortogonais, sua inversa  $\nu^{-1} = \nu^T$  porque  $\nu \nu^T = \nu^T \nu = I$ , pois  $\nu$  é um operador real ortogonal [30]. Como

$$\Lambda^{-1} = \text{diag}\left(\frac{1}{\lambda_1}, \frac{1}{\lambda_2}, \dots, \frac{1}{\lambda_n}\right)$$

pode-se concluir que

$$\begin{aligned} \Lambda &= \nu^T C \nu \\ I &= \nu^T C \nu \Lambda^{-1} \\ \nu &= C \nu \Lambda^{-1} \\ C^{-1} \nu &= \nu \Lambda^{-1} \\ C^{-1} &= \nu \Lambda^{-1} \nu^T. \end{aligned}$$

Então, considerando o vetor transladado  $\Phi = \Gamma - \Psi$  e utilizando o fato de  $C^{-1} = \nu \Lambda^{-1} \nu^T$ , onde  $\Lambda^{-1} = \text{diag}\left(\frac{1}{\lambda_1}, \frac{1}{\lambda_2}, \dots, \frac{1}{\lambda_n}\right)$ , pode-se apresentar o formato do quadrado da distância de Mahalanobis como sendo

$$\begin{aligned} M^2(\Gamma) &= \Phi^T C^{-1} \Phi \\ &= \Phi^T \nu \Lambda^{-1} \nu^T \Phi \end{aligned}$$

$$\begin{aligned}
&= \varphi^T \Lambda^{-1} \varphi \\
&= \sum_{i=1}^n \frac{z_i^2}{\lambda_i}
\end{aligned}$$

onde  $\varphi = (z_1, z_2, \dots, z_n) = \nu^T \Phi$ , isto é,  $\varphi$  é o vetor  $\Phi$  escrito nas coordenadas da base dos autovetores.

### 4.3.1 Interpretação Geométrica

Considere o exemplo dos pontos que distam  $r$  fixo, segundo a distância de Mahalanobis. Temos que

$$M(\Gamma) = \sqrt{\sum_{i=1}^n \frac{z_i^2}{\lambda_i}} = r \Leftrightarrow \sum_{i=1}^n \frac{z_i^2}{\lambda_i} = r^2 \Leftrightarrow \sum_{i=1}^n \frac{z_i^2}{r^2 \lambda_i} = 1.$$

A última equação trata de um elipsóide  $n$ -dimensional. A figura abaixo dá uma noção dos pontos equidistantes ao centróide a distância de Mahalanobis.

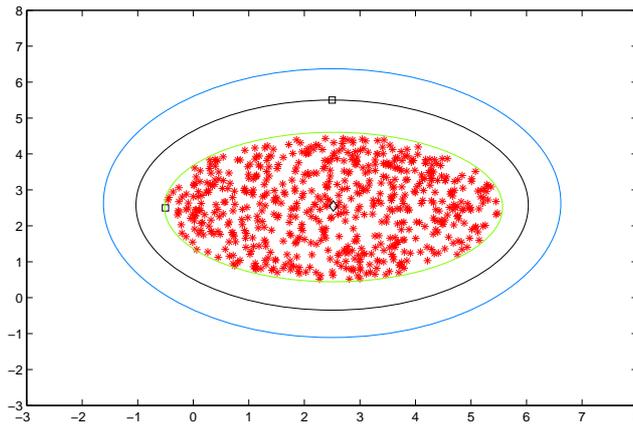


Figura 4.4: Noção de distância ao conjunto de treinamento utilizando elipses.

Tais distribuições são casos isolados e como não se sabe *a priori* o comportamento da distribuição das faces, este será representado como uma união de elipsóides  $n$ -dimensionais.

## 4.4 Algoritmo PCA

Considere uma conjunto de  $N$  imagens e uma imagem-teste, todas com dimensões  $N \times M$ . Seja  $F_m$  a face média do banco calculada por meio de média aritmética simples.

**Algoritmo 4.4.1.** *Rotina para extração das características principais*

*Passo 1) Leia todas as imagens do banco e a imagem-teste. Armazene em formato vetorial:*

- $v_i$  - a  $i$ -ésima imagem do banco de imagens.
- $testea$  - imagem-teste.

*Passo 2) Normalize a amostra através da diferença entre cada imagem e a média do banco armazenando seus resultados nas colunas de  $L$ :*

$$L_i = (v_i - F_m), i = 1, \dots, N$$

*Passo 3) Calcule a decomposição SVD da matriz:*

$$L * L^T = U * S * V^T;$$

*Passo 4) Projete  $testea$  normalizada no eigenspace:  $w = U^T * (testea - F_m)$ ;*

*Passo 5) Projete as imagens do banco normalizadas no eigenspace:  $P = U^T * L$ ;*

*Passo 6) Calcule a distância de Mahalanobis:*

$$D_i = \sqrt{(P_i - w)^T (S(i, i))^{-1} (P_i - w)},$$

onde  $P_i$  é a  $i$ -ésima coluna da matriz  $P$ ;

*Passo 7) Se  $\min_i \{D_i\} < \text{tolerância}$ , então reconheça a  $testea$  como compatível com o banco de imagens.*

## 4.5 Resumo dos objetivos do PCA

Como descrito em [52], os principais objetivos pela análise dos componentes principais são:

- Gerar novas variáveis que possam expressar a informação contida em um conjunto original de dados;
- Reduzir a dimensionalidade do problema que se está estudando, como passo prévio para análises futuras;
- Eliminar, o quanto possível, algumas das variáveis originais se elas traduzem poucas informações.

# Capítulo 5

## Abordagem OVO acoplada à técnica PCA

Neste capítulo será apresentada a junção dos dois métodos usados para o problema do reconhecimento facial, uma vez que o uso da técnica PCA permite uma redução da dimensionalidade e retira as informações menos significativas.

### 5.1 Motivação para unir o Algoritmo OVO à técnica PCA

A principal vantagem de aplicar o PCA junto à abordagem OVO é que o PCA extrai as principais características de cada face e as projeta em um novo espaço denominado *eigenspace*. Desta forma, os detalhes que são mais visualmente evidentes em cada imagem ficarão salientados no novo espaço e serão mais fáceis de serem distinguidos. Após feito isto, o reconhecimento através do Algoritmo 3.1.1 tem maior probabilidade de ser bem sucedido, pois se o programa encontrar características na imagem-teste muito parecidas com as características de uma imagem do banco, é bastante provável que tal imagem-teste seja de uma pessoa que tenha fotos no banco de imagens.

Pode-se dizer que com as características ressaltadas e tão distantes umas das outras, ao se utilizar a abordagem OVO acoplada à técnica PCA, as distâncias entre a imagem-teste e as imagens do banco devem aumentar, já que o Algoritmo 3.1.1 partirá desses dados pré-filtrados, causando diferença significativa entre os *pixels* do banco e os *pixels* a serem ajustados na imagem-teste após as transformações geométricas.

A forma clássica de implementação para a técnica PCA utiliza o algoritmo da decomposição em valores singulares que está apresentado em [26]. Utilizaremos este método em nossa abordagem.

## 5.2 Detalhes sobre a abordagem OVO acoplado à rotina PCA

No problema acoplado, a construção da função objetivo vai se alterar uma vez que será aferido o ajuste (o valor das  $f_j$ 's) após a projeção no *eigenspace*. Para tanto, é necessário que:

- Com as imagens do banco seja construído o *eigenspace* usando o Algoritmo 4.4;
- Ainda antes do ajuste, sejam projetadas as imagens do banco no *eigenspace*.

