

UNIVERSIDADE FEDERAL DE SÃO CARLOS
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM
CIÊNCIA DA COMPUTAÇÃO

**Análise de Estruturas de Rede Neocognitron para
Aplicação no Reconhecimento Facial**

Cristiane Oliveira de Santana

São Carlos
Agosto/2008

**Ficha catalográfica elaborada pelo DePT da
Biblioteca Comunitária da UFSCar**

S232ae

Santana, Cristiane Oliveira de.

Análise de estruturas de rede neocognitron para aplicação no reconhecimento facial Cristiane Oliveira de Santana. -- São Carlos : UFSCar, 2008.

82 f.

Dissertação (Mestrado) -- Universidade Federal de São Carlos, 2008.

1. Processamento de imagens. 2. Redes neurais. 3. Rede neural neocognitron. 4. Reconhecimento facial. I. Título.

CDD: 006.42 (20^a)

Universidade Federal de São Carlos
Centro de Ciências Exatas e de Tecnologia
Programa de Pós-Graduação em Ciência da Computação

**“Análise de Estruturas de Rede Neocognitron para
Aplicação no Reconhecimento Facial”**

CRISTIANE OLIVEIRA DE SANTANA

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de São Carlos, como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação.

Membros da Banca:



Prof. Dr. José Hiroki Saito
(Orientador – DC/UFSCar)



Prof. Dr. Nelson Delfino d'Ávila Mascarenhas
(DC/UFSCar)



Prof. Dr. João Batista Destro Filho
(UFU)

São Carlos
Agosto/2008

Dedico a realizaço deste sonho
a toda minha famlia.

Agradecimentos

Inicialmente, agradeço a Deus pela oportunidade de crescimento concedida e por me dar forças para superar todas as barreiras.

A minha mãe e meu padrasto por todo o apoio diante das dificuldades, sem vocês não teria chegado até aqui.

Ao prof. Saito pelos ensinamentos, pela paciência e por ter acreditado em meu potencial.

A meu filho Guilherme por ter se comportado desde a gestação me deixando prosseguir com este sonho. Eu te amo!

Aos meus grandes amigos: Deyse, Jú, Laia, Li, Márcio, Mila, Paula, Rê, Ricardo, Tati e Val por todo apoio durante minha gestação e pelo amor dedicado ao meu filho... não sei o que seria de mim sem vocês (Ah...está em ordem alfabética para não dar problema depois!).

A família Trevelin pela acolhida... vocês são muito importantes para mim. Verinha, você é como uma mãezona.

Ao grupo GEME (Grupo de Estudos do MEstrado) pela amizade e companhia...altas discussões do Lema do bombeamento...hahaha

Aos amigos gapinianos pelas conversas, risadas e discussões. Além da excelente companhia no Restaurante Universitário.

A todos os amigos encontrados no Departamento de Computação e no PPGCC da UFS-Car.

A FAPESB pelo apoio concedido através da bolsa de mestrado e ao prof. Simon Baker da *Carnegie Mellon University* pela gentileza de ter enviado a base de imagens *CMU-PIE database*, utilizada neste trabalho.

Se esqueci alguém, me perdoe, são tantos para agradecer que precisaria de mais um capítulo para descrever todo o carinho recebido durante o mestrado. Contudo, saibam que todos possuem um lugar garantido em meu coração.

Meu sincero agradecimento.

*"Sonho que se sonha só é só
um sonho que se sonha só, mas
sonho que se sonha junto é rea-
lidade."*

Raul Seixas

Resumo

O presente trabalho consiste na utilização da combinação de redes neocognitron na tarefa de reconhecimento facial integrando um sistema de reconhecimento facial, categorizado como método holístico, pelo fato de abordar a face como um todo na extração das características da imagem de entrada. A utilização da face como atributo biométrico no reconhecimento de indivíduos tem crescido principalmente devido a sua característica de não necessitar da cooperação do indivíduo para análise. Os grandes desafios na área de reconhecimento facial trata de sua aplicação em ambientes não-controlados em que a variação de iluminação e pose podem degradar o desempenho do sistema. A fim de trabalhar os desafios (variação de iluminação e distorção nos padrões) utilizou-se imagens frontais da base CMU-PIE (*Carnegie Mellon University - Pose, Illumination and Expression*) aproveitando sua característica de variação da iluminação e expressão. O modelo neural aplicado ao trabalho é a rede neocognitron que é capaz de reconhecer padrões sem que sua capacidade de reconhecimento seja afetada por deformações, mudanças no tamanho ou posição do padrão de entrada. Com o objetivo de obter a estrutura da rede neocognitron mais adequada à tarefa de reconhecimento facial, dentro de um conjunto de estruturas, foram analisadas algumas estruturas de rede neocognitron com uma saída e com diferentes resoluções de imagens. O melhor resultado obtido consiste na taxa de reconhecimento de 78% para um conjunto com 30 classes com trinta e seis padrões cada na fase de reconhecimento. Para a melhoria dos resultados aplicou-se a combinação de classificadores utilizando: *Decision Templates* e o método *Decision Templates Modificado* desenvolvido neste trabalho. O desempenho dos classificadores foi analisado através da estimação de erro pelo método *hold-out* e pelo coeficiente Kappa. Os resultados finais apontaram que a combinação de classificadores, aplicados a este modelo não resultou em melhoras significativas devido às características inerentes ao modelo aplicado.

Abstract

The present work consists in the use of the combination of neocognitron networks for the face recognition tasks integrating a system of face recognition, categorized as holistic method, by the fact of to approach all the face in the extration of the characteristics of the input image. The use of the face as biometric attribute in the recognition of individuouss has grown mainly to its characteristic that dispense their cooperation for analysis. The great challenges in the area of face recognition are about the application in non-controlled environments in which the variation of illumination and pose can decrease the performance of the system. In order to work the challenges (variation of illumination and distortion in the patterns) it was used frontals images of CMU-PIE (University Carnegie Mellon - Pose, Illumination and Expression) database using the advantage of illumination and expression variations. The neural network model applied in the work is the neocognitron network that is capable to recognize patterns without its capacity of recognition been affected by deformations, changes in the size or position, of the input pattern. With the goal of to get the structure more adjusted of the neocognitron for the face recognition task, inside of a set of structures, it was analyzed some structures of neocognitron network with one output class and with different images resolutions. The best result corresponds to a recognition rate of 78% for a set with 30 classes with thirty and six patterns each one in the recognition phase. To improve the results it was applied combination of classifiers using: *Decision Templates* and the Modified *Decision Templates* method developed in this work. The performance of the classifiers was analyzed through the error estimation using the hold-out method and the Kappa coefficient. The final results pointed that the combination of classifiers applied to this model did not result in significant improvements due the inherent characteristics of the applied model.

Sumário

1	Introdução	1
1.1	Contexto, motivação e definição do problema	1
1.2	Organização do trabalho	4
2	Reconhecimento Facial	6
2.1	Contexto e características	6
2.2	Processo de reconhecimento facial	9
2.3	Técnicas utilizadas no reconhecimento facial	11
2.4	Desafios da área	14
2.5	Bancos de dados	15
2.6	Avaliações	17
2.6.1	Protocolo FERET	18
2.6.2	Protocolo FRVT 2002	18
2.6.3	Protocolo FRVT 2006	19
2.7	Considerações Finais	20
3	Redes Neurais Artificiais	21
3.1	Introdução	21
3.2	Breve histórico	22
3.3	Motivação: redes biológicas	23

3.4	O neurônio artificial	24
3.5	Arquitetura	26
3.6	Aprendizado	26
3.7	Exemplos de redes neurais	27
3.7.1	Perceptron	27
3.7.2	MultiLayer Perceptron - MLP	28
3.7.3	Redes recorrentes	30
3.7.4	Redes auto-organizáveis	31
3.7.5	Redes convolucionais	34
3.8	Redes Neurais e reconhecimento facial	35
3.9	Considerações Finais	36
4	Rede Neocognitron	38
4.1	Breve histórico	38
4.2	Arquitetura da rede	39
4.3	Processamento	41
4.4	Treinamento não supervisionado	43
4.5	Considerações Finais	47
5	Classificadores: Combinação e medidas de desempenho	48
5.1	Combinação de classificadores	48
5.1.1	Combinadores de saída contínua	50
5.2	Medidas de desempenho	53
5.2.1	<i>Hold-out</i>	53
5.2.2	<i>Coefficiente Kappa</i>	54
5.3	Considerações finais	55
6	Proposta de sistema de reconhecimento facial usando neocognitron	56
6.1	Sistema de reconhecimento facial	57
6.2	Base de dados: CMU-PIE <i>Database</i>	60
6.3	Pré-processamento de imagens	62
6.3.1	Detecção da região facial	62
6.3.2	Redimensão: Interpolação bicúbica	63

6.4	Estrutura-base de simulação dos experimentos	64
6.5	Neocognitron multi-classes X neocognitron de uma classe	66
6.6	Considerações finais	67
7	Trabalho realizado e resultados obtidos	68
7.1	Sistema de reconhecimento de faces implementado	68
7.2	Experimentos	69
7.2.1	Experimentos variando o tamanho do padrão de entrada sem usar combinadores	70
7.2.2	Experimentos para imagens de tamanho 57x57 usando combinadores	71
7.3	Resultados	71
7.3.1	Resultado dos experimentos 7.2.1 variando o tamanho das amostras .	72
7.3.2	Resultado dos experimentos 7.2.2 usando combinadores	72
7.4	Discussões	73
7.4.1	Discussão sobre os experimentos 7.2.1	73
7.4.2	Discussão sobre os experimentos 7.2.2	74
8	Conclusões	76
8.1	Trabalhos futuros	77

Lista de Figuras

2.1 Porcentagem de rendimentos biométricos por tecnologia, 2006 (exceto AFIS/Live Scan) (LOCKIE, 2006)	7
2.2 Redução da taxa de erro para o estado-da-arte dos algoritmos de reconhecimento facial (PHILLIPS et al., 2007).	8
2.3 Fluxo do processo de reconhecimento facial (LI;JAIN, 2005)	10
3.1 Modelo de um neurônio biológico	24
3.2 Modelo do neurônio artificial	25
3.3 Gráficos das funções apresentadas na tabela 3.1 (LOESCH;SARI, 1996)	25
3.4 Topologia da rede MLP (HAYKIN, 2001)	29
3.5 Topologia da rede de Hopfield	31
3.6 Topologia dos Mapas de Kohonen	32
3.7 Raios de influência de um neurônio	33
3.8 Exemplo de uma rede neural convolucional (LAWRENCE et al., 1996)	35
4.1 Estrutura da rede Neocognitron	40
4.2 Funcionamento das conexões <i>forward</i> e <i>backward</i>	40
4.3 Fluxo de pesos	41

4.4 (A) Característica global de um padrão formada por três características locais. (B) Padrão deformado reconhecido. (C) Características locais fora dos respectivos campos de tolerância (FUKUSHIMA, 1988)	42
5.1 Obtenção dos <i>templates</i> de decisão.	53
5.2 Teste de reconhecimento	54
6.1 Fluxo de reconhecimento adaptado ao projeto realizado.	57
6.2 Processo realizado para extração dos vetores de características.	58
6.3 Processo realizado na fase de reconhecimento dos padrões.	59
6.4 Processo realizado na fase de combinação utilizando o método <i>Decision Template</i>	59
6.5 (A)CMU 3D <i>Room</i> (B) Posicionamento da cabeça, flashes e câmeras CMU 3D <i>Room</i>	60
6.6 Exemplos de variação de pose e expressão na base CMU-PIE.	61
6.7 Exemplo de código utilizado para obtenção da região facial.	62
6.8 Código utilizado para redimensionar imagens	63
6.9 Estrutura implementada pelo simulador (SAITO et al., 2005).	64
6.10 Estrutura-base dos experimentos.	65

Lista de Algoritmos

1	Algoritmo de treinamento do Perceptron (BRAGA et al., 2000)	28
2	Algoritmo da fase de propagação da MLP (BRAGA et al., 2000)	29
3	Algoritmo da fase de retropropagação da MLP (BRAGA et al., 2000)	30
4	Algoritmo de treinamento da MLP (BRAGA et al., 2000)	30
5	Algoritmo de treinamento da rede de Hopfield (BRAGA et al., 2000)	32
6	Algoritmo de treinamento da rede auto-organizável de Kohonen (BRAGA et al., 2000)	34
7	Algoritmo para cálculo das células-S de um plano celular k	45
8	Algoritmo para cálculo das células-C de um plano celular k	45
9	Algoritmo para treinamento de um estágio da rede	46

Lista de Tabelas

1.1	Típicas aplicações do reconhecimento facial (ZHAO et al., 2002)	2
2.1	Categorização de métodos de reconhecimento facial (ZHAO et al., 2002) . . .	12
2.2	Métodos utilizados para detecção facial (YANG et al., 2002)	13
2.3	Principais bancos de dados utilizados para reconhecimento facial (GROSS et al., 2005)	16
3.1	Funções de ativação mais utilizadas (LOESCH;SARI, 1996)	24
5.1	Interpretação para coeficiente Kappa	55
7.1	Estuturas utilizadas nos experimentos	70
7.2	Erro estimado e coeficiente Kappa para a estrutura de tamanho de 57x57 . .	72
7.3	Erro estimado e coeficiente Kappa para a estrutura de tamanho de 60x60 . .	72
7.4	Erro estimado e coeficiente Kappa para a estrutura de tamanho de 74x74 . .	72
7.5	Erro estimado e coeficiente Kappa para a estrutura de tamanho de 120x120	73
7.6	Erro estimado e coeficiente Kappa para a estrutura de tamanho de 160x160	73
7.7	Erro estimado para aplicação da combinação de classificadores	73
7.8	Coeficiente Kappa para aplicação da combinação de classificadores	74

Lista de Abreviaturas

AFIS: *no-Automated Fingerprint Identification System*

ATM: *Automated Teller Machine*

BD: *Banco de Dados*

BMP: *Bitmap*

CCTV: *Closed Circuit Television*

CMU-PIE: *Carnegie Mellon University - Pose, Illumination and Expression*

CGM: *Constrained Generative Model*

DLA: *Dynamic Link Architecture*

EBGM: *Elastic Bunch Graph Matching*

FAR: *False Accept Rate*

FERET: *Facial Recognition Technology*

FRR: *False Reject Rate*

FRT: *Face Recognition Techniques*

FRVT: *Face Recognition Vendor Test*

HMM: *Hidden Markov Model*

HyperBF: *Hyper Basis Function*

HCInt: *High Computational Intensity Test*

ICA: *Independent Component Analysis*

JAFEE: *Japanese Female Facial Expression*

LDA: *Liner Discriminant Analysis*

MCP: *McCulloch e Pitts*

MCInt: *Medium Computational Intensity Test*

MIT: *Massachusetts Institute Technology*

MLP: *Multi-Layer Perceptron*

ORL: *Olivetti Reserch Laby*

PCA: *Principal Component Analysis*

PPM: *Portable Pixmap*

RNA: *Rede Neural Artificial*

SSP: *Seed Selection Plane*

Introdução

1.1 Contexto, motivação e definição do problema

Os métodos convencionais de reconhecimento de usuário baseados na utilização de cartões e senhas não são altamente confiáveis. Com isso, a demanda por sistemas que trabalhem com o reconhecimento automático vem aumentando.

Neste contexto, os identificadores biométricos, por serem inerentemente confiáveis e mais capazes do que técnicas baseadas em conhecimento e senhas na identificação de pessoas autorizadas, vem ganhando atenção. Esse crescimento do interesse por esta tecnologia está também ligado ao fato de se tratar de uma tecnologia que não possui a desvantagem de esquecimento, perda ou roubo, pois se utiliza de atributos que são próprios ao indivíduo (JAIN et al., 2000).

Muitas são as tecnologias biométricas que garantem, eficientemente, a identificação automática de um indivíduo. Entretanto, em algumas áreas, como na imigração, investigação de suspeitos, entre outros, muitas vezes se faz necessário a utilização de sistemas que realizem esta verificação sem que haja a cooperação do indivíduo. Neste momento, dentre as tecnologias biométricas existentes, o reconhecimento facial destaca-se como a única que

consegue realizar verificações e identificações sem a cooperação do indivíduo além de ser considerado como uma tecnologia natural, discreta e, sobretudo, amigável (LI;JAIN, 2005).

Apesar de não existir nenhuma técnica que forneça uma solução robusta para todas as situações e diferentes áreas de aplicações do reconhecimento facial, segundo HUANG et al. (2005) trata-se de uma área considerada promissora para pequenas e médias aplicações como controle de acesso ou aplicações que combinem reconhecimento facial e *smart cards* com um pequeno banco de dados. Entretanto, trata-se de um grande desafio para aplicações de grande porte como na área de segurança em aeroportos, por exemplo, onde existe um grande número de faces a serem analisadas.

O reconhecimento de faces por máquinas é uma área de pesquisa ativa que envolve várias disciplinas como processamento de imagens, reconhecimento de padrões, visão computacional e redes neurais. Existem numerosas aplicações comerciais de Técnicas de Reconhecimentos Faciais (*Face Recognition Techniques - FRT*) tal como pode ser observado na Tabela 1.1 abaixo.

Tabela 1.1: Típicas aplicações do reconhecimento facial (ZHAO et al., 2002)

Área	Aplicação
Biometria	Carteira de motorista, imigração, identidade, passaportes, eleição.
Segurança da Informação	<i>Logon</i> de <i>desktop</i> , aplicações de segurança, segurança de dados, criptografia, segurança de intranet, acesso a internet, registros médicos
Vigilância e <i>law applications</i>	Vigilância avançada com vídeo, controle de portos, investigação, rastreamento de furtos e suspeitos.
Cartões Inteligentes	Segurança de valor armazenado, autenticação de usuário.
Controle de acesso	Acesso a instalações , acesso veicular.

Recentes avaliações de sistemas comerciais de reconhecimento facial mostram que os níveis de desempenho para verificação da face (frontal) dos melhores sistemas estão lado a lado com o reconhecimento por impressão digital, em ambiente uniformemente iluminado

(GROSS et al., 2005).

Existem muitas pesquisas na área de reconhecimento facial realizadas nos últimos anos, como em VALENTIN et al. (1997) que utilizou *Principal Component Analysis* (PCA) e redes neurais para análise de imagens faciais, WISKOTT et al. (1997) que apresenta uma técnica para o reconhecimento com apenas uma imagem por classe utilizando *Elastic Bunch Graph Matching* (EBGM) ou ainda em BIANCHINI (2001) e HIRAKURI (2003) que utilizam a rede neocognitron para reconhecimento utilizando pontos de controle para a extração de fatores localizados em posições estratégicas. Além disso, encontram-se trabalhos com a combinação de reconhecimento de expressões faciais e detecção da face aplicando redes convolucionais (MATSUGU et al., 2003), além de trabalhos mais atuais com o reconhecimento de faces tridimensional (3D) (ABATE et al., 2006) (CHEN et al., 2007). Entretanto, diante dos maiores desafios da área (variação de pose e iluminação) percebe-se que muitos trabalhos ainda devem ser desenvolvidos a fim de que seja alcançado um sistema robusto para a tarefa de reconhecimento facial.

Várias são as abordagens que podem ser utilizadas na tarefa de reconhecimento facial. A abordagem utilizando redes neurais, foco deste trabalho, tem sido aplicada com sucesso nas mais diversas formas como pode ser verificado em trabalhos anteriormente citados assim como em AL-AKKAD (2004) que realiza o reconhecimento facial baseado na estimação de curvatura e rede MultiLayer Perceptron, FÉRAUD (2001) utiliza uma nova rede neural denominada *Constrained Generative Model* (CGM). Entretanto, as redes convolucionais, por possuírem a capacidade de ser tolerante a distorções dos padrões têm sido objeto de estudo em muitas pesquisas (FUKUSHIMA; WAKE, 1992), (LAWRENCE et al., 1996), (FASEL, 2002), (FUKUSHIMA, 2004), (SAITO et al., 2005), entre outros.

A rede neocognitron é uma rede convolucional desenvolvida para o reconhecimento de caracteres manuscritos e que vem obtendo resultados satisfatórios na área de reconhecimento facial (SAITO et al., 2005).

De acordo ao cenário apresentado, e sabendo-se que os grandes desafios da área de reconhecimento de faces está na variação de iluminação e pose, este trabalho propõe a aplicação da rede neocognitron no reconhecimento facial devido à sua tolerância a padrões distorcidos¹. Num primeiro momento são analisadas as estruturas da rede neocognitron

¹Configurados neste trabalho pela variação de iluminação e expressão dos indivíduos da base de imagens

para implementação do processo de reconhecimento facial. Num segundo momento ao melhor resultado obtido é aplicada uma combinação de classificadores a fim de obter um melhor resultado da estrutura considerada como mais adequada à tarefa de reconhecimento facial dentre as analisadas.

1.2 Organização do trabalho

No primeiro capítulo é apresentada a motivação, relevância do trabalho e definido o problema. No capítulo 2 encontra-se a fundamentação teórica sobre reconhecimento facial, onde são abordadas as principais razões para seu crescimento, limitações, cenários, tarefas dentro do processo de reconhecimento facial, além de várias abordagens e técnicas encontradas.

A seguir, no terceiro capítulo, são abordadas as redes neurais. Inicialmente é apresentado um breve histórico da área, em seguida é realizada a apresentação do neurônio artificial, as características das redes neurais e realizada uma revisão sobre alguns modelos: Perceptron, MultiLayer Perceptron (MLP), redes recorrentes, redes auto-organizáveis e redes convolucionais além de realizada uma breve pesquisa sobre o relacionamento entre redes neurais e o reconhecimento facial.

O quarto capítulo descreve a pesquisa referente ao neocognitron, rede neural utilizada para o desenvolvimento do projeto exibindo sua topologia, processamento, principais características da rede e treinamento, além de limitações. Seqüencialmente, no quinto capítulo são apresentadas uma revisão sobre combinação de classificadores e as medidas de desempenho utilizadas para comparação das estruturas.

Em seguida, no sexto capítulo, é apresentado o trabalho proposto, detalhando todo o funcionamento das fases de treinamento e reconhecimento. Além disso, toda a metodologia desenvolvida no trabalho bem como a base de imagens utilizada (CMU-PIE *Database: Carnegie Mellon University - Pose, Illumination and Expression Database*) e as operações realizadas no pré-processamento das imagens. Por fim, é apresentada a implementação utilizadas e realiza uma comparação entre Neocognitron multi-classes e neocognitron de uma classe.

