

Arquiteturas Construtivas Autoassociativas para a Verificação Automática de Assinaturas

Juliana Ribeiro, Germano Vasconcelos
Departamento de Informática – UFPE
Cx Postal 7851, CEP 50.732-970, Recife-PE
E-mails: jngr@di.ufpe.br, gcv@di.ufpe.br

Abstract

In this paper, four autoassociator-based neural networks are investigated on an automatic signature verification task. The first is a neural network based on the traditional Multilayer-Perceptron (MLP) architecture, and the other architectures are the constructive structures based on the Cascade-Correlation, Tower and Pyramid networks. Extensive comparative results are carried out in a real off-line signature verification problem, indicating that all the constructive networks investigated can be efficiently applied in difficult real world pattern verification problems.

1. Introdução

Um problema fundamental em reconhecimentos de padrões é o da verificação da autenticidade de um dado padrão. Exemplos de tais problemas podem ser facilmente observados no mundo real, onde máquinas são cada vez mais empregadas no controle de acesso de pessoas com base na voz, face e impressões digitais [6]. Outro importante exemplo, objeto deste trabalho, envolve padrões como manuscritos humanos, mais especificamente assinaturas, que devem não apenas serem classificadas como também verificadas, no sentido de evitar imitações e fraldes. Um sistema de verificação decide sobre a identidade de um autor através de um processo de comparação de um padrão para uma classe, uma vez que se sabe de antemão a classe a qual a assinatura supostamente pertence, e deseja-se verificar sua autenticidade.

Assinaturas são um caso particular de manuscritos contendo caracteres especiais ou distorcidos e desenhos. É natural não existir homogeneidade quanto ao tamanho e distribuição dos caracteres, possuindo, em muitos casos, uma semântica ininteligível a seres humanos. Todavia o principal interesse reside no fato de que diversas características próprias dos autores são consciente e inconscientemente colocadas no papel, quando estes assinam, tornando possível uma posterior identificação através do processamento destas características singulares. Um sistema para reconhecimento de assinaturas requer uma preocupação com cinco etapas: (i) a aquisição dos dados, onde a

imagem da assinatura é produzida através de câmeras, *scanners* ou digitalizadores, e a assinatura ou texto aparece como uma imagem bidimensional $F(x, y)$ adquirida através destes meios ópticos; (ii) o pré-processamento, onde os dados brutos são pré-processados a fim de se filtrar os sinais significativos ou imagens, validando a aquisição, (iii) a extração de características, onde são selecionadas as propriedades mais relevantes que representam a assinatura, (iv) o processo de comparação, onde um conjunto de referência de assinaturas é gerado para se realizar o processo comparativo e (v) o processo de decisão, onde é avaliada a saída do processo de comparação, determinando se assinatura deverá ser considerada como pertencente ou não à determinada classe. Para tal decisão, respeita-se um certo limiar.

O processo de verificação de assinaturas não pode ser considerado uma tarefa trivial de reconhecimento de padrões. Este problema pode ser apresentado como sendo particionado em duas classes: a classe referente às assinaturas verdadeiras de um autor, e a classe referente às assinaturas falsificadas. Assinar não é um processo determinístico que sempre produz características precisas e idênticas após muitas tentativas. Pode-se observar uma vasta variabilidade em assinaturas, conforme o país, a idade, os hábitos, as condições físico-psíquicas, entre outros aspectos. Esta variabilidade pode ser enquadrada em dois principais tipos: a variabilidade intra-classe (variabilidade observada dentro de uma classe de assinaturas genuínas de um autor) e a variabilidade inter-classe (diferenças existentes entre as classes de assinaturas genuínas obtidas a partir de dois autores distintos).

O desempenho de um sistema de verificação é geralmente avaliado através da representação de dois tipos de erros: taxa de falsa rejeição e taxa de falsa aceitação, as quais variam de acordo com um determinado limiar de aceitação/rejeição. A situação ideal ocorrerá, se as melhores características forem selecionadas no sentido de separar completamente as assinaturas verdadeiras das falsas. É importante a inclusão de diferentes tipos de falsificações para validar o sistema, tais como:

- Falsificações simples: o falsificador não faz esforço algum para simular uma assinatura genuína;

- Falsificações aleatórias: o falsificador utiliza sua própria assinatura ao invés da assinatura a ser testada;
- Falsificações habilidosas: o falsificador tenta simular uma assinatura genuína, chegando o mais próximo possível de seu traçado original.

