

UNIVERSIDADE CATÓLICA DE PELOTAS  
ESCOLA DE INFORMÁTICA  
CURSO CIÊNCIA DA COMPUTAÇÃO

**Proposta de um Padrão Manuscrito para Reconhecimento  
Automático dos Símbolos do Sistema SignWriting (SW)**

por

Fabiana Zaffalon Ferreira Rocha

Projeto de Graduação submetido como requisito  
parcial à obtenção do grau de Bacharel em Ciência  
da Computação.

Orientador: Prof. Antônio Carlos da Rocha Costa

Co-Orientador: Prof<sup>ª</sup>. Graçaliz Pereira Dimuro

Colaboradores: Alexsandro Teixeira Gonçalez

Diogo Madeira

Pelotas, julho de 2003.

**Dedicado a meu marido**

João Batista Rocha

**meu filho**

Gabriel Zaffalon Ferreira Rocha

**meus pais**

Florenil Maia Ferreira

Leci Zaffalon Ferreira

**e minha irmã**

Roberta Zaffalon Ferreira

## AGRADECIMENTO

Ao meu filho por simplesmente encher minha vida de felicidade.

Ao meu marido que com amor e carinho soube transformar as dificuldades, os medos e os sofrimentos em êxito, coragem e vitória; por sempre caminhar ao meu lado me apoiando nas minhas decisões. Esta conquista também é tua.

À minha mãe que além do dom da vida me cobriu de amor. Que na hora da insegurança, não esmoreceu em relação às suas expectativas, e com amor, carinho e dedicação lutou por mim. Obrigada pelo apoio, pela certeza da vitória, pela força na hora do desânimo e pela firmeza diante dos obstáculos. Teu impulso me deu coragem e determinação na busca de meus ideais. As alegrias de hoje são tuas também, pois teu estímulo e amor foram as armas que me levaram a esta conquista.

Ao meu pai pelos exemplos de dedicação e entusiasmo pelo trabalho, honestidade, valores éticos e outros tantos exemplos que me ajudam a viver com garra e dignidade.

À minha irmã pela amizade, pelo amor e, acima de tudo, por sempre acreditar em mim me dando todo o apoio necessário para que eu pudesse atingir meus objetivos.

Ao Alessandro Gonzalez, pelo carinho, pela amizade, pelo companheirismo durante esta jornada e pela forte contribuição para a realização deste trabalho.

Ao Diogo Madeira, pela amizade, carinho e, por estar presente nos momentos em que mais precisei da tua ajuda. Sem tua contribuição este trabalho não teria sido realizado.

Ao orientador e amigo professor Rocha pelo apoio, experiência e incentivo na elaboração deste trabalho, como também ao professor Cava que acima de tudo contribuiu para que minha formação fosse além do conhecimento: um aprendizado de vida.

Agradeço também a todas as pessoas que estiveram ao meu lado, aos meus amigos, aos meus colegas de graduação e de trabalho, aos meus professores e todas as pessoas que conviveram comigo durante esta jornada.

Obrigada a todos vocês!

**A VIDA**

*“A vida são deveres,  
que nós trouxemos para fazer em casa.  
Quando se vê, já são seis horas!  
Quando se vê, já é sexta-feira...  
Quando se vê, já terminou o ano...  
Quando se vê, passaram-se 50 anos!  
Agora, é tarde demais para ser reprovado...  
Se me fosse dado, um dia,  
outra oportunidade, eu nem olhava o relógio.  
Seguiria sempre em frente e iria jogando,  
pelo caminho, a casca dourada e inútil das horas...  
Dessa forma, eu digo:  
não deixe de fazer algo que gosta devido à falta de tempo.  
A única falta que terá, será desse tempo que  
infelizmente não voltará mais.”*

*Mário Quintana*

**SUMÁRIO**

**LISTA DE FIGURAS**

## **LISTA DE ABREVIATURAS**

**RNA** Redes Neurais Artificiais

**SW** Sistema SignWrintig

## RESUMO

Este trabalho consiste no desenvolvimento de um padrão manuscrito para o reconhecimento automático dos símbolos do Sistema SignWriting. O reconhecimento do padrão é feito por uma rede Neural Backpropagation utilizando o padrão alfabético proposto como padrão de escrita.

**Palavras-chaves:** Sistema Signwriting, Rede Neural Backpropagation, Alfabeto “Graffiti”, Símbolos.

**ABSTRACT**

This work consists of the development of a standard manuscript for the automatic recognition of the symbols of the SignWriting System. The recognition of the standard is made by a Neural net Backpropagation having used the considered alphabetical standard as writing standard.

**Keywords:** Signwriting System, Neural Net Backpropagation, Alphabet “Graffiti”, Symbols.

## 1 INTRODUÇÃO

Pessoas surdas possuem qualidades, dificuldades e linguagem própria, a qual não é levada em consideração quando se trata da escrita dos símbolos utilizados na comunicação entre estas pessoas. Com isto elas são forçadas a se adaptarem a uma linguagem que não é natural delas, e que, a qual, possuem muitas dificuldades de entender e utilizar.

A principal motivação deste trabalho é criar um padrão manuscrito para os símbolos do sistema SW utilizados por pessoas surdas, de modo a simplificar, permitir o reconhecimento automático desses símbolos.

O padrão manuscrito criado neste trabalho está baseado no alfabeto “Graffiti”, que também é utilizado no computador de bolso Palm. É um alfabeto simplificado e muito parecido com os caracteres do alfabeto regular, porém a ordem da escrita influencia no reconhecimento dos caracteres.

## 2 SISTEMA SIGNWRITING

Pessoas surdas utilizam os sinais para comunicação, porém, quando precisam recorrer à escrita, elas precisam aprender a representação escrita da língua sonora utilizada pelas pessoas falantes. Desta forma, Valérie Sutton do Deaf Action Commite, da Califórnia, USA desenvolveu uma notação gráfica para a Língua de Sinais, o sistema SignWriting.[5]. Um dos motivos do SignWriting ter sido criado é a dificuldade que pessoas surdas tem em ler textos na forma oral.

O SW expressa os movimentos, as formas das mãos, as marcas não manuais e os pontos de articulação, ou seja, mostra a forma da Língua de Sinais

O SignWriting pode ser comparado com um alfabeto, como um conjunto de símbolos visuais que podem descrever os movimentos realizados por qualquer língua de sinal no mundo. Assim, seus símbolos são internacionais e podem ser utilizados para descrever qualquer língua de sinais.

Da mesma forma que um alfabeto, o SW não é uma nova linguagem, são ferramentas usadas para escrever línguas que já existem pelo mundo.

O sistema de escrita SignWriting se organiza de forma similar à forma de escrita oral. Ao invés de possuir letras, existem os símbolos que representam as letras e os movimentos. A escrita da língua dos sinais não é feita de desenhos, mas de símbolos. Se diz “eu escrevi o sinal carro” e não eu “desenhei o sinal carro”. Um conjunto destes símbolos representa um sinal.

O SW como escrita de língua de sinais é o mais rápido, mais fácil e mais claro do que os desenhos de sinais que normalmente se utilizam em aulas de surdos. Em muitas escolas e classes de surdos se costuma desenhar os sinais quando se quer fixar vocabulário, é um procedimento trabalhoso e demorado e nem sempre os desenhos mostram claramente todos os elementos que compõem o sinal. Também na escrita de sinais existe a forma “bastão” (imprensa que é a mais utilizada) e a forma cursiva (“shorthand” ou estenográfica). Exemplos desta última podem ser vistos em <http://atlas.ucpel.tche.br/~fabinha/Sinais.html> .

O SW é dividido em 10 categorias:

- Categoria 1 - Mãos
- Categoria 2 - Contato das mãos
- Categoria 3 - Faces
- Categoria 4 - Movimentos do corpo e da cabeça
- Categoria 5 - Ombro
- Categoria 6 - Membros
- Categoria 7 - Inclinação da cabeça
- Categoria 8 - Localização
- Categoria 9 - Movimento de dinâmicas
- Categoria 10 - Pontuação

Neste trabalho, em função do tempo disponível, desenvolveremos a escrita apenas das categorias 1, 2, 3 e 9, pois, segundo o estudante surdo que colaborou com este trabalho, Diogo Madeira, estes são os mais utilizados.

Os símbolos serão codificados, como mostra a figura 1.

0101001 onde: 

01	01	001
Categoria	Grupo	Símbolo

**Figura 1** Representação do código de cada símbolo

**Tabela 1** Demonstração dos símbolos em libras e do Sistema SW com seu respectivo padrão manuscrito.

Símbolo (Libras)	Símbolo (SW)	Padrão Manuscrito
		
		
		
		
		
		

### 3 ALFABETO "GRAFFITI"

O problema da entrada de dados num sistema digital sem uso do teclado é um assunto cada vez mais relevante à medida que o tamanho dos sistemas diminui. A potência de cálculo das máquinas tendo vindo a aumentar, várias técnicas foram investigadas para a entrada de dados, mas, até hoje não se pode considerar o problema como resolvido. Duas abordagens podem ser diferenciadas: a entrada de dados através do reconhecimento da fala e através do reconhecimento da escrita [8].