No sétimo capítulo é realizado uma síntese da implementação utilizada e exibidos os ex-

perimentos realizados, seus resultados e realizada uma discussão a respeito das análises realizadas. A seguir, no oitavo capítulo são apresentadas as conclusões realizadas e abordados os trabalhos futuros que podem ser desenvolvidos a partir deste projeto. Por fim são apresentadas as referências bibliográficas.

Reconhecimento Facial

Este capítulo apresenta os conceitos básicos relativos ao reconhecimento facial. São expostas suas motivações, vantagens sobre as demais tecnologias biométricas, limitações, abordagens e métodos, além de uma revisão a respeito dos bancos de dados faciais e protocolos de avaliação mais importantes.

2.1 Contexto e características

Em meio à necessidade dos indivíduos de assegurarem sua autenticidade em muitas situações cotidianas como a realização de transações financeiras, controle de acesso às dependências de empresas, etc, muitas tecnologias vem sendo estudadas e desenvolvidas.

Do ponto de vista tecnológico, os métodos biométricos têm se mostrado efetivos à demanda por sistemas relacionados ao contexto ao qual a sociedade está inserida. Segundo dados de 2006 da *Biometric Technology Today*¹ dentre os métodos biométricos existentes, os sistemas que utilizam a impressão digital e a face como atributos biométricos acumulam juntos mais de 60% dos rendimentos de sistemas biométricos (figura 2.1) não contabi-

¹A *Biometric technology Today* é uma revista especializada da área e publicada pela editora Elsevier.

lizadas suas porcentagens quando utilizados em sistemas multi-biométricos e os Sistemas de Identificação por Impressão Digital não Automatizados (AFIS- *no-Automated Fingerprint Identification System*) que não foram incluídos na pesquisa (LOCKIE, 2006).

Apesar dos dados apresentados mostrarem a face como segundo atributo biométrico de maior rendimento, deve-se observar que aplicações na área de imigração, passaportes, investigações, controle de portos, por exemplo, muitas vezes, não podem contar com a cooperação do indivíduo descartando assim métodos que utilizam a impressão digital como atributo biométrico.

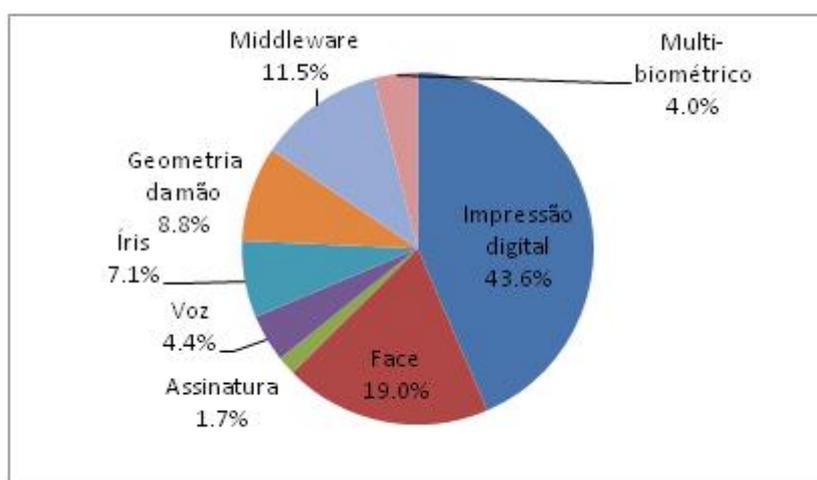


Figura 2.1: Porcentagem de rendimentos biométricos por tecnologia, 2006 (exceto AFIS/Live Scan) (LOCKIE, 2006)

Historicamente, a maioria dos trabalhos existentes formulavam os problemas de reconhecimento de objetos tridimensionais(3D) com imagens bidimensionais(2D) resultando assim em pesquisas com abordagens voltadas apenas para os problemas de reconhecimento de padrões 2D.

Na década de 70, as técnicas de classificação tipicamente utilizavam como medidas os atributos das características das faces. Em meados da década de 80, o trabalho na área de reconhecimento facial esteve enfraquecido. Entretanto, desde o início da década de 90 os interesses em técnicas de reconhecimento facial têm crescido significativamente (ZHAO;CHELLAPPA, 2006).

As opiniões são as mais variadas em relação aos fatores que desencadearam o crescimento da área. Em 1995, CHELLAPPA et al. (1995) consideravam que o crescimento

da área podia ser atribuído à ênfase em pesquisa de projetos civil/comercial; emergência de classificadores de rede neural com ênfase em computação de tempo-real e adaptação; a disponibilidade de hardware de tempo-real; e necessidade crescente de aplicações relacionadas à vigilância. Enquanto isso YANG et al. (2002) consideram que sua rápida expansão se baseia na premissa de que a identidade do usuário e sua intenção (comportamento) pode ser obtida através da análise da imagem da face e que os computadores podem gerar reações a partir desta análise. Ainda em 2002, ZHAO et al. (2002) defendia a posição de que a significativa atenção despendida à área de reconhecimento facial está no fato de que existe uma diversidade de aplicações que podem ser desenvolvidas, como visto na tabela 1.1, além da atual disponibilidade de tecnologia que oferece suporte ao seu desenvolvimento.

Nos últimos 15 anos, as pesquisas focaram no desenvolvimento de sistemas totalmente automatizados como mostra a figura 2.2 que quantifica a melhoria do reconhecimento facial em quatro marcos (PHILLIPS et al., 2007).

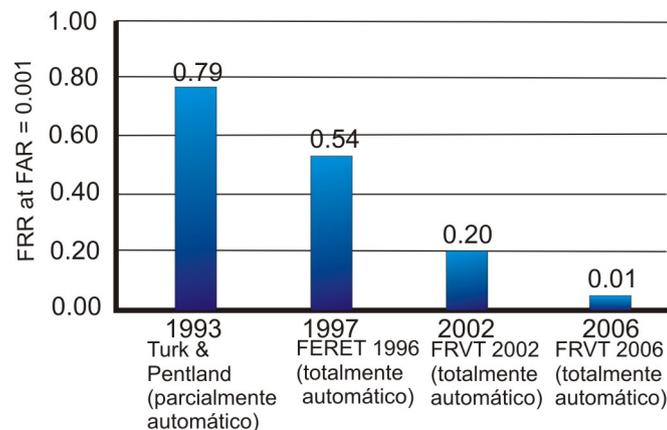


Figura 2.2: Redução da taxa de erro para o estado-da-arte dos algoritmos de reconhecimento facial (PHILLIPS et al., 2007).

Para cada marco, a taxa de falsos erros (FRR - *False Reject Rate*) e a taxa de falsos acertos (FAR - *False Accept Rate*) de 0.001 (1 em 1000) é dada por um algoritmo representativo. O marco de 1993 é uma retrospectiva da implementação do algoritmo de *eigenface* de Turk e Pentland que era parcialmente automático por requerer a coordenada dos olhos. O marco de 1997 representa a avaliação de setembro de 97 do FERET (*Face Recognition Technology*), que foi conduzida na conclusão do programa do FERET com a submissão

de algoritmos inteiramente automáticos. Em seguida, os marcos de 2002 com a avaliação FRVT 2002 (*Face Recognition Vendor Test*) e por último a avaliação do FRVT 2006. A tecnologia melhorou de algoritmos parcialmente automático aos inteiramente automáticos, quando a taxa de erro declinou aproximadamente por um terço (PHILLIPS et al., 2007).

A comunidade do reconhecimento facial beneficiou-se de uma série de esforços do desenvolvimento da tecnologia financiado pelo governo norte-americano e de ciclos de avaliações, começando com o programa do FERET em setembro 1993. Uma das contribuições e dos legados chave destes esforços de desenvolvimento são as grandes bases de dados coletadas para suportar estes esforços. As bases de dados estimularam o desenvolvimento de novos algoritmos e as avaliações independentes forneceram uma avaliação sem influências do estado-da-arte da tecnologia e têm identificado as abordagens mais promissoras.

Mais recentemente, significativos avanços têm sido realizados no reconhecimento baseado em imagens 3D que tem se mostrado como uma nova alternativa às limitações na utilização de imagens 2D, principalmente voltado ao problema de variação de pose. Em YUAN et al. (2005) foi realizado o desenvolvimento de um sistema de reconhecimento baseado em textura, através de extração 2D, e forma, através de extração 3D, dos dados apresentados identificando assim imagens faciais com variação de pose. Em ABATE et al. (2006) foi apresentado um método de reconhecimento de face 3D baseado numa esfera normal onde sua superfície esférica é utilizada para representar a curvatura local de uma malha poligonal 3D e o processo de classificação 2D é usado para conservar uma parte significativa da complexidade computacional enquanto que preserva a propriedade de invariância da rotação. Mais recentemente o trabalho de CHEN et al. (2007) utilizou o algoritmo proposto para estimar a pose da face de cada pessoa numa galeria, neste trabalho é realizada a adaptação de um modelo facial 3D para todas as faces de cada pessoa.

2.2 Processo de reconhecimento facial

Tendo como base a premissa de que um sistema de reconhecimento facial deve ser capaz de identificar faces apresentadas em imagens estáticas e vídeos, automaticamente, o cenário de reconhecimento facial pode ser classificado em dois tipos (LU, 2003):

- Verificação: dado um conjunto de imagens de faces e uma imagem de face a ser

investigada tendo sua classe declarada, a tarefa de verificação é certificar ou não a informação declarada (autenticação).

- Identificação: compara uma imagem de entrada com as imagens contidas num banco de dados de faces a fim de determinar a identidade da face em questão (reconhecimento).

O processo de reconhecimento facial é tratado como um problema de reconhecimento de padrões onde a face é um objeto tridimensional que é submetido a variações (pose, iluminação, expressões, etc.) e pode ser identificada através de imagem bidimensional (LI;JAIN, 2005).

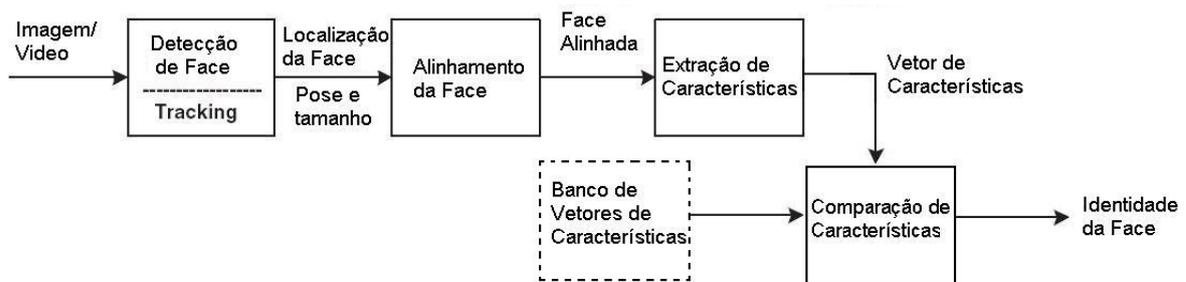


Figura 2.3: Fluxo do processo de reconhecimento facial (LI;JAIN, 2005)

Este processo geralmente consiste de quatro módulos como mostrado na figura 2.3, que são detecção, alinhamento, extração de características e comparação. No primeiro módulo é realizada a detecção da face, em que a área da face é segmentada do fundo da imagem (no caso do vídeo é feito o rastreamento da face; *tracking*). No segundo módulo é realizado o alinhamento da face, responsável por conseguir maior exatidão da localização e normalização da face. No próximo módulo é realizada a extração de características onde os componentes faciais (olhos, boca, nariz, etc.) são localizados. O módulo de extração de características gera um vetor de características que provê informações efetivas, que são utilizadas para distinguir faces de diferentes pessoas, e estáveis com respeito a variações geométricas e fotométricas. No último módulo é efetuada a comparação do vetor

de características da imagem de entrada com os vetores existentes no banco de dados de faces mostrado em pontilhado na figura 2.3. O banco de dados de faces contém vetores de características previamente extraídos das imagens. Assim, a identificação de uma face é realizada quando uma combinação (*matching*) é encontrada com suficiente confiança, caso contrário, a face é indicada como desconhecida. O resultado do reconhecimento facial depende muito das características que são extraídas para representar o padrão de face e dos métodos de classificação usados para distingui-las (LI;JAIN, 2005).

O maior problema desta tecnologia está no fato de que raramente suas condições são perfeitas, normalmente são encontrados desafios se a imagem é obscura, muito pequena, clara ou escura, ou o ângulo da pessoa dificulta a visibilidade da face, como em caixas eletrônicos (ATM -*Automated Teller Machine*) ou em redes de circuito fechado de televisão (CCTV - *Closed Circuit Television*) (LOCKIE, 2003).

2.3 Técnicas utilizadas no reconhecimento facial

As técnicas de reconhecimento facial podem ser divididas em dois grupos: técnicas estáticas e dinâmicas. As primeiras dizem respeito ao reconhecimento utilizando imagens estáticas enquanto que as técnicas dinâmicas são utilizadas com imagens de vídeo. Existem diferenças significativas entre esses grupos que irão depender da aplicação específica. Em geral, essas diferenças são em termos de qualidade da imagem, *background* da cena, disponibilidade de critérios bem definidos para a verificação, e a natureza, tipo e quantidade de entradas (ZHAO et al., 2002).

Segundo LI;JAIN (2005) as abordagens sobre reconhecimento facial em imagens estáticas podem ser agrupadas em duas áreas: comparação geométrica e comparação de padrões. Inicialmente, utilizou-se intensamente a abordagem baseada em comparação geométrica onde as características faciais como olhos, nariz, boca e queixo eram detectadas, sendo que as propriedades e relações entre estas características eram usadas como descritores para o reconhecimento facial. Contudo, estas técnicas iniciais mostraram-se limitadas por descartarem informações de textura facial e aparência, consideradas ricas em informação. No segundo grupo, comparação de padrões, as imagens faciais são representadas como uma matriz bidimensional de valores de intensidade de pixels que são

comparadas com um ou vários padrões que representam a face.

Em ZHAO et al. (2002) é realizado um detalhamento maior das abordagens sobre reconhecimento facial que podem ser categorizadas em três grandes grupos:

- Abordagem Holística: a face é identificada usando a imagem de entrada como um todo, sem localizar pontos individuais;
- Abordagem Baseada em características: as características locais da face são utilizadas como dados de entrada para classificadores estruturais; e
- Abordagem Híbrida: combinação de métodos da abordagem holística com métodos da abordagem baseada em características para reconhecer a face.

Na tabela 2.1 são apresentados algoritmos representativos para cada abordagem.

Tabela 2.1: Categorização de métodos de reconhecimento facial (ZHAO et al., 2002)

Abordagem	Exemplos de algoritmos representativos
Métodos Holísticos	<i>Eigenface</i> , <i>Fisherface</i> , Análise de Componentes Principais (PCA), Análise Discriminante Linear (LDA), Análise de Componentes Independentes (ICA), <i>Template Matching</i>
Métodos Baseados em características	Campos de Markov (HMM - <i>Hidden Markov Model</i>), Redes Neurais, <i>Dynamic link Architecture</i> (DLA)
Métodos Híbridos	<i>Elastic Bunch Graph Matching</i> (EBGM), Modelos Morfológicos <i>Wavelets</i>

Os métodos baseados em PCA e LDA têm promovido significativo avanço das técnicas de reconhecimento facial. Relatórios comparativos mostram que os métodos baseados no LDA geralmente têm resultados melhores pois é construído um subespaço discriminante a fim de distinguir, da melhor forma, faces de diferentes pessoas (LI;JAIN, 2005).

Segundo YANG et al. (2002), na fase de detecção pode-se observar quatro categorias de métodos. São elas:

- Baseados em conhecimento: codificam o conhecimento humano do que seria uma face típica. Onde as regras utilizadas capturam os relacionamentos entre as características faciais;

- Abordagens de características invariantes: apontam as características estruturais que existem mesmo quando a pose, o ponto de vista ou as condições da luz variam, utilizando-as para localizar as faces;
- *Template matching*: (comparação de modelos) Diversos padrões de uma face são armazenados para descrever a face ou as características faciais separadas. As similaridades entre uma imagem de entrada e os padrões armazenados são computadas para a detecção; e
- Baseados em aparência: Os modelos são formulados de um conjunto de treinamento das imagens que capturam a variedade representativa da aparência facial.

Dentro de cada categoria acima especificada, vários métodos vêm sendo estudados (Tabela 2.2).

Tabela 2.2: Métodos utilizados para detecção facial (YANG et al., 2002)

Abordagem	Métodos
Baseados em conhecimento	Multiresolução baseada em regras
Características invariantes	Características Faciais, textura, cor da pele, múltiplas características
<i>Templates</i>	<i>Templates</i> pré-definidos, <i>templates</i> deformáveis
Aparência	<i>Eigenfaces</i> , distribuições, redes neurais, classificador de Bayes, HMM (Hidden Markov Model)

Em ZHAO et al. (2002), encontram-se outras técnicas que vêm sendo utilizadas para outras fases do processo de reconhecimento facial. Na tarefa de extração de características, por exemplo, têm sido desenvolvidas técnicas utilizando a expansão de Karhunen-Loève, *Wavelets* de Gabor, operadores generalizados de simetria, entre outros. O autor cita ainda algumas técnicas utilizadas para atividade de reconhecimento utilizando imagens de vídeos como o método em que o rastreamento da face é feito com base na cor da pele, em seguida, a modelagem 3D é utilizada para normalizar a imagem e por fim a construção de um *eigensubspace* para realizar a análise de diferença de imagem e assim conseguir identificar uma determinada face.

A utilização das *eigenfaces* é um método bem difundido, no qual as faces do banco são representadas como um vetor de pesos. Quando é requerida a identificação de uma imagem, é feita sua representação por um vetor de pesos a fim de, através da aplicação da distância euclidiana, realizar a localização da imagem no banco. Existem pesquisas em que este método sofreu modificações e o subespaço utilizado foi restringido, utilizando medidas probabilísticas de similaridade como a análise bayesiana da imagem (ZHAO et al., 2002).

Alguns métodos utilizam a abordagem das redes neurais que são caracterizadas pelo uso de um algoritmo para determinar a similaridade entre imagens. Em CHELLAPPA et al. (1995) são citados trabalhos na área de reconhecimento facial com a utilização dos mapas de Kohonen para reconhecer uma imagem mesmo quando esta imagem é muito ruidosa ou faltam partes da mesma; ou ainda, trabalhos em que foram utilizados duas HyperBFs (*Hyper Basis Function*) interconectadas para a classificação por sexo sendo que cada rede foi treinada para o reconhecimento de um sexo, DLAs (*Dynamic Link Architecture*) que mostraram reduzir o tempo de treinamento devido a sua capacidade de permitir a discriminação de padrões com a ajuda de um conjunto padrão de detectores de características, entre outras.

Alguns trabalhos vêm sendo desenvolvidos utilizando as redes neurais convolucionais como em LAWRENCE et al. (1997) que é apresentada uma solução neural híbrida para o reconhecimento facial utilizando rede auto-organizável e rede convolucional. Em BIANCHINI (2001) foi desenvolvida uma arquitetura de rede neural para o reconhecimento facial baseado no neocognitron. Em FASEL (2002) foi utilizada uma rede neural convolucional na aplicação da análise facial. HIRAKURI (2003) aplicou a rede neocognitron especificamente no reconhecimento de atributos faciais e em MATSUGU et al. (2003) foi utilizada uma rede convolucional para a detecção de face e posterior análise de expressão facial através das características extraídas.

2.4 Desafios da área

Uma das grandes dificuldades do reconhecimento facial encontra-se na delimitação do problema (CHELLAPPA et al., 1995). Para restringir o problema a proporções práticas,

tanto a imagem de entrada quanto o espaço de busca devem possuir limites. Quanto à imagem de entrada podem ser definidos limites com o controle de qualidade da imagem, do formato e do *background*, a fim de simplificar a tarefa de segmentação. Para a limitação do espaço de busca pode-se pensar no tamanho da base de dados.

Dentre os grandes desafios da área encontram-se a variação da iluminação e pose. Segundo ZHAO et al. (2002), ambos podem causar séria degradação do desempenho do sistema. Tanto a variação da iluminação como da pose pode induzir um sistema baseado em comparação de imagens, por exemplo, a má classificação da identidade da imagem de entrada fazendo com que haja uma queda significativa no desempenho do sistema.

Pesquisas mostram que já existem algumas abordagens para trabalhar com o problema da variação da iluminação, dentre elas encontram-se métodos heurísticos, que realizam a comparação de imagens tendo como premissa que a diferença entre imagens do mesmo objeto é menor do que entre a imagem de objetos diferentes, abordagens baseadas em classes, onde são utilizadas múltiplas imagens de uma mesma classe com uma pose e diferentes iluminações, e a abordagem baseada em modelos (ZHAO et al., 2002).

Quanto ao desafio da variação da pose, algumas abordagens também vêm sendo utilizadas: múltiplas classes, na qual são utilizadas múltiplas bases da imagem de cada pessoa; a abordagem híbrida, que também utiliza múltiplas bases, porém sendo uma base por pessoa; e a abordagem por imagem única, que possui métodos que se baseiam em características de baixo-nível, características invariantes e modelos 3D (ZHAO et al., 2002).

2.5 Bancos de dados

O desenvolvimento de algoritmos robustos a variações de pose da face, iluminação, expressão facial, idade, entre outras, necessitam de um banco de dados (BD) que possua tamanho suficiente à inclusão das variações cuidadosamente controladas destes fatores, sendo necessária, para a avaliação destes algoritmos, a obtenção de bancos de dados comuns a fim de permitir comparações entre o desempenho de algoritmos.

Com o desenvolvimento de algoritmos de reconhecimento facial, um número relativamente grande de bancos de dados tem sido desenvolvido, por exemplo, *AR Database*, *BANCA Database*, *FERET (Facial Recognition Technology) Database*, *Carnegie Mellon Uni-*

iversity - Pose Illumination and Expression (CMU-PIE) Database e XM2VTS entre outros. Na Tabela 2.3 resumem-se os principais bancos de dados usados no reconhecimento facial, onde consta o nome do banco, o número de sujeitos, a quantidade de poses, a variação da iluminação, expressão, o número de sessões ao longo do tempo(*Time*) e a resolução.