Várias técnicas têm sido utilizadas na realização de verificação de assinaturas. Neste trabalho, são investigadas, em particular, redes neurais artificiais baseadas no modelo autoassociativo [4, 6, 7]. A grande vantagem de redes neurais em relação às diversas técnicas é o fato delas serem capazes de aprender para realizar separação de classes. Redes Neurais têm sido empregadas tanto na verificação dinâmica de assinaturas como na verificação estática. Observa-se, na literatura, que o modelo de redes neurais mais comumente utilizado na verificação de assinaturas é a rede *perceptron* multicamadas (MLP) juntamente com o algoritmo de aprendizagem *backpropagation*. Neste trabalho, investiga-se a tarefa de verificação de assinaturas, utilizando-se as arquiteturas construtivas *Cascade-Correlation* [1], *Tower* e *Pyramid* [2, 3] sob a abordagem autoassociativa e realizando uma análise comparativa entre os resultados obtidos entre tais redes e a rede MLP também sob a abordagem autoassociativa. As redes neurais construtivas, ao contrário das redes de topologia fixa, constroem sua própria estrutura incluindo unidades e/ou camadas intermediárias, conforme a demanda do processo de treinamento. O mecanismo de inserção de unidades e o algoritmo de treinamento utilizado para treinar as redes são os parâmetros principais na distinção entre os vários modelos construtivos.

Nas seções seguintes, são descritos os modelos autoassociativos utilizados, a base de dados de assinaturas adotada nos experimentos, o procedimento experimental e os resultados alcançados.

2. Modelos Autoassociativos

Os modelos de redes *feedforward* que atuam como classificadores (classificação de 1 para N) não são adequados para tarefas de verificação de padrões. Há um argumento claro para tal afirmação: a discriminação de classes em redes *feedforward* é realizada por superfícies de separação no espaço de padrões as quais são geradas pelo algoritmo de aprendizagem sob o único objetivo de discriminar os dados exemplos [5]. As superfícies de separação resultantes são geradas de forma a particionar os dados de entrada propriamente, no entanto tais superfícies não são necessariamente fechadas, ou seja, elas não cobrem os exemplos capturando suas distribuições de probabilidade. O modelo usado nos experimentos realizados neste trabalho resolve as inconveniências advindas das superfícies de separação abertas. É baseado em redes *feedforward* que agem como autoassociadores, onde as saídas são forçadas a reproduzirem as entradas. Os

critérios de rejeição estão relacionados à maneira com que a entrada é aproximada pela saída. Resultados teóricos recentes oferecem suporte direto a tal escolha: os autoassociadores geram superfícies de separação fechadas cobrindo os dados de treinamento [4]. A verificação é estabelecida observando-se a distância entre os vetores de entrada e saída, através da equação do erro médio quadrado que é comparado a um dado limiar de verificação

$$E = 1/2 \sum (I_i - O_i)^2 < L \quad (1)$$

onde I_i e O_i são os padrões de entrada e saída respectivamente.

Na literatura, os autoassociadores são sugeridos, principalmente, para problemas de compressão de imagens. Neste caso, a compressão é realizada na camada escondida, sempre contendo um número menor de unidades que as demais camadas, onde a informação pode ser subsequenteemente reproduzida pela computação realizada na última camada. O modelo explora a propriedade dos autoassociadores de produzir superfícies de separação fechadas, devido ao fato das camadas intermediárias serem definidas com número menor de unidades. Estas regiões são obtidas através do treinamento da rede apenas com exemplos positivos. Desta forma, o autoassociador realiza compressão dos dados de entrada, mantendo a informação necessária para representar os dados relevantes. Quanto menor o número de elementos na camada intermediária, maior a probabilidade dos padrões serem mapeados para representações cada vez mais próximas nesta camada, visto que há um número reduzido de unidades respondendo para todo o conjunto de padrões de entrada [5].

O modelo baseado em autoassociadores tem sido aplicado com sucesso em outras tarefas de verificação de padrões, tais como verificação de locutor [6] e verificação de papel moeda [7]. Quatro redes autoassociativas foram utilizadas nos experimentos realizados neste trabalho: a arquitetura *MLP* autoassociativa convencional, a arquitetura *Cascade-Correlation* (*Cascor*) autoassociativa e as arquiteturas *Tower-autoassociativa* e *Pyramid-autoassociativa* como ilustram as Figuras 1 e 2.