Para tornar mais eficiente o reconhecimento dos caracteres, um alfabeto simplificado é usado, chamado alfabeto "Graffiti", que é usado no computador de bolso Palm [14]. Em vez de usar um teclado, o utilizador escreve diretamente no ecrã.

A maioria dos textos podem ser incorporados rapidamente, mas requer alguns minutos da prática. A escrita do "Graffiti" inclui alguns caracteres, e assemelham-se às letras do alfabeto regular, o que torna a escrita fácil de aprender, como mostra a figura 2.



**Figura 2 Exemplos do Alfabeto "Graffiti"**

O ponto pesado em cada forma, na figura 3, mostra onde começar o curso. A figura 4 mostra que determinados caracteres têm formas similares, mas pontos diferentes do começo e da extremidade. A maioria dos caracteres têm somente um único curso. Quando a caneta é retirada da área da escrita, os caracteres são reconhecidos e indicados.



**Figura 3 Indicação do início do curso**

	Back Space
	Space
	Letra N
	Acento Til

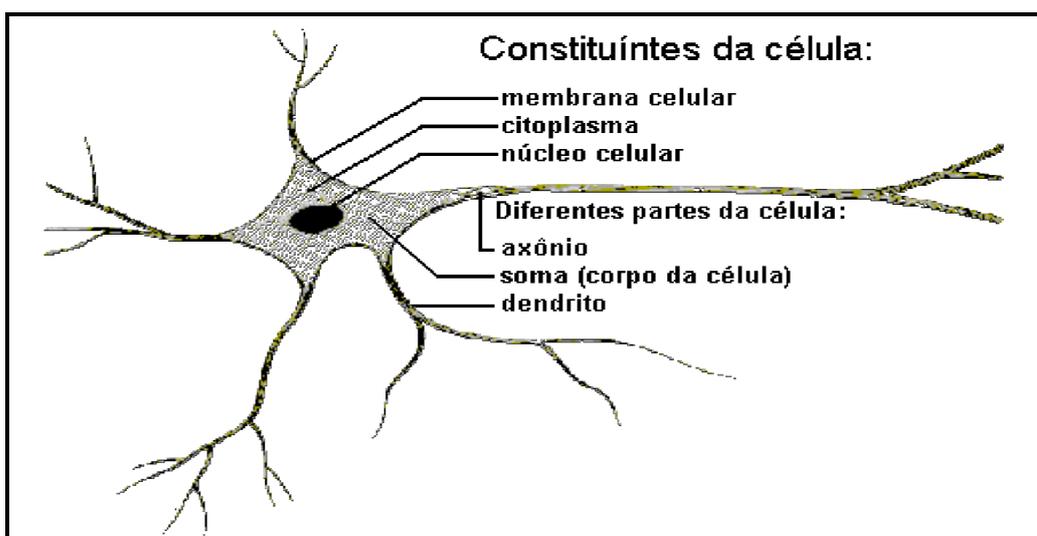
**Figura 4 Caracteres com formas similares**

#### 4 REDES NEURAIAS

Redes Neurais Artificiais (RNAs) [9] são técnicas computacionais que apresentam um modelo matemático inspirado na estrutura neural de organismos inteligentes e que adquirem conhecimento através da experiência. As Redes Neurais Artificiais surgiram como uma tentativa de reproduzir o funcionamento do cérebro humano, e assim se desenvolver máquinas capazes de realizar muitas funções que antes só eram possíveis de serem realizadas através da intervenção humana.

As RNAs foram originalmente desenvolvidas baseadas nos estudos realizados sobre a forma como o conhecimento é armazenado no cérebro humano e a forma como ocorre o aprendizado. Nestes estudos foi constatado que nos seres humanos o conhecimento é armazenado nas ligações que os neurônios realizam uns com os outros, chamadas de sinapse, e a medida que o aprendizado ocorre mais ligações vão se formando e se fortalecendo entre estes neurônios.

Para que seja possível compreender o funcionamento das Redes Neurais Artificiais, é necessário que se compreenda um pouco do funcionamento do cérebro humano [10]. O núcleo do sistema nervoso é composto por milhões de células que são responsáveis pela geração, recepção e transmissão dos impulsos elétricos, conhecidas como neurônios. A figura 5 mostra o desenho de um neurônio biológico humano.



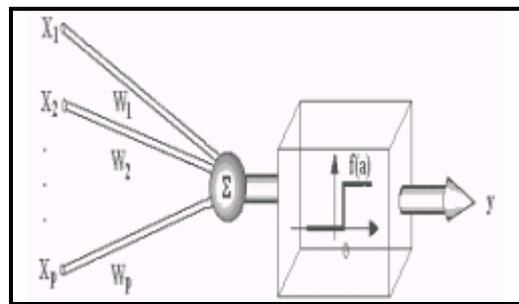
**Figura 5 Neurônio Biológico Humano**

Um neurônio biológico é formado por três partes principais, que são uma árvore

dendrítica, um corpo celular e um prolongamento fino e longo, chamado de axônio. Os dendritos são as entradas pelas quais o neurônio recebe os impulsos elétricos provenientes de outros neurônios, e o axônio é o meio por onde os sinais elétricos saem do neurônio em direção aos outros neurônios. A ligação existente entre dois neurônios, conhecida como sinapse, é que irá influenciar sobre a transmissão do sinal recebido, podendo atuar de forma a atenuar ou amplificar os impulsos elétricos recebidos.

Baseado no funcionamento do neurônio biológico, foi desenvolvido o neurônio formal, que é uma simplificação matemática que tenta reproduzir as principais características dos neurônios humanos referentes a forma em que se processa a aquisição de conhecimentos.

Assim como um neurônio biológico, o neurônio artificial (figura 6) é constituído de entradas, que recebem os sinais provenientes do exterior, uma função de ativação, que realiza a combinação dos valores obtidos nas entradas, e saídas, que transmitem os sinais recebidos e processados pelo neurônio para o exterior. Associado a cada entrada de um neurônio artificial existem pesos, que são valores numéricos que representam a força das conexões existentes entre os neurônios.



**Figura 6 Modelo de Neurônio Artificial**

As Redes Neurais possuem a capacidade de aprendizado a partir de exemplos e a capacidade de generalização, de forma que a partir dos exemplos analisados elas conseguem extrair as regras gerais que descrevem um problema e assim podem aplicar esta regra para a solução de novos exemplos não analisados anteriormente. Outras vantagens que as Redes Neurais apresentam em relação as demais técnicas de Inteligência Artificial são o fato de serem tolerantes a dados incorretos ou incompletos e também de poderem lidar tanto com informações quantitativas quanto com informações qualitativas.

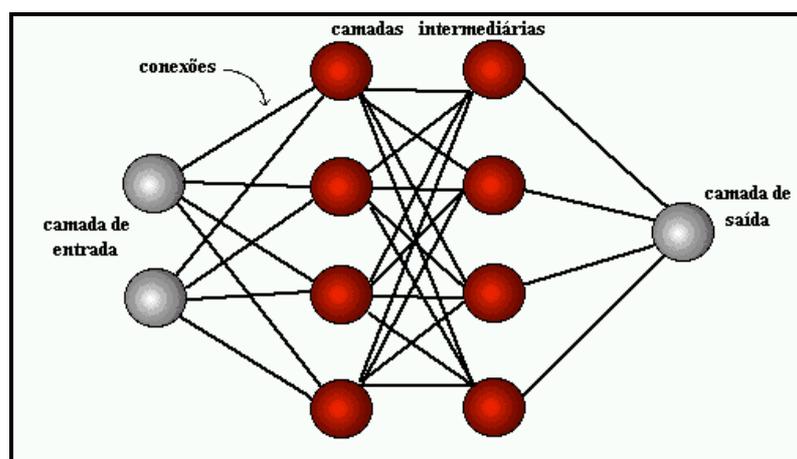
A propriedade mais importante das redes neurais é a *habilidade* de aprender de seu ambiente e com isso melhorar seu desempenho. Isso é feito através de um processo iterativo de ajustes aplicado a seus pesos, o treinamento. As redes neurais têm uma capacidade inata de *adaptar* seus pesos sinápticos a modificações do meio ambiente.

Denomina-se algoritmo de aprendizado a um conjunto de regras bem definidas para a solução de um problema de aprendizado. Existem muitos tipos de algoritmos de aprendizado específicos para determinados modelos de redes neurais, estes algoritmos diferem entre si principalmente pelo modo como os pesos são modificados.