Tabela 2.3: Principais bancos de dados utilizados para reconhecimento facial (GROSS et al., 2005)

Banco	Nº de Sujeitos	Pose	Iluminação	Expressão	<i>Time</i>	Resolução
AR	116	1	4	4	2	768x576
BANCA	208	1	-	1	12	720x576
CAS-PEAL	66-1040	21	9-15	6	2	360x480
CMU Hyper	54	1	4	1	1-5	640x480
CMU-PIE	68	13	43	3	1	640x480
Equinox IR	91	1	3	3	1	240x320
FERET	1199	9-20	2	2	2	256x384
KFDB	1000	7	16	5	1	640x480
MIT	15	3	3	1	1	120x128
MPI	200	3	3	1	1	256x256
ND HID	300+	1	3	2	10/13	720x480
ORL	10	1	-	-	-	925x110
UMIST	20	-	1	-	1	220x220
U. Texas	284	-	1	-	1	720x480
U. Oulu	125	1	16	1	1	428x569
XM2VTS	295	-	1	-	4	720x576
Yale	12	1	3	6	1	320x243
Yale B	10	9	64	1	1	640x480

Muitos BDs são feitos sob medida para uma necessidade específica do algoritmo em desenvolvimento. Atualmente existem bancos para tarefas específicas como detecção de face, análise de expressão facial, além dos bancos para o reconhecimento facial.

Os bancos para detecção facial devem conter imagens consideradas faces e não-faces, o que torna a coleta uma tarefa difícil, mas não impossível de ser realizada, pois a caracterização de imagens não-face requer dedicação. Para este propósito normalmente tem sido utilizados os bancos como o FERET, MIT (*Massachusetts Institute Technology*), ORL (*Olivetti Reserch Lab*) Database e AR Database. Outro recurso que vem sendo utilizado é a *World Wide Web* e os arquivos da *Washington University* (GROSS et al., 2005).

Os trabalhos para coletar uma base de dados que exemplifique o espaço da variedade de expressões estão concentrados em dois grupos: os que utilizam as seis emoções básicas definidas por Ekman e Friesen (1971) (felicidade, tristeza, medo, revolta, surpresa e raiva) e os que têm focado na extração de descrições mais suaves de expressões. São bancos

utilizados na análise de expressões faciais o *Japanese Female Facial Expression* (JAFEE) e *University of Maryland Database* (GROSS et al., 2005).

2.6 Avaliações

A avaliação e *benchmarking* das teorias e técnicas que são aplicáveis ao reconhecimento facial é crucial devido, principalmente, a grande variedade de técnicas disponíveis (ZHAO et al., 2002). Contudo, alguns requisitos são necessários à avaliação de um sistema de reconhecimento facial. Segundo YANG et al. (2002), é importante utilizar um padrão e um conjunto de testes representativo na avaliação do desempenho de um método. No caso de reconhecimento em sistemas de tempo-real é importante considerar quais os recursos computacionais necessários para que não haja sacrifício do desempenho do método empregado e de quanto é o tempo gasto em treinamento.

ZHAO et al. (2002) ainda afirma que para uma avaliação adequada é necessário um grande conjunto de imagens testes e que as imagens que o compõem sejam similares estatisticamente. Quanto à pontuação na avaliação, o autor afirma que a mesma deve ser realizada de forma a refletir o custo do erro no reconhecimento.

Em CHELLAPPA et al. (1995) é feito um levantamento dos requisitos necessários à avaliação:

- Restrições em amostras;
- Velocidade e requisitos de hardware: sugere-se que as avaliações sejam realizadas em diversos tipos de hardware;
- Interface humana; e
- Exatidão dos requisitos.

A seguir são apresentados, ainda, segundo CHELLAPPA et al. (1995) alguns métodos de avaliação:

- Tamanho do banco e uniformidade;
- Tamanho da amostra;
- Método teste para a exatidão do algoritmo e probabilidade de combinação;

- Medidas de Similaridade; e
- *Rank* de comparação estatística

A estimativa do *status* das técnicas de reconhecimento facial era dificultada pela inexistência de um protocolo de avaliação comum que possuísse um grande banco e métodos de avaliação padrão. Alguns protocolos foram criados e dentre eles o FERET (ZHAO et al., 2002) (PHILIPS et al., 2005), o XM2VTS (ZHAO et al., 2002) e posteriormente o FRVT (Face Recognition Vendor Test) (PHILIPS et al., 2005).

Um protocolo de avaliação reúne os princípios a serem utilizados no projeto e condução de uma avaliação provendo os detalhes da implementação da avaliação (PHILIPS et al., 2005).

2.6.1 Protocolo FERET

O FERET tem como metas medir a performance das técnicas de reconhecimento facial, aprimorá-las e aumentar sua base de imagens. Seu banco de dados é dividido em duas partes: *development*, que está disponível para pesquisadores para o desenvolvimento de algoritmos e *sequestered*, para avaliação independente e teste de algoritmos. A primeira avaliação com o FERET aconteceu em agosto de 1994 e estabeleceu a linha-base para os algoritmos de reconhecimento facial além de projetar as medidas de desempenho dos algoritmos que podiam localizar, normalizar e identificar faces automaticamente. A segunda avaliação ocorreu em março de 1995 e consistiu de um único teste que mediu o desempenho na tarefa de identificação. Sua terceira e última avaliação foi realizada entre setembro de 1996 e março de 1997 e avaliou que o desempenho depende da categoria de investigação além da descoberta de que o cenário também possui um impacto no desempenho do algoritmo (PHILIPS et al., 2005).

2.6.2 Protocolo FRVT 2002

O FRVT 2002 é um protocolo de avaliação baseado no FERET e foi projetado para ser utilizado em avaliações de biometria em geral não apenas nos algoritmos de reconhecimento facial. Seu objetivo inicial era prover uma medida de desempenho para estimar a habilidade dos sistemas automáticos de reconhecimento facial conhecendo as exigências

do mundo real. Está dividido em duas partes: teste de intensidade computacional alta (HCInt) e média (MCInt), sendo que seu carro-chefe é o HCInt.

Sua avaliação mostrou que dado um razoável controle da iluminação, o atual estado da arte no reconhecimento facial era 90% para verificação e 1% para taxa de falsos acertos. Em imagens externas, o reconhecimento facial mostrou-se um desafio que necessita de pesquisa. Foi percebido que o uso de modelos morfológicos melhora significativamente o reconhecimento de faces não-frontais e que o desempenho na identificação decresce linearmente ao logaritmo do tamanho da galeria. Descobriu-se ainda que em aplicações de reconhecimento facial as características tais como idade e o sexo podem afetar significativamente o desempenho do reconhecimento (PHILIPS et al., 2005).

2.6.3 Protocolo FRVT 2006

Em geral o FRVT 2006 documentou o progresso no reconhecimento facial em quatro tópicos: obteve um desempenho melhor do que o FRVT 2002; estabeleceu a primeira avaliação de reconhecimento com imagens 3D; mostrou um progresso significativo no reconhecimento em imagens com variação de condições de iluminação e mostrou que existem algoritmos de reconhecimento que possuem desempenho melhor do que de seres humanos.

Especificamente, o objetivo era melhorar o desempenho das taxas FRR (*False Reject Rate*) para 0.02 e de FAR (*False Accept Rate*) para 0.001 em imagens sob condições controladas de iluminação. Entretanto, para bases com imagens de alta resolução esta taxa foi melhor do que o objetivo, alcançando uma taxa de falsos erros de 0.01. Em se tratando das imagens 3D, os melhores algoritmos alcançaram uma FRR de 0.005 a 0.015 e o FAR de 0.001.

O FRVT 2006 é a avaliação biométrica que comparou pela primeira vez o desempenho humano e das máquinas no reconhecimento facial. Os resultados mostraram que, em experiências com mudança de condições de iluminação na fase de reconhecimento, seis de sete algoritmos têm desempenho igual ou superior aos seres humanos (PHILLIPS et al., 2007).

2.7 Considerações Finais

Muitas são as aplicações que utilizam a biometria em sua composição. O reconhecimento facial mostra-se como a tecnologia biométrica que possui vantagem sobre as demais por não necessitar da cooperação do participante.

No intuito de aprimorar o reconhecimento facial várias são as abordagens e muitos são os métodos desenvolvidos. Dentre as abordagens citadas, este trabalho será desenvolvido sob a abordagem das redes neurais que, apesar de normalmente requerer a utilização de técnicas de pré-processamento e redução da imagem a fim de tratar o problema de dimensionalidade da imagem vem obtendo significativos avanços como observado com o grande número de trabalhos desenvolvidos na área e respectivos resultados (FASEL, 2002) (MAT-SUGU et al., 2003) (AL-AKKAD, 2004) (SMACH et al., 2005) (SAITO et al., 2005).

O próximo capítulo traz uma revisão sobre os principais tópicos das redes neurais, alguns exemplos e seu relacionamento com o reconhecimento facial.

Redes Neurais Artificiais

O presente capítulo contém a revisão bibliográfica a respeito das redes neurais artificiais (RNAs). Inicialmente é realizada uma introdução apresentando um conceito sobre redes neurais e os principais atrativos desta abordagem. Em seguida é realizado um breve histórico e apresentada uma revisão sobre redes biológicas que são a motivação para as RNAs. Na sequência apresentado o neurônio artificial seguido pelas principais características das redes neurais e por fim são apresentados alguns exemplos de redes e a relação das redes neurais com o reconhecimento facial.

3.1 Introdução

Na sua forma mais geral, a rede neural é uma máquina que é projetada para modelar a maneira como o cérebro realiza uma tarefa particular ou função de interesse sendo normalmente implementada utilizando componentes eletrônicos ou simulada por programação em computador digital. Seu poder computacional está vinculado à sua estrutura maciçamente paralelamente distribuída e capacidade de generalização (HAYKIN, 2001).

Em LOESCH;SARI (1996) são apresentadas as características que tornam as RNAs atra-

tivas na resolução de diversos problemas computacionais:

- Aprendizagem por exemplo: característica relevante quando o problema a ser resolvido não possui todas as suas variáveis conhecidas sendo assim um problema de complexidade computacional alta;
- Independência do problema: uma rede neural pode ser aplicada a diversos problemas sem necessidade de qualquer alteração na sua estrutura;
- Obtenção dos resultados desejados: uma rede neural pode ter sua arquitetura modificada em busca de otimizações caso não tenha obtido os resultados desejáveis em um nível aceitável;
- Processamento distribuído, paralelo e local: características inerentes à sua natureza. O paralelismo ocorre nos níveis de módulos de redes neurais, camadas, neurônios e conexões; e
- Implementação realística: novas arquiteturas de sistemas de computação sem utilizar a arquitetura Von Neumann e vários modelos de redes neurais têm sido implementados em hardware, otimizando e aumentando a faixa de utilização.

3.2 Breve histórico

Os primeiros estudos sobre a neurocomputação foram realizados na década de 1940 por McCulloch e Pitts (1943) que fizeram uma analogia entre células vivas e o processamento eletrônico, simulando o comportamento do neurônio natural em um neurônio artificial. No final da década de 1940, Donald Hebb, escreveu um livro denominado "*The organization of behavior*"(1949) onde propôs uma lei de aprendizagem específica para as sinapses dos neurônios.

Em 1958, Roseblatt criou o modelo dos perceptrons, no qual os neurônios eram organizados em camadas de entrada e saída e os pesos das conexões eram adaptados para atingir o máximo de desempenho. Em seu livro "*Principles of Neurodynamics*"(1961), Roseblatt escreveu todos os seus resultados sobre os perceptrons. A partir das críticas ao modelo Perceptron houve uma queda nas pesquisas na década seguinte, havendo uma

retomada das publicações a partir da década de 1980 quando Rumelhart, Hilton e Williams (1986) introduziram o método do *backpropagation*.

Hopfield (1982) relatou a utilização de redes simétricas para otimização através de algoritmos de aprendizagem que estabilizam uma rede binária simétrica com realimentação. Atualmente, os avanços nesta área se concentram no desenvolvimento de novos algoritmos no intuito de melhorar os modelos existentes.

3.3 Motivação: redes biológicas

A rede biológica do cérebro humano tem a capacidade de reconhecer padrões e relacioná-los, usar e armazenar conhecimento por experiência, além de interpretar observações. Apesar do estudo contínuo, o funcionamento das redes biológicas ainda não foi totalmente desvendado pelo homem. As RNAs tentam reproduzir as funções das redes biológicas, buscando implementar seu comportamento básico e sua dinâmica (BRAGA et al., 2000).

A estrutura fisiológica básica das redes biológicas é conhecida e é exatamente nesta estrutura fisiológica que se baseiam as RNAs. Ou seja, as RNAs são baseadas em unidades de computação paralela e distribuída que se comunicam através de conexões sinápticas assim como as redes biológicas e conseguem reproduzir várias funções encontradas nos cérebros biológicos. O modelo básico de uma célula nervosa, o neurônio (figura 3.1), é formado por dendritos que possuem a função de receber os estímulos transmitidos pelos outros neurônios, o corpo do neurônio, chamado de soma, que é responsável por coletar e combinar as informações recebidas pelos dendritos e finalmente o axônio que é responsável por transmitir os estímulos para outras células, sendo que a região de contato entre o axônio de um neurônio e o dendrito de outro é chamado de sinapse.

Cada neurônio envia impulsos para muitos outros neurônios e recebem impulsos de muitos neurônios. Este simples conceito é a base para toda a atividade no sistema nervoso central e forma a base da maioria dos modelos de redes neurais artificiais (FREEMAN;SKAPURA, 1992). O funcionamento do neurônio é relativamente simples: os impulsos que chegam ao neurônio através dos dendritos são "somados" e, caso a soma seja maior do que um determinado valor limiar, o neurônio é ativado e dispara um impulso através do axônio para outras células.

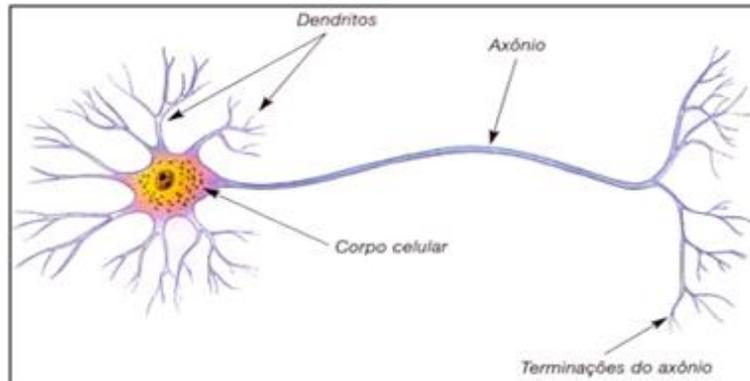


Figura 3.1: Modelo de um neurônio biológico

3.4 O neurônio artificial

O modelo do neurônio artificial (figura 3.2) é uma simplificação do neurônio biológico. Nele, os sinais (u_1, u_2, \dots, u_n) são apresentados à entrada, cada sinal é multiplicado por um peso (w_1, w_2, \dots, w_n) , que indica a sua influência na saída da unidade. É realizada uma soma ponderada (s) dos sinais que tem como resposta um nível de atividade. Caso este nível exceda o limiar (θ), obtido a partir da função de ativação, a unidade produz uma determinada resposta de saída (y).

As principais inovações no modelo do neurônio vieram com a utilização de novos tipos de função de ativação, responsável por processar as entradas ponderadas pelos seus respectivos pesos e medir o estado de ativação para o neurônio. Qualquer função matemática pode ser utilizada como função de ativação, porém algumas têm sido mais estudadas e utilizadas como a função linear, degrau e sigmóides que são encontradas na construção de diversos modelos de redes neurais (Tabela 3.1).

Tabela 3.1: Funções de ativação mais utilizadas (LOESCH;SARI, 1996)

Função	Equação
Linear	$f(s) = s$
Passo	$f(s) = \begin{cases} +1, & \text{se } s \geq 0 \\ 0, & \text{se } s < 0 \end{cases}$
Logística	$f(s) = \frac{1}{1+e^{-s}}$
Hiperbólica	$f(s) = \frac{1-e^{-2s}}{1+e^{-2s}}$

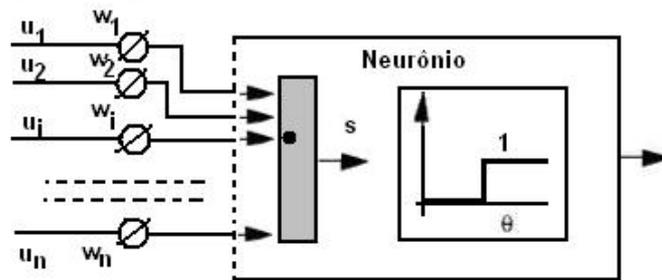


Figura 3.2: Modelo do neurônio artificial

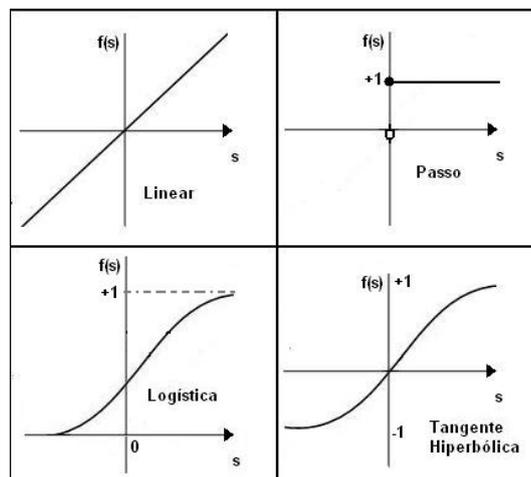


Figura 3.3: Graficos das funções apresentadas na tabela 3.1 (LOESCH;SARI, 1996)

3.5 Arquitetura

A definição da arquitetura de uma RNA é um parâmetro importante na sua concepção, uma vez que ela restringe o tipo de problema que pode ser tratado pela rede (BRAGA et al., 2000). Segundo HAYKIN (2001), são identificados três tipos de arquiteturas fundamentais:

- Rede com conexões progressivas com camada única: A "camada única" refere-se à camada de saída de nós computacionais, não sendo contada a camada de entrada onde não é realizada computação;
- Rede com conexões progressivas de múltiplas camadas: possui uma ou mais camadas ocultas que situam entre a camada de entrada e a camada de saída. Esta rede pode ainda ser totalmente conectada (cada nó de uma camada está conectado a todos os nós da camada seguinte) ou parcialmente conectada (o nó de uma camada se encontra conectado a alguns dos nós da camada seguinte); e
- Redes recorrentes: possui pelo menos um laço de realimentação. Esta realimentação possui uma grande influência na capacidade de aprendizagem da rede e no seu desempenho.

3.6 Aprendizado

A habilidade de aprender de seu ambiente e posteriormente melhorar seu desempenho é a principal propriedade de uma rede neural. Esse processo de aprendizagem é realizado com o ajuste aplicado aos pesos nas conexões do neurônio baseado em algoritmos.

Os principais paradigmas para treinamento de redes neurais são: aprendizado supervisionado e o não-supervisionado. O paradigma supervisionado é o mais comum, neste caso, é utilizado um agente externo que indica a resposta desejada para o padrão de entrada. A cada padrão submetido à rede, a resposta obtida é comparada com a resposta desejada. Sua desvantagem é que na ausência de um supervisor (agente externo) a rede não consegue aprender novas estratégias para situações que não foram simuladas no treinamento.

O paradigma não-supervisionado por sua vez, não possui um agente externo para supervisionar e acompanhar o processo de aprendizado. A habilidade de formar represen-

tações internas para codificar características da entrada e criar novas classes automaticamente acontece a partir do momento em que a rede estabelece uma harmonia com as regularidades estatísticas de entrada dos dados. Entretanto, este tipo de aprendizado só é possível quando existe redundância nos dados de entrada para que seja possível encontrar padrões (BRAGA et al., 2000).

Existem diferentes métodos para implementar o aprendizado de uma rede neural. A seguir estão listadas algumas das regras mais utilizadas:

- Regra Hebb: propõe que o peso de uma conexão sináptica deve ser ajustado se houver sincronismo entre os "níveis de atividade" das entradas e saídas, ou seja, o fortalecimento da sinapse só ocorrerá se dois neurônios, em lados distintos da sinapse, estão ativados;
- Regra Delta: esta regra procura minimizar a diferença da função de ativação aplicada sobre a soma ponderada das entradas pelos pesos (saída obtida) e a saída desejada que representa o erro da resposta atual da rede;
- Retropropagação: trata-se de uma variação da regra delta. O ajuste dos pesos é feito de trás para frente, isto é, da última camada em direção à camada de entrada; e
- Competitivo: os neurônios são inibidos por outros neurônios de modo que a competição entre eles leva a somente um vencedor.

3.7 Exemplos de redes neurais

3.7.1 Perceptron

O modelo Perceptron é composto por uma estrutura de rede, que possui nodos como, os exibidos na figura 3.2, como unidade básica e por uma regra de aprendizado. Roseblatt desenvolveu o Teorema da convergência, mostrando que um nodo treinado com o algoritmo de aprendizado do perceptron sempre converge para uma solução, num tempo finito, caso o problema seja linearmente separável (HAYKIN, 2001).

A topologia original possui três níveis (retina, unidades de associação e unidades de resposta). Porém, apenas o nível de saída (resposta) possui propriedades adaptativas, por

isso ela é conhecida como perceptron de uma única camada. As unidades de associação possuem pesos fixos.

A regra de aprendizado do Perceptron permite a adaptação dos seus pesos de forma que a rede execute uma determinada tarefa de classificação. Esta regra está baseada na obtenção do incremento dos pesos (Δw) a ser aplicado no vetor de pesos de forma que o valor atualizado seja mais próximo da solução.

O algoritmo de execução do treinamento do perceptron pode ser observado no algoritmo 1 (BRAGA et al., 2000), onde p denota o número de entradas; x o valor da entrada; η a taxa de aprendizado; d o valor da saída desejada; $w(t)$ o valor do pesos no tempo t e e o erro obtido pela subtração da saída desejada pela saída obtida.