Logo após a realização de tais procedimentos pela rotina acoplada, inicia-se o processo de otimização:

- São fornecidos os parâmetros iniciais;
- É realizado o ajuste da imagem-teste com esses parâmetros iniciais;
- Logo após, projeta-se tal imagem-teste ajustada no *eigenspace*;
- Calcula-se a  $f_i$  como sendo a distância da imagem-teste projetada à  $i$ -ésima imagem do banco projetada, para cada  $i$ .
- Com tal  $f_i$ , determina-se o valor da função OVO de cada iteração durante o processo de determinação dos parâmetros ótimos via minimização pelo Algoritmo 3.1.1.

## 5.3 Sobre os Experimentos Numéricos

Algumas características práticas com respeito às implementações numéricas são dadas abaixo:

- O Problema (2.1) é não linear. Além disso, em nossas aplicações, as restrições serão lineares. Para tal implementação (sem uso do PCA) foram feitas quatro rotinas principais, responsáveis pela avaliação da função, do gradiente, montagem do conjunto de índices  $I_\epsilon(x_k)$  e a rotina principal;
- No próximo capítulo serão feitos testes numéricos acoplando a rotina do PCA às quatro rotinas principais citadas anteriormente;
- Todos os experimentos numéricos foram rodados em um computador Pentium 3, 1.86 Ghz, 640MB RAM utilizando o software Matlab 7.0;

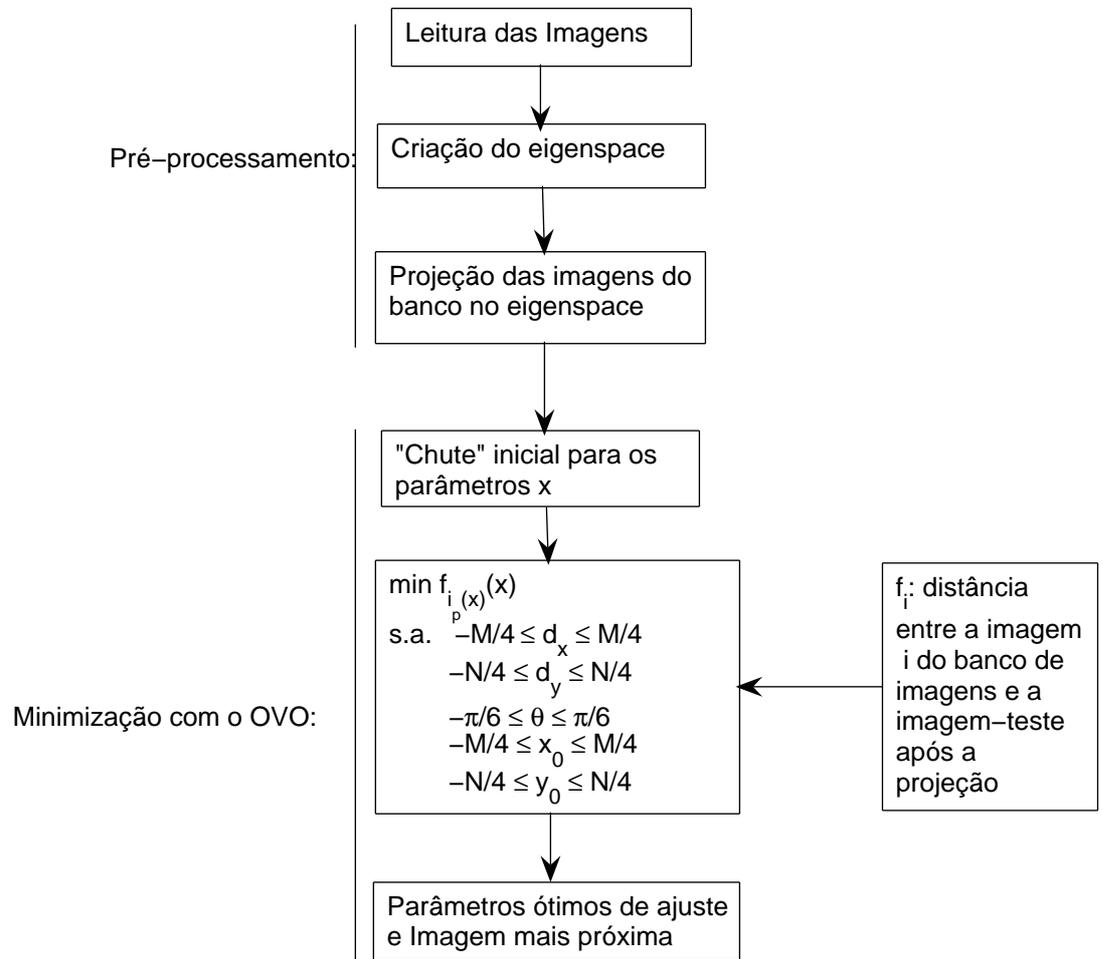


Figura 5.1: Estrutura em caixa da rotina OVO acoplada à técnica PCA

# Capítulo 6

## Testes Numéricos

Neste capítulo serão expostos os resultados dos testes numéricos utilizando a rotina OVO (Algoritmo 3.1.1) isolada e a abordagem OVO acoplada à rotina do PCA (Algoritmo 4.4). Todas as imagens dos bancos utilizados e as imagens-teste possuem  $NC$  colunas e  $NL$  linhas. O vetor que aparece nas tabelas representa os parâmetros ótimos obtidos ( $x_{sol}$ ) e tem a ordem  $(d_x, d_y, \theta, x_0, y_0)$ . Para evitar confusões, o conjunto de parâmetros iniciais foi denominado  $x_a$  e  $x_b$ . Devemos observar o fato de que o  $x^*$  (solução ótima), com o aumentar do valor de  $p$  produz valores de  $f_{i_p(x^*)}(x^*)$  maiores.

### 6.1 Resultados com um banco de 24 imagens

O banco de 24 imagens utilizado [47] neste trabalho possui duas imagens de cada pessoa, fazendo com que o  $p$  ideal seja  $p = 2$ . Foram feitos 4 tabelas de testes numéricos com a imagem-teste *teste1*. As imagens foram ordenadas por linha. As imagens compatíveis com a pessoa da foto imagem-teste *teste1* são as imagens 3 e 4 deste banco. Foram dados dois “chutes” iniciais:  $x_a = (1; 1; 0; NC/2; NL/2)$  e  $x_b = (2; 5; -0.1; NC/2; NL/2)$ .



Figura 6.1: Imagem-teste: *teste1*.



Figura 6.2: Banco com 24 imagens - Imagens ordenadas por linha (de 1 a 24).

A Tabela 6.1 trata da rotina OVO sendo aplicada isoladamente e para esta,  $x_a$  é bastante conveniente pois é um “chute” inicial muito próximo da solução ideal. Pode-se ver que o algoritmo não faz grandes modificações para encontrar  $x_{sol}$ , e que as imagens reconhecidas após o otimização ter sido feita pertencem a mesma pessoa da imagem-teste  $teste1$ , que é a imagem 3 do banco de imagens. Com este  $x_a$  inicial, os resultados surgiram como se era esperado, ou seja, com o crescimento do valor de  $p$  a  $p$ -ésima distância aumentou.