A operação de uma unidade de processamento, proposta por McCulloch e Pitts em 1943, pode ser resumida da seguinte maneira:

- sinais são apresentados à entrada;
- cada sinal é multiplicado por um número, ou peso, que indica a sua influência na saída da unidade;
- é feita a soma ponderada dos sinais que produz um nível de atividade; se este nível de atividade exceder um certo limite a unidade produz uma determinada resposta de saída.

Usualmente as camadas são classificadas em três grupos, ilustrados na figura 7:



**Figura 7 Camadas das RNAs**

- **Camada de Entrada:** onde os padrões são apresentados à rede;

- **Camadas Intermediárias ou Escondidas:** onde é feita a maior parte do processamento, através das conexões ponderadas; podem ser consideradas como extratoras de características;
- **Camada de Saída:** onde o resultado final é concluído e apresentado.

Outro fator importante é a maneira pela qual uma rede neural se relaciona com o ambiente. Nesse contexto existem os seguintes paradigmas de aprendizado:

- **Aprendizado Supervisionado**, quando é utilizado um agente externo que indica à rede a resposta desejada para o padrão de entrada;
- **Aprendizado Não Supervisionado** (auto-organização), quando não existe uma agente externo indicando a resposta desejada para os padrões de entrada.

Em um sistema de rede neural, a informação pode parecer ter representação redundante, porém, o fato de que ela se encontre distribuída por todos os elementos da rede significa que mesmo que parte da rede seja destruída, a informação contida nesta parte ainda estará presente na rede, e poderá ser recuperada. Portanto, a redundância na representação de informações em uma rede neural, diferente de outros sistemas, transforma-se em uma vantagem, que torna o sistema tolerante à falhas. Os atributos de uma rede neural, tais como aprender através de exemplos, generalizações redundantes, e tolerância à falhas, proporcionam fortes incentivos para a escolha de redes neurais como uma escolha apropriada para aproximação para a modelagem de sistemas biológicos.

## 5 REDES NEURAS APLICADAS À AUTENTICAÇÃO DE ASSINATURAS

A autenticação de assinaturas [10] faz parte do reconhecimento de padrões, onde o padrão a ser reconhecido é a assinatura de determinado usuário, e as classes nas quais este padrão pode ser atribuído são as classes **Pertence** e **Não\_Pertence**, relativas a determinado usuário. Para a autenticação de assinaturas, é necessário que se identifiquem nos padrões das assinaturas quais os elementos que podem diferenciá-las em relação as assinaturas dos demais usuários, e também é necessário que se abstraiam as características que não são relevantes para o processo de classificação.

Diversas características presentes nas Redes Neurais Artificiais as tornam muito eficazes na autenticação de assinaturas, sendo que as mais importantes são:

- Aprendizado a partir de exemplos: basta submeter a base de dados de assinaturas a uma Rede Neural que a rede irá aprender quais as características que melhor identificam as assinaturas de determinado usuário, sem que seja necessário explicitar as regras de classificação manualmente;
- Generalização: as assinaturas não precisam ser exatamente iguais para serem reconhecidas como pertencendo a um determinado usuário, pois o sistema é capaz de aprender as regras gerais que definem as assinaturas e realizar a classificação a partir destas regras;
- Possibilidade de trabalhar com informações incorretas e/ou incompletas: mesmo que o usuário tenha assinado de forma diferente em uma ou mais assinaturas da base de dados, isto não afetará de forma significativa o desempenho do sistema;
- Segurança dos dados: como a Rede Neural só precisa do arquivo de pesos para fazer a classificação, a base de dados original pode ser descartada ou mantida a salvo em outro local após do treinamento, o que torna o sistema mais seguro em relação ao ataque por parte de *hackers*.

Para a coleta de assinaturas é necessário utilizar um dispositivo de hardware chamado *Tablet*. Este dispositivo é composto de duas partes, que são uma caneta especial e uma superfície sensível sobre a qual a caneta se desloca quando uma

assinatura é desenhada. O funcionamento de um *tablet* é similar ao funcionamento de um *mouse*, que envia as coordenadas em que a caneta se encontra a cada  $n$  milissegundos, de forma que a aplicação que faz a coleta das assinaturas precisa coletar e armazenar estas informações em tempo real para poder montar posteriormente a imagem da assinatura.

### 5.1 Procedimentos para a autenticação de assinaturas:

Os procedimentos usuais para autenticação de assinaturas [10] são os seguintes:

- *Montagem da assinatura:* o *tablet* disponibiliza apenas uma lista seqüencial dos pontos que foram coletados durante a assinatura. Portanto, a primeira tarefa é a montagem da assinatura a partir dos pontos coletados pelo *tablet*. O primeiro passo para a montagem da assinatura é desenhar o pontos coletados em um *bitmap*.
- *Ajuste da posição:* O ajuste de posição é uma alteração que se faz nas assinaturas de forma que elas fiquem todas na mesma posição na tela, o que facilita o processo de autenticação. O ajuste de posição pode ser feito de duas formas, pelo canto superior esquerdo ou pelo centro de massa da assinatura. O ajuste pelo canto superior esquerdo consiste em deslocar a assinatura de forma que as suas extremidades superior e esquerda coincidam com as extremidades superior e esquerda da tela. O ajuste de posição pelo centro de massa da assinatura altera a posição da assinatura de forma que o centro de massa da assinatura coincida com o centro da imagem. O centro de massa da assinatura é o local que corresponde a média das coordenadas de todos os pontos coletados da assinatura.
- *Ajuste de escala:* O ajuste de escala visa minimizar as variações de tamanho que podem ocorrer entre assinaturas de uma mesma pessoa. Ele é realizado através da alteração das coordenadas dos pontos coletados pelo *tablet* de forma que a assinatura sofra um incremento no tamanho, e este incremento é cessado quando um dos extremos da assinatura coincidir com um dos extremos da tela.
- *Tempo de duração da assinatura:* É o tempo que o usuário utilizou para a realização da assinatura. O cálculo é realizado diminuindo-se o instante final da assinatura pelo instante inicial, obtendo-se assim o tempo de duração da assinatura em termos de milissegundos. Este atributo é importante principalmente para a detecção de fraudes,

pois assinaturas falsificadas costumam levar muito mais tempo para serem desenhadas que as assinaturas originais.

- *Número de vezes que a caneta foi levantada:* Este atributo conta quantas vezes o usuário afastou a caneta do *tablet* durante a assinatura, excluindo-se os extremos. O usuário costuma levantar a caneta para separar nome e sobrenome ou para colocar acentos. Este atributo, em conjunto com outros, é muito útil diferenciar assinaturas de diferentes usuários, pois ele praticamente não varia entre assinaturas de um mesmo usuário.
- *Velocidade média da assinatura:* É a velocidade média de deslocamento da caneta sobre o *tablet*, calculada dividindo-se a distância total que a caneta percorreu durante o traçado pelo tempo de duração da assinatura. Este atributo é muito útil em processos de detecção de fraudes, pois assinaturas falsificadas costumam ser desenhadas muito mais lentamente que as assinaturas originais.
- *Velocidade máxima da assinatura:* Este atributo é similar a velocidade média, mas com a diferença que ao invés de se considerar a assinatura como um todo, o conjunto dos pontos amostrados é dividido em diversos subconjuntos e é calculada a velocidade média para cada subconjunto. A velocidade máxima será a maior velocidade encontrada nestes subconjuntos. Este atributo é útil para diferenciar assinaturas que possuam características especiais como um pequeno trecho muito mais rápido que o restante da assinatura, e em sistemas de autenticação de assinaturas, quanto mais informações estiverem disponíveis, melhores serão os resultados.
- *Amostragem seqüencial da assinatura:* Em assinaturas simples, onde o usuário apenas escreve seu nome, a trajetória que a caneta percorre é fácil de ser identificada quando a assinatura é inspecionada visualmente. Mas em assinaturas mais complexas, é difícil determinar com precisão qual a trajetória realizada pela caneta apenas com a inspeção visual. Em sistemas de autenticação de assinaturas, as informações relativas a trajetória da assinatura são muito úteis para o processo de classificação e de detecção de fraudes, pois se um falsário fizer uma assinatura muito semelhante a original, mas com uma trajetória diferente, a falsificação poderá ser detectada facilmente.