Algoritmo 1 Algoritmo de treinamento do Perceptron (BRAGA et al., 2000)

- 1: Inicializar taxa de aprendizado (η) e o vetor de pesos w ;
 - 2: **Enquanto** $e \neq 0$ **Faça**
 - 3: **Para** Cada par do conjunto de treinamento $\Gamma = \{(x^i, y_d^i)\}_{i=1}^p$ **Faça**
 - 4: Atualizar o vetor de pesos para cada um dos nodos da rede segundo a regra
 $w(t+1) = w(t) + \eta ex(t)$
 - 5: **Fim Para**
 - 6: **Fim Enquanto**
-

3.7.2 MultiLayer Perceptron - MLP

Como redes de uma camada (Perceptron) resolvem apenas problemas linearmente separáveis, a solução encontrada para a classificação de mais de duas classes foi utilizar mais de uma camada equivalendo ao aumento da dimensionalidade do problema.

A rede MLP consiste de um conjunto de unidades sensoriais que formam a camada de entrada, uma ou mais camadas ocultas e uma camada de saída (figura 3-3). Um perceptron de múltiplas camadas, denominação traduzida do MLP, possui três características bem particulares (HAYKIN, 2001):

- Cada neurônio possui uma função de ativação diferenciável e em muitos casos não-linear. A presença da não-linearidade é importante para que a relação de entrada-saída da rede não seja reduzida a uma rede do modelo Perceptron.
- Contém uma ou mais camadas de neurônios ocultos que capacitam a rede a aprender tarefas complexas, extraíndo progressivamente as características mais significativas

dos padrões.

- Possui um alto grau de conectividade determinado pelas sinapses da rede.

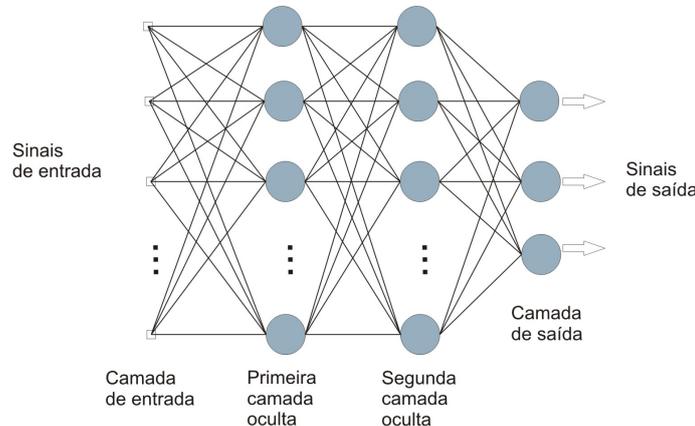


Figura 3.4: Topologia da rede MLP (HAYKIN, 2001)

Segundo HAYKIN (2001), "a diferenciabilidade é a única exigência que a função de ativação deve satisfazer". Utilizam-se normalmente como função de ativação as funções sigmóides como a logística e a tangente hiperbólica.

Sua aplicação tem sido focada no reconhecimento de caracteres, previsões do comportamento, verificação de assinaturas, diagnóstico médico, processamento de imagens, etc (BRAGA et al., 2000).

O algoritmo de aprendizado mais aplicado ao MLP é o algoritmo de retropropagação (*backpropagation*) que é uma variação da fórmula para treinamento da regra delta. Basicamente, este algoritmo de retropropagação (algoritmo 4) consiste de dois passos através das diferentes camadas: a propagação (algoritmo 2) e a retropropagação (algoritmo 3).

Algoritmo 2 Algoritmo da fase de propagação da MLP (BRAGA et al., 2000)

- 1: A entrada é apresentada à primeira camada da rede;
 - 2: **Para** C^i **Faça**
 - 3: O sinal de saída é calculado;
 - 4: Após os nodos da camada C^i ($i > 0$) calcularem seus sinais de saída, estes servem para a definição das saídas produzidas pelos nodos da camada C^{i+1} ;
 - 5: **Fim Para**
 - 6: As saídas produzidas pelos nodos da última camada são comparadas às saídas desejadas;
-

Vale ressaltar que durante a propagação, os pesos são fixos. Seu erro é calculado e posteriormente retro propagado com a finalidade de ajuste de pesos.

Algoritmo 3 Algoritmo da fase de retropropagação da MLP (BRAGA et al., 2000)

- 1: A partir da última camada;
 - 2: **Enquanto** não chegar na camada de entrada **Faça**
 - 3: Os nodos da camada atual ajustam seus pesos de forma a reduzir os erros;
 - 4: O erro de um nodo das camadas intermediárias é calculado utilizando os erros dos nodos da camada seguinte conectados a ele, ponderados pelos pesos das conexões entre eles;
 - 5: **Fim Enquanto**
-

Algoritmo 4 Algoritmo de treinamento da MLP (BRAGA et al., 2000)

- 1: Inicializar os pesos e parâmetros;
 - 2: **Enquanto** Erro não for mínimo ou chegar num limite de ciclos **Faça**
 - 3: **Para** Cada padrão de treinamento **Faça**
 - 4: Definir saída através da fase de propagação;
 - 5: Comparar saída obtida com saída desejada;
 - 6: Atualizar pesos dos nodos através da fase de retropropagação;
 - 7: **Fim Para**
 - 8: **Fim Enquanto**
-

O principal problema do *backpropagation* diz respeito à lentidão do algoritmo para superfícies mais complexas. Segundo (AZEVEDO et al., 2000), uma forma de minimizar este problema é utilizar o termo *momentum* (μ) em conjunto com o algoritmo. Neste caso, o algoritmo segue os mesmos passos, porém, a função de ajuste dos pesos é modificada para ser incluído o *momentum* (equação 3.1).

$$w_i(t+1) = w_i(t) - \eta \delta x + \mu(w_i(t) - w_i(t-1)) \quad (3.1)$$

3.7.3 Redes recorrentes

As redes recorrentes são redes neurais com um ou mais ciclos de realimentação, que influenciam profundamente na capacidade de aprendizado da rede e em sua performance (HAYKIN, 2001). São utilizadas como memórias associativas ou rede de mapeamento de entrada em saídas. Uma rede recorrente responde temporalmente à aplicação de um sinal externo. Uma das redes neurais recorrentes mais utilizadas é a rede de Hopfield na qual a resposta de rede depende sempre de seu estado no intervalo de tempo anterior (BRAGA et al., 2000).

A estrutura básica da rede consiste de um conjunto de neurônios e um conjunto correspondente de atrasos unitários, formando um sistema realimentado (figura 3.5). Cada

neurônio tem um valor de atividade (estado) binário (HAYKIN, 2001). Sua topologia, como se pode perceber na figura 3.5, é interconectada completamente porém, não existe auto-realimentação para que a rede se mantenha estável. Cada interconexão tem um peso associado, considerado como a força da conexão e é representado por um valor escalar, sendo que w_{ij} representa o peso da unidade j para a unidade i onde, $w_{ij} = w_{ji}$ (AZEVEDO et al., 2000).

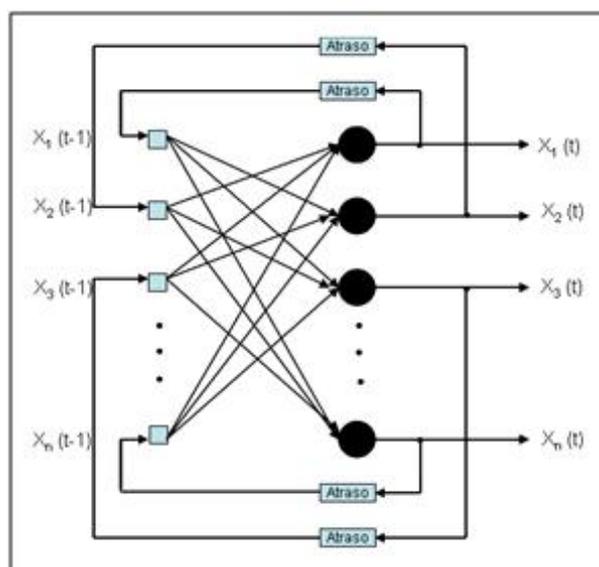


Figura 3.5: Topologia da rede de Hopfield

Esta rede é caracterizada pela operação assíncrona com a alteração de estado de apenas um neurônio por instante de tempo. A probabilidade de um neurônio ser o escolhido para ser analisado numa possível transição de estado é a mesma para todos eles. A atualização das unidades continua até que a rede venha a convergir para um estado estável.

A seguir é apresentado o algoritmo de treinamento da rede de Hopfield (algoritmo 5)

3.7.4 Redes auto-organizáveis

Uma rede neural auto-organizável tem como principal característica a capacidade de auto-organização caracterizada como a capacidade de reconhecer relações entre padrões apresentados pela competição entre células vizinhas para ativação por meio das interações laterais mútuas (KARAYIANNIS; VENETSANOPOULOS, 1993)(AZEVEDO et al., 2000). Ou seja, para executar a auto-organização utiliza a competição como regra de aprendizado.

Algoritmo 5 Algoritmo de treinamento da rede de Hopfield (BRAGA et al., 2000)

1: Calcular os pesos através da equação:

$$w_{ij} = \frac{1}{k} x_i x_j$$

2: Apresentação do vetor de entrada, no qual cada entrada do vetor será um valor de entrada da rede;

3: Aleatoriamente, cada neurônio é atualizado individualmente a partir de:

$$x(t+1) = \text{sgn} \sum_i^n w_{ij} x_i(t)$$

onde

$$\text{sgn}(u) = \begin{cases} -1, & \text{se } u < 0 \\ +1, & \text{se } u \geq 0 \end{cases}$$

4: **Enquanto** Houver variação no estado dos neurônios **Faça**

5: Repetir passo 3;

6: **Fim Enquanto**

As redes auto-organizáveis têm sido utilizadas em aplicações de reconhecimento de fonemas, diagnósticos financeiros, controle de processos, processamento de imagens, exploração de bases de dados, entre outras (BRAGA et al., 2000).

Provavelmente, a rede mais representativa nesta categoria é o mapa de Kohonen. Este modelo é baseado no mapa topológico presente no córtex cerebral, onde, os neurônios estão espacialmente ordenados dentro de áreas que são responsáveis por funções específicas.

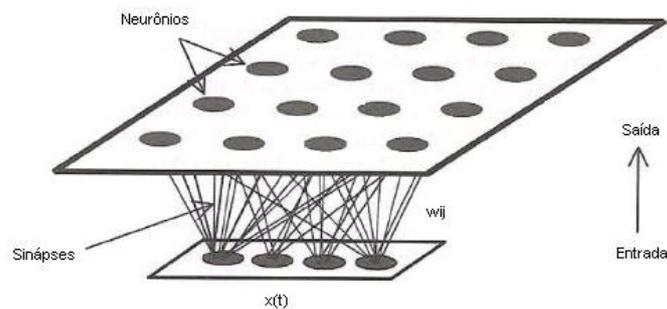


Figura 3.6: Topologia dos Mapas de Kohonen

A topologia básica da rede possui duas camadas: uma camada de entrada com n unidades, onde a camada de entrada irá conter tantos neurônios quanto o número de elementos do vetor de entrada ($x(t)$). E uma camada de saída composta por uma matriz $m \times m$ de neurônios formando um reticulado. Cada nodo recebe todas as entradas e funciona

como um discriminador ou extrator de características (figura 3.6).

Os Neurônios topologicamente próximos tendem a responder a padrões ou estímulos semelhantes e cada neurônio influencia o estado de ativação de neurônios vizinhos de três formas: excitatória (dentro de um mesmo raio R_1), inibitória (dentro de um segundo raio R_2 e fora do raio R_1) e levemente excitatória (dentro de uma terceira área de raio R_3 e fora de R_1 e R_2) com $R_1 < R_2 < R_3$, conforme figura 3.7 (BRAGA et al., 2000).

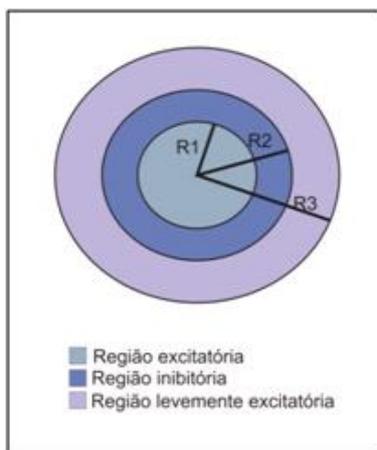


Figura 3.7: Raios de influência de um neurônio

Segundo HAYKIN (2001), é aconselhável atribuir aos pesos, valores pequenos tomados de um gerador de números aleatórios para que nenhuma organização prévia seja imposta ao mapa. Sua formação pode ser dividida em três processos:

- Processo de competição: para cada padrão de entrada os neurônios calculam os valores da função discriminante que fornece a base para a competição entre os neurônios, em que o neurônio com maior valor da função é declarado vencedor;
- Processo de cooperação: neste processo, o neurônio vencedor determina a localização espacial de uma vizinhança topológica de neurônios excitados. Logo, desta forma é fornecida a base para a cooperação; e
- Processo adaptativo: neste último processo é permitido aos neurônios excitados aumentarem seus valores individuais da função discriminante em relação ao padrão de entrada através de ajustes aplicados a seus pesos.

Seu algoritmo de treinamento funciona segundo o algoritmo apresentado a seguir (algoritmo 6):

Algoritmo 6 Algoritmo de treinamento da rede auto-organizável de Kohonen (BRAGA et al., 2000)

- 1: Inicializar pesos e parâmetros;
 - 2: **Enquanto** Houver mudança no mapa de características **Faça**
 - 3: **Para** Cada padrão de treinamento **Faça**
 - 4: Definir nodo vencedor;
 - 5: Atualizar os pesos deste nodo e seus vizinhos;
 - 6: **Se** Número do ciclo for múltiplo de N **Então**
 - 7: Reduzir taxa de aprendizado e área de vizinhança
 - 8: **Fim Se**
 - 9: **Fim Para**
 - 10: **Fim Enquanto**
-

3.7.5 Redes convolucionais

As redes convolucionais consistem em um conjunto de camadas que possuem um ou mais planos. Como mostrado na figura 3.8, cada unidade de uma camada recebe entrada de um conjunto de unidades localizada numa vizinhança da camada anterior. Sucessivas camadas de convolução e *subsampling* são tipicamente alternadas e o número de mapas de características cresce com a diminuição da resolução espacial (LECUN;BENGIO, 2003).

As redes convolucionais são redes que possuem a habilidade de lidar com deformações e variações nos dados de entrada (BIANCHINI, 2001).

Combinam três idéias em sua arquitetura a fim de assegurar sua invariância ao deslocamento e distorção de padrões: campos receptivos locais; pesos compartilhados; e, às vezes, *subsampling* temporal ou espacial (LECUN;BENGIO, 2003). Com os campos receptivos locais os neurônios podem extrair características visuais elementares como a orientação de bordas, extremidades e cantos que posteriormente serão combinadas pelas camadas mais elevadas.

Um dos exemplos representativos deste modelo neural é a rede neocognitron, foco deste trabalho, apresentado a seguir no capítulo 4.

3.8 Redes Neurais e reconhecimento facial

Além das potencialidades em problemas de classificação, predição, modelamento e categorização, as RNA's ganharam atenção na área de processamento de imagens e sinais como ferramenta alternativa e às vezes complementar às técnicas clássicas.

O diferencial dos problemas de visão computacional e reconhecimento de imagens está na quantidade de informação manipulada que normalmente requerem pré-processamento e redução de dimensionalidade a fim de tornar a imagem tratável pela RNA (BRAGA et al., 2000).

Grandes foram os avanços nas técnicas utilizadas. Em 1996, Lawrence desenvolveu uma solução neural híbrida com a combinação de redes auto-organizáveis, redes convolucionais e *local image sampling*, para o reconhecimento facial, capaz de classificar rapidamente e com desempenho superior ao método de *eigenface*, tendo como limitação o fato de ser parcialmente invariante a deformações, escala, translação, entre outros (LAWRENCE et al., 1996).

Em 1998, Rowley apresentou um algoritmo baseado em rede neural para detectar visões perpendiculares e frontais de faces em imagens em escala de cinza aplicando uma ou mais redes neurais diretamente em porções da imagem de entrada e arbitrando seus resultados sendo capaz de classificar a presença ou ausência de uma face com uma taxa de 77.9% e 90.3% de detecção em um conjunto de 130 imagens (ROWLEY et al., 1998).

Apesar da habilidade conhecida da MLP para trabalhar com mapeamento não linear

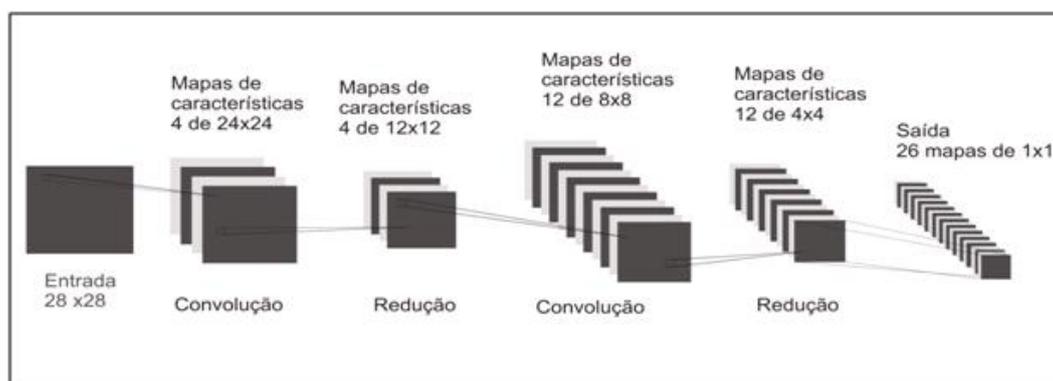


Figura 3.8: Exemplo de uma rede neural convolucional (LAWRENCE et al., 1996)

para grandes coleções de exemplos, aprendizado complexo e alta dimensionalidade colocá-la como candidata óbvia para trabalhar com a tarefa de reconhecimento de imagem, não se deve esquecer que as imagens são grandes e geralmente possuem centenas de variáveis podendo gerar problemas na aplicação deste modelo no reconhecimento facial. Por exemplo, uma estrutura que possui a primeira camada totalmente conectada com a segunda camada, em que ambas possuem centenas de unidades irá gerar dezenas de milhares de pesos associados. Além disso, podem ocorrer problemas de *underfitting* caso o número de dados de treinamento seja escasso, o custo computacional pode ser grande e principalmente existe uma deficiência para trabalhar de forma invariante com respeito a translação ou distorção local (LECUN;BENGIO, 2003).

Nos últimos anos muitos trabalhos vêm sendo realizados utilizando as redes convolucionais como em FASEL (2002) que combinou redes convolucionais e MLP para a detecção facial. MATSUGU et al. (2003) que combinou reconhecimento de expressão facial e detecção de face numa rede convolucional e SAITO et al. (2005) que trabalhou especificamente com a rede convolucional neocognitron para o reconhecimento facial, motivados pela sua característica de ser invariante a forma, obtida automaticamente devido a replicação forçada da configuração dos pesos através do espaço e por possuir correlação local que garante a extração e combinação de características locais (LECUN;BENGIO, 2003).

3.9 Considerações Finais

Neste capítulo foram apresentados os principais modelos neurais. Dentre os modelos apresentados alguns têm ganho destaque no desenvolvimento de métodos a serem utilizados para o reconhecimento facial e detecção facial. São eles as redes: MLP, na detecção de face; as redes auto-organizáveis, com a utilização dos mapas de Kohonen que se mostraram eficientes no reconhecimento de imagens ruidosas e faltando partes e a utilização de redes convolucionais devido sua tolerância a deformações e variações dos dados de entrada.

O presente projeto irá trabalhar com o modelo convolucional, com a utilização específica da rede neocognitron que foi desenvolvida para o reconhecimento de caracteres manuscritos e vem sendo utilizada em pesquisas para reconhecimento facial. O próximo capítulo aborda a rede neocognitron mostrando suas características, arquitetura, proces-

samento e limitações.

Rede Neocognitron

Esse capítulo apresenta um estudo da rede neocognitron, escolhida para realização do presente trabalho, oferecendo um levantamento das principais características da rede, sua arquitetura, processamento e treinamento além de limitações.

4.1 Breve histórico

A rede neocognitron foi desenvolvida por Fukushima para o reconhecimento de caracteres manuscritos. Foi modelada com base no modelo biológico da visão de modo qualitativo e foi desenvolvido a partir da rede Cognitron (1975) que possuía a limitação de reconhecer um padrão apresentado com uma posição diferente como um novo padrão. Esse modelo inicial da neocognitron tem a habilidade de reconhecer padrões corretamente mesmo se o padrão sofrer variação de posição ou distorção da forma (FUKUSHIMA, 1980).

Entretanto, esta versão de 1980 possuía um reconhecimento limitado quando padrões eram apresentados simultaneamente. Para melhorar seu funcionamento, conexões de retorno (*backward*) foram adicionadas acrescentando a função de atenção seletiva à rede (FUKUSHIMA, 1988). Este modelo de 1988 pode automaticamente segmentar e reconhecer

padrões individuais apresentados simultaneamente além de restaurar padrões imperfeitos e eliminar ruído de padrões corrompidos.

Em 1992, Fukushima e Wake aperfeiçoaram a neocognitron com a adição de células para detecção de curvas em que após a extração das bordas e das linhas, são detectados os pontos de curvatura e final de linha (FUKUSHIMA;WAKE, 1992). Posteriormente em 2003 a rede foi aprimorada para que seu treinamento pudesse ser incremental (FUKUSHIMA, 2004). Anteriormente, após uma fase completa de treinamento era obtida uma estrutura de rede contendo um número fixo de neurônios de tal forma que uma estrutura não podia ser modificada com uma segunda fase de treinamento.

4.2 Arquitetura da rede

O neocognitron é formado por vários estágios hierarquicamente posicionados que possuem duas camadas cada um, exceto o estágio que representa a camada de entrada (U_0): uma camada de células-S (U_S) e uma camada de células-C (U_C) (figura 4.1). Cada camada é formada por vários planos de células, sendo que, cada plano de uma mesma camada possui a mesma quantidade de células. Quanto ao número de planos S e C, esta informação depende do número de características extraídas pelo estágio e seu tamanho vai diminuindo de forma que no último estágio os planos-C possuem apenas uma célula.

Cada estágio possui ainda um plano de células-V que é responsável por prover informações para as células-S sobre a quantidade de atividades presente em cada região de entrada de cada célula-S. As células-V têm um efeito inibitório e normaliza as respostas da célula-S com respeito a atividade da sua região de entrada (LOVELL et al., 1997).