Na Tabela 6.2 foi dado o mesmo “chute” inicial porém na rotina OVO acoplada ao PCA. Observa-se que o algoritmo modifica bastante os parâmetros iniciais até encontrar  $x_{sol}$ . Apesar disto, a rotina funciona bem e o valor da  $p$ -ésima distância aumenta com o crescimento do valor de  $p$ , muito embora tenha reconhecido a imagem correta para todo  $p$ , pode-se observar um “salto” no valor da distância de  $p = 2$  para  $p = 3$ .

As imagens reconhecidas são todas da mesma pessoa (imagens 3 e 4 do banco de imagens), inclusive do  $p$  ideal, que é  $p = 2$ . Percebe-se também que da Tabela 6.1 para a Tabela 6.2 houve um aumento significativo nos valores da  $p$ -ésima distância. Isto era de se esperar, pois com a rotina OVO acoplada ao PCA, as características principais são ressaltadas. Com características tão distantes de uma pessoa para outra para serem comparadas, é normal que os valores das distâncias aumentem.

$xsol$	p	Valor da $p$ -ésima distância	Imagem reconhecida mais próxima	Valor da menor distancia
$(1, 1, 0, 62.5, 75)^T$	1	0.0001	3	0.0001
$(1, 1, 0, 62.5, 75)^T$	2	0.0067	3	0.0002
$(1, 1, 0, 62.5, 75)^T$	3	0.0188	3	0.0002
$(1, 1, 0, 62.5, 75)^T$	4	0.0553	3	0.0002
$(1, 1, 0, 62.5, 75)^T$	5	0.1034	3	0.0002
$(1, 1, 0, 62.5, 75)^T$	24	0.9534	3	0.0002

Tabela 6.1: Rotina OVO - Imagem-teste:  $teste1 - x_a = (1; 1; 0; NC/2; NL/2)$  - banco com 24 imagens -  $p = 2$  .

$xsol$	p	Valor da $p$ -ésima distância	Imagem reconhecida mais próxima	Valor da menor distancia
$(0.009, 0.009, -0.0001, 1.23, 1.47)^T$	1	17.568	4	17.568
$(0.009, 0.009, -0.0001, 1.23, 1.47)^T$	2	18.955	4	17.568
$(0.007, 0.007, -0.001, 1.093, 1.312)^T$	3	51.922	3	44.656
$(0.007, 0.007, -0.001, 1.093, 1.312)^T$	4	54.363	3	44.656
$(0.007, 0.007, -0.001, 1.093, 1.312)^T$	5	59.453	3	44.656
$(0, 0, -0.52, 62.500, 75)^T$	24	93.775	4	56.214

Tabela 6.2: Rotinas OVO e PCA acopladas - Imagem-teste:  $teste1 - x_a = (1; 1; 0; NC/2; NL/2)$  - banco com 24 imagens -  $p = 2$  .

A Tabela 6.3 mostra resultados (rotina OVO) um pouco díspares devido a  $x_b$  ter parâmetros bem distantes do  $x$  ideal para o reconhecimento. Observa-se que com o crescimento do valor de  $p$  a  $p$ -ésima distância não diminui, o que era de se esperar, porém as imagens do banco reconhecidas não são compatíveis com a pessoa da imagem-teste  $teste1$ .

A Tabela 6.4 mostra que com a rotina OVO acoplada à técnica PCA, os resultados que antes não estavam bons (Tabela 6.3) ficam muito melhores, devido ao fato do PCA fazer a “filtragem” das características principais e depois haver o processo de otimização pela rotina OVO. As imagens reconhecidas são todas compatíveis com o banco (imagem 3 do banco), inclusive para o  $p$  ideal, ou seja,  $p = 2$ . Com o crescimento do valor de  $p$ , também aumenta o valor da  $p$ -ésima distância. Como descrito antes, as distâncias da rotina acoplada são maiores teoricamente.

$x_{sol}$	p	Valor da $p$ -ésima distância	Imagem reconhecida mais próxima	Valor da menor distancia
$(4.033, 11.951, -0.016, 62.177, 74.985)^T$	1	0.0167	18	0.0167
$(13.71, 10.44, -0.002, 49.62, 83.89)^T$	2	0.0167	21	0.0049
$(13.71, 10.44, -0.002, 49.62, 83.89)^T$	3	0.0271	21	0.0049
$(13.22, 14.85, -0.033, 70.44, 92.70)^T$	4	0.041	18	0.016
$(13.22, 14.85, -0.033, 70.44, 92.70)^T$	5	0.064	18	0.016
$(1.013, 3.988, 0.011, 61.48, 74.01)^T$	24	0.364	8	$1.38e-004$

Tabela 6.3: Rotina OVO - Imagem-teste: *teste1* -  $x_b = (2; 5; -0.1; NC/2; NL/2)$  - banco com 24 imagens -  $p = 2$  .

$x_{sol}$	p	Valor da $p$ -ésima distância	Imagem reconhecida mais próxima	Valor da menor distancia
$(0.019, 0.049, -0.001, 1.249, 1.499)^T$	1	44.062	3	44.062
$(0.019, 0.049, -0.001, 1.249, 1.499)^T$	2	44.433	3	44.062
$(0.019, 0.049, -0.001, 1.249, 1.499)^T$	3	46.280	3	44.062
$(0.019, 0.049, -0.001, 1.249, 1.499)^T$	4	47.951	3	44.062
$(0.019, 0.049, -0.001, 1.249, 1.499)^T$	5	53.803	3	44.062
$(0.015, 0.045, -0.003, 0.937, 1.125)^T$	24	81.715	3	58.767

Tabela 6.4: Rotina OVO acoplada ao PCA - Imagem-teste: *teste1* -  $x_b = (2; 5; -0.1; NC/2; NL/2)$  - banco com 24 imagens -  $p = 2$  .

## 6.2 Resultados com um banco de 32 imagens

O banco de 32 imagens [47] também possui duas imagens de cada pessoa, fazendo com que o  $p$  ideal seja  $p = 2$ . Seguindo o mesmo modelo do banco de 24 imagens, foram feitos 4 tabelas de testes numéricos com a imagem-teste *teste2*. As imagens compatíveis com a pessoa da foto imagem-teste *teste2* são as imagens 29 e 30 deste banco. Os parâmetros iniciais permaneceram os mesmos:  $x_a = (1; 1; 0; NC/2; NL/2)$  e  $x_b = (2; 5; -0.1; NC/2; NL/2)$ .



Figura 6.3: Imagem-teste: *teste2*.