- *Simetria da assinatura:* Cada assinatura possui uma forma particular de se distribuir geograficamente em relação ao centro de massa, algumas sendo mais simétricas e outras sendo distribuídas mais para um lado do que para outro. Este atributo é mais importante para a identificação das assinaturas devido ao fato dele ter relação direta com o formato de uma assinatura, mas também acrescenta importantes informações no processo de autenticação de assinaturas.
- *Densidade da assinatura com informações de grade:* Este atributo acrescenta importantes informações relacionadas com a forma de uma assinatura. Ele é calculado da seguinte forma: divide-se a imagem da assinatura em diversas células de um determinado tamanho, como se fosse um tabuleiro de xadrez. Para cada célula, calcula-se o número de pontos da assinatura que se encontram dentro desta célula. Os valores encontrados dentro de cada célula são normalizados entre 0 e 1, sendo 0 para a célula que tiver o menor número de pontos e 1 para a célula que tiver o maior número de pontos. O tamanho das células é um importante fator a ser considerado, pois se as células forem muito grandes não será possível diferenciar todas as assinaturas, e se as células forem muito pequenas o sistema terá dificuldades para generalizar, pois como cada célula representa uma entrada na Rede Neural, o número de entradas necessário seria muito grande.
- *Interseções de linhas verticais em relação a assinatura:* A partir da assinatura desenhada em um *bitmap* são traçadas linhas verticais imaginárias cortando a assinatura em intervalos fixos em relação ao tamanho total da imagem. Para cada linha imaginária, são contadas quantas vezes esta linha intersecciona a assinatura.
- *Interseções de linhas horizontais em relação a assinatura:* Similar a interseção de linhas verticais, só que utilizando linhas horizontais. Estes dois atributos costumam ser utilizados em conjunto.
- *Número de vetores apontando para cada um dos pontos geográficos:* Uma assinatura é uma figura bastante complexa, com diversas curvaturas em sua trajetória. Para uma análise dos atributos geométricos de uma assinatura, um artifício que pode ser utilizado é a extração da forma estrutural da assinatura, que é uma simplificação da forma original. Para se obter a forma estrutural de uma assinatura, são calculados os pontos chave da assinatura, que são o início e o

término de cada segmento e os pontos onde houve troca de sentido do traçado em relação aos eixos  $x$  ou  $y$ . Depois de calculados os pontos chave, estes são interligados através de retas, formando a estrutura básica da assinatura. No cálculo dos pontos onde houve trocas de sentido do traçado, uma distância mínima entre cada ponto chave é estipulada para evitar que pequenas ondulações na trajetória da assinatura tornem a forma estrutural muito detalhada. Quanto maior for esta distância, mais simplificada será a forma estrutural da assinatura. As retas que compõe a forma estrutural de uma assinatura são uma coleção de vetores, que possuem comprimento, orientação e sentido. Um atributo que pode ser utilizado na autenticação de assinaturas é o número de vetores que apontam para cada ponto geográfico da assinatura.

- *Soma do comprimento dos vetores para cada ponto geográfico*: Similar ao atributo anterior, mas ao invés de se calcular o número de vetores que apontam para cada ponto geográfico, calculamos a soma do comprimento dos vetores que apontam para cada ponto geográfico. Os valores obtidos são normalizados entre 0 e 1 para que fiquem independentes da escala da assinatura. O diferencial introduzido por este atributo é a noção de grandeza, que indica para qual sentido a assinatura se desloca com mais intensidade, tendo assim uma relação direta com a trajetória da assinatura.

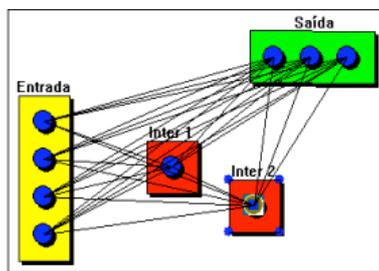
Para a autenticação de assinaturas, o modelo de Redes Neurais mais indicado é o *Cascade-Correlation*, devido as diversas vantagens que ele apresenta em relação aos demais modelos de Redes Neurais.

## 5.2 Correlação em Cascata

O algoritmo **Cascade Correlation** [15] utiliza uma técnica de aprendizado supervisionado para treinar RNAs. Esse algoritmo começa com uma rede mínima (apenas camadas de entrada e saída) e, durante o treinamento, insere novas unidades intermediárias, uma por vez, criando uma estrutura de múltiplas camadas.

A nova unidade é conectada à entrada, à saída e às unidades intermediárias que já fazem parte da rede. Quando uma nova unidade é inserida, congela-se os peso de suas entradas (conexões). Esse neurônio começa então a influenciar as operações da rede, sendo utilizado para detectar novas características no grupo de padrões.

A unidade a ser incluída na rede pode ser selecionada de um conjunto de candidatos organizados em uma camada. Essa camada é conectada à camada de entrada e às camadas intermediárias já existentes, mas não à camada de saída, pois não deve influir diretamente no resultado da rede. O critério de seleção do candidato é a correlação que a saída desse candidato tem com a saída da rede. Portanto, o peso de conexão entre os candidatos e as camadas de entrada e intermediárias deve ser definido de modo a maximizar a correlação entre o candidato e a camada de saída. Assim, o candidato que apresentar a maior correlação será inserido na rede como uma camada intermediária, ligando-se a todas as camadas da rede.



**Figura 8 Rede Cascade Correlation com duas unidades intermediárias inseridas**

As principais vantagens que o *Cascade-Correlation* [10] apresenta em relação aos demais modelos de Redes Neurais são:

- *Definição da arquitetura*: no *Cascade-Correlation*, não é necessário determinar o número de neurônios da camada oculta nem as interligações entre estes neurônios, pois o próprio algoritmo do *Cascade-Correlation* se encarrega de determinar a melhor topologia de rede possível para solucionar um determinado problema.
- *Menos parâmetros*: no *Cascade-Correlation*, o passo não precisa ser ajustado, pois o algoritmo altera o valor do passo durante o aprendizado com uma técnica chamada *Quickprop* de forma a acelerar ao máximo o aprendizado.
- *Continuidade do aprendizado*: quando surgirem novos exemplos, pode-se treinar uma Rede Neural para aprender estes novos casos a partir de uma rede já treinada sem perder os conhecimentos adquiridos anteriormente.
- *Velocidade de aprendizado*: o *Cascade-Correlation* converge muito rapidamente para um ponto mínimo da curva de erro, e pelo fato de não serem necessárias várias

simulações anteriores para ajustar a arquitetura da rede e os parâmetros, o tempo necessário para o aprendizado de um problema é reduzido drasticamente.

Todas estas vantagens fazem com que o *Cascade-Correlation* seja muito simples de ser utilizado em comparação a outros modelos de Redes Neurais, além de ser extremamente eficaz em termos de aprendizado. Uma das poucas limitações que o *Cascade-Correlation* apresenta é que ele só pode ser utilizado para problemas de classificação, e não para aproximação de funções. Uma vez que o *Cascade-Correlation* é muito adequado para resolver problemas de classificação, isto permite o seu uso em sistemas de autenticação de assinaturas.

## 6 REDES NEURAI APLICADAS AO RECONHECIMENTO DE CARACTERES

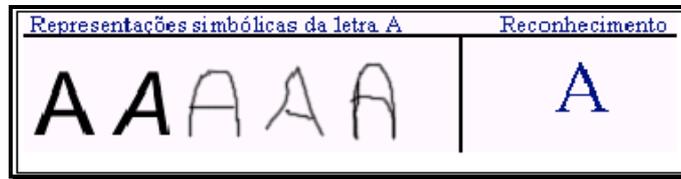
O reconhecimento de caracteres [16] é uma tarefa simples executada por uma rede neural e pode ser implementado em redes Perceptron de Múltiplas Camadas (MLP) treinadas com o algoritmo BackPropagation. No reconhecimento de caracteres, o computador tem que associar um conjunto de pontos a uma letra do alfabeto, sinais, símbolos e algarismos. Unindo os caracteres obtidos através da leitura e reconhecimento de suas características, são formadas sílabas e posteriormente palavras, que podem ser tratadas por algoritmos de correção ortográfica e gramatical, recriando textos em mídia digital. O reconhecimento de caracteres pode ser aplicado também em interface que permitem capturar palavras manuscritas diretamente de um periférico acoplado a um processador de textos, permitindo ao usuário escrever e obter o resultado digitalizado do texto em tempo real.

Ao analisar a forma com que o modelo pedagógico tradicional propõe a alfabetização, verificamos que o sujeito da aprendizagem faz um movimento progressivo na aprendizagem das letras para as sílabas e posteriormente para palavras e frases.

A aprendizagem se dá por um processo de laços de repetição aplicados sucessivamente onde letras são apresentadas para o reconhecimento visual e os sons característicos deste reconhecimento são associados. Num segundo momento, os símbolos que representam as letras são desenhados pelo sujeito da aprendizagem que passa a traduzir símbolos em letras. Os próximos passos são a organização em sílabas, depois em palavras e frases.

O experimento proposto aplica este paradigma tradicional para ensinar um computador a reconhecer o alfabeto. Considera-se aqui que o sujeito da aprendizagem (S) deve passar a reconhecer todas as letras do alfabeto (O), mesmo com a ocorrência de variações na forma como estão escritas, após submetido a um treinamento que apresentará os símbolos e sua tradução correta em letras. A complexidade da proposta está no fato de que é impossível determinar com exatidão a forma como a letra será escrita, exatamente como acontece no dia-a-dia. A aplicação da inferência, ou seja, a generalização de um conhecimento já adquirido é que permitirá o reconhecimento dos

caracteres mesmo com variações de forma. A figura 8 ilustra a aplicação da generalização do conhecimento no reconhecimento da letra A.