As células-S contém pesos variáveis nas suas conexões que necessitam de treinamento, enquanto que as células-C e as células-V apenas possuem conexões com pesos fixos. Durante a fase de teste as células-S são ativadas apenas se as características ou fatores extraídos e aprendidos durante a fase de treinamento estiverem posicionados na região da área de conexão na camada de entrada. As células-C, por sua vez, servem para obter valores correspondentes a uma operação de borrimento que define uma área de tolerância num grupo de células-S. Logo, é responsável pela tolerância a deformação do neocognitron.

Trata-se de uma rede que possui conexões para frente (*forward*) e para trás (*backward*)

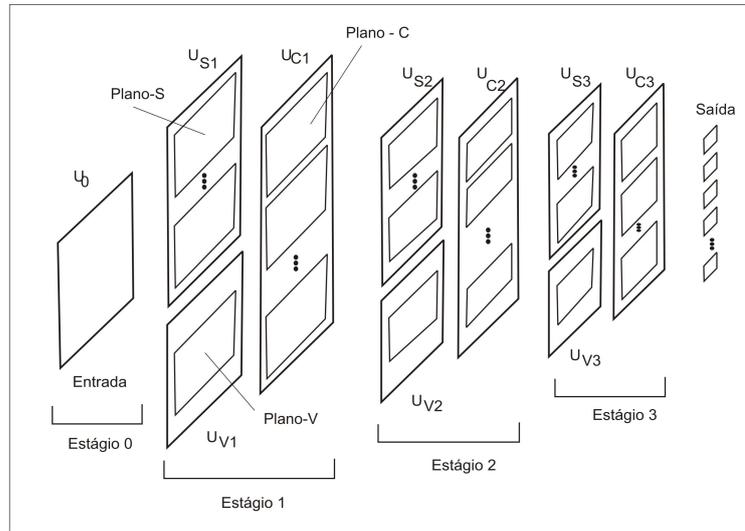


Figura 4.1: Estrutura da rede Neocognitron

entre as células. Nesta hierarquia, os sinais forward gerenciam a função de reconhecimento do padrão enquanto que os sinais backward gerenciam a função de atenção seletiva, segmentação de padrões e lembrança associativa (Figura 4-2) (FUKUSHIMA, 1988).

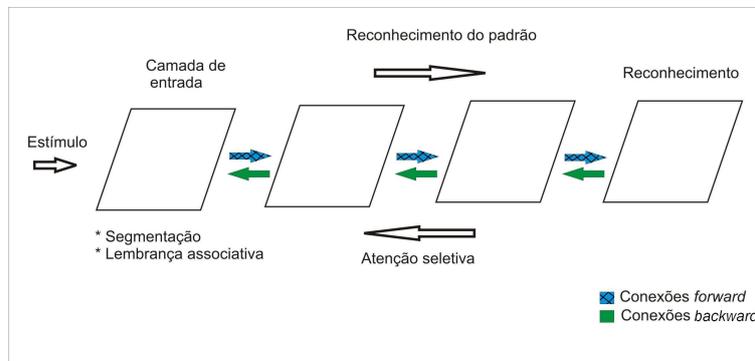


Figura 4.2: Funcionamento das conexões *forward* e *backward*

A habilidade de extrair características é influenciada por um limiar θ (*threshold*). Quanto maior o limiar, maior é o fator de inibição, ou seja, menor a generalização ocasionando uma maior seletividade à extração do fator. Durante o treinamento, o θ deve ser alto o suficiente para garantir a geração de células suficientes de extração. No reconhecimento, o θ deve ser mantido baixo para manter a habilidade de generalização (FUKUSHIMA; TANIGAWA, 1996).

Para cada conexão existe um peso associado que é utilizado para influenciar na quantidade de informação transferida: os pesos- a codificam as características a serem extraídas;

os pesos- b são utilizados nas conexões das células- V com as células- S e são proporcionais ao fator de inibição; os pesos- c são invariantes durante o treinamento e são utilizados nas conexões das células- C com as células- V ; e os pesos- d , também são fixos e utilizados nas conexões das células- S com as células- C , conforme figura 4.3. O conjunto de pesos de uma célula é igual ao de todas as outras células num mesmo plano, ou seja, todas as células de um mesmo plano procuram uma mesma característica.

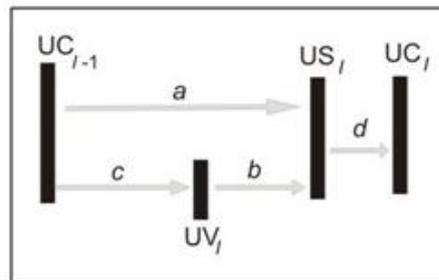


Figura 4.3: Fluxo de pesos

4.3 Processamento

Para realizar o reconhecimento do padrão, um padrão é apresentado à camada de entrada U_0 , que consiste de uma matriz bidimensional de células receptoras, sendo passado pelo modelo através das conexões *forward* até a camada de reconhecimento. Em casos de implementação do neocognitron com atenção seletiva, a saída da camada de reconhecimento é enviada através das conexões *backward* para a camada inicial até chegar a camada de *recall* (lembrança)(Figura 4-2) (FUKUSHIMA, 1988).

Inicialmente, as conexões *forward* e *backward* são nulas e, durante o processo de aprendizado, as conexões *forward* são reforçadas primeiro. Em seguida as conexões *backward* também são reforçadas, em mesma quantidade e formando par com as conexões *forward* (FUKUSHIMA, 1988).

Seu reconhecimento ocorre baseado em similaridade de forma, não sendo afetado por distorções devido a tolerância ao erro de posicionamento dentro de um campo de tolerância conforme figura 4-4 que representa o princípio de reconhecimento de padrões deformados. Na figura 4.4 (A) está representada uma característica global formada por três caracterís-

ticas locais e respectivos campos de tolerância ao erro de posicionamento. Na figura 4.4 (B) é mostrado o reconhecimento de padrões deformados. Na figura 4.4 (C) exemplifica um padrão que pode ser reconhecido erroneamente como um padrão "A" no caso de haver uma grande área de tolerância ao erro de posicionamento.

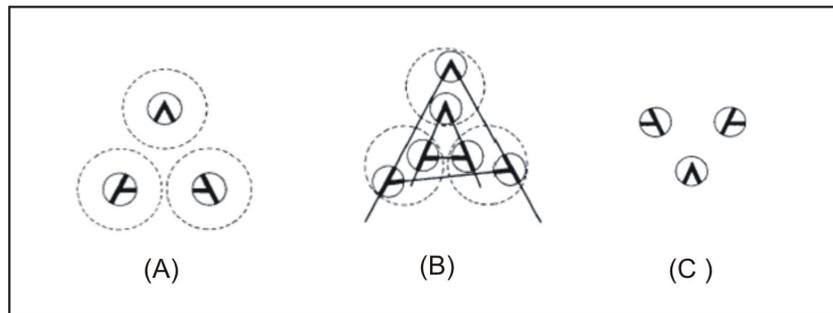


Figura 4.4: (A) Característica global de um padrão formada por três características locais. (B) Padrão deformado reconhecido. (C) Características locais fora dos respectivos campos de tolerância (FUKUSHIMA, 1988)

A rede pode ser treinada por aprendizado supervisionado ou não supervisionado. Vários métodos de treinamento, dentro destes dois paradigmas, vêm sendo propostos (FUKUSHIMA, 2003). No aprendizado supervisionado um supervisor (professor) apresenta os padrões de treinamento para a rede e indica a posição das características que serão extraídas. A célula cujo centro do campo receptivo coincide com a posição da característica torna-se a célula semente (*seed cell*).

O segundo método apresentado trata-se de um método por aprendizado não supervisionado em que é realizada a auto-organização da rede baseado em dois princípios. O primeiro diz respeito a regra do aprendizado por competição, onde uma quantidade de células competem pela extração de uma característica mais relevante. Através de conexões excitatórias a célula-S recebe sinais indicando a existência de uma característica. Ao mesmo tempo em que recebe um sinal inibitório através de sua conexão inibitória. Caso a característica seja relevante a célula-S responde com valor maior que zero. A célula-S com maior valor é a vencedora. O segundo princípio diz que uma célula vencedora não apenas cresce, mas também controla sua vizinhança. Esta célula vencedora é a célula semente, que é replicada para todas as posições do plano celular, responsável pelo reconhecimento da característica correspondente.

4.4 Treinamento não supervisionado

Devido à importância do treinamento não-supervisionado nesse trabalho, resolveu-se inserir essa seção no presente capítulo para abordar de forma mais detalhada o processo de treinamento aplicado.

Como apresentado anteriormente, no aprendizado não-supervisionado, também conhecido por auto-organizado, não é necessário um supervisor. No caso do neocognitron sua auto-organização é baseada em dois princípios: aprendizado por competição onde as células competem pela extração de uma característica e pelo princípio de que uma célula vencedora não apenas cresce como também controla a sua vizinhança.

Cada plano celular da camada S representa um novo fator extraído. No início não existe nenhum plano na camada S. Para dar início ao treinamento é utilizado um plano especial denominado SSP (*Seed Selection Plane*) habilitado a reconhecer todos os fatores de uma camada precedente.

Inicialmente um padrão de entrada é apresentado para a rede. Todas as características do padrão de entrada são reconhecidos pelo SSP e várias células ficam ativas. Usando a regra da competição mencionada, é escolhido um neurônio de maior intensidade. Se o fator correspondente a esse neurônio não tiver sido treinado até aquele momento, então é acrescentado um novo plano celular para o reconhecimento daquele fator. Caso este fator já tenha sido treinado então existe um plano celular já treinado ativo naquela posição. Nesse caso, escolhe-se um próximo neurônio vencedor. Os pesos- a e $-b$ do neurônio vencedor são reforçados e faz-se com que todos os neurônios do novo plano celular compartilhe os mesmos pesos do neurônio vencedor.

Após todos os fatores correspondentes à camada S em questão terem sido treinados prossegue-se com o treinamento da próxima camada S do próximo estágio pois a camada C não necessita de treinamento. Apenas a última camada C recebe um treinamento supervisionado para o estabelecimento das classes de reconhecimento.

Por motivo de legibilidade o algoritmo de treinamento da rede foi dividido em algoritmo de cálculo das células-S (algoritmo 7) e células -C (algoritmo 4.4) e no algoritmo de treinamento do estágio da rede(algoritmo 9).

O algoritmo de cálculo das células-S num plano celular inicia com o cálculo dos so-

matórios auxiliares $e(n, k)$ e $h(n, k)$, conforme apresentados nas equações 4.1 e 4.2 a seguir, de todas as entradas conectadas aos K_{l-1} planos celulares da camada precedente representados pelo primeiro somatório, numa dada área de conexão S_l que circunda a posição da célula n da camada de células-C do estágio anterior ou da camada de entrada representada pelo segundo somatório.

No cálculo de $e(n, k)$, o termo $a_l(k_{l-1}, i, k)$ corresponde ao peso- a mostrado na figura 4.3, em que o subscrito l corresponde ao estágio l , k_{l-1} é o índice que indica o plano do estágio $l - 1$, i é o índice que indica a posição da célula conectada na área S_l e k indica o plano celular que está sendo calculado. O termo $u_{Cl-1}(n + i, k_{l-1})$ corresponde ao valor da célula-C do estágio $l - 1$ conectado à célula $u_{Sl}(n, k)$, onde o índice $n + i$ indica a posição da célula no plano e k_{l-1} indica o plano da estágio $l - 1$.

No cálculo de $h(n, k)$, as áreas calculadas são idênticas ao anterior. O que difere é o termo $c_l(i)$ que corresponde ao peso- c exibido na figura 4.3 e o termo $u_{Cl-1}^2(n + i, k_{l-1})$ que é o quadrado do valor da célula-C do estágio $l - 1$.

$$e(n, k) = \sum_{k_{l-1}=1}^{K_{l-1}} \sum_{i \in S_l} a_l(k_{l-1}, i, k) \cdot u_{Cl-1}(n + i, k_{l-1}) \quad (4.1)$$

$$h(n, k) = \sum_{k_{l-1}=1}^{K_{l-1}} \sum_{i \in S_l} c_l(i) \cdot u_{Cl-1}^2(n + i, k_{l-1}) \quad (4.2)$$

O cálculo da raiz quadrada de $h(n, k)$, dada por $\text{sqrt}(h(n, k))$, resulta em $u_{Vl}(n, k)$. Em seguida é calculado o valor de $u_{Sl}(n, k)$ segundo equação 4.3 e aplicada a função de ativação $\varphi(x)$ conforme equação 4.4, onde θ representa o *threshold* cujos valores ficam no intervalo $[0, 1]$ e $b(k)$ representa o coeficiente de inibição.

$$u_{Sl}(n, k) = \frac{\theta}{1 - \theta} \cdot \varphi \left(\frac{1 + e(n, k)}{1 + \theta \cdot b(k) \cdot u_{Vl}(n, k)} - 1 \right) \quad (4.3)$$

$$\varphi(x) = \begin{cases} x, & \text{se } x \geq 0 \\ 0, & \text{se } x < 0 \end{cases} \quad (4.4)$$

Para obter o valor de $u_{Cl}(n, k)$ (algoritmo 4.4), inicialmente é computado o somatório de todas as entradas correspondentes aos valores $u_{Sl}(n, k)$, previamente obtidos numa área de conexão S_l (equação 4.5) seguido pela aplicação da função de ativação $\psi(x)$ (equação

4.6) limitando a saída das células-C no intervalo $[0, 1]$. O termo $d_l(i)$ corresponde ao peso- d como apresentado na figura 4.3.

$$u_{Cl}(n, k) = \psi \left(\sum_{i \in S_l} d_l(i) \cdot u_{Sl}(n+1, k_l) \right) \quad (4.5)$$

$$\psi(x) = \begin{cases} \frac{x}{1+x}, & \text{se } x \geq 0 \\ 0, & \text{se } x < 0 \end{cases} \quad (4.6)$$

Conforme apresentado nos algoritmos 7 e 8, em pseudo-código, a seguir:

Algoritmo 7 Algoritmo para cálculo das células-S de um plano celular k

- 1: **Para** $n = 1$ to N **Faça**
 - 2: **Para** Planos celulares da camada precedente **Faça**
 - 3: **Para** Uma dada área de conexão S_l **Faça**
 - 4: Cálculo dos valores de $e(n, k)$ e $h(n, k)$;
 - 5: **Fim Para**
 - 6: **Fim Para**
 - 7: É realizado o cálculo das células-V ;
 - 8: É realizado o cálculo das células-S;
 - 9: **Fim Para**
-

Algoritmo 8 Algoritmo para cálculo das células-C de um plano celular k

- 1: **Para** $n = 1$ to N **Faça**
 - 2: **Para** Uma dada área de conexão S_l **Faça**
 - 3: Cálculo das células-C;
 - 4: **Fim Para**
 - 5: **Fim Para**
-

No algoritmo 9 é apresentado o treinamento dentro de um estágio que consiste em acrescentar novos planos celulares a cada novo fator ou característica extraída, incrementando assim o número K_l de planos celulares do estágio l . O treinamento da rede Neocognitron inicia computando os valores de células-S para os planos celulares $k = 1$ a $k = K_l$, relacionados aos fatores já existentes, e $k = K_{l+1}$, que corresponde ao plano SSP. O plano de seleção de semente é um plano auxiliar para a obtenção de um novo plano celular, correspondente a uma nova característica, cujos pesos iniciais das conexões de entrada são valores pequenos, porém diferentes de zero, para que os seus neurônios possam ser

excitados para quaisquer fatores apresentados nos padrões de entrada.

Algoritmo 9 Algoritmo para treinamento de um estágio da rede

```

1: Enquanto Houver fatores do padrão de entrada a ser encontrado Faça
2:   Para Planos celulares inclusive o SSP Faça
3:     Computar algoritmos de cálculo das células-S (algoritmos 7);
4:     Iniciar variável de controle de seleção (sel) como falsa;
5:     Se Um candidato a vencedor tiver valor maior que zero Então
6:       sel é acionada como verdadeira indicando que existe um candidato;
7:       Para todos os planos precedentes Faça
8:         Verificar se já existe um plano que reconhece o fator extraído (responsável);
9:         Se Existir um responsável pelo fator extraído Então
10:          Reiniciar sel como falso;
11:        Fim Se
12:      Fim Para
13:    Fim Se
14:    Enquanto Não houver um vencedor Faça
15:      Voltar ao passo 5;
16:    Fim Enquanto
17:    Se Houver um vencedor Então
18:      Para Todos os planos Faça
19:        Para Toda a área  $S_l$  Faça
20:          Reforço dos pesos variáveis (a e b);
21:        Fim Para
22:      Fim Para
23:    Fim Se
24:  Fim Para
25: Fim Enquanto

```

Conforme apresentado no algoritmo 9 é verificada no SSP a célula vencedora, ou seja, aquela em que o valor de resposta é máximo. Se na posição da célula vencedora, existe alguma resposta maior que zero nos K_l planos celulares anteriormente treinados (fatores já existentes), o algoritmo segue na procura de um novo vencedor; caso contrário, cada conexão de entrada da célula vencedora é reforçada, por meio do cálculo de $\Delta a(v, k_{l-1}, K_l)$ e $\Delta b(K_l)$, equações 4.7 e 4.8, respectivamente, que devem ser adicionadas aos valores de a e b .

$$\Delta a(v, k_{l-1}, K_l) = q_l \cdot c_{l-1}(v) \cdot u_{Cl-1}(k_{l-1}, n + v) \quad (4.7)$$

$$\Delta b(K_l) = q_l \cdot u_{Cl-1}(n) \quad (4.8)$$

Assim, um novo fator é obtido e um novo plano celular é adicionado à camada, incrementando K_l de um. Uma vez feito o reforço das conexões de entrada na célula vencedora, essa célula passa a ser a célula semente, pois todas as demais células do novo plano celu-

lar terão os mesmos pesos nas conexões de entrada. Daí surge o nome *seed selection plane* ao plano auxiliar usado para obter a célula vencedora. O procedimento de treinamento da camada-S é repetido até que todos os novos fatores sejam detectados, com a apresentação dos padrões de treinamento na camada de entrada.

4.5 Considerações Finais

Os principais estudos já realizados na rede neocognitron são voltados ao reconhecimento de caracteres manuscritos. Alguns trabalhos têm sido desenvolvidos na área de reconhecimento facial, porém ainda não foi significativamente verificada sua performance nesta área. Sua característica de tolerância a padrões distorcidos torna-a uma importante ferramenta no trabalho do reconhecimento de padrões.

A seguir é apresentado o capítulo sobre combinação de classificadores a fim de elucidar os métodos aplicados em experimentos na execução deste projeto com a finalidade de melhorar os resultados obtidos pelos classificadores utilizados.

Classificadores: Combinação e medidas de desempenho

Este capítulo faz uma revisão sobre os principais métodos empregados na combinação dos classificadores propostos no trabalho com o intuito de melhorar os resultados obtidos e descreve as medidas de desempenho utilizadas para avaliar a performance dos classificadores nos experimentos realizados. Inicialmente é feita uma introdução apresentando a justificativa para o uso de combinação de classificadores e subsequentemente são apresentados métodos para aplicar combinação de classificadores que forneçam saída de valor contínuo. São explicados também os métodos *hold-out* e coeficiente Kappa para avaliar o desempenho dos classificadores.

5.1 Combinação de classificadores

XU et al. (1992) afirma que a combinação de múltiplos classificadores pode ser interessante para a resolução de vários tipos de problemas envolvendo classificação de padrões em que muitas vezes são necessários vários tipos de classificadores para o processamento.

Existem casos em que os tipos de características são drasticamente diferentes em sua forma de representação e são necessários vários tipos de classificadores para a resolução do problema, ou as características possuem o mesmo formato porém são medidas físicas totalmente diferentes como temperatura e volume, ou ainda, apesar de representadas de uma mesma forma, o vetor que as representa pode possuir uma dimensionalidade muito alta sendo mais fácil reduzir sua dimensionalidade e utilizar vários classificadores.

HO (2002) reforça o uso de combinação de classificadores afirmando que antes de desenvolver um novo método de classificação deve-se fazer o melhor uso das ferramentas e metodologias existentes. Os tipos possíveis de combinação de classificadores depende do tipo da informação obtida por cada classificador. Em KUNCHEVA (2004) são apresentados quatro tipos de classificadores:

1. Oráculo: a saída de um classificador D_i para um dado x é conhecida apenas como correta ou incorreta;
2. Abstrato: cada classificador D_i produz um rótulo para uma classe $s_i \in \Omega, i = 1, \dots, L$, onde i define o número dos classificadores utilizados que chega ao máximo de L classificadores. Então, para todo objeto $x \in \mathfrak{R}^n$ a ser classificado, a saída dos L classificadores define um vetor $s = [s_1, \dots, s_L]^T \in \Omega^L$;
3. Ranking: a saída do classificador D_i é um subconjunto de Ω , com um *rank* das classes de forma ordenada; e
4. Medida: cada classificador D_i produz um vetor c -dimensional $[d_{i,1}, \dots, d_{i,c}]^T$ cujo valor $d_{i,j}$ representa a probabilidade de que o vetor x submetido à classificação seja da classe w_j . Sem perda de generalidade, pode-se assumir que cada valor neste vetor possui um valor no intervalo $[0, 1]$.

De acordo com a classificação apresentada, pode-se afirmar que o classificador utilizado neste trabalho enquadra-se no quarto tipo de classificador (medida) pois a neocognitron implementada ao final do treinamento gera uma saída com valor contínuo no intervalo $[0, 1]$.

Existem muitos métodos para realizar a combinação podendo ser agrupados em duas categorias: combinadores de saídas discretas e de saídas contínuas. Como já justificado,

os métodos analisados enquadram-se para a aplicação em saídas contínuas. Maiores informações sobre a combinação de classificadores com saídas de valores discretos encontram-se em (KITTLER et al., 1998), (KUNCHEVA, 2004), (XU et al., 1992), (DOMINGOS;PAZZANI, 1997), (MERZ, 1999), entre outros .

5.1.1 Combinadores de saída contínua

Existem duas abordagens gerais para esta tarefa de combinar classificadores com saída contínua (KUNCHEVA, 2004):

1. Conscientes da classe: nesta abordagem, os métodos usam o fato dos valores na coluna j serem os suportes individuais por classe w_j e derivam um valor de suporte geral para a classe.
2. Indiferentes da classe: nesta abordagem é ignorado o contexto do perfil de decisão $DP(x)$ e considera os valores $d_{i,j}(x)$ como atributos em um novo espaço de atributos chamado de espaço de atributos intermediário. A decisão final é tomada por outro classificador que recebe o espaço intermediário como entrada e possui como saída um rótulo de classe.