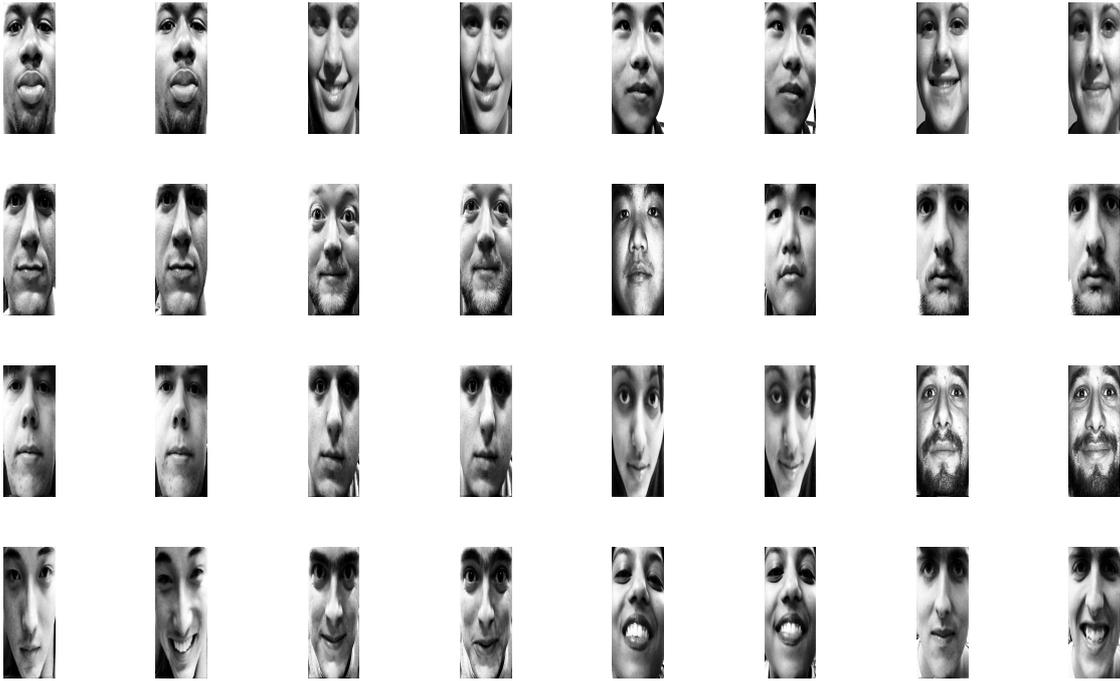


Figura 6.4: Banco com 32 imagens - Imagens ordenadas por linha (de 1 a 32).

A Tabela 6.5 mostra que, assim como na Tabela 6.1, os resultados da rotina OVO são bem sucedidos quando os valores dos parâmetros de  $x_a$  estão próximos da solução. A imagem reconhecida em todos os casos é da mesma pessoa da imagem-teste *teste2* e, como previsto, o valor da  $p$ -ésima distância não diminui com o aumento do valor de  $p$ .

A Tabela 6.6 mostra que a rotina funcionou bem para este caso, pois reconheceu a pessoa correta para o  $p$  ideal, que é  $p = 2$ . Observa-se que a partir de  $p > 2$ , as fotos do banco que são reconhecidas não foram relativas à pessoa da imagem-teste *teste2*, o que também é um resultado esperado e normal, já que as distâncias da rotina OVO acoplada ao PCA aumentam muito, salientando bastante as diferenças entre as imagens. O valor da  $p$ -ésima distância também não diminui com o aumento do valor de  $p$ . O valor da  $p$ -ésima distância “salta” de  $p = 2$  para  $p = 3$ , como era esperado.

$xsol$	p	Valor da $p$ -ésima distância	Imagem reconhecida mais próxima	Valor da menor distancia
$(1, 1, 0, 62.50, 75)^T$	1	$6.151e-005$	30	$6.151e-005$
$(1, 1, 0, 62.50, 75)^T$	2	0.003	30	$6.151e-005$
$(1, 1, 0, 62.50, 75)^T$	3	0.004	30	$6.151e-005$
$(1, 1, 0, 62.50, 75)^T$	4	0.006	30	$6.151e-005$
$(1, 1, 0, 62.50, 75)^T$	5	0.007	30	$6.151e-005$
$(1, 1, 0, 62.50, 75)^T$	24	0.301	30	$6.151e-005$
$(1, 1, 0, 62.50, 75)^T$	32	0.460	30	$6.151e-005$

Tabela 6.5: Rotina OVO - Imagem-teste:  $teste2 - x_a = (1; 1; 0; NC/2; NL/2)$  - banco com 32 imagens -  $p = 2$  .

$xsol$	p	Valor da $p$ -ésima distância	Imagem reconhecida mais próxima	Valor da menor distancia
$(0.009, 0.009, -0.00001, 1.248, 1.498)^T$	1	15.889	30	15.889
$(0.009, 0.009, -0.0001, 1.23, 1.47)^T$	2	31.878	30	27.995
$(0.005, 0.005, -0.002, 0.937, 1.125)^T$	3	65.700	11	59.948
$(0.005, 0.005, -0.002, 0.937, 1.125)^T$	4	66.879	11	59.948
$(0.005, 0.005, -0.002, 0.937, 1.125)^T$	5	67.014	11	59.948
$(0, 0, -0.52, 62.500, 75)^T$	24	82.909	8	61.011
$(0, 0, -0.52, 62.500, 75)^T$	32	87.940	8	61.011

Tabela 6.6: Rotina OVO acoplada ao PCA - Imagem-teste:  $teste2 - x_a = (1; 1; 0; NC/2; NL/2)$  - banco com 32 imagens -  $p = 2$  .

Pode-se ver observar na Tabela 6.7 que a rotina OVO quando implementada com  $x_b = (2; 5; -0.1; NC/2; NL/2)$  não foram bem sucedidos quanto ao reconhecimento em si.

A Tabela 6.8 a rotina OVO acoplada à técnica PCA com o mesmo  $x_b$  utilizado para a Tabela 6.7, e os resultados obtidos também não foram bem sucedidos quanto ao reconhecimento.

Como em todos os casos anteriores, a magnitude da  $p$ -ésima distância do programa OVO acoplado ao PCA é maior que se apenas o programa OVO tentar otimizar o processo.

Tais resultados reforçam que o método apresenta grande sensibilidade ao conjunto de parâmetros iniciais, convergindo de acordo com este, como visto nas oito tabelas anteriores deste capítulo.

$xsol$	p	Valor da $p$ -ésima distância	Imagem reconhecida mais próxima	Valor da menor distancia
$(-2.561, 5.276, -0.101, 62.500, 75)^T$	1	$1.537e-005$	23	$1.537e-005$
$(-2.561, 5.276, -0.101, 62.500, 75)^T$	2	$6.151e-005$	23	$1.537e-005$
$(-2.561, 5.276, -0.101, 62.500, 75)^T$	3	0.001	23	$1.537e-005$
$(-2.561, 5.276, -0.101, 62.500, 75)^T$	4	0.002	23	$1.537e-005$
$(-2.561, 5.276, -0.101, 62.500, 75)^T$	5	0.005	23	$1.537e-005$
$(13.828, -1.360, 0.074, 61.948, 73.229)^T$	24	0.206	2	0.001
$(2.374, 4.803, 0.518, 62.229, 74.797)^T$	32	0.263	28	$3.844e-004$

Tabela 6.7: Rotina OVO - Imagem-teste:  $teste2 - x_b = (2; 5; -0.1; NC/2; NL/2)$  - banco com 32 imagens -  $p = 2$  .

$xsol$	p	Valor da $p$ -ésima distância	Imagem reconhecida mais próxima	Valor da menor distancia
$(0.019, 0.049, -0.001, 1.210, 1.453)^T$	1	42.941	11	42.941
$(0.019, 0.049, -0.001, 1.210, 1.453)^T$	2	55.729	11	42.941
$(0.019, 0.049, -0.001, 1.230, 1.476)^T$	3	57.678	11	44.908
$(0.019, 0.049, -0.001, 1.249, 1.499)^T$	4	59.076	11	44.142
$(0.019, 0.049, -0.001, 1.249, 1.499)^T$	5	62.085	11	44.142
$(1, 4, -0.523, 62.500, 75)^T$	24	82.872	7	59.590
$(1, 4, -0.523, 62.500, 75)^T$	32	88.536	7	59.590

Tabela 6.8: Rotina OVO acoplada ao PCA - Imagem-teste:  $teste2 - x_b = (2; 5; -0.1; NC/2; NL/2)$  - banco com 32 imagens -  $p = 2$  .