**Figura 9 Representação Simbólica para o reconhecimento da letra A**

Para o reconhecimento de caracteres, o modelo de Redes Neurais mais indicado é o Backpropagation (RNA supervisionada).

### 6.1 Algoritmo Backpropagation

A camada de entrada do algoritmo Backpropagation [10] não possui pesos associados nem função de ativação, ela é apenas um *buffer* que recebe os valores externos e os repassa para a camada seguinte. Depois da camada de entrada, existem uma ou mais camadas ocultas, que possuem um número variável de neurônios que depende do problema que está sendo analisando. Por último temos a camada de saída, que recebe os valores da(s) camada(s) oculta(s), realiza a ativação e emite os resultados de saída da Rede Neural. O algoritmo do *Backpropagation* trabalha da seguinte forma: a camada de entrada recebe os valores externos e os repassa para os neurônios da camada seguinte, que é a primeira camada oculta. Estes neurônios fazem a ativação e o resultado é enviado para os neurônios da camada seguinte. Cada camada recebe como entrada os valores obtidos na saída dos neurônios da camada anterior. A camada de saída serve para combinar os valores obtidos nos neurônios das camadas anteriores e assim emitir o valor de saída da Rede Neural.

No *Backpropagation*, o ajuste dos pesos ocorre do fim para o início, sucessivamente em camadas. Para o ajuste dos pesos da camada de saída, calcula-se o erro comparando-se a saída obtida com a saída desejada, e se utiliza este erro para ajustar os pesos desta camada. Para o ajuste dos pesos da camada seguinte, é necessário calcular o nível de participação de cada neurônio na saída da Rede Neural. Este cálculo é realizado multiplicando-se o erro obtido na camada anterior (analisando-se no sentido da saída para a entrada) com o valor do peso que interliga os neurônios destas camadas,

e assim se obtém um erro estimado, que é utilizado para o ajuste dos pesos desta camada. Pelo fato do erro ser estimado e não absoluto, Redes Neurais com mais de uma camada oculta costumam apresentar mais dificuldades no aprendizado.

O número de neurônios na camada de entrada de uma Rede Neural depende do número de variáveis envolvidas no problema a ser abordado. O número de neurônios na camada de saída depende do tipo problema a ser abordado, pois para problemas que exigem uma resposta do tipo sim ou não, uma única saída é suficiente, mas se houver a necessidade de se classificar os exemplos em diversas categorias, podem ser necessárias várias saídas, uma para cada categoria de valores possíveis. Na camada oculta pode ser utilizada qualquer quantidade de neurônios, independente do número de entradas e de saídas utilizadas. Problemas linearmente separáveis não necessitam de neurônios na camada oculta, mas problemas de classificação ou de aproximação de funções mais complexos podem exigir vários neurônios na camada oculta. Para a determinação do número de neurônios da camada oculta, não existe uma regra fixa, pois tudo depende do problema que está sendo analisado, mas o que se costuma fazer é começar com poucos neurônios e ir aumentando o número até que seja possível realizar o aprendizado, o que exige que sejam feitas diversas simulações. O ajuste dos pesos em uma Rede Neural ocorre ao longo de muitas épocas, onde em cada época são submetidos todos os exemplos a Rede Neural, e assim os pesos vão sendo lentamente ajustados até que se chegue aos valores ideais para um determinado problema.

Outro fator importante em uma Rede Neural é o grau de generalização. Quando uma Rede Neural Artificial é treinada, os pesos vão sendo ajustados lentamente até que ela responda de forma adequada aos exemplos presentes na base de dados. Mas simplesmente responder de forma correta aos exemplos que estão sendo analisados não é o suficiente, é necessário que a Rede Neural consiga responder de forma correta a outros exemplos que sigam o mesmo padrão. Por esta razão, quando se realiza o aprendizado, costuma-se dividir a base de dados de exemplos em dois conjuntos de dados distintos, um conjunto de dados para o aprendizado, com a qual a Rede Neural é ativada e tem os seus pesos ajustados, e um conjunto de dados de teste, com a qual a Rede Neural é ativada mas não sofre ajuste nos pesos. A base de dados de teste serve para medir como a Rede Neural se comporta com exemplos para os quais ela não foi treinada ao longo do processo de aprendizado, o que torna possível medir o grau de

generalização da Rede Neural. A medida que as épocas vão passando, o erro em ambas as bases de dados vai sendo reduzido, como mostra a figura 3.9. (pag 33 do artigo assinaturas) Para a base de dados de aprendizado o erro se reduz indefinidamente, já que é a partir dela que os pesos são ajustados. Mas para a base com os dados de teste, chega um ponto em que o erro volta a subir, como pode ser visualizado na figura 3.9. Isto se deve ao fato de que a partir deste ponto a Rede Neural está começando a *decorar* os exemplos da base de aprendizado, e os novos conhecimentos adquiridos não representam mais uma regra geral, mas sim regras específicas para cada exemplo. O ideal é parar o treinamento no ponto ótimo de generalização, que é o instante imediatamente anterior ao ponto onde o erro começa novamente a subir na base de dados de teste.

## 7 REDES NEURAIIS APLICADAS A RECONHECIMENTO DE SÉRIES TEMPORAIS

Um dos aspectos fundamentais da inteligência natural é a habilidade em processar informação temporal. A aprendizagem e a produção de padrões temporais está intimamente associada com nossa habilidade em perceber e gerar movimentos do corpo e de suas partes, fala e linguagem, música, etc [11].

Entre as tarefas que lidam com seqüências temporais podemos citar: *reconhecimento de seqüências, reprodução de seqüência, associação temporal e geração de seqüências temporais.*

Em muitas aplicações científicas e de engenharia é necessário modelar processos dinâmicos que lidam com seqüências temporais. O tipo de informação que se deseja extrair da seqüência vai depender da aplicação. Normalmente, quando se processa algum tipo de padrão temporal, se está interessado em:

**Reconhecimento de Seqüências:** neste caso, deseja-se gerar um padrão de saída particular quando uma seqüência de entrada específica é apresentada. A seqüência de entrada deve ser apenas identificada. Uma aplicação típica é o reconhecimento de voz, em que a saída indicaria a palavra que foi falada.

**Reprodução de Seqüências Temporais:** aqui, o sistema deve ser capaz de gerar a seqüência de entrada quando parte dela (um ou mais estados) lhe é apresentada.

Este seria o caso apropriado quando se deseja que a rede aprenda uma seqüência melódica, ou seja capaz de prever o curso futuro de uma série temporal a partir de partes desta melodia.

**Associação Temporal de Seqüências:** para esta situação, uma seqüência de saída particular deve ser gerada em resposta a uma seqüência específica de entrada. A seqüência de entrada e a de saída podem ser bastante diferentes. Este caso inclui as duas classes anteriores como casos especiais.

**Geração de Seqüências Temporais:** neste caso, o sistema gera uma sucessão de estados entre dois pontos quaisquer não consecutivos dados: o ponto inicial e o ponto final. Este conceito está estreitamente ligado ao de interpolação de estados.

Até o presente momento tem-se falado exaustivamente no termo seqüência temporal. Entretanto, não se comentou ainda que a variável tempo pode ser considerada explícita ou implicitamente em tal seqüência temporal. Desta forma, é interessante perceber que a variável *tempo* pode vir embutida em uma seqüência temporal de duas maneiras básicas [11]:

(i) **Ordem temporal:** se as componentes de um padrão temporal são retirados de um alfabeto específico, a ordem temporal se refere à posição relativa destas componentes dentro da seqüência. Por exemplo, a seqüência a-b-c é considerada diferente da c-b-a por causa do ordenamento diferente. Ordem temporal também pode se referir a uma estrutura sintática, tal como sujeito-verbo-objeto, na qual cada componente é escolhido entre um número de símbolos possíveis.

(ii) **Duração do tempo:** assumindo uma taxa de amostragem uniforme, a duração do tempo é inversamente proporcional à taxa de apresentação (ou de observação) da seqüência. A duração desempenha um papel crítico em algumas tarefas de processamento temporal, tanto no reconhecimento quanto na reprodução de padrões temporais.

Sistemas invariantes à taxa de apresentação são sensíveis tanto à ordem dos eventos quanto à duração relativa dos eventos, enquanto que sistemas invariantes ao intervalo de duração são sensíveis apenas à ordem dos eventos.

O uso de RNAs com aprendizagem não-supervisionada para reconhecimento de séries temporais pode ser justificado pela necessidade de o sistema ser capaz de extrair por si só a informação necessária ao seu correto funcionamento sem (ou com mínima) intervenção de um operador humano. Esta propriedade recebe o nome de auto-organização.