A arquitetura *Cascor* é um modelo de rede *feedforward* em que novas unidades de processamento são inseridas uma a uma nas camadas intermediárias, conforme a demanda do processo de aprendizagem [1]. A configuração inicial apresenta apenas as camadas de entrada e saída, neste caso, ambas contendo o mesmo número de unidades, e a topologia da rede vai sendo formada durante o treinamento, ao contrário das redes cujas topologias são fixadas *a priori*.

A rede *Tower* constrói uma torre de *TLUs* (*Threshold Logic Units*) [2]. A extensão do algoritmo *Tower* para tratar de múltiplas classes é obtida pela simples inclusão de M unidades em cada camada adicionada à torre, onde M representa o número de classes. Cada unidade na

camada mais recentemente adicionada, a qual serve como camada de saída da rede, recebe entradas dos $N + 1$ componentes de entrada como também das M unidades na camada precedente [3].

A rede *Pyramid* constrói uma rede de maneira similar à rede *Tower*, exceto pelo fato de cada nova unidade adicionada receber entradas tanto dos N componentes do vetor de entradas como também de todas as unidades em cada uma das camadas previamente adicionadas à rede. A unidade mais recentemente incluída torna-se a saída da rede [2]. Como no caso da rede *Tower*, a extensão da rede *Pyramid* para tratar de múltiplas categorias funciona da seguinte forma: cada nova camada adicionada contém M unidades, onde M é o número de classes. Cada unidade recebe as $N + 1$ entradas bem como as saídas de cada unidade das camadas previamente inseridas na rede [3].

Estes dois modelos foram modificados durante este trabalho para permitir que suas estruturas tipicamente classificadoras fossem adaptadas ao paradigma autoassociativo e, conseqüentemente, pudessem ser empregadas em tarefas de verificação.

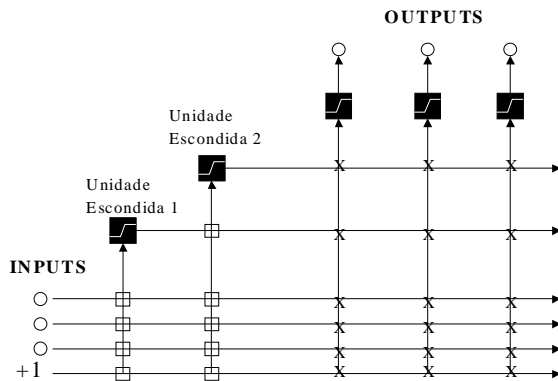


Figura 1: *Cascor* autoassociativa

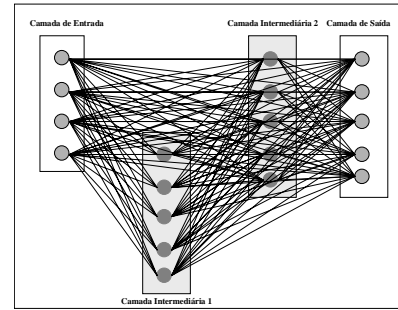
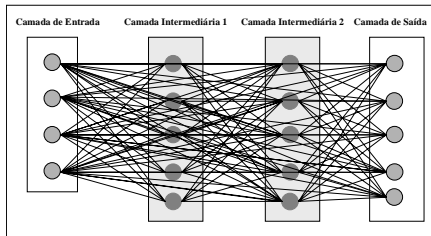


Figura2: *Tower*- autoassociativa e *Pyramid*-autoassociativa

3. Extração de Características

Um problema fundamental em reconhecimento de padrões é a definição das características dos dados as quais são realmente importantes para os processos posteriores de classificação e verificação de padrões, ou seja, a chamada extração de características. O principal objetivo desta seleção é transformar amostras de entrada em um novo espaço (espaço de características), onde a informação acerca das amostras é retida, mas a ordem de dimensão é reduzida [9]. A extração de características torna a tarefa de reconhecimento de padrões muito mais simples e fácil.

Apesar de em muitas aplicações os dados não sofrerem quaisquer etapas de extração de informações, no caso da aplicação de verificação de assinaturas em questão, esta etapa é essencial, pois o número de exemplos em problemas de verificação é geralmente reduzido. Com a extração de características, há uma redução na dimensão da rede, o número de parâmetros livres a serem estimados é menor, possibilitando a calibragem da rede com um número menor de dados de treinamento. A base de dados utilizada neste trabalho é formada por imagens de assinaturas digitalizadas, cujas características foram extraídas através de duas técnicas muito conhecidas: Técnica de Momentos [10] e PCA (Análise dos Componentes Principais) [9]. Neste artigo, serão apresentados apenas os resultados relativos aos modelos cujas características foram extraídas através de momentos.