A seguir é apresentado o método *Decision Template*, classificado como indiferente da classe, utilizado em experimentos realizados neste trabalho.

Decision Template

Seja $x \in \mathfrak{R}^n$ o vetor de características e $\Omega = w_1, w_2, \dots, w_c$ o conjunto de rótulos das classes. Cada classificador D_i do conjunto $D = D_1, D_2, \dots, D_L$ fornece c graus de suporte. A notação $d_{i,j}(x)$ representa o suporte que o classificador D_i oferece à hipótese de que x pertença à classe w_j , e o perfil de decisão $DP(x)$ consiste na matriz dada pela equação 5.1:

$$DP(x) = \begin{bmatrix} d_{1,1}(x) & \dots & d_{1,j}(x) & \dots & d_{1,c}(x) \\ \vdots & & \vdots & & \vdots \\ d_{i,1}(x) & \dots & d_{i,j}(x) & \dots & d_{i,c}(x) \\ \vdots & & \vdots & & \vdots \\ d_{L,1}(x) & \dots & d_{L,j}(x) & \dots & d_{L,c}(x) \end{bmatrix} \quad (5.1)$$

A idéia básica neste método é relembrar o perfil de decisão mais típico para cada classe w_j chamando-o de *template* de decisão (DT_j) e então comparar com o perfil de decisão atual de $DP(x)$ usando alguma medida de similaridade S . Como medidas de similaridade pode-se utilizar, por exemplo, distância Euclidiana ou diferença simétrica, entre outras.

- Distância Euclidiana

$$\mu_j(x) = 1 - \frac{1}{L \times c} \sum_{i=1}^L \sum_{k=1}^c [DT_j(i, k) - d_{i,k}(x)]^2 \quad (5.2)$$

onde $DT_j(i, k)$ é a (i, k) -ésima entrada no template de decisão DT_j .

- Diferença Simétrica

$$\mu_j(x) = 1 - \frac{1}{L \times c} \sum_{i=1}^L \sum_{k=1}^c \max \{ \min [DT_j(i, k) - (1 - d_{i,k}(x))], \min [(1 - DT_j(i, k)) - d_{i,k}(x)] \} \quad (5.3)$$

Para realizar o treinamento para o método *Decision Template* deve-se para $j = 1, \dots, c$ calcular a média dos perfis de decisão $DP(Z_k)$ para todos $w_j \in Z$. A seguir, DT_j é calculado com base nos valores $DP(Z_k)$ segundo a equação 5.4.

$$DT_j = \frac{1}{N_j} \sum DP(Z_k) \quad (5.4)$$

onde N_j é o número de elementos de Z para w_j .

Na operação propriamente dita do método, dada uma entrada $x \in \mathfrak{R}^n$ é construída a $DP(x)$ e em seguida calculada sua similaridade com DT_j aplicando-se a equação 5.5.

$$\mu(x) = S(DP(x), DT_j) \quad (5.5)$$

onde $j = 1, \dots, c$.

Decision Template Modificado

O método aqui denominado *Decision Template Modificado* foi desenvolvido durante a fase das análises experimentais devido às características inerentes ao modelo proposto. Nosso modelo possui como saída um valor contínuo que representa a ativação do classifi-

gador segundo uma amostra apresentada. Entretanto, para amostras pertencentes a uma mesma classe podem ocorrer uma grande variação nos valores de ativação das amostras numa mesma classe.

Denominou-se *Decision Template* Modificado pois é um método que utiliza a DP para auxiliar na inserção do rótulo da amostra teste, entretanto não é utilizado um *template* único de decisão devido à variação de valores citada anteriormente pois ao aplicarmos a média dos suportes muitas amostras podem ficar distantes do *template* gerando assim resultados que não são satisfatórios.

No método desenvolvido são realizadas três fases de processamento:

- Treinamento: fase em que é obtida a topologia da rede para cada classe utilizada, configurando assim o classificador D_j como o responsável por uma determinada classe j ;
- Combinação: fase em que é realizado um primeiro reconhecimento de amostras previamente apresentadas no momento de treinamento, a fim de obter vetores de saída para o treinamento do combinador, correspondentes aos *templates* de decisão; e
- Teste: na fase de teste são apresentadas individualmente amostras à rede e realizado o reconhecimento no conjunto de classificadores (D_1, D_2, \dots, D_L) a fim de obter o vetor de reconhecimento que deverá ser comparado com todos os *templates* de decisão obtidos na fase de combinação.

Para realizar a comparação entre os vetores de *templates* de decisão e o vetor de reconhecimento, é realizado o cálculo da distância entre eles considerando pertencentes à um espaço euclidiano M-dimensional segundo a equação 5.6.

$$D_{L2} = \|x - y\| = \sqrt{\sum_{i=1}^M (x_i - y_i)^2} \quad (5.6)$$

Segundo a figura 5.1 é possível visualizar melhor a combinação proposta para os experimentos realizados neste trabalho. Como apresentado, na fase de combinação são gerados vetores de saída (y) por amostra apresentada que são os *templates* de decisão. Em seguida na fase de teste, o vetor de reconhecimento obtido com a apresentação da amostra de teste é comparado com cada um dos *templates* de decisão gerados e calculadas suas distâncias.

O vetor de saída que obtiver menor distância irá rotular a amostra teste com o rótulo de sua classe (Figura 5.2).

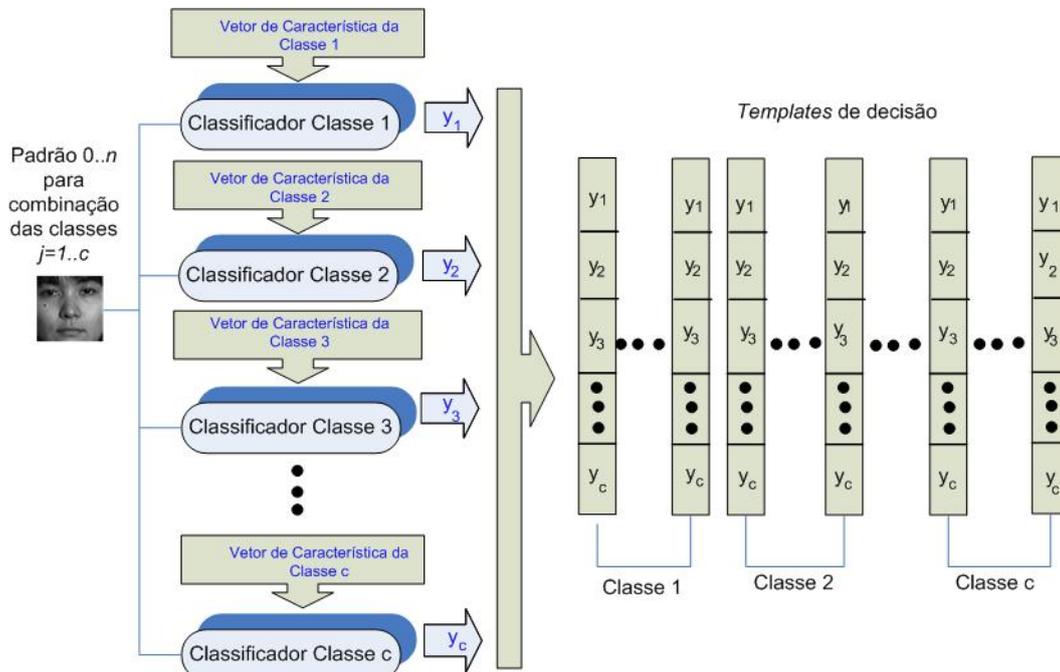


Figura 5.1: Obtenção dos *templates* de decisão.

5.2 Medidas de desempenho

A avaliação do desempenho dos classificadores é importante pois apresenta a performance do classificador auxiliando na escolha de um determinado classificador para um processo. Dentre os vários métodos para avaliar o desempenho de classificadores serão utilizados neste trabalho dois métodos: *hold-out* e coeficiente Kappa.

5.2.1 *Hold-out*

Tradicionalmente, o método *hold-out* separa o conjunto de dados disponível em duas metades: dados para treinamento e dados para teste e obtém-se como medida a taxa de erro obtida. Este método é apropriado para avaliar a performance do classificador em condições reais, mas como os dados são geralmente escassos, não utilizar parte deles para o treino pode ser catastrófico, principalmente quando existem exemplos raros. Segundo

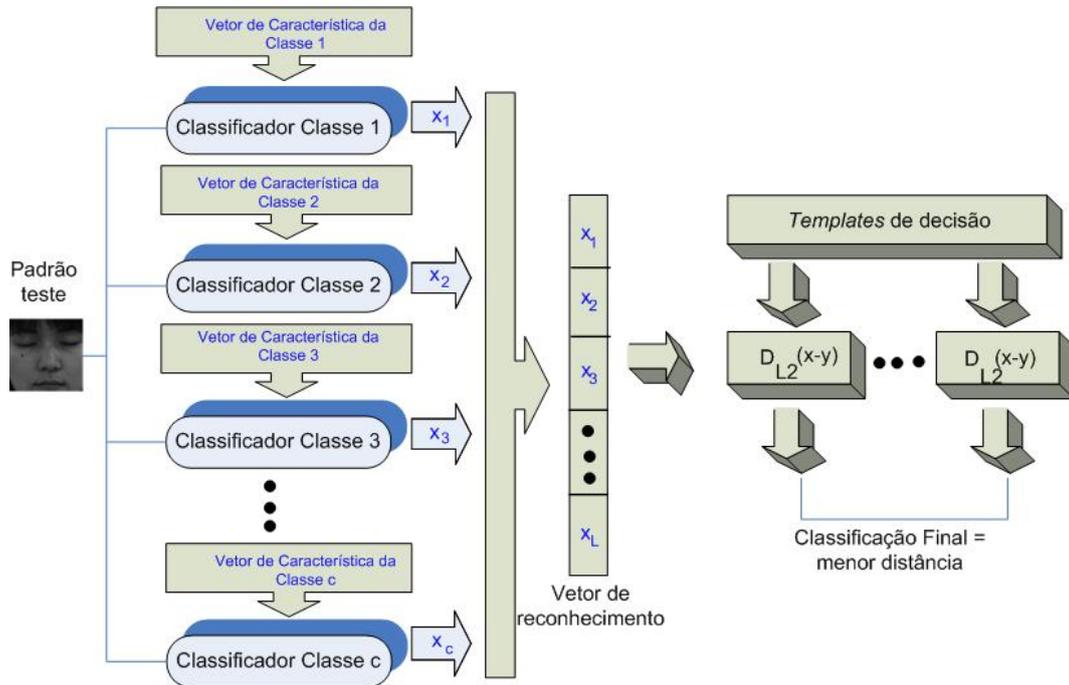


Figura 5.2: Teste de reconhecimento

esta configuração podemos considerá-lo pessimista por utilizar apenas uma parte dos dados, KOHAVI (1995) explica que quanto maior for o conjunto teste, maior é o desvio de estimação e que um número reduzido de dados para teste pode significar que o intervalo de precisão será maior.

5.2.2 Coeficiente Kappa

Este método determina a concordância entre os rótulos das amostras teste com as saídas do classificador. Para obter o valor do coeficiente Kappa é utilizada a equação 5.7:

$$K = \frac{P(A) - P(E)}{1 - P(E)} \quad (5.7)$$

onde $P(A)$ é a proporção de vezes em que o classificador acerta e $P(E)$ é a proporção de vezes que se espera que um classificador acerte se as classificações forem feitas ao acaso. O coeficiente Kappa apresenta valores no intervalo $[-1, 1]$. Sua interpretação é subjetiva. Entretanto foi utilizado como base para determinar a interpretação do resultado do Kappa a tabela 5.1 sugerida por CONGALTON (1991).

Tabela 5.1: Interpretação para coeficiente Kappa

Resultado (Kappa)	Interpretação do desempenho do classificador
$K \leq 0$	Péssimo
$0 < K \leq 0,2$	Mau
$0,2 < K \leq 0,4$	Razoável
$0,4 < K \leq 0,6$	Bom
$0,6 < K \leq 0,8$	Muito Bom
$0,8 < K \leq 1$	Excelente

5.3 Considerações finais

Partindo do fato de que os classificadores utilizados geram valores contínuos em seu neurônio ativo na saída, não foi objeto de estudo as técnicas que utilizam classificadores que possuem como saída valores discretos. Sendo assim, neste capítulo foram apresentados métodos que utilizam classificadores que possuem na saída valores contínuos. Os dois métodos explicados no decorrer deste capítulo são utilizados em experimentos na análise realizada em busca da estrutura mais adequada ao processo de reconhecimento.

Quanto às métricas para medir o desempenho dos classificadores, foram apresentadas duas medidas. O método *hold-out* é uma métrica mais tradicional que separa o conjunto de dados disponível em subconjuntos para treinamento e teste, no caso da aplicação da combinação é acrescentado um subconjunto para realizar a configuração da combinação, também conhecido como conjunto de validação. Além disso utilizando o coeficiente Kappa é possível determinar a concordância entre as amostras previamente rotuladas pelo treinamento com as saídas do classificador.

No próximo capítulo é apresentada a proposta de sistema de reconhecimento facial usando neocognitron bem como a apresentação da base de imagens utilizada e operações realizadas no pré-processamento das imagens. Por fim, é apresentado o simulador que serviu de base ao desenvolvimento do trabalho e as alterações efetuadas que levam à implementação.

Proposta de sistema de reconhecimento facial usando neocognitron

Diante da limitação na área de reconhecimento facial concentrar-se no fato de ser uma tecnologia que sofre degradação da performance com a variação da iluminação e pose, além de obter resultados com baixa satisfação do usuário quando existe diferença da época de obtenção entre a imagem que está sendo analisada e a imagem contida no banco de imagens devido ao envelhecimento do ser humano, propõe-se analisar diversas estruturas da rede neocognitron para aplicação no reconhecimento de faces e posteriormente aplicadas técnicas de combinação de classificadores a fim de obter um melhor resultado pela estrutura que se mostrou mais adequada à tarefa.

Como explicado no capítulo 4, a rede neocognitron possui a habilidade de reconhecer padrões deformados e por este motivo, propõe-se aplicar à rede neocognitron imagens da base CMU-PIE com variação de iluminação e expressão a fim de obter um sistema que

consiga trabalhar sobre estas variações e manter uma baixa taxa de erro de classificação. Ressaltando que o sistema proposto deve ser aplicado para imagens faciais estáticas.

6.1 Sistema de reconhecimento facial

Como apresentado na figura 6.1, a adaptação no fluxo do processo de reconhecimento de LI;JAIN (2005) mostra o foco do sistema desenvolvido no presente trabalho com a aplicação de estruturas de rede neocognitron nos módulos de extração e comparação de características.

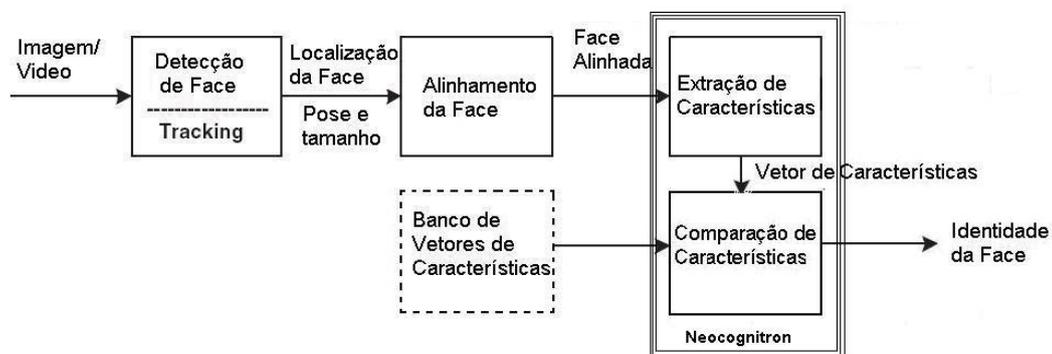


Figura 6.1: Fluxo de reconhecimento adaptado ao projeto realizado.

Com a utilização da rede neocognitron e seu aprendizado não-supervisionado, as características faciais são extraídas como explicado na seção 4.4 do capítulo 4. O conjunto de características extraídas que consistem de detalhes da face, representados pelos pesos das conexões variáveis da rede para cada característica extraída durante o treinamento, resultam num vetor de características utilizado para identificação de um indivíduo (classe).

Durante o treinamento, cada classe deve ser apresentada individualmente conforme apresentado na figura 6.2 para obter a estrutura de rede capaz de reconhecê-la juntamente com seu vetor de características a ser utilizado na fase de reconhecimento.

Na fase de reconhecimento é realizada a comparação das características do padrão de entrada com todos os vetores do banco de vetores de características, sendo classificado como pertencente a uma determinada classe segundo a combinação dos classificadores efetuada após apresentação do padrão a todos os classificadores representantes de classes

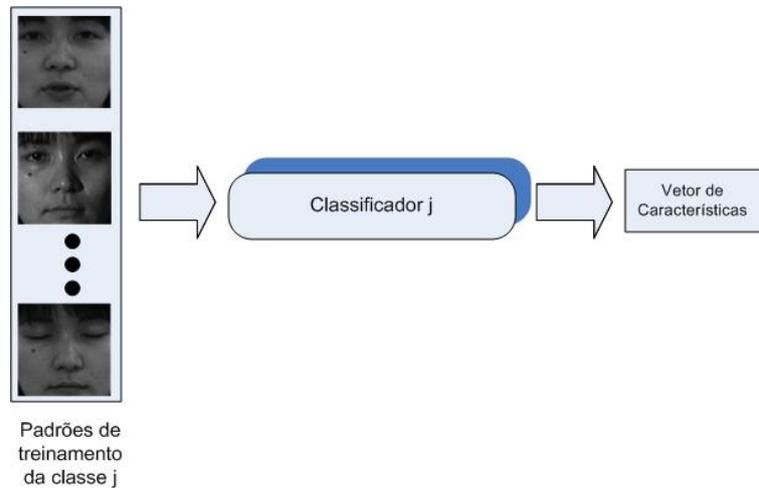


Figura 6.2: Processo realizado para extração dos vetores de características.

(figura 6.3). Ressaltando que cada classificador é caracterizado por uma estrutura de rede para uma determinada classe em conjunto com seu vetor de característica contido no banco de vetores tornando-se classificadores especialistas por classe.

No caso da aplicação de combinação de classificadores, uma fase intermediária denominada fase de combinação é realizada entre a fase de treinamento e reconhecimento (teste) a fim de obter os templates de decisão utilizados no método de combinação (figura 6.4 na aplicação do *Decision Template* e figura 5.1 na aplicação do *Decision Template Modificado*). Nesta fase são aplicadas imagens de padrões de classes conhecidas a todos os classificadores a fim de determinar, segundo equações apresentadas no capítulo 5, os templates de decisão que serão utilizados para obter os suportes à classificação dos padrões na fase de reconhecimento.

Para realização dos experimentos foi utilizada a base de dados CMU-PIE (*Carnegie Mellon University - Pose, Illumination and Expression*). Foram trabalhadas, em operações de pré-processamento, as imagens na posição frontal com variação de iluminação e expressão.

Para obter as medidas de desempenho dos classificadores aplicou-se os métodos *hold-out* com obtenção do erro estimado dos classificadores e o coeficiente Kappa.

A seguir é apresentada a base utilizada bem como o pré-processamento realizado para obter as imagens com as características necessárias à realização dos experimentos.

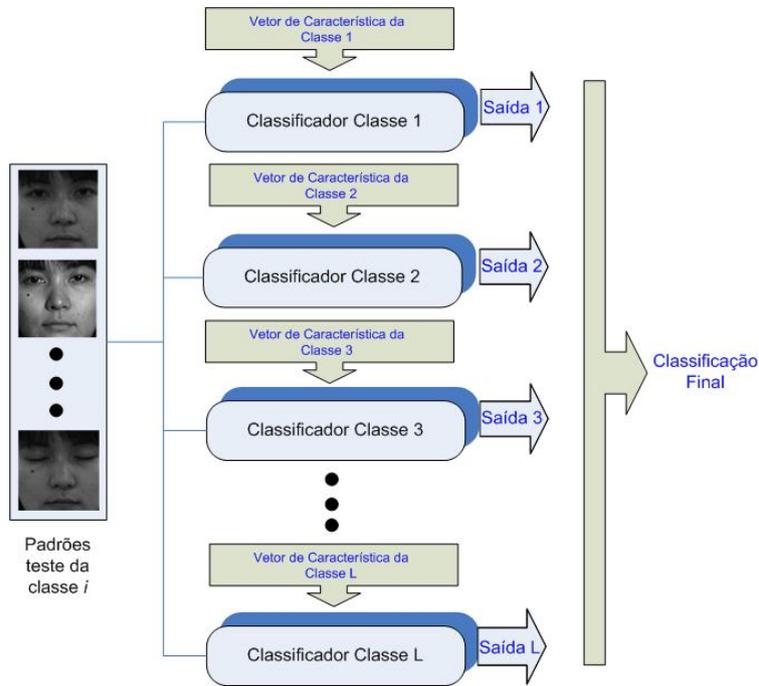


Figura 6.3: Processo realizado na fase de reconhecimento dos padrões.

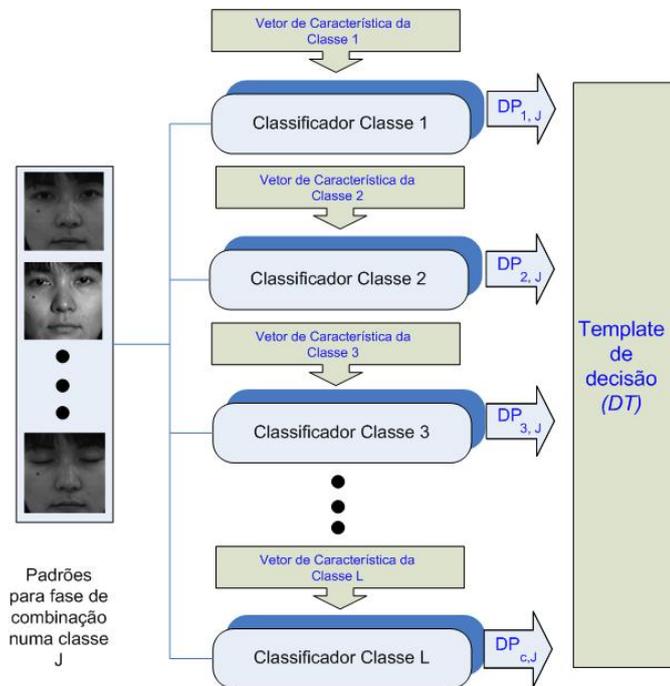


Figura 6.4: Processo realizado na fase de combinação utilizando o método *Decision Template*.