## 8 UM PADRÃO MANUSCRITO PARA O SISTEMA SW

Este capítulo apresenta um manual do padrão manuscrito. Para cada símbolo do Sistema SW, das categorias 1, 2, 3 e 9, foi criado um padrão manuscrito seguindo o padrão “Graffiti”.

Foram propostas várias escritas para cada símbolo e permaneceu a mais simplificada e a que melhor representou o símbolo. A seguir, serão apresentados os padrões manuscritos dos símbolos.

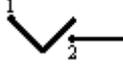
### Categoria Mãos

<i>Código</i>	<i>Símbolo</i>	<i>Padrão Manuscrito</i>
0101001		
0101002		
0101005		

**Tabela 2 Grupo 1 – Indicador**

0102001		
0102002		
0102003		
0102006		

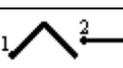
**Tabela 3 Grupo 2 - Indicador-Médio**

0103001		
0103005		
0103008		

**Tabela 4 Grupo 3 - Indicador-Médio-Polegar**

0104001		
0104003		
0104004		

**Tabela 5 Grupo 4 - Quatro Dedos**

0105001		
0105003		
0105007		
0105009		
0105016		
0105018		
0105020		
0105026		
0105027		
0105030		

**Tabela 6 Grupo 5 - Cinco Dedos**

0106004		
0106007		
0106009		
0106011		
0106012		

**Tabela 7 Grupo 6 - Dedo Mínimo**

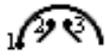
0109013		
0109014		
0109018		
0109020		
0109022		
0109023		

**Tabela 8 Grupo 9 - Dedo Indicador-Polegar**

0110001		
0110003		
0110004		

**Tabela 9 Grupo 10 – Polegar**

### Categoria Expressões Faciais

0301002		
0301004		

**Tabela 10 Grupo 1 - Sobrancelhas**

0302001		
0302002		
0302003		
0302004		

**Tabela 11 Grupo 2 - Olhos**

0305001		
---------	---	--

**Tabela 12 Grupo 5 - Nariz**

0306001		
0306003		
0306009		

**Tabela 13 Grupo 6 - Boca**

As tabelas 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12 e 13 mostram os padrões manuscritos para alguns símbolos do Grupo 1 da Categoria Mãos. Estes padrões foram a maneira mais simplificada que encontramos para representar os respectivos símbolos.

Devido os símbolos conter muitos detalhes ficou difícil de criar um manuscrito que os representassem de maneira clara e simplificada, por isto, os mesmos precisam ser reavaliados para encontrarmos um manuscrito que melhor represente cada um. As tabelas 14, 15, 16, 17, 18 e 19 mostram estes símbolos com seus respectivos manuscritos.

0103002		
0103004		

**Tabela 14 Grupo 3 – Indicador – Médio – Polegar**

0105005		
---------	---	--

**Tabela 15 Grupo 5 – Cinco Dedos**

0106001		
---------	---	--

**Tabela 16 Grupo 6 – Dedo Mínimo**

0107001		
---------	---	--

**Tabela 17 Grupo 7 - Dedo Anular**

0108001		
0108003		
0108004		

**Tabela 18 Grupo 8 - Dedo Médio**

0109001		
0109009		

**Tabela 19 Grupo 9 Dedo Indicador - Polegar**

Como há símbolos com semelhanças muito grande, resolvemos classificá-los em grupo, ou seja, um manuscrito representará um grupo de símbolos, como mostram as tabelas 20 e 21.

0109012		
0109011		
0109010		

**Tabela 20 Grupo 9 Dedo Indicador - Polegar**

### Categoria Contato

0201001	*	
0201002	+	
0201003	*	
0201004	#	
0201005	⊙	
0201006	@	
0202001	•	
0202002	◦	
0202003	^	

**Tabela 21 Contato das mãos**

### Categoria Movimentos

0203001		
0205001		
0203002		
0205002		
0203003		
0205003		
0203014		
0203015		
0206001		
0206005		
0206009		
0210001		
0210003		
0210004		

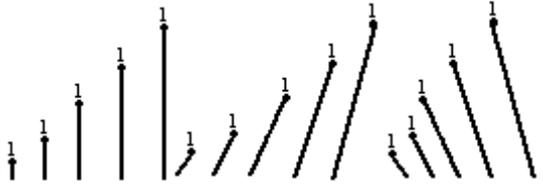
**Tabela 22 Símbolos de Movimentos**

#### 8.1 Variações dos manuscritos

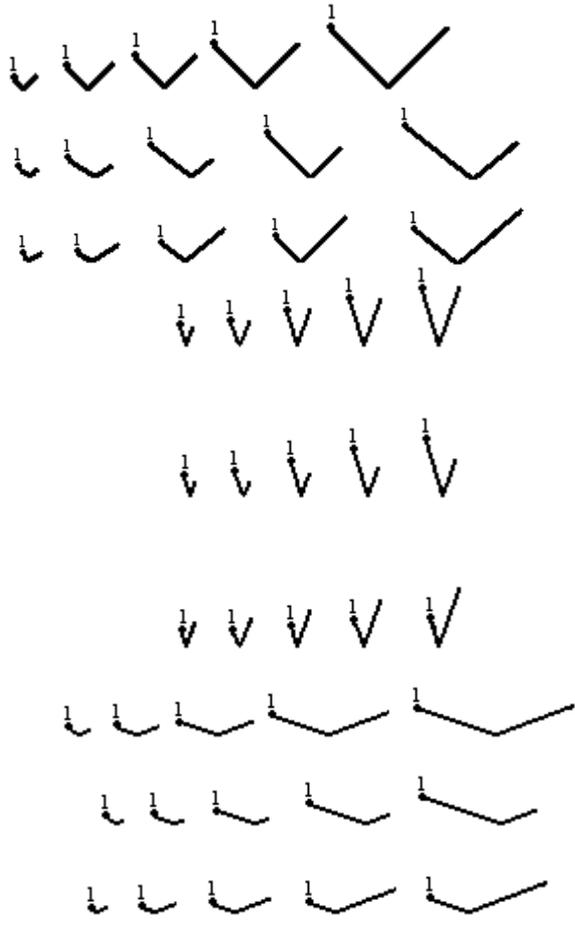
Como a escrita à mão livre nunca é igual, foi necessário fazer as possíveis variações dos manuscritos do Sistema SW, para apresentar à rede apenas algumas destas variações e, assim, obter um bom resultado no reconhecimento.

Em função do tempo disponível, foram feitas as variações apenas dos manuscritos com 1 e 2 traços da categoria mãos. As tabelas mostram as possíveis variações destes manuscritos.

### Categoria Mãos

<i>Código</i>	<i>Símbolo</i>	<i>Manuscrito</i>	<i>Variações da Manuscrito</i>
0101001			

**Tabela 23** Variações do manuscrito com um traço do grupo 1 – Indicador

0102001			
0102003			

<p>0102006</p>			



--	--	--	--

**Tabela 24** Variações do manuscrito com dois traços do grupo 2 – Indicador – Médio

0104004	☐	$\begin{matrix} 1 \leftarrow \\ 2 \leftarrow \end{matrix}$	
---------	---	--	--

**Tabela 25** Variações do manuscrito com dois traços do grupo 4 – Quatro Dedos

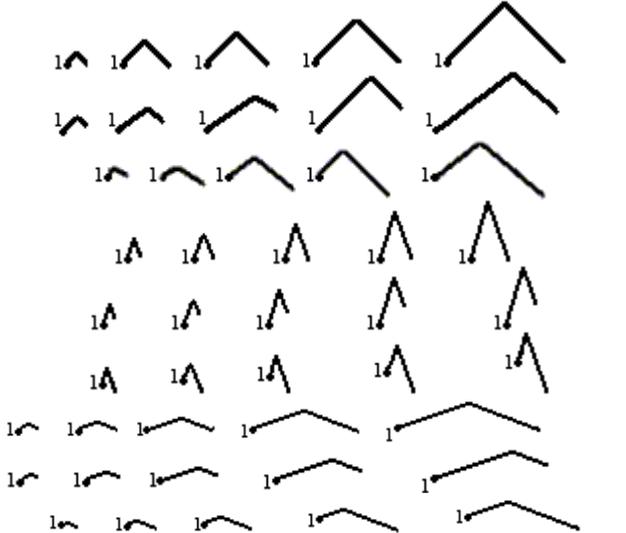
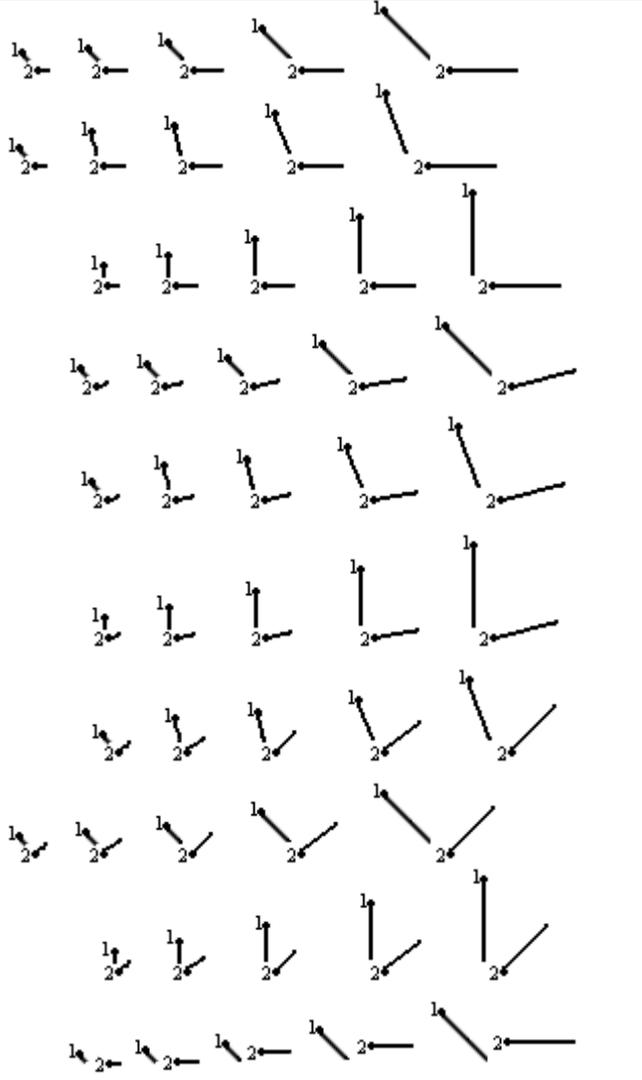
0105007			
---------	---	---	--