### 6.3 Resultados com um banco de 120 imagens

Foram realizados novos testes numéricos para o banco de 120 imagens [7], com uma composição mais heterogênea (permitindo maior variação de posicionamento da face). Nele, o valor de  $p$  ideal é  $p = 3$ , por ter 3 imagens de cada pessoa no banco. A imagem-teste utilizada foi a  $teste3$  e esta é compatível com as imagens 7, 8 e 9 do banco de imagens. Os parâmetros iniciais foram  $x_a = (1; 1; 0; NC/2; NL/2)$  e  $x_b = (0.7; -0.4; 0.05; NC/2; NL/2)$ . Observe que nos testes anteriores o banco de imagens era composto por imagens tratadas de modo a excluir o máximo possível o “fundo” (*background*) e os cabelos das pessoas, o que facilita o reconhecimento. Este último banco é composto por imagens mais heterogêneas, no sentido de que abrange uma região maior do rosto, tornando os métodos de reconhecimento mais suscetíveis a erros oriundos das variações mais intensas de posicionamento do rosto, variações de expressões e intensidade de cor e luz.

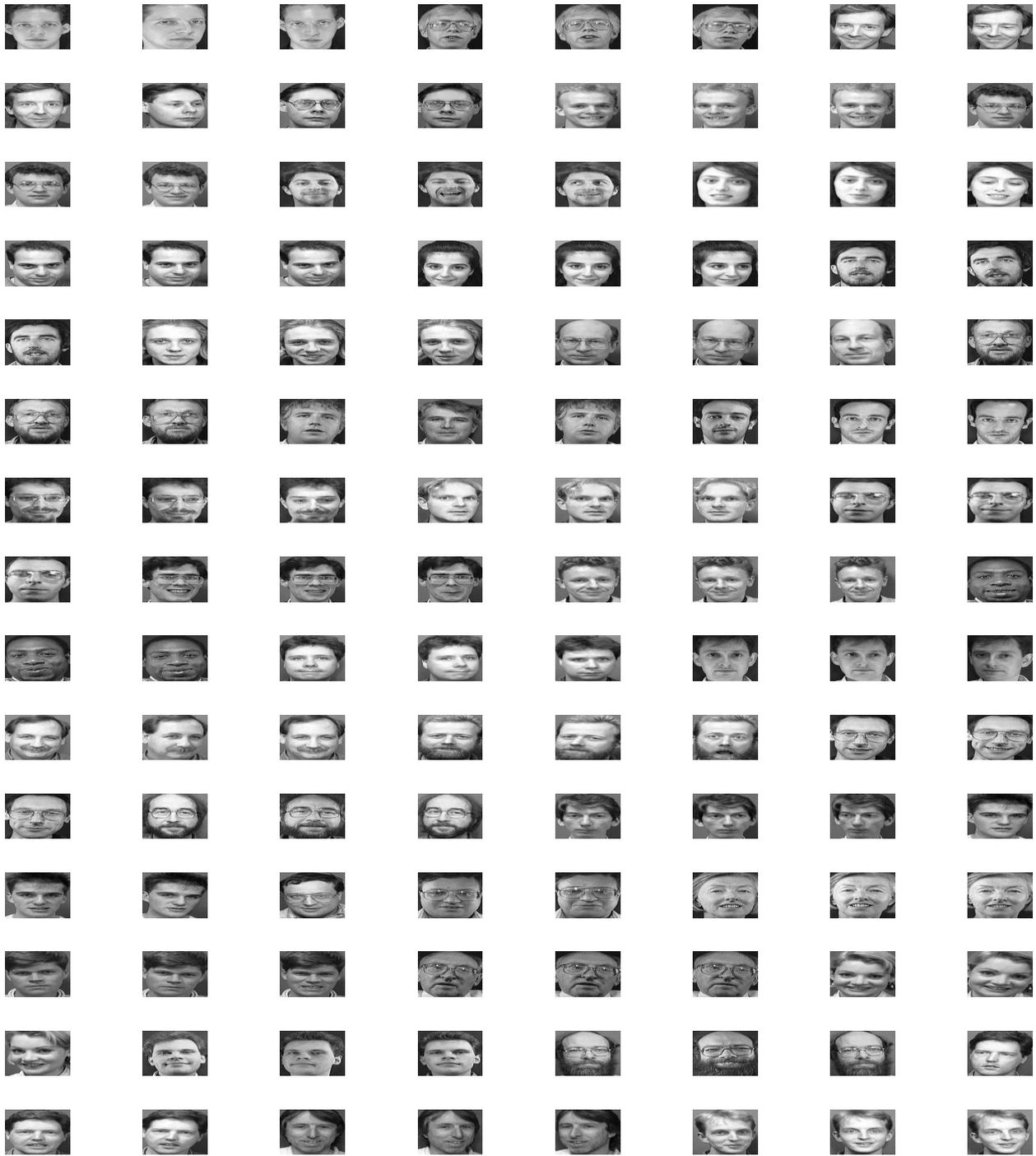


Figura 6.5: Imagens ordenadas por linha (de 1 a 120).



Figura 6.6: Imagem-teste: *teste3*.

As Tabelas 6.9, 6.10, 6.11 e 6.12 ajudam na confirmação do que constatamos anteriormente: a rotina OVO sozinha tem uma probabilidade muito menor de ser bem sucedida do que a rotina OVO acoplada ao PCA. Houve um caso raro na Tabela 6.10 em que de  $p = 4$  para  $p = 5$ , a  $p$ -ésima distância diminuiu um pouco (um valor desprezível, na verdade, para a rotina acoplada que deve ter grandes valores para as distâncias), embora os resultados práticos continuem bons.

$x_{sol}$	$p$	Valor da $p$ -ésima distância	Imagem reconhecida mais próxima	Valor da menor distancia
$(1, 1, 0, 23, 28)^T$	1	$1.121e-009$	11	$1.121e-009$
$(1, 1, 0, 23, 28)^T$	2	$2.398e-008$	11	$1.121e-009$
$(1, 1, 0, 23, 28)^T$	3	$2.398e-008$	11	$1.121e-009$
$(1, 1, 0, 23, 28)^T$	4	$1.537e-005$	11	$1.121e-009$
$(1, 1, 0, 23, 28)^T$	5	$1.537e-005$	11	$1.121e-009$
$(1, 1, 0, 23, 28)^T$	30	$5.535e-004$	11	$1.121e-009$
$(1, 1, 0, 23, 28)^T$	60	0.005	11	$1.121e-009$
$(1, 1, 0, 23, 28)^T$	90	0.015	11	$1.121e-009$
$(1, 1, 0, 23, 28)^T$	120	0.369	11	$1.121e-009$

Tabela 6.9: Rotina OVO - Imagem-teste: *teste3* -  $x_a = (1; 1; 0; NC/2; NL/2)$  - banco com 120 imagens -  $p = 3$  .