Figura 6.6: Exemplos de variação de pose e expressão na base CMU-PIE.

tativas de cada caso, SIM et al. (2003) decidiram capturar imagens com a iluminação da sala ligada e desligada considerando que são mais apropriadas para determinar a robustez de algoritmos de reconhecimento facial sob variação de luminosidade.

Quanto à variação de expressões foi solicitado a cada indivíduo que fizesse uma expressão neutra, uma sorrindo, uma piscando e uma conversando (Figura 6.6). Para as três primeiras expressões apresentadas foram capturadas 13 imagens (uma para cada câmera) nas expressões obtidas enquanto o indivíduo conversava foram capturados 2 segundos de vídeo (60 frames), entretanto, o espaço para armazenamento ficaria muito grande então os desenvolvedores consideraram apenas os dados de 3 câmeras (c27, c22 e c05). Com os indivíduos que geralmente utilizam óculos foram coletados um conjunto extra de 13 imagens sem seus óculos e em expressão natural.

As imagens obtidas são coloridas e possuem um tamanho de 640x486 pixels, totalizando uma base com aproximadamente 40Gb composta de imagens no formato PPM (*Portable Pixmap*). Como dito, são utilizadas treze variações de pose e três variações de expressão além de quarenta e três variações de iluminação.

6.3 Pré-processamento de imagens

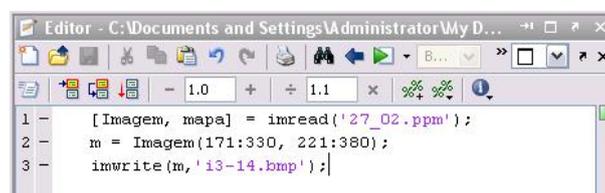
A rotina de pré-processamento consiste de uma série de operações que são aplicadas à imagem no nível de intensidade do pixel. O objetivo deste pré-processamento, no presente trabalho, é a obtenção da região facial dos indivíduos em diversos tamanhos para serem utilizados como imagem de entrada da rede neocognitron visando a eliminação de possíveis distorções, principalmente em seu redimensionamento.

Para a realização deste trabalho são utilizadas apenas as imagens referentes à câmera c27 da CMU 3D Room na base CMU-PIE por se tratar da câmera que obteve as imagens frontais dos indivíduos, foco deste trabalho.

Como uma imagem facial da base de dados CMU-PIE tem 640x486 pixels, e a região facial corresponde a 1/4 desse tamanho total (em torno de 160x160 pixels) na posição central da imagem, a imagem original teve que passar por uma rotina para obtenção da região facial e em seguida foi submetida a uma técnica de redução a fim de obter o tamanho correspondente à imagem de entrada do estágio inicial da rede.

6.3.1 Detecção da região facial

O processo de obtenção da região facial foi realizado "manualmente" utilizando o MATLAB[®] como ferramenta. Na figura 6.7 é apresentado o código utilizado. Na linha 1 a imagem é carregada para a variável *Imagem* através da função *imread()* em seguida é obtida uma subimagem na variável *m* adquirida pelo intervalo de pixels determinados nos dois eixos da imagem e por último a imagem é salva num determinado arquivo em formato BMP (*Bitmap*).



```

Editor - C:\Documents and Settings\Administrator\My D...
1 - [Imagem, mapa] = imread('27_02.ppm');
2 - m = Imagem(171:330, 221:380);
3 - imwrite(m, '13-14.bmp');
  
```

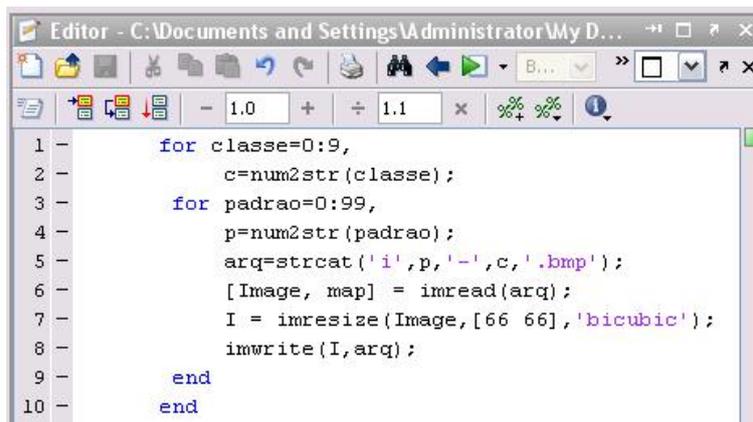
Figura 6.7: Exemplo de código utilizado para obtenção da região facial.

²MATLAB[®] é uma linguagem de alto nível e ambiente interativo desenvolvida pela The MathWorks TM que permite executar tarefas computacionalmente intensivas mais rápido do que a tradicional programação em linguagens como C, C++ e Fortran.

As imagens resultantes foram geradas em escalas de cinza (retorno *default* da função utilizada) e possui resolução de 160x160 pixels que deverá ser redimensionada para ser aplicada como imagem de entrada das estruturas a serem analisadas. É importante destacar que a transformação da imagem para escalas de cinza não prejudica sua utilização como entrada da rede pois a mesma, antes de passar para a fase de treinamento, passa por um processo de obtenção da imagem de contraste.

6.3.2 Redimensão: Interpolação bicúbica

A redimensão da imagem foi executada automaticamente segundo o código apresentado na figura 6.8, também com a ferramenta MATLAB ®. Partindo da imagem de 160x160 pixels da região facial foram obtidas as demais imagens empregadas nos experimentos.



```

1 -     for classe=0:9,
2 -         c=num2str(classe);
3 -         for padrao=0:99,
4 -             p=num2str(padrao);
5 -             arq=strcat('i',p,'-',c,'.bmp');
6 -             [Image, map] = imread(arq);
7 -             I = imresize(Image,[66 66],'bicubic');
8 -             imwrite(I,arq);
9 -         end
10 -     end

```

Figura 6.8: Código utilizado para redimensionar imagens

Inicialmente a imagem é carregada através da função *imread()* (linha 6) retornando a imagem *Image*. Aplicando-se essa imagem como entrada para a função *imresize()* é realizado o redimensionamento numa escala $[x y]$ utilizando como método a interpolação bicúbica.

A interpolação bicúbica tem a característica de usar a média dos quatro valores de pixel próximos para realizar a interpolação para cada valor do pixel da saída, garantindo assim um resultado melhor do que com a aplicação dos demais métodos suportados por esta função, vizinhança mais próxima e bilinear.

6.4 Estrutura-base de simulação dos experimentos

A realização dos experimentos é baseada num simulador de rede neocognitron do departamento de Computação da UFSCar que possui quatro estágios e recebe como entrada imagens, da base CMU-PIE, previamente pré-processadas de 57×57 pixels (SAITO et al., 2005). São necessárias alterações em sua estrutura para realização dos experimentos pois a estrutura existente apresenta-se limitada para aplicação a um grande número de classes devido ao crescimento da complexidade da rede com o aumento do número de classes. A comparação entre as estruturas pode ser observada na figura 6.9 com a apresentação da antiga estrutura e a atual estrutura na figura 6.10. Anteriormente a rede realizava um treinamento com todas as classes não aplicando o conceito de vetor de características por classe.

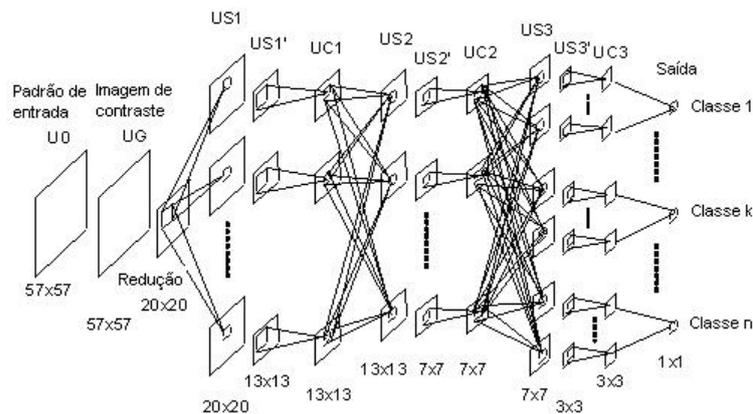


Figura 6.9: Estrutura implementada pelo simulador (SAITO et al., 2005).

Sua implementação foi realizada utilizando o ambiente de desenvolvimento DelphiTM da Borland® e está constituído de duas telas principais: treinamento e reconhecimento. No treinamento as classes são apresentadas separadamente e treinadas conforme uma mesma configuração de parâmetros especificados por: q , taxa de aprendizado que possui valor em torno de 10.000; limiares de treinamento ($train1$, $train2$ e $train3$); e limiares de reconhecimento ($theta1$, $theta2$ e $theta3$). Os limiares (*thresholds*) regulam a capacidade de generalização nos estágios um, dois e três, respectivamente, gerando um plano a cada nova característica extraída.

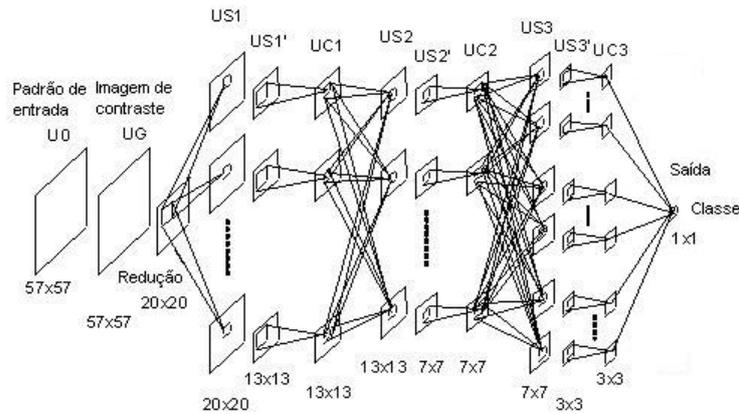


Figura 6.10: Estrutura-base dos experimentos.

Os limiares de treinamento são utilizados para a fase de extração de características e obtenção dos neurônios vencedores no treinamento de U_S enquanto que os limiares de reconhecimento são utilizados para a transição do treinamento de um estágio para o estágio subsequente. Esses limiares estão num intervalo $[0, 1]$ e os valores dos limiares de treinamento devem ser sempre maiores que seus respectivos limiares de reconhecimento, caso contrário, a rede não consegue manter a habilidade de generalizar configurando a falta de neurônios ativos para o treinamento do estágio posterior.

Inicialmente é apresentada uma imagem de 57×57 pixels, em seguida é obtida, utilizando um filtro, a imagem de contraste positivo e negativo, de mesmo tamanho. Contudo, apenas a imagem de contraste positivo é utilizada no processo pois obtém um melhor resultado para visualização das características da face na imagem. Posteriormente, é realizada a redução da imagem para 20×20 pixels aplicando um borramento espacial e em seguida eliminando-se a vizinhança das células resultantes. A imagem reduzida é então aplicada como entrada ao primeiro estágio da rede. Utilizando uma área de conexão de 7×7 pixels para uma imagem de 20×20 pixels e utilizando o limiar *train1* como parâmetros de controle da capacidade de generalização é realizado o treinamento da camada U_{S_1} . A saída da camada U_{S_1} é reduzida para planos celulares de 13×13 pixels antes mesmo da aplicação do cálculo da camada U_{C_1} (U_{S_1}).

A partir da camada U_{S_1} o treinamento prossegue com o parâmetro *theta1* para o cálculo das células-C de U_{C_1} . O mesmo procedimento é realizado para o treinamento dos demais

estágios, sendo que são utilizados os limiares $train2$ e $theta2$ para as camadas U_{S_2} e U_{C_2} e $train3$ e $theta3$ para as camadas U_{S_3} e U_{C_3} .

Como visto, a camada U_{S_2} é reduzida para planos de 7x7 pixels e U_{S_3} para 3x3 pixels finalizando um neurônio na camada de saída, com um valor contínuo correspondendo a classe treinada.

À medida que o treinamento é realizado são inseridos planos às camadas e ao final de cada estágio são indicados quantos planos foram gerados em cada estágio através das variáveis $K1$, $K2$ e $K3$, sendo que o número de planos por camada é limitado a 201 planos.

No reconhecimento o usuário deve inserir os limiares de generalização $theta1$, $theta2$ e $theta3$ e ao iniciar a fase de reconhecimento, automaticamente, os padrões a serem reconhecidos são apresentados e comparados aos vetores de características existentes no banco de vetores de características. O padrão é rotulado como pertencente a uma determinada classe de acordo à estrutura que obtiver maior valor de ativação na camada de saída.

6.5 Neocognitron multi-classes X neocognitron de uma classe

No caso do neocognitron para reconhecimento de múltiplas classes (figura 6.9), a última camada (U_C) da rede tem vários planos pertencentes a uma classe devido ao aprendizado não-supervisionado adotado pela rede. Esses vários planos fornecem dados para a obtenção do neurônio que representa a classe respectiva. Esse último neurônio é do tipo perceptron.

À medida que o número de classes aumenta, o número de planos celulares nos diversos estágios da rede cresce e o desempenho da rede diminui.

No caso de se aplicar o neocognitron para o treinamento de uma classe, é possível obter dois tipos de saída. A primeira é a saída semelhante ao caso anterior da neocognitron de múltiplas classes, onde um único neurônio do tipo perceptron recebe sinais de vários planos celulares, e é ativo quando a classe é reconhecida e é inativo, caso contrário.

Uma outra possibilidade é a obtenção de um vetor de saídas que coincide com os valores dos planos celulares da última camada (U_C). Nesse caso é conveniente que os planos dessa última camada tenham a sua dimensão reduzida para um único neurônio.

Para o reconhecimento de diversas classes com essa estrutura é necessário treinar

várias redes, cada uma para uma classe específica. Um padrão de teste é então apresentado para todas as redes e aquela rede que tiver a saída ativa corresponde à classe reconhecida.

Uma outra possibilidade é o uso de uma técnica de combinação de classificadores para verificar a classe correspondente ao padrão de teste apresentado. Nota-se que usando estruturas de redes neocognitron de uma única classe e uma técnica de combinação de classificadores, o número de classes pode aumentar sem prejudicar o desempenho final de classificação, desde que a estrutura de rede neocognitron para cada classe não cresça. Nesse caso, o limite de crescimento depende do combinador de classificadores.

6.6 Considerações finais

O presente capítulo apresentou o sistema proposto para o reconhecimento facial usando neocognitron. Discutiu-se a implementação utilizando multi-classes e uma classe buscando justificar a utilização da implementação com uma classe.

Além disso, foi apresentada a base CMU-PIE utilizada, desenvolvida na Universidade de Carnegie Mellon, e as operações de pré-processamento realizadas para obtenção das imagens segundo as estruturas analisadas.

No próximo capítulo é apresentado um resumo das principais características da implementação utilizada, o trabalho realizado e os resultados obtidos. São feitas considerações a respeito dos resultados obtidos.

Trabalho realizado e resultados obtidos

Esse capítulo apresenta os resultados dos experimentos realizados utilizando diversas estruturas de rede neocognitron na tarefa de reconhecimento facial. Com base na metodologia apresentada no capítulo 6 foram realizados os experimentos e calculadas as medidas de desempenho dos classificadores além de discutidos os resultados obtidos.

7.1 Sistema de reconhecimento de faces implementado

Como apresentado no decorrer do trabalho, nosso foco está na aplicação da rede neocognitron nos módulos de extração e comparação de características de imagens frontais. A implementação desenvolvida utilizou como base o treinamento não-supervisionado da rede neocognitron numa estrutura desenvolvida para treinamento de uma classe por vez, sendo o sistema facial constituído da combinação de um conjunto de redes neocognitron. Com esta implementação, não é necessária a presença de um supervisor para aprendizado pois a rede realiza auto-organização baseada no princípio da competição além de conseguirmos

aumentar o número de classes sem que o número de planos celulares cresça indiscriminadamente para cada rede neocognitron do sistema, garantindo que o desempenho da rede não diminua.

Para garantir que com o aumento do número de classes não haverá aumento do custo computacional por classe treinada foi aplicado o armazenamento de vetores de características por classe treinada num banco de vetores de características. Desta forma, a cada nova classe inserida no processo não resulta em aumento do número de planos celulares nos estágios da rede, e sim ao treinamento de uma nova rede neocognitron.

Na fase de treinamento da rede cada neocognitron é treinada para uma classe específica com a apresentação de um conjunto de amostras de treinamento contendo 36 padrões a cada classe. Como resultado é obtida a estrutura topológica da rede com a definição do número de planos por estágio e o vetor de características da classe com os pesos das conexões variáveis da rede para cada característica extraída durante o treinamento. Realizado o treinamento de todas as classes, passamos ao processo de reconhecimento em que amostras não-coincidentes com o conjunto de treinamento é utilizado para obter as medidas de desempenho do sistema. Ressalta-se que cada amostra é apresentada a todos os classificadores (neocognitron), sendo o classificador que possuir em seu neurônio de saída maior valor para ativação será o responsável pelo rótulo da amostra.

Para definição do tamanho de imagem mais adequado ao reconhecimento e sua estrutura de redimensionamento de imagens foram realizados diversos experimentos que são abordados na seção a seguir.

7.2 Experimentos

Para realização dos experimentos foram utilizadas 30 classes de padrões de imagens frontais da região facial sendo que para cada classe foram disponibilizados 108 padrões. Utilizando o método *hold-out* como base para divisão do banco de imagens, foram separados 1/3 das imagens para cada conjunto envolvido nos experimentos: conjunto de treinamento; combinação; e teste.

Para os primeiros experimentos em que não foi realizada combinação de classificadores foram utilizados apenas 72 padrões por classe, representados pelos conjuntos de treina-

mento e teste. Contudo, nos experimentos em que foram aplicadas a combinação de classificadores, um conjunto de padrões distintos para combinação foi adicionado contendo 36 padrões. Ressalta-se que para que a comparação se tornasse mais confiável foram utilizados, tanto nos experimentos sem combinação quanto nos experimentos com combinação, os mesmos conjuntos de treinamento e teste.

7.2.1 Experimentos variando o tamanho do padrão de entrada sem usar combinadores

Para uma primeira análise foram escolhidos aleatoriamente 5 tamanhos de imagens de entrada: 57x57, 60x60, 74x74, 120x120 e 160x160 *pixels*, sendo que para os dois últimos tamanhos de imagens foram aplicadas redes com quatro estágios. A partir destas imagens foram realizadas as reduções dentro da estrutura de rede neocognitron conforme apresentado na tabela 7.1 a seguir:

Tabela 7.1: Estruturas utilizadas nos experimentos

Tamanho de imagem	Estágios	U_0	U_{S1}	U_{S2}	U_{S3}	U_{S4}	Saída
57x57	3	57x57	21x21	13x13	7x7	-	1x1
60x60	3	60x60	41x41	27x27	13x13	-	1x1
74x74	3	74x74	49x49	33x33	17x17	-	1x1
120x120	4	120x120	91x91	69x69	47x47	23x23	1x1
160x160	4	160x160	121x121	91x91	61x61	31x31	1x1

No processo de treinamento, os padrões foram aplicados e obteve-se a topologia da rede neocognitron e seus respectivos vetores de características (figura 6.2) para cada classe. Após o treinamento, cada padrão do conjunto de teste (1080 imagens, sendo 30 classes e 36 imagens por classe) foi apresentado para o processo de reconhecimento, a cada um dos classificadores de classe, conforme apresentado pela figura (figura 6.3). A escolha do rótulo da amostra (classe) é dada pelo classificador cuja saída obteve uma maior ativação.

A fim de obter os resultados de estrutura com taxas mais adequadas ao reconhecimento, os limiares de treinamento foram variados de acordo com o número de planos gerados pela classe. De acordo com a literatura (SAITO et al., 2005), uma camada com número de planos em torno de cinquenta possui melhores resultados de reconhecimento. Os limiares iniciaram com os valores: $\theta_1^t = 0.75$, $\theta_2^t = 0.70$ e $\theta_3^t = 0.65$ para os experimentos com redes de 3 estágios e $\theta_1^t = 0.75$, $\theta_2^t = 0.70$, $\theta_3^t = 0.65$ e $\theta_4^t = 0.60$ para redes de 4 estágios na fase

de treinamento, onde o sobrescrito t indica que o limiar referido foi utilizado na fase de treinamento.

Para a fase de reconhecimento (teste) foram utilizados limiares iniciais $\theta_1^r = 0.70$, $\theta_2^r = 0.65$ e $\theta_3^r = 0.60$ para os experimentos com redes de 3 estágios e $\theta_1^r = 0.70$, $\theta_2^r = 0.65$, $\theta_3^r = 0.60$ e $\theta_4^r = 0.55$ para redes de 4 estágios, onde o sobrescrito r atribui o limiar à fase de reconhecimento, e realizados vários experimentos com a variação destes limiares até os valores dos limiares utilizados para treinamento pois, como explicado anteriormente, os limiares de reconhecimento devem possuir valores menores do que os limiares de treinamento a fim de manter a habilidade de generalização.

7.2.2 Experimentos para imagens de tamanho 57x57 usando combinadores

Após realizados os primeiros experimentos e encontrada a estrutura com menor erro estimado de acordo com o maior coeficiente Kappa foi aplicada a combinação de classificadores a fim de melhorar os resultados obtidos utilizando três métodos: *Decision Template* com distância Euclidiana (DTDE), *Decision Template* com Diferença Simétrica (DTDS) e o método *Decision Template* Modificado (DTM) (apresentados na seção 5.1.1) sendo que no último método citado foi aplicado como conjunto de combinação o mesmo conjunto de treinamento dos classificadores a fim de garantir que os classificadores pudessem ter o máximo de desempenho na combinação.