Tabela 26 Variações do manuscrito com dois traços do grupo 5 – Cinco Dedos

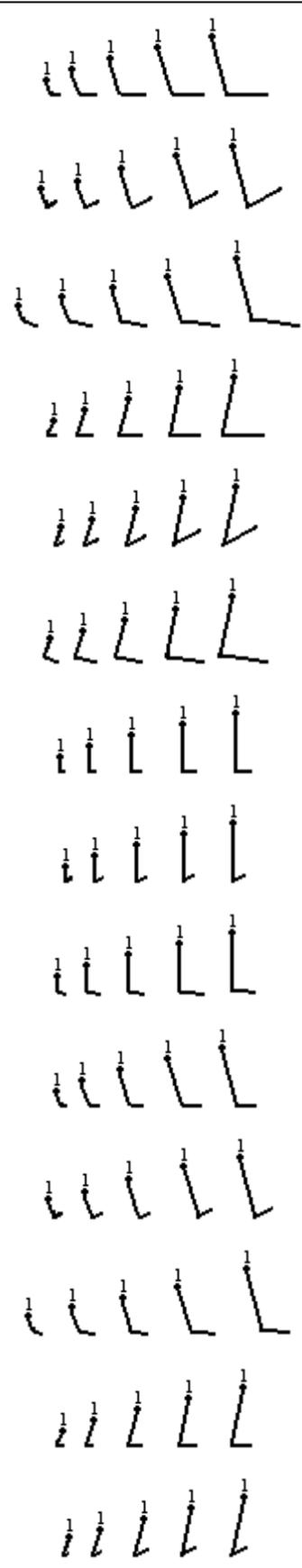
0106009			
---------	---	---	---

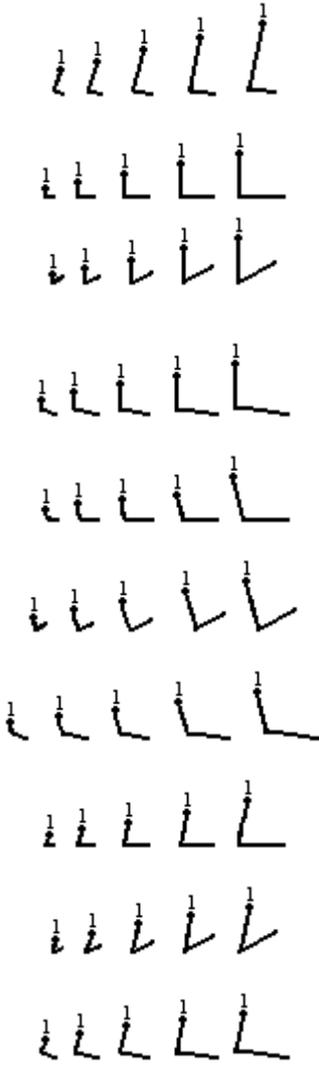
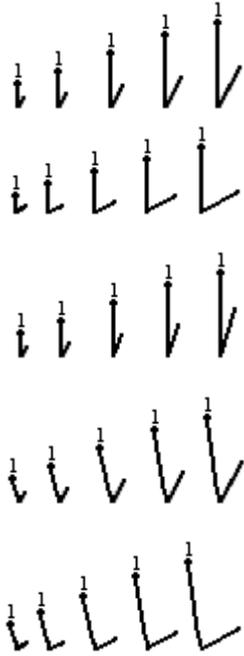
<p>0106012</p>			

--	--	--	--

Tabela 27 Variações do manuscrito com dois traços do grupo 6 – Dedo Mínimo

0109013			
---------	--	--	--

			
--	--	--	---

			
0109014			

			<p>Handwriting practice for the letter 'l'. The page contains 12 rows of the letter 'l' in a cursive style. Each row consists of five letters. Each letter has a small number '1' at the top with a downward-pointing arrow, indicating the starting point and direction of the single stroke used to form the letter. The rows are arranged in a grid within the rightmost column of a table.</p>
--	--	--	--

--	--	--	--

**Tabela 28** Variações do manuscrito com dois traços do grupo 9 – Dedo Indicador - Polegar

0110001			
0110004			


			↗1 ↗1 ↗1 ↗1 ↗1
			↘2 ↘2 ↘2 ↘2 ↘2
			↖1 ↖1 ↖1 ↖1 ↖1
			↙2 ↙2 ↙2 ↙2 ↙2
			→1 →1 →1 →1 →1
			→2 →2 →2 →2 →2
			↗1 ↗1 ↗1 ↗1 ↗1
			↘2 ↘2 ↘2 ↘2 ↘2
			↖1 ↖1 ↖1 ↖1 ↖1
			↙2 ↙2 ↙2 ↙2 ↙2

**Tabela 29** Variações do manuscrito com dois traços do grupo 10 – Polegar

## 9 RECONHECIMENTO DOS SÍMBOLOS MANUSCRITOS DO SISTEMA SW

Para o reconhecimento dos símbolos manuscritos do sistema SW utilizaremos uma Rede Neural Backpropagation utilizando o mesmo modelo para reconhecimento de séries temporais (obs: **dizer o porquê de utilizar séries temporais e não o mesmo modelo de reconhecimento de assinaturas e caracteres**), por ser uma rede supervisionada facilitando o entendimento no momento do aprendizado. A rede foi desenvolvida por Alexandro Teixeira Gonzalez e possui as seguintes características:

- **Número máximo de neurônios na camada de entrada:** 20 neurônios.
- **Número de neurônios na camada escondida:** 41 neurônios.
- **Número de camadas:** 4 camadas, pois este número foi o que apresentou melhor configuração.
- **Número de neurônios na camada de saída:** 1 neurônio.
- Para reconhecer o símbolo foi utilizada a maior função de ativação para ter o neurônio vencedor, isto é, o neurônio que reconheceu o símbolo, sendo que este neurônio ficará “adormecido” para não interferir no resto do treinamento.

As respostas referentes a cada símbolo e os respectivos códigos devem ser editados em arquivos separados com extensão \*.txt na ordem em que os arquivos de dados serão apresentados ao programa.

Para o treinamento serão pedidos estes dois arquivos com as respostas e com os códigos que serão apresentados ao programa, e os símbolos salvos em um outro arquivo com extensão \*.txt, para depois utilizar o botão Processar e saber se o software reconheceu os símbolos corretamente ou não.

Uma Rede Neural não é fácil de ser ensinada, assim como uma criança também não é, mas como uma criança devemos mostrar mais de uma vez a expressão e a resposta referente, por exemplo:

$$1+1 = 2 \quad 2+2 = 4 \quad 3+3 = 6 \quad \dots\dots\dots 100+100 = 200$$

É preciso tomar muito cuidado para que quando chegar no "100+100" a rede não tenha esquecido o "1+1".