3.1 Técnica de Momentos

A partir de uma imagem pré-processada, são produzidas três outras imagens: esqueleto, borda e regiões de pressão. Um total de 12 características numéricas são consideradas. A partir do esqueleto, são extraídos 6 valores de momentos, e o número de componentes de uma assinatura. A partir da imagem de borda, calculam-se as inclinações globais de uma assinatura, e, a partir das regiões de pressão, são extraídos um limiar de pontos de pressão na imagem e um fator de pressão, conforme ilustrado na Figura 3.

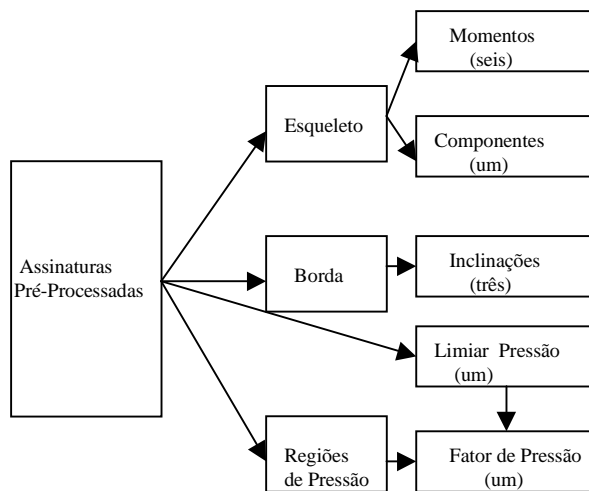


Figura 3: Extração das características que formam a base de dados

4. A Base de Dados

A base de dados de assinaturas utilizada neste trabalho encontra-se disponibilizada no Departamento de Informática da UFPE. Tais assinaturas foram colhidas de voluntários e digitalizadas no próprio Departamento de Informática por Gomes [8], durante a preparação de sua dissertação de Mestrado.

Os objetos da base de dados são imagens digitalizadas de assinaturas juntamente com as características de interesse extraídas destas imagens. Estas características foram extraídas utilizando-se a técnica de momentos. A fim de permitir a avaliação de sistemas de verificação, além das assinaturas genuínas de cada escritor, são também incluídas algumas falsificações tanto habilidosas como aleatórias para estas assinaturas. As falsificações aleatórias foram colhidas de voluntários que criaram assinaturas alheias sem nunca terem visto as originais, enquanto as assinaturas falsas habilidosas foram colhidas através de voluntários que se propunham a observar as assinaturas genuínas de cada classe e tentar reproduzi-las o mais próximo da original. Nas Figuras 4 e 5, observam-se exemplos de duas classes.

Obviamente, a construção de um sistema verificador torna-se bem mais complexa, quando os objetos a serem verificados são semelhantes aos objetos originais. A robustez do sistema pode ser medida, por conseguinte, observando-se seu comportamento em relação aos exemplos simulados. Durante os experimentos, um subconjunto das assinaturas verdadeiras de um dado autor (classe) é usado como conjunto de referência para este autor, formando o conjunto de treinamento, um outro subconjunto igual ao primeiro forma o conjunto de validação, e as assinaturas verdadeiras restantes junto com as falsas constituem o conjunto de teste.

Figura 4 (a): Assinatura genuína – classe 20

Figura 4 (b): Assinatura falsa habilidosa – classe 20

Figura 4 (c): Assinatura falsa aleatória – classe 20

5. Os Modelos *Tower-autoassociativo* e *Pyramid-autoassociativo*

Através dos resultados preliminares obtidos com a rede *Tower* e *Pyramid*, foi demonstrado que tais redes, quando usadas em sua forma original, não são apropriadas aos mecanismos de verificação de padrões. No mecanismo natural de funcionamento destas redes, as novas camadas que são inseridas (com a demanda do processo de treinamento) possuem o mesmo número de unidades da camada de saída original, contrariando o princípio da autoassociação. Como já mencionado, na abordagem autoassociativa as camadas intermediárias possuem um pequeno número de unidade de processamento para que haja uma espécie de filtragem das informações mais relevantes dos exemplos de treinamento positivos apresentados à rede. Para contornar tal problema, foi proposta uma modificação no mecanismo de inserção de camadas, no qual cada nova camada adicionada passa a não mais possuir o mesmo número de unidades que a camada de saída [11]. Nesta solução, o número de unidades das camadas intermediárias que venham a ser inseridas é proporcionalmente menor que o número de unidades da camada de saída. Com esta mudança, os algoritmos *Tower* e *Pyramid* passam a funcionar também sob a abordagem da autoassociação – *Tower-autoassociativo* e *Pyramid-autoassociativo*.