Para realização dos experimentos foram utilizados dois computadores AMD Athlon 64x Dual Core Processor 4200+ com 2.21GHz e 2GB de RAM. O tempo de processamento para o treinamento de uma rede neocognitron com o tamanho 57x57 de padrão de entrada, foi de aproximadamente, 2 horas, enquanto que o treinamento para a rede neocognitron com o tamanho 160x160, teve o tempo de duração de 36 horas. O tempo de processamento para a fase de reconhecimento é substancialmente menor em relação ao tempo de treinamento, sendo menor que 1 segundo, para ambos os casos.

7.3 Resultados

A seguir são apresentados os resultados obtidos aos dois experimentos descritos.

7.3.1 Resultado dos experimentos 7.2.1 variando o tamanho das amostras

Da tabela 7.2 até a tabela 7.6 são apresentados os resultados dos experimentos iniciais realizados segundo o erro estimado e o coeficiente Kappa encontrado em cada experimento realizado. Ressaltamos que o erro estimado foi calculado segundo equação 7.1 (KUNCHEVA, 2004) a seguir, onde N_{error} representa o número de erros de classificação apresentados pelo classificador D e N_{ts} o número de amostras totais de teste, sendo que foram considerados erros de classificação tanto a não classificação de uma amostra ou uma classificação errônea da amostra.

$$Error(D) = \frac{N_{error}}{N_{ts}} \quad (7.1)$$

Tabela 7.2: Erro estimado e coeficiente Kappa para a estrutura de tamanho de 57x57

Limiares	Erro Estimado	Coeficiente Kappa
$\theta_1^r = 0.72; \theta_2^r = 0.69; \theta_3^r = 0.62$	0.3675	0.6198
$\theta_1^r = 0.74; \theta_2^r = 0.69; \theta_3^r = 0.63$	0.3008	0.6888
$\theta_1^r = 0.74; \theta_2^r = 0.69; \theta_3^r = 0.64$	0.3038	0.6857
$\theta_1^r = 0.75; \theta_2^r = 0.72; \theta_3^r = 0.70$	0.2167	0.7758

Tabela 7.3: Erro estimado e coeficiente Kappa para a estrutura de tamanho de 60x60

Limiares	Erro Estimado	Coeficiente Kappa
$\theta_1^r = 0.65; \theta_2^r = 0.62; \theta_3^r = 0.58$	0.6959	0.2801
$\theta_1^r = 0.70; \theta_2^r = 0.66; \theta_3^r = 0.62$	0.6901	0.2861
$\theta_1^r = 0.76; \theta_2^r = 0.72; \theta_3^r = 0.68$	0.9601	0.0067
$\theta_1^r = 0.77; \theta_2^r = 0.73; \theta_3^r = 0.68$	0.9612	0.0057

Tabela 7.4: Erro estimado e coeficiente Kappa para a estrutura de tamanho de 74x74

Limiares	Erro Estimado	Coeficiente Kappa
$\theta_1^r = 0.68; \theta_2^r = 0.62; \theta_3^r = 0.57$	1.000	-0.0341
$\theta_1^r = 0.68; \theta_2^r = 0.63; \theta_3^r = 0.58$	0.9050	0.0982
$\theta_1^r = 0.69; \theta_2^r = 0.63; \theta_3^r = 0.58$	1.000	-0.0341
$\theta_1^r = 0.72; \theta_2^r = 0.67; \theta_3^r = 0.62$	0.7425	0.2319

7.3.2 Resultado dos experimentos 7.2.2 usando combinadores

Como explicado anteriormente, com base nos resultados das tabelas apresentadas, aplicou-se a combinação de classificadores à estrutura que se mostrou mais adequada

Tabela 7.5: Erro estimado e coeficiente Kappa para a estrutura de tamanho de 120x120

Limiares	Erro Estimado	Coeficiente Kappa
$\theta_1^r = 0.65; \theta_2^r = 0.62; \theta_3^r = 0.58; \theta_4^r = 0.53$	0.9147	0.0537
$\theta_1^r = 0.68; \theta_2^r = 0.63; \theta_3^r = 0.60; \theta_4^r = 0.55$	0.8883	0.0811
$\theta_1^r = 0.70; \theta_2^r = 0.65; \theta_3^r = 0.60; \theta_4^r = 0.55$	0.8845	0.0850
$\theta_1^r = 0.72; \theta_2^r = 0.67; \theta_3^r = 0.62; \theta_4^r = 0.58$	0.8673	0.1028

Tabela 7.6: Erro estimado e coeficiente Kappa para a estrutura de tamanho de 160x160

Limiares	Erro Estimado	Coeficiente Kappa
$\theta_1^r = 0.65; \theta_2^r = 0.62; \theta_3^r = 0.58; \theta_4^r = 0.53$	0.99047	-0.0245
$\theta_1^r = 0.68; \theta_2^r = 0.63; \theta_3^r = 0.60; \theta_4^r = 0.55$	0.9774	-0.0113
$\theta_1^r = 0.70; \theta_2^r = 0.65; \theta_3^r = 0.60; \theta_4^r = 0.55$	0.9721	-0.0055
$\theta_1^r = 0.72; \theta_2^r = 0.67; \theta_3^r = 0.62; \theta_4^r = 0.58$	0.9647	0.0020

ao reconhecimento facial com a obtenção de menor erro estimado e melhor coeficiente kappa representado pela estrutura de tamanho 57x57.

Nas tabelas 7.7 e 7.8 encontram-se o desempenho da estrutura de tamanho 57x57 com a aplicação da combinação de classificadores. Diante de todos os resultados apresentados, concluímos que a estrutura de tamanho 57x57 obteve melhor resultado obtendo um menor erro estimado e melhor classificação no coeficiente Kappa.

Tabela 7.7: Erro estimado para aplicação da combinação de classificadores

Limiares	DTDE	DTDS	DTM
$\theta_1^r = 0.75; \theta_2^r = 0.70; \theta_3^r = 0.65$	0.9693	0.9972	0.9666
$\theta_1^r = 0.72; \theta_2^r = 0.69; \theta_3^r = 0.62$	0.9972	0.9791	0.9666
$\theta_1^r = 0.72; \theta_2^r = 0.68; \theta_3^r = 0.63$	0.9943	0.9722	0.9647

7.4 Discussões

Através dos resultados obtidos e apresentados anteriormente foram realizadas análises e algumas observações devem ser discutidas.

7.4.1 Discussão sobre os experimentos 7.2.1

Inicialmente, percebeu-se que a rede possui uma variação de desempenho diante da variação da resolução da imagem de entrada, áreas de conexões utilizadas e limiares. Entretanto seria necessário um número muito grande de experimentos para conseguir abranger uma representação mais exata da variedade de desempenho conforme a modifi-

Tabela 7.8: Coeficiente Kappa para aplicação da combinação de classificadores

Limiares	DTDE	DTDS	DTM
$\theta_1^r = 0.75; \theta_2^r = 0.70; \theta_3^r = 0.65$	-0.0026	-0.0315	0.0001
$\theta_1^r = 0.72; \theta_2^r = 0.69; \theta_3^r = 0.62$	-0.0315	-0.0066	0.0001
$\theta_1^r = 0.72; \theta_2^r = 0.68; \theta_3^r = 0.63$	-0.02855	-0.0056	0.0020

cação dos parâmetros citados. Em relação ao aumento de um estágio na rede neocognitron era esperada uma melhoria dos resultados explicada melhor distribuição dos fatores processados. Entretanto, não foi possível obter uma melhora significativa nos resultados como era esperado. Talvez a alteração das áreas de conexões utilizadas consiga obter uma melhora no desempenho destas estruturas.

Nas tabelas 7.2 a 7.6 são mostrados os resultados alcançados e foi possível observar que uma pequena alteração no tamanho da imagem (57x57 para 60x60) obteve uma grande variação do desempenho que pode ser explicada não pela variação do tamanho da imagem de entrada e sim pela variação do tamanho dos planos celulares nos estágios da rede.

7.4.2 Discussão sobre os experimentos 7.2.2

Ao utilizarmos o combinador de classificadores *Decision Template* encontramos uma dificuldade para obter resultados satisfatórios pelo fato de numa mesma classe haver uma grande variação no valor de ativação da célula de saída (suporte). Esta observação está associada à premissa de que para a obtenção dos *templates* de decisão é necessária a utilização da média dos suportes para posteriormente ser calculada a distância entre a saída obtida na fase de reconhecimento e o *template* de decisão. Com a variação de suportes encontrada, a distância entre as saídas e o *template* mostraram-se grandes não retornando a classificação correta da maioria dos padrões como pode ser observado nas tabelas 7.7 e 7.8. No segundo método também encontramos esta dificuldade pois continuamos trabalhando com a média dos suportes apesar da alteração do cálculo da distância para a regra da diferença simétrica.

Diante desta característica inerente aos classificadores das classes de obter uma grande variação na ativação da saída numa mesma classe, foi desenvolvida uma modificação, conforme apresentado na seção 5.1.1, no método *Decision Template* em que, ao invés de trabalhar com um *template* de decisão obtido pela média das saídas, assumimos que cada

vetor de saída obtido pela apresentação de uma amostra a todos os classificadores (figura 5.1) seria um *template* de decisão de uma classe. A partir desta premissa foi calculada a distância entre o vetor de saída obtido pela amostra de teste e todos os templates adquiridos na fase de combinação dos classificadores. Contudo, apesar do esforço para aplicar a combinação de classificadores não foi possível obter uma melhora significativa nos resultados, o que pode ser atribuída à uma possível similaridade existente entre as classes utilizadas. Observou-se que o somatório realizado no cálculo da distância entre o vetor de saída e os *templates* descarta a informação de qual classificador corresponde aquela saída podendo ocorrer erros no momento de rotular a amostra de acordo com a classe.

Com os dados apresentados na tabela 7.8, apesar dos resultados não serem satisfatório percebemos que o método DTM obteve, dentre os métodos de combinação utilizados, o melhor resultado reforçando a observação realizada a respeito da obtenção dos templates a partir da média.

A avaliação dos classificadores segundo a tabela de interpretação do coeficiente Kappa (tabela 5.1) mostra que os classificadores obtiveram desempenho de razoável a muito bom, enquanto que a estimação de erro na maioria dos experimentos mostrou-se alta em relação à melhor estrutura encontrada.

Conclusões

Após a realização de experimentos e sua discussão apresentamos neste capítulo as conclusões a respeito do trabalho realizado.

Neste trabalho foi possível reforçar a utilização da rede neural neocognitron na aplicação à tarefa de reconhecimento facial. A rede neocognitron apresenta a importante habilidade de ser capaz de generalizar padrões distorcidos diante de ambientes não-controlados que a colocam como forte candidata a ser utilizada no reconhecimento de faces sob variação de iluminação expressão.

Foi revelado que a variedade das imagens do conjunto de treinamento é fundamental no processo, pois a partir destas amostras é que são extraídas as características relevantes da classe pela neocognitron. A alteração na base de imagens utilizadas exercerá uma forte influência sobre o desempenho dos classificadores utilizados.

Diante dos experimentos conclui-se que o fato de uma imagem ter um tamanho maior não implica em melhor desempenho automaticamente, em relação a uma imagem de tamanho menor quando se aplica a rede neocognitron, existindo uma diversidade de parâmetros que pode influenciar no desempenho dos classificação.

Muitos trabalhos têm sido desenvolvidos aplicando combinação de classificadores e tem

obtido resultados satisfatórios na melhoria do desempenho da classificação. Contudo, neste trabalho, as características do modelo aplicado requer um estudo mais aprofundado dos combinadores que são passíveis de utilização neste tipo de implementação. Acredita-se otimizando o uso dos limiares de cada ativação no vetor de saída melhores resultados possam ser alcançados. Desta forma pode-se aplicar a comparação apenas com *templates* referentes às classes candidatas a classificação, descartando a possibilidade do vetor de saída obter uma distância menor em relação a um *template* de uma classe que não chegou a possuir ativação em relação a amostra aplicada.

Conclui-se também que a utilização do método *hold-out* e coeficiente Kappa foram de grande valia na análise dos resultados, o método *hold-out* no fornecimento do erro estimado, enquanto que o coeficiente Kappa, após aplicação da interpretação encontrada na tabela 5.1, fornece uma informação mais abstrata a respeito dos classificadores utilizados, porém não menos importante.

8.1 Trabalhos futuros

Como extensão deste trabalho sugerimos:

- Estudo de métodos de combinação que melhor se adequem ao modelo da rede neocognitron;
- Realização de outros experimentos a fim de realizar um estudo mais detalhado das variáveis que influenciam o desempenho da rede; e
- Implementar um sistema completo de reconhecimento facial com todos os módulos referentes ao fluxo do processo, estudando a possibilidade da utilização da neocognitron em outros módulos, como na detecção tendo como base a utilização de pontos de controle.

Referências Bibliográficas

- ABATE, A. F.; NAPPI, M.; RICCIO, D.; SABATINO, G. 3d face recognition using normal sphere and general fourier descriptor. **18th International Conference on Pattern Recognition (ICIPR 2006)**. v. 3, p. 1183–1186, 2006.
- AL-AKKAD, A. Face recognition based on curvature and neural network. In: Proceedings of International Conference on Computer Graphics, 2004., 2004.
- AZEVEDO, F. M.; BRASIL, L. M.; OLIVEIRA, R. C. L. **Redes Neurais com Aplicações em Controle e em Sistemas Especialistas**. Bookstore, 2000.
- BIANCHINI, A. **Arquitetura de Redes Neurais para o Reconhecimento Facial Baseado no Neocognitron**. Dissertação (Mestrado) - Universidade federal de são carlos, 2001.
- BRAGA, A.; LURDEMIR, T.; CARVALHO, A. **Redes Neurais Artificiais: teoria e aplicações**. Rio de Janeiro: Livros Técnicos e Científicos Editora, 2000.
- CHELLAPPA, R.; WILSON, C.; SIROHEY, S. Human and machine recognition of face: a survey. In: Proceedings of the IEEE, 5, 1995., 1995. p. 705–740.
- CHEN, Q.; YAO, J.; CHAM, W. K. 3d model-based pose invarriant face recognition from multiple views. **IET Conmput. Vis.** v. 1, p. 25–34, 2007.
- CONGALTON, R. A review assessing the accuracy of classification of remotely sensed data. **Remote Sensing Environment**. v. 37, p. 35–46, 1991.

- DOMINGOS, P.; PAZZANI, M. On the optimality of the simple bayesian classifier under zero-one loss. **Machine Learning**. v. 29, p. 103–130, 1997.
- FASEL, B. Robust face analysis using convolutional neural networks. In: Proceedings of IEEE Pattern Recognition, 2002., 2002. p. 40–43.
- FÉRAUD, R. A fast and accurate face detector based on neural networks. In: IEEE Transactions on Pattern Analysis and Machine Intelligence, 1, 2001., 2001. p. 42–53.
- FREEMAN, J.; SKAPURA, D., 1992. **Neural networks Algorithms, Applications and Programming Techniques**, chapter Neocognitron, p. 373–393. Pearson Education: India.
- FUKUSHIMA, K. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected shift in position. **Biological Cybernetics**. v. 36, n. 4, p. 193–202, 1980.
- FUKUSHIMA, K. A neural network for visual pattern recognition. In: IEEE Computer, 3, 1988., 1988. p. 65–75.
- FUKUSHIMA, K., 2003. **The Handbook of Brain Theory and Neural Network**, chapter Neocognitron: a Model for Visual Pattern Recognition, p. 715–719. Brandford Books. MIT Press, 2^a. ed.
- FUKUSHIMA, K. Neocognitron capable of incremental learning. **Neural Networks**. v. 17, n. 1, p. 37–46, 2004.
- FUKUSHIMA, K.; TANIGAWA, M. Use of different threshld in learning and recognition. **Neurocomputing**. v. 11, p. 1–17, 1996.
- FUKUSHIMA, K.; WAKE, N. Improved neocognitron with bend-detecting cells. In: IEEE International Join Conference on Neural Networks, 1992., 1992. p. 190–195.
- GROSS, R.; BAKER, S.; MATTHEWS, I.; KANADE, T., 2005. **Handbook of Face Recognition**, chapter Face Recognition Across Pose and Illumination, p. 193–216. Springer.
- HAYKIN, S. **Redes Neurais: Princípios e Prática**, 2^a. ed. Bookman, 2001.
- HIRAKURI, M. **Aplicação da rede Neural Neocognitron para Reconhecimento de Atributos Faciais**. Dissertação (Mestrado) - Universidade federal de são carlos, 2003.

- HO, T. K., 2002. **Hybrid Methods in Pattern Recognition**, chapter Multiple Classifier Combination: Lessons and the Next Steps, p. 171–198. World Scientific Publishing.
- HUANG, T.; XIONG, Z.; ZHANG, Z., 2005. **Handbook of Face Recognition**, chapter Face Recognition Applications, p. 371–390. Springer.
- JAIN, A.; L.HONG; PANKANTI, S. Biometric identification. **Communications of the ACM**. v. 43, p. 91–98, 2000.
- KANADE, T.; SAITO, J.; VEDULA, S. **The 3D Room: Digitizing Time-Varying 3D Events by Synchronized Multiple Video Stream**, 1998. Relatório técnico.
- KARAYIANNIS, N. B.; VENETSANOPOULOS, A. N. **Artificial Neural Networks: Learning Algorithms, Performance Evaluation and Applications**. Kluwer Academic Publisher, 1993.
- KITTLER, J.; HATEF, M.; DUIN, R.; MATAS, J. On combining classifiers. **Transactions on Pattern Analysis and Machine Intelligence**. v. 20, p. 226–239, 1998.
- KOHAVI, R. A study of cross-validation and bootstrap for accuracy estimation and model selection. In: Proceedings of 14th International Joint Conference on A. I., 1995., 1995. p. 1137–1143.
- KUNCHEVA, L. I. **Combining Pattern Classifiers**. Wiley-Interscience, 2004.
- LAWRENCE, S.; GILES, C.; TSOI, L. Convolutional neural networks for face recognition. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1996., 1996. p. 217–222.
- LAWRENCE, S.; GILES, C.; TSOI, L.; BACK, A. Face recognition: Convolutional neural networks approach. **IEEE Transaction on Neural Networks, Especial Issue on Neural Network and Pattern Recognition**. v. 8, n. 1, p. 98–113, 1997.
- LECUN, Y.; BENGIO, Y., 2003. **The Handbook of Brain and Neural Networks**, chapter Convolutional Networks for Images, Speech and Neural Networks, p. 276–279. Bradford Books. MIT Press, 2^a. ed.
- LI, S.; JAIN, A., 2005. **Handbook of Face Recognition**, chapter Introduction, p. 1–11. Springer.

- LOCKIE, M. Face recognition. **Biometric Technology Today**. v. 11, n. 11, p. 10–11, 2003.
- LOCKIE, M. Biometric statistics in focus. **Biometric Technology Today**. v. 14, n. 2, p. 7–9, 2006.
- LOESCH, C.; SARI, S., 1996. **Redes Neurais Artificiais: fundamentos e modelos**, chapter Escolha de redes e desenvolvimento de aplicações, p. 45–53. FURB.
- LOVELL, D.; DOWNS, T.; TSOI, A. An evaluation of the neocognitron. **IEEE Transaction on Neural Networks**. v. 8, n. 5, p. 1090–1105, 1997.
- LU, X., 2003. Image analysis for face recognition. Personal notes.
- MATSUGU, M.; MORI, K.; MITARI, Y.; KANEDA, Y. Facial expression recognition combined with robust face detection in a convolutional neural network. In: International Joint Conference on Neural Networks, 2003., 2003.
- MERZ, C. Using correspondence analysis to combiner classifiers. **Machine Learning**. v. 36, p. 33–58, 1999.
- PHILIPS, J.; GROTH, P.; MICHEALS, R., 2005. **Handbook of Face Recognition**, chapter Evaluations Methods in Face Recognition, p. 329–348. Springer.
- PHILLIPS, P. J.; SCRUGGS, W. T.; O'TOOLE, A. J.; FLYNN, P. J.; BOWYER, K. W.; SCHOTT, C. L.; SHARPE, M. Fvt 2006 and ice 2006 large-scale results. **National Institute of Standards and Technology**, p. 1–56, 2007.
- ROWLEY, H.; BAJULA, S.; KANADE, T. Neural network-based face detection. **IEEE Transaction on pattern analysis and machine intelligence**. v. 20, n. 1, p. 23–38, 1998.
- SAITO, J. H.; CARVALHO, T. V.; HIRAKURI, M.; SAUNITE, A.; IDE, A. N.; ABIB, S. Using cmu-pie human face database to a convolutional neural network - neocognitron. In: Proceedings of European Symposium on Artificial Neural Networks, 2005., 2005.
- SIM, T.; BAKER, S.; BSAT, M. The cmu pose, illumination and expression database. **IEEE Transaction on Pattern Analysis and Machine Intelligence**. v. 25, n. 2, p. 1615–1618, 2003.

- SMACH, F.; ATRI, M.; MITERAN, J.; ABID, M. Design of a neural network classifier for face detection. **IEEE Transaction on Engineering, Computing and Technology**. v. 5, p. 274–277, 2005.
- VALENTIN, D.; ABDI, H.; EDELMAN, B.; O'TOOLE, A. Principal component and neural network analyses of face images: What can be generalized in gender classification. **Journal of Mathematical Psychology**. v. 42, p. 175–194, 1997.
- WISKOTT, L.; FELLOUS, J.; KRUGER, N. Face recognition by elastic bunch graph matching. **IEEE Transactions on Pattern Analysis and Machine Intelligence**. v. 19, p. 775–779, 1997.
- XU, L.; KRZYZAK, A.; C.SUEN. Methods of combining multiple classifiers and their applications to handwriting recognition. **IEEE Transactions on Systems, Man, and Cybernetics**. v. 22, n. 3, p. 418–435, 1992.
- YANG, M.; KRIEGMAN, D.; AHUJA, N. Detecting faces in images: a survey. **Transaction on Pattern Analysis and Machine Intelligence**. v. 24, p. 34–58, 2002.
- YUAN, X.; LU, J.; YAHAGI, T. A method of 3d face recognition based on principal component analysis algorithm. In: IEEE International Symposium on Circuits and Systems, 2005., 2005.
- ZHAO, W.; CHELLAPPA, R., 2006. **Face Processing:Advanced Modeling and Methods**, chapter A Guider Tour of Face Processing, p. 3–54. Elsevier.
- ZHAO, W.; CHELLAPPA, R.; ROSENFELD, A.; PHILLIPS, P. **Face Recognition: A Literature Survey**, 2002. Relatório técnico, CART-TR-948.University of Maryland.