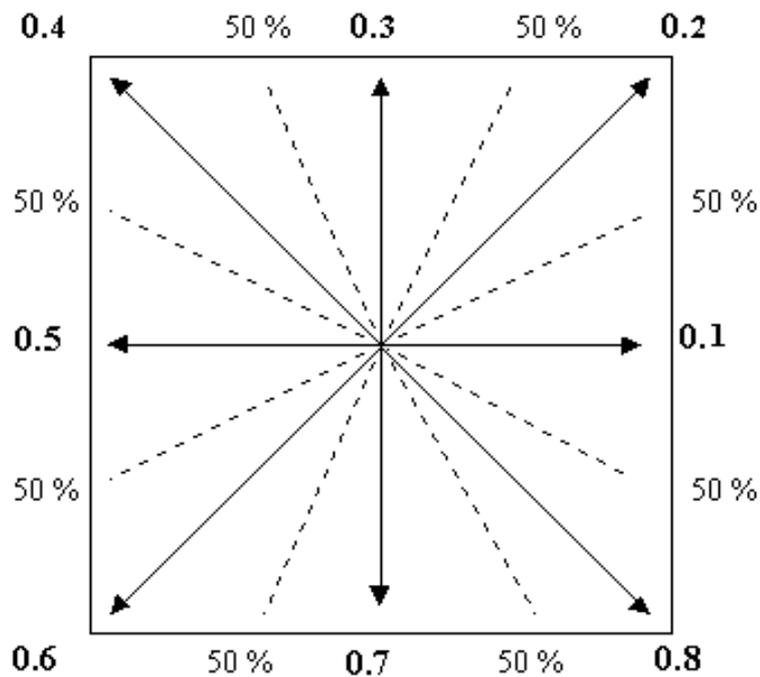
Não se deve medir esforços para que este aprendizado seja feito da melhor maneira possível pelo usuário, pois caso contrário não se tem um software eficiente. Ele aprende mas depende da maneira que ele é treinado.

Para um melhor aprendizado é recomendável trabalhar com valores entre [0..1], mas não deve-se utilizar os valores 0 e 1, pois a rede nunca vai reconhecer nada (0) e ela nunca vai aprender 100% (1).

### **9.1 Dados de entrada**

Os dados de entrada são os manuscritos do sistema SW, que serão expressos na tela do editor, para que posteriormente sejam processados pela Rede Neural. O diferencial deste programa é que os símbolos são desenvolvidos a mão livre, de maneira dinâmica, e não estática, seguindo a seqüência que eles forem dispostos, isto é, à medida que eles vão sendo escritos.

Os dados de entrada serão separados através do grau de inclinação da reta (figura 10), isto é, um mínimo e máximo de pontos podem estar localizados para serem considerados de uma região.



**Figura 10 Dados de entrada separados através da inclinação da reta**

Para dividir os dados de entrada de acordo com a inclinação da reta, foi feito um teste para saber a que quadrante pertence a reta e em seguida outro teste para descobrir o coeficiente angular. Este processo é repetido para todos os quadrantes.

Como pode ser visto na figura 10, a inclinação da reta é de 45 em 45 graus, então se for dada pouca inclinação para as retas, será armazenado o mesmo valor sempre, isto é, se for feita uma reta com inclinação de  $0^\circ$  e uma reta com inclinação de  $30^\circ$ , ambas terão o mesmo código de entrada o que irá diferenciá-las é o tamanho das retas.

## 9.2 Editor utilizado para o reconhecimento dos manuscritos

O editor é composto pelos seguintes botões (figura 11):

- **Gravar:** Serve para gravar o símbolo num arquivo \*.txt, para que o usuário possa visualizá-lo mais tarde.
- **Salvar:** Serve para salvar os dados de entrada, ou seja, o código gerado pela rede no momento da escrita. Estes códigos é quem são responsáveis pelo aprendizado do software.

- **Treinamento:** Este item trabalha com a parte de aprendizado do software, sendo nesta parte necessário apresentar arquivos com os dados de entrada (salvos no botão salvar), o arquivo com os dados de saída ( as respostas) e o arquivo dos códigos que serão apresentadas à rede, pois a mesma necessita destas características para adquirir seu conhecimento.
- **Processar:** Este botão serve para que depois que a rede já tenha adquirido seu aprendizado (através do item anterior), possamos desenvolver o nosso aprendizado dos manuscritos.
- **Limpar:** A principal função deste botão é limpar a tela do editor.



**Figura 11 Interface do editor onde é feita a entrada de dados**

### 9.3 Resultados obtidos

Ainda estou tentando treinar a rede para concluir este item

## CONCLUSÕES

Este trabalho é de extrema importância para as pessoas que possuem deficiência auditiva. Existe o sistema SW que facilita muito para estas pessoas quando estas pessoas precisam recorrer a leitura na forma oral.

A ideia da criação de manuscrito para cada símbolo do sistema SW surgiu com o intuito de facilitar ainda mais a escrita destes sinais, pois eles são uma maneira simplificada de representar cada símbolo.

Na elaboração dos manuscritos procuramos fazer o mais parecido possível para que quando o usuário for usar ele possa associar o manuscrito a cada símbolo sem precisar recorrer ao manual.

O reconhecimento automático destes manuscritos também é de extrema importância, pois os usuários poderão escrever à mão livre, dispensando o uso do teclado. Porém, esta fase do trabalho não foi possível concluir com êxito, devido a falta de tempo e a dificuldade de se treinar uma rede neural.

### **Dificuldades**

Uma das principais dificuldades encontradas foi como treinar a rede neural, que é um processo lento e requer tempo e paciência. Além disso, a comunicação com o autor da rede se dava somente através de e-mail, o que dificultava ainda mais na hora de tentar entender o que estava sendo feito por ele.

A outra dificuldade foi a comunicação com Diogo Madeira, um dos colaboradores, que é surdo-mudo. Como foi citado neste trabalho, ele tem muita dificuldade de interpretar textos na forma oral.

Como alguns símbolos do sistema SW são parecidos, ficou difícil a criação de manuscritos diferentes, por isto criamos um manuscrito para um grupo de símbolos parecidos.

**Desenvolvimentos futuros**

Com a conclusão deste trabalho, ficaram algumas opções para trabalhos futuros, tais como:

- Implementação de uma rede neural mais eficiente para o reconhecimento dos manuscritos em PC,
- Implementação de uma rede neural para o reconhecimento dos manuscritos em palm,
- Adaptação deste trabalho no editor para textos em Língua de Sinais Escritos em SW, desenvolvido por Rafael Torchelsen,

**REFERÊNCIAS BIBLIOGRÁFICAS**

- [1] SUTTON, Valerie. *Um Sistema de Escrita para Línguas de Sinais*. DAC – Deaf Action Committee For SignWriting. Tradução Marianne Rossi Stumpf. Colaboração Antônio Carlos da Rocha Costa e Ronice Muller de Quadros. 2000. Manuscrito não publicado.
- [2] SUTTON, Valerie. *SignWriting Shorthand for Sign Language Stenography*. The Sutton Movement Writing Press. 1982.
- [3] SUTTON, Valerie. *Lessons in Sign Writing*. The Deaf Action Committee For Sign Writing, Inc.. Second Edition. 1982.
- [4] ILS – O Intérprete e a Escrita. Disponível por WWW em <http://www.interpretels.hpg.ig.com.br/7.htm>, 2002. (29/10/02).
- [5] JORNAL do Surdo>Jornalzinho Especial. Disponível por WWW em <http://www.sentidos.com.br/jornaldosurdo/jornalzinho-19-libras-sign.html> 2002. (29/10/02).
- [6] PLASTICIDADE em Redes Neurais Artificiais. Disponível por WWW em <http://www.epub.org.br/cm/n05/tecnologia/plasticidade2.htm>, [s.d.]. (25/10/02).
- [7] UMA Introdução às Redes Neurais. Disponível por WWW em <http://www.din.uem.br/ia/neurais/>, [s.d.]. (25/10/02).
- [8] Disponível por WWW em <http://neural.inesc-id.pt/~tl/tfc.html>, [s.d.]. (25/10/02)
- [9] Haykin, Simon. *Redes Neurais: princípios e prática*. Porto Alegre. Bookman, 2 ed. 2001. trad. Paulo Martins Engel
- [10] Heinen, Milton Roberto. *Autenticação On-line de assinaturas utilizando Redes Neurais*. São Leopoldo: UNISINOS, 2002. (Monografia)
- [11] Barreto, Guilherme de Alencar. *Redes neurais não-supervisionadas para processamento de seqüências temporais*. USP, 1998. (dissertação de mestrado)
- [12] Disponível por WWW em <http://www.ec.ucdb.br/~marco/courses02a/ai2/papers/rnc-pruning.pdf>, [s.d.]. (12/04/03)

- [13] REDES Neurais Artificiais. Disponível por WWW em <http://www.icmc.sc.usp.br/~andre/neural2.html>, [s.d.]. (14/04/03)
- [14] 3COM. *Handbook for the Pam III<sup>TM</sup> Organizer*. [s.d.].
- [15] Disponível por WWW em <http://www.ifqsc.sc.usp.br/grad/artigos/simulacao.doc>, [s.d.]. (15/04/03)
- [16] Disponível por WWW em <http://www.univir-mg.br/engconh/rna.pdf>, [s.d.], (15/04/03)