$x_{sol}$	p	Valor da $p$ -ésima distância	Imagem reconhecida mais próxima	Valor da menor distancia
$(0.996, 0.996, -0.002, 45.910, 55.890)^T$	1	7.492	8	7.492
$(0.984, 0.9841, -0.008, 45.640, 55.562)^T$	2	8.265	8	7.767
$(0.937, 0.937, -0.032, 45.562, 55.250)^T$	3	10.956	8	10.080
$(0.937, 0.937, -0.032, 45.562, 55.250)^T$	4	12.113	7	10.080
$(0.968, 0.968, -0.016, 45.281, 55.125)^T$	5	12.085	7	8.462
$(0.937, 0.937, -0.032, 44.562, 54.250)^T$	30	16.026	7	10.082
$(0.875, 0.875, -0.065, 43.125, 52.500)^T$	60	18.128	67	11.282
$(0.996, 0.996, -0.002, 45.910, 55.890)^T$	90	20.042	8	7.492
$(0.968, 0.968, -0.016, 45.281, 55.125)^T$	120	25.048	8	8.462

Tabela 6.10: Rotina OVO acoplada ao PCA - Imagem-teste:  $teste3 - x_a = (1; 1; 0; NC/2; NL/2)$  - banco com 120 imagens -  $p = 3$  .

$x_{sol}$	p	Valor da $p$ -ésima distância	Imagem reconhecida mais próxima	Valor da menor distancia
$(2.128, -1.514, 0.05, 23, 27.999)^T$	1	$1.121e-009$	33	$1.121e-009$
$(2.128, -1.514, 0.05, 23, 27.999)^T$	2	$1.537e-005$	33	$1.121e-009$
$(2.128, -1.514, 0.05, 23, 27.999)^T$	3	$1.537e-005$	33	$1.121e-009$
$(2.128, -1.514, 0.05, 23, 27.999)^T$	4	$6.151e-005$	33	$1.121e-009$
$(2.128, -1.514, 0.05, 23, 27.999)^T$	5	$6.151e-005$	33	$1.121e-009$
$(2.128, -1.514, 0.05, 23, 27.999)^T$	30	0.003	33	$1.121e-009$
$(2.128, -1.514, 0.05, 23, 27.999)^T$	60	0.013	33	$1.121e-009$
$(2.128, -1.514, 0.05, 23, 27.999)^T$	90	0.108	33	$1.121e-009$
$(2.128, -1.514, 0.05, 23, 27.999)^T$	120	0.418	33	$1.121e-009$

Tabela 6.11: Rotina OVO - Imagem-teste:  $teste3 - x_b = (0.7; -0.4; 0.05; NC/2; NL/2)$  - banco com 120 imagens -  $p = 3$  .

$x_{sol}$	p	Valor da $p$ -ésima distância	Imagem reconhecida mais próxima	Valor da menor distancia
$(0.575, -0.525, -0.021, 43.125, 52.5)^T$	1	9.941	7	9.941
$(0.575, -0.525, -0.021, 43.125, 52.5)^T$	2	9.986	7	9.941
$(0.575, -0.525, -0.021, 43.125, 52.5)^T$	3	11.065	7	9.941
$(0.575, -0.525, -0.021, 43.125, 52.5)^T$	4	11.447	7	9.941
$(0.575, -0.525, -0.021, 43.125, 52.5)^T$	5	11.732	7	9.941
$(0.575, -0.525, -0.021, 43.125, 52.5)^T$	30	15.917	7	9.941
$(0.575, -0.525, -0.021, 43.125, 52.5)^T$	60	17.539	7	9.941
$(0.637, -0.462, 0.014, 44.562, 54.250)^T$	90	20.045	8	8.766
$(0.684, -0.415, 0.041, 45.640, 55.562)^T$	120	25.277	9	11.348

Tabela 6.12: Rotina OVO acoplada ao PCA - Imagem-teste:  $teste3 - x_b = (0.7; -0.4; 0.05; NC/2; NL/2)$  - banco com 120 imagens -  $p = 3$  .

# Capítulo 7

## Conclusões

Este trabalho consistiu da proposição de uma nova estratégia para o problema do reconhecimento facial utilizando a abordagem da Otimização do Valor Ordenado. O Algoritmo 3.1.1 foi usado para o reconhecimento.

Os resultados numéricos evidenciam a necessidade de uma rotina de pré-processamento que ressalte as características peculiares de cada face. Para tanto, acoplou-se o Algoritmo 3.1.1 ao Algoritmo 4.4, este sendo muito bem sucedido em relação ao Algoritmo 3.1.1 aplicado isoladamente, mesmo para um banco de imagem mais heterogêneo.

A principal contribuição deste trabalho é uma nova estratégia de otimização para o problema do reconhecimento facial que pode ou não ser aplicada juntamente com a técnica PCA. Como a abordagem OVO mostrou-se muito suscetível ao conjunto de parâmetros iniciais, utilizamos a técnica PCA, e apesar de não sanar totalmente o problema, os resultados numéricos evidenciam uma menor sensibilidade ao ponto inicial, apresentando êxito no reconhecimento em muitos casos.

Dos resultados numéricos nós concluímos que a abordagem OVO acoplada à técnica PCA mostrou-se uma poderosa ferramenta, quando os parâmetros iniciais estão ligeiramente próximos da solução ideal, mostrando que se trata de um problema sensível a tais parâmetros iniciais.

Essa modelagem proposta é inédita, uma vez que desconhece-se na literatura acadêmica uma aplicação da abordagem OVO ao reconhecimento facial (embora já existam algumas aplicações de tal problema em outras áreas da ciência).

Embora não tenhamos argumentos teóricos rigorosos para apoiar o ponto de vista de que esta estratégia desenvolvida neste trabalho é a melhor possível para a abordagem OVO (e provavelmente existe outra abordagem melhor ainda não elaborada), uma experimentação numérica particular (aqui detalhada) sugere que o uso da formulação acoplada é uma estratégia competitiva, visto que todos os métodos conhecidos atualmente apresentam restrições de funcionamento como influência da luminosidade, expressões faciais provindas de estímulos-resposta (condicionamentos), etc.

Para o procedimento de ajuste das imagens foram utilizadas transformações clássicas da geometria (rotação e translação), compondo uma função objetivo que depende de 5 parâmetros. Neste trabalho nos concentramos apenas em dois tipos de transformações geométricas, e provavelmente em um futuro próximo haverá uma extensão da modelagem desenvolvida neste trabalho, englobando também operadores de escala. O algoritmo local introduzido em [2] não é suficiente para o reconhecimento de maneira mais eficaz, portanto a técnica PCA foi introduzida junto a abordagem OVO.

Acredita-se que grande parte dos testes que não obtiveram êxito tem como causa o fato da luminosidade influenciar no reconhecimento. Por ordenar primeiro as funções e depois otimizar, o Algoritmo 3.1.1 confunde características errôneas caso a intensidade de coloração dos pixels da imagem-teste e alguma imagem do banco estejam próximas (olhos pretos e cabelos pretos; fundo preto e cor de pele negra; etc.) ao deslocar e rotacionar e, logo em seguida otimizar, sendo necessárias portanto um melhoramento da abordagem proposta nesse sentido.