Figura 5 (a): Assinatura genuína – classe 03

Figura 5 (b): Assinatura falsa habilidosa – classe 03

Figura 5 (c): Assinatura falsa simples – classe 03

6. Descrição dos Experimentos

Foram realizados experimentos com 21 classes de assinaturas. A taxa de erro adotada tanto para a rede *MLP* quanto para as redes *Cascor*, *Tower-autoassociativa* e *Pyramid autoassociativa* foi de 0.01. O número máximo de épocas para as redes *MLP* e *Cascor* foi igual a 200, enquanto o das outras duas redes foi igual a 1500. O número máximo de unidades candidatas inseridas na rede *Cascor* foi 8, o número máximo de camadas inseridas nas redes *Tower-autoassociativa* e *Pyramid-autoassociativa* foi 5, e a taxa de *momentum* nas redes *MLP* foi igual a 0.6. O número de unidades intermediárias nestas duas redes foi definido como sendo 50% do número de unidades da camada de saída das redes. Estes parâmetros foram considerados adequados através da realização de experimentos preliminares. O limiar utilizado como critério de verificação consiste em um valor definido como *maxerr*, ou erro máximo obtido com os padrões de validação de cada classe. Este critério foi adotado de modo a tornar a verificação um processo mais robusto. No caso das redes *MLP*, o número de unidades na camada intermediária variou entre 5 e 10 unidades, no entanto são apresentados apenas os resultados com 10 unidades intermediárias.

As Tabelas 1-7 apresentam os resultados de desempenho das redes com 7 das 21 classes, em porcentagem. A taxa de reconhecimento de assinaturas genuínas é indicada por I, enquanto a taxa rejeição para assinaturas falsas habilidosas é indicada por II e para assinaturas falsas simples por III.

Tabela 1: Resultados obtidos relativos à classe 0

Algoritmo	I	II	III
<i>MLP</i>	60	100	100
<i>Cascor</i>	70	100	100
<i>Tower-autoassociativa</i>	100	50	100
<i>Pyramid-autoassociativa</i>	100	75	100

Tabela 2: Resultados obtidos relativos à classe 1

Algoritmo	I	II	III
<i>MLP</i>	60	75	100
<i>Cascor</i>	100	75	100
<i>Tower-autoassociativa</i>	90	75	75

<i>Pyramid-autoassociativa</i>	100	50	75
--------------------------------	-----	----	----

Tabela 3: Resultados obtidos relativos à classe 6

Algoritmo	I	II	III
<i>MLP</i>	80	75	100
<i>Cascor</i>	80	50	100
<i>Tower-autoassociativa</i>	90	75	75
<i>Pyramid-autoassociativa</i>	90	75	75

Tabela 4: Resultados obtidos relativos à classe 8

Algoritmo	I	II	III
<i>MLP</i>	60	75	100
<i>Cascor</i>	100	75	75
<i>Tower-autoassociativa</i>	100	75	75
<i>Pyramid-autoassociativa</i>	100	75	100

Tabela 5: Resultados obtidos relativos à classe 10

Algoritmo	I	II	III
<i>MLP</i>	80	50	100
<i>Cascor</i>	90	100	100
<i>Tower-autoassociativa</i>	90	100	100
<i>Pyramid-autoassociativa</i>	100	75	100

Tabela 6: Resultados obtidos relativos à classe 15

Algoritmo	I	II	III
<i>MLP</i>	50	75	75
<i>Cascor</i>	90	50	100
<i>Tower-autoassociativa</i>	80	75	75
<i>Pyramid-autoassociativa</i>	90	75	75

Tabela 7: Resultados obtidos relativos à classe 20

Algoritmo	I	II	III
<i>MLP</i>	60	75	100
<i>Cascor</i>	70	75	100
<i>Tower-autoassociativa</i>	90	75	100
<i>Pyramid-autoassociativo</i>	90	75	75