# Apêndice A

## O Método Potencial

Como descrito na Seção (1.3.1), o método potencial foi um dos primeiros métodos que se utilizou para encontrar componentes principais. Em sua forma simplificada, o método potencial é uma técnica para encontrar o maior autovalor e seu correspondente autovetor de uma matriz  $T_{p \times p}$ . A idéia consiste em escolher um vetor  $p$ -dimensional  $u_0$ , e formar a sequência:

$$\begin{aligned}u_1 &= Tu_0 \\u_2 &= Tu_1 = T^2u_0 \\&\vdots \\u_r &= Tu_{r-1} = T^ru_0 \\&\vdots\end{aligned}$$

Se  $v_1, v_2, \dots, v_p$  são autovetores de  $T$ , então eles formam uma base para um espaço  $p$ -dimensional, e pode-se escrever, para  $u_0$  arbitrário:

$$u_0 = \sum_{k=1}^p \sigma_k v_k$$

para algum conjunto de constantes  $\sigma_1, \sigma_2, \dots, \sigma_p$ . Então

$$u_1 = Tu_0 = \sum_{k=1}^p \sigma_k T v_k = \sum_{k=1}^p \sigma_k \lambda_k v_k,$$

onde  $\lambda_1, \lambda_2, \dots, \lambda_p$  são autovalores de  $T$ . Seguindo o processo para  $r = 2, 3, \dots$  obtemos

$$u_r = \sum_{k=1}^p \sigma_k \lambda_k^r v_k$$

e

$$\frac{u_r}{\sigma_1 \lambda_1^r} = \left( v_1 + \frac{\sigma_2}{\sigma_1} \left( \frac{\lambda_2}{\lambda_1} \right)^r v_2 + \dots + \frac{\sigma_p}{\sigma_1} \left( \frac{\lambda_p}{\lambda_1} \right)^r v_p \right).$$

Se  $|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_p|$ , diz-se que  $\lambda_1$  é *autovalor dominante*. Segue que com a normalização conveniente tem-se que  $u_r \rightarrow v_1$  quando  $r \rightarrow \infty$ .

Então, para o método convergir é necessário que  $\lambda_1$  seja autovalor dominante e que  $u_0$  tenha uma componente na direção do correspondente autovetor  $v_1$ .

O método potencial fornece um algoritmo simples para encontrar o maior autovalor da matriz de covariância e seu correspondente autovetor. Ele funciona bem se  $\lambda_1 \gg \lambda_2$ , mas converge vagarosamente se  $\lambda_1$  está próximo de  $\lambda_2$ . A velocidade de convergência depende do vetor inicial  $u_0$ ; a convergência é mais rápida quando  $u_0$  está próximo de  $v_1$  [26]. Se  $\lambda_1 = \lambda_2 > \lambda_3$ , um argumento similar para o problema pode ser consultado em [33].

# Apêndice B

## O caso da Prosopagnosia

Em relação aos seres humanos, há casos raros de certas deficiências em que a capacidade de reconhecer rostos é perdida, tal como a prosopagnosia. Esta doença foi descoberta no *front* de batalha em 1944, durante a 2ª Guerra Mundial, quando um soldado nazista se feriu e alguns estilhaços de bomba causaram lesões cerebrais. Este foi tratado pelo neurologista Joachim Bodamer, que fez uma operação para remover os estilhaços e depois aplicou um teste para avaliar o estado do paciente. O médico simplesmente pediu para que a esposa do soldado vestisse um uniforme de enfermeira e ficasse à frente dele junto a outras enfermeiras reais.

O resultado disto foi que o soldado não identificou diferença alguma entre as mulheres uniformizadas, ou seja, o soldado não identificou a própria esposa. Então chegou-se a conclusão que o paciente estava normal, porém não identificava mais rostos. O médico criou o termo prosopagnosia: uma junção de *prosopo* (rosto) e *agnosia* (sem conhecimento). Atualmente foi descoberto que a prosopagnosia também pode ser genética e afetar pessoas que não sofreram traumas cerebrais [53].

Contudo esses indivíduos são capazes de identificar pessoas usando outros centros do cérebro, como o centro responsável pelo reconhecimento de padrões (sem essa deficiência) usada para reconhecer objetos. As imagens abaixo foram retiradas de [43].



Figura B.1: Prosopagnosia.

# Referências Bibliográficas

- [1] T. W. Anderson. Asymptotic theory for principal component analysis. *Ann. Math. Statist.* 34, pp. 122-148 (1963).
- [2] R. Andreani, C. Dunder and J. M. Martínez. Order-Value Optimization: formulation and solution by means of a primal Cauchy method. *Mathematical Methods of Operations research* 58, pp. 387-399 (2003).
- [3] R. Andreani, C. Dunder and J. M. Martínez. Nonlinear-Programming Reformulation of the Order-Value Optimization Problem. *Mathematical Methods of Operations Research* 61, pp. 365-384 (2005).
- [4] R. Andreani, J. M. Martínez, M. Salvatierra and F. Yano. Global Order-Value Optimization by means of a multistart harmonic oscillator tunneling strategy. *Global Optimization: from Theory to Implementation*, edited by L. Liberti and N. Maculan, Kluwer, pp. 379-397 (2006).
- [5] R. Andreani, J. M. Martínez, M. Salvatierra and F. Yano. Quasi-Newton methods for Order-Value Optimization and Value-at-Risk calculations. *Pacific Journal of Optimization* 2, pp. 11-33 (2006).
- [6] T. Ahonen, A. Hadid and M. Pietikainen. Face recognition with local binary patterns. *In Proceedings of the European Conference on Computer Vision*, pp. 469-481, Praga, Czech (2004).
- [7] AT&T - Face Database, <http://www.uk.research.att.com/facedatabase.html>
- [8] J. C. Bartlett and J. Searcy. Inversion and configuration of faces. *Cognitive Psychology* 25, pp. 281-316 (1993).
- [9] P. V. Beek, M. Reinders, B. Sankur and J. V. D. Lubbe. Semantic segmentation of video-phone image sequences. *SPIE'92 Vol. 1818 Visual Communications and Image Processing Boston, USA*, pp. 1182-1193 (1992).
- [10] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19(7):711-720 (1997).

- [11] M. T. Berg, M. J. V. Kreveld, M. Overmars and O. S. Cheong. A procedure for generations of three-dimensional halftone computer graphics presentations. *Communications of the ACM* (1970).
- [12] M. Bichsel and A. P. Pentland. Human and machine recognition and the face image set's topology. *CVGIP: Image Understanding* 59 pp. 254-261 (1994).
- [13] I. Biederman and P. Gerhardstein. Recognizing depth-rotated objects: evidence and conditions for three-dimensional viewpoint invariance. *J. Exp. Psychol. Hum. Percep. Perform.* 19, pp. 1162-1183 (1993).
- [14] V. Blanz and A. J. O'Toole. A morphable model for the synthesis of 3D faces. *In SIG-GRAPH'99 Proceedings, ACM: Computer Society Press*, pp. 187-194 (1999).
- [15] W. W. Bledsoe. The model method in facial recognition. Tech. rep. PRI:15, Panoramic Research Inc, Palo Alto, CA.
- [16] W. L. Braje, D. L. Kersten, M. J. Tarr and N. F. Troje. Illumination effects in face recognition. *Psychobiology* 26, pp. 371-380 (1999).
- [17] R. Brunelli and T. Poggio. Face Recognition: Features versus templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(10): pp.1042-1052 (1993).
- [18] V. Bruner and R. Tagiuri. The perception of people. *In Handbook of Social Psychology, Vol. 2, G. Lindzey.* Addison - Wesley, Reading MA, pp. 634-654 (1954).
- [19] <http://pcworld.uol.com.br/noticias/2009/05/25/projeto-brasileiro-de-reconhecimento-facial-e-premiado-na-italia/>
- [20] R. Chellappa, C. Wilson, and S. Sirohey. Human and machine recognition of faces: A survey. *Proceedings of IEEE* 83, pp.705-740 (1995).
- [21] P. J. Ekman and W. V. Friesen. The facial action coding system: A technique for the measurement of facial movement. Consulting Psychology Press, San Francisco (1978).
- [22] L. Farkas and I. Munro. Anthropometric facial proportions in medicine. Charles C. Thomas, Publisher Ltd (1987).
- [23] K. Fukunaga. Introduction to statistical pattern recognition. Academic Press, Boston, 2nd edition (1990).
- [24] D. S. Furl, P. J. Phillips and A. J. O'Toole. Face recognition algorithms as models of the other-race effect. *Cognitive Science* 96, pp. 1-19 (2002).
- [25] M. A. Girshick. Principal Components. *J. Amer. Statist. Assoc.*, 31, pp. 519-528 (1936).
- [26] G. H. Golub and C. F. V. Loan. Matrix Computations. 3<sup>a</sup>. The Johns Hopkins University Press, Baltimore and London (1996).

- [27] V. Govindaraju. Location human faces in photographs. *Int. J. Comput. Vision*, vol. 19, 2, pp.129-146, ISSN: 0920-5691 (1996).
- [28] J. C. Gower. Some distance properties of latent root and vector methods used in multivariate analysis. *biometrika* 53, pp. 325-338 (1966).
- [29] G. D. Guo, S. Z. Li and K. L. Chan. Face recognition by support vector machines. *In Proc. Fourth IEEE Int. Conf. on Automatic Face and Gesture Recognition*, pp. 196-201, Grenoble (2000).
- [30] G. Hadley. Álgebra linear. Forense - Universitária. Rio de Janeiro (1974).
- [31] R. Herpers, H. Kattner, H. Rodax and G. Sommer. Gaze: An attentive processing strategy to detect and analyze the prominent facial regions. *International workshop on face and gesture recognition* (1995).
- [32] H. Hotelling. Analysis of a complex of statistical variables into principal components. *J. Educ. Psychol.* 24, pp. 417-441, pp. 498 - 520 (1933).
- [33] H. Hotelling. Simplified calculation of principal components. *Psychometrika* 1, pp. 27-35 (1936).
- [34] A. F. Izmailov e M. V. Solodov. Otimização, Vol.1: Condições de otimalidade, elementos de análise convexa e de dualidade. IMPA, Rio de Janeiro (2005).
- [35] J. N. R. Jeffers. Two cases studies in the application of principal component analysis. *Appl. Statist.*, 16, pp. 225 - 236 (1967).
- [36] A. K. Jain. Fundamentals of digital image processing. Prentice-Hall, Inc. (1989).
- [37] Computer recognition of human faces. Birkhauser, Basel, Switzerland, and Stuttgart, Germany (1973).
- [38] M. D. Kelly. Visual identification of people by computer. Tech. Rep. AI - 130, Stanford AI Project, Standford, CA (1970)
- [39] R. Kjeltdsen and J. Kender. Finding skin in color images. *Proceedings of the 2nd International Conference on Automatic Face and Gesture Recognition(FG'96)*, pp.321, IEEE Computer Society, ISBN:0-8186-7713-9 (1996).
- [40] H. Kruppaand, M. Bauer and B. Schiele. Skin patch detection in real-world images. Percerp-tual Computing and Computer Vision Group, ETH Zurich, Switzerland (2002).
- [41] M. Lades, J. Vorbruggen, J. Buhmann, J. Lange, C. von der Malsburg, R. P. Wurtz and W. Ronen. Distortion invariation object recognition in the dynamic link architecture. *IEEE Transactions on Computers*, 42 pp. 300-311 (1993).
- [42] D. S. Lindsay, P. C. Jack and M. A. Christian. Other-race face perception. *Journal of Applied Psychology* 76, pp. 587-589 (1991).

- [43] Macquarie University. <http://maccs.mq.edu.au/research/projects/prosopagnosia/>
- [44] R. S. Malpass and J. Kravitz. Recognition for faces of own and other race faces. *Journal of Personality and Social Psychology* 13, pp. 330-334 (1969).
- [45] P. Meyer. Probabilidade - Aplicações à Estatística. Ao Livro Técnico, Rio de Janeiro (1974).
- [46] S. Mika, G. Ratsch, J. Weston, B. Scholkopf and K.-R. Müller. Fisher discriminant analysis with Kernels. *Neural Networks for Signal Processing IX*, pp. 41-48 (1999).
- [47] <http://www.owl.net.rice.edu/elec301/Projects99/faces/algo.html>
- [48] K. Pearson. On lines and planes of closest fit to systems of points in space. *Phil. Mag.* (6), 2, pp. 559 - 572 (1901).
- [49] P. Penev and J. Atick. Local feature analysis: A general statistical theory for object representation. *Neural Systems* 7(3), pp. 477-500 (1996).
- [50] A. Pentland, B. Moghaddam and T. Starner. View-based and modular eigenspace for face recognition. *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition s.n.* (1994).
- [51] P. J. Philips, R. M. McCabe and R. Chellappa. Biometric image processing and recognition. *In Proceedings European Signal Processing Conference.* (1998).
- [52] L. E. Pla. Análisis Multivariado: Método de componentes principales. Serie de Matemática - Monografía nº 27, Secretaria General de la Organización de los Estados Americanos (1986).
- [53] Prosopagnosia Research Centers at Harvard University. <http://www.faceblind.org>
- [54] J. P. M. Sá. Pattern Recognition - Concepts, Methods and Applications. Springer, United States (2001).
- [55] A. Pentland and M. Turk. Face Recognition using *eigenfaces*. *In Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 586-591 (1991).
- [56] C. R. Rao. The use and interpretation of principal component analysis in applied research. *Sankhya A* 26, pp. 329-358 (1964).
- [57] <http://gadiguetes.com/2009/03/09/reconhecimento-facial-ja-e-realidade-em-colegio-do-reino-unido/>
- [58] H. Rowley, S. Bajula and T. Kanade. Human face detection in visual scenes. *Advances in Neural Information Processing Systems Number 8* (1996).
- [59] T. Sakai, M. Nagao and T. Kanade. Computer analysis and classification of photographs of human faces. *Proc. First USA-JAPAN Computer Conference*, pp. 55-62 (1972).
- [60] M. Salvatierra Jr., *Otimização Global Usando Trajetórias Densas e Aplicações*, Tese de Doutorado, Unicamp, Campinas - SP (2005).

- [61] B. Scholkopf, A. Smola and K. R. Müller. Nonlinear component analysis as a Kernel eigenvalue problem. *Neural Computation* 10, pp. 1299-1319 (1999).
- [62] G. D. Smith. Numerical solution of partial differential equations: finite difference methods. Oxford: OUP (1978).
- [63] L. D. Silva, K. Aizawa, M. Hatori. Detection and tracking of facial features by using a facial feature model and deformable circular templates. *IEICE Transactions on Information and Systems, Vol. E78-D, No. 9*, pp. 1195-1207, IEEE Computer Society (1995).
- [64] M. A. Turk and A. P. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience* 3(1), pp. 71-86 (1991).
- [65] L. Wiskott, J.-M. Fellous, N. Kruger and C. V. D. Malsburg. Face recognition by elastic bunch graph matching. *7th Conference on Computer Analysis of Images and Patterns, CAIP'97, Kiel*, pp. 456-463, Springer-Verlag (1997).