Observa-se, nas tabelas, que os resultados obtidos com os modelos de redes construtivas apresentam, para todas as classes apresentadas, índices de reconhecimento para padrões verdadeiros superiores aos apresentados pela rede de topologia fixa tradicional *perceptron* multicamadas (MLP). Tais índices obtidos com as redes *Tower* e *Pyramid* adaptadas ao modelo autoassociativo são ainda superiores aos obtidos com a rede construtiva *Cascor*. No entanto, para os erros do tipo II e III, ou seja, rejeição de padrões falsos habilidosos e aleatórios, as redes apresentam índices bem próximos. Para os casos de assinaturas falsas habilidosas, os índices obtidos pelas redes não são sempre satisfatórios (em alguns casos, rejeita-se apenas 50% das assinaturas falsas). Este fato pode ser explicado pela dificuldade inerente à construção de um sistema de verificação que rejeite este tipo de assinatura. Muitas vezes, até a análise humana não consegue fazer a distinção entre assinaturas genuínas e assinaturas falsas habilidosas.

7. Conclusões

Neste trabalho, foi realizada uma investigação das arquiteturas *Cascade-Correlation* autoassociativa, *Tower-autoassociativa* e *Pyramid-autoassociativa* do ponto de vista prático, estabelecendo-se uma avaliação experimental destes modelos em uma aplicação real de verificação de padrões. Para efeito de comparação, também foi investigada a arquitetura *feedforward* convencional MLP, também sob a abordagem da autoassociação. Os resultados mostram um melhor desempenho das arquiteturas construtivas consideradas quanto ao reconhecimento dos padrões genuínos, sendo os índices obtidos com as redes *Tower* e *Pyramid* modificadas superiores aos obtidos com a rede *Cascor*. Em relação às taxas de rejeição para as assinaturas falsas habilidosas e simples, os valores por todos os modelos foram bem próximos. Dos resultados obtidos, conclui-se que os modelos de redes construtivas podem ser considerados como uma alternativa à arquitetura tradicional *perceptron* multicamadas (MLP) em tarefas de verificação de padrões.

Referências

- [1] Fahlman, S.E., Lebiere, C., "The cascade-correlation learning architecture", *Advances in Neural Information Processing Systems II*, Morgan Kaufmann 1990.
- [2] GALLANT, S. *Neural Network Learning and Expert Systems*. Cambridge, MA: MIT Press, 1993.
- [3] PAREKH, R., YANG, J. E HONAVAR, V. *Constructive Neural Network Learning Algorithms for Multi-Category real-Valued Pattern Classification*. Technical Report TR # 97-06, Artificial Intelligence Research Group, Department of Computer Science, Iowa State University, 1997.
- [4] BIANCHINI, M., FRASCONI, P. E GORI, M. *Learning in Multilayered Networks Used as Autoassociators*. IEEE

- Transactions on Neural Networks*, vol. 6, pp.512-515, 1995.
- [5] VASCONCELOS, G. *Why is Neural Network Validation Important?* Anais do III Simpósio Brasileiro de Redes Neurais, pp. 159-166, 1996.
- [6] GORI, M., LASTRUCCI, L. E SODA, G. *Autoassociator-based Models for Speaker Verification*. *Pattern Recognition Letters*, vol17, 241-250, 1996.
- [7] FROSINI, A., GORI, M. E PRIAMI, P. *A Neural Network-Based Model for Paper Currency Recognition and Verification*. *IEEE Transactions on Neural Networks*, 1997.
- [8] GOMES, H. *Investigação de Técnicas Automáticas para Reconhecimento Off-line de Assinaturas*. Dissertação de Mestrado, Departamento de Informática, Universidade Federal de Pernambuco, 1995.
- [9] BISHOP, M. *Neural Networks for Pattern Recognition*. Clarendon Press Oxford, 1995.
- [10] PROKOP, R. E REEVES, A. *A Survey of Moment-Based Techniques for Unoccluded Object Representation and Recognition*. School of Electrical Engineering, Cornell University, Ithaca, New York, 14853, 1992.
- [11] RIBEIRO, J. *Modelos de Redes Neurais Construtivas para Classificação e Verificação de Padrões*. Dissertação de Mestrado, Departamento de Informática, UFPE, concluída em 23 de abril de 1999.