



Relatório Técnico

**Núcleo de
Computação Eletrônica**

Extração de Características para o Reconhecimento de Dígitos Cursivos - Uma Nova Abordagem

**Gizelle K. Vianna
Roberto J. Rodrigues
Antonio C. G. Thomé**

NCE - 19/2000

Universidade Federal do Rio de Janeiro

EXTRAÇÃO DE CARACTERÍSTICAS PARA O RECONHECIMENTO DE DÍGITOS CURSIVOS – UMA NOVA ABORDAGEM

GIZELLE K. VIANNA¹

ROBERTO J. RODRIGUES²

ANTONIO C. G. THOMÉ³

AEP/NCE- Núcleo de Computação Eletrônica/UFRJ, Caixa Postal 2324, Ilha do Fundão, Rio de Janeiro, RJ, Brasil

¹kupac@posgrad.nce.ufrj.br

²cracky@correio.nce.ufrj.br

³thome@correio.nce.ufrj.br

Resumo

Este artigo propõe um novo método de extração de vetores de características de imagens textuais, obtidos a partir do cálculo de distâncias tomadas do contorno das imagens até pontos de observação dispostos ao redor da mesma. Avaliou-se o método comparativamente com o de representação através de matrizes de bits, concluindo-se que, além da superioridade de desempenho, o método proposto resolve problemas de escala, rotação e translação, que prejudicam o reconhecimento das imagens. A avaliação foi feita por meio de modelos neurais e de figuras de mérito estatísticas tais como o Discriminante de Fisher e distâncias entre centros de massa.

Palavras-Chave

Reconhecimento de Caracteres Manuscritos, Reconhecimento de Padrões, Segmentação de Imagem, Vetores Característicos, Descritores de Formato, Análise de Discriminantes, Redes Neurais, RNA, Inteligência Artificial.

1. Introdução

O reconhecimento de caracteres cursivos é uma tarefa bastante complexa e difícil de ser realizada. Devido à grande quantidade de parâmetros e características individuais da escrita cursiva, não existe, até o presente momento, um procedimento de reconhecimento cujo desempenho se mostre satisfatório em todas as situações existentes.

O desempenho de um sistema automático de reconhecimento depende fortemente da qualidade dos

documentos originais, antes e após sua digitalização. A fim de minimizar os efeitos decorrentes da baixa qualidade dos dados, é comum fazer-se uso de técnicas de compensação da imagem. Técnicas essas que incluem: realce de contornos, remoção de linhas extras, de sublinhados e de ruído, além de outros. Os problemas mais comuns [6], relacionados com a qualidade e a dificuldade de tratamento de uma imagem textual são os ruídos, distorções, translações, rotações e variações de estilo, escala, textura e traço.

O processo de reconhecimento da escrita manuscrita, cursiva ou não, ou de texto impresso, inclui diversas fases, que vão desde a análise dos caracteres individuais, até a utilização de informações léxicas, sintáticas, semânticas e contextuais, para a validação do texto como um todo. Geralmente, a primeira fase é constituída pela segmentação da imagem, que vem a ser uma das tarefas mais complexas e relevantes do processo. Nesta estratégia, o reconhecimento em si é feito ao nível de cada segmento, com posterior validação do conjunto de segmentos, através da utilização de bibliotecas de contexto.

A segmentação de caracteres cursivos precisa lidar com problemas como a segmentação de caracteres inclinados, sobrescritos e com traços conectados. Algumas técnicas existentes utilizam a extração de características baseadas nos dígitos, através de processos que refinam a imagem, afinando o caractere. Existem ainda processos que geram vetores de características, baseando-se nas características das curvas que formam os caracteres [5,8], tais como: a direção dos segmentos, pontos terminais, intersecções e ciclos.

Uma das soluções mais utilizadas atualmente é a extração de pontos e linhas para posterior análise de contorno. Os resultados são satisfatórios, entretanto, o

método mostra-se complexo e requer parametrizações extremamente interdependentes. A maior vantagem das técnicas deste tipo é sua tolerância às distorções e variações de estilo, superior a de outras técnicas.

A investigação relatada neste artigo se concentra unicamente na extração de características e geração de padrões para o reconhecimento de dígitos cursivos do CEP (Código de Endereçamento Postal), utilizado pelo correio brasileiro. A escrita cursiva de padrões

numéricos pode ser vista como um caso mais simples do cursivo genérico, uma vez que, por sua natureza, os dígitos geralmente não são escritos de forma conectada. O processo de reconhecimento que vem sendo adotado nesta investigação inclui diversas etapas, conforme mostra o diagrama da *figura 1*.

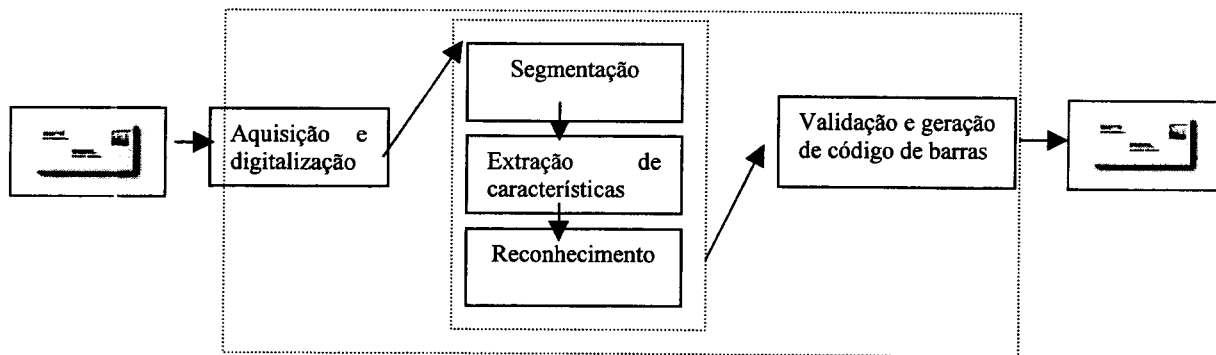


Figura 1: Diagrama geral

2. Segmentação

A segmentação de caracteres cursivos precisa lidar com problemas diversos, que geram enormes dificuldades de tratamento. Dentre eles pode-se citar os casos de caracteres conectados, inclinação do traço, traços extras, elementos não-numéricos, e caracteres sobrescritos.

O método de segmentação proposto e utilizado nesta pesquisa baseia-se no uso de histogramas de projeção numérica, conforme descrito em [7]. Histograma de projeção é uma estrutura que armazena o resultado da projeção da imagem em cada uma das dimensões existentes. Cada dimensão utiliza um vetor para armazenar o número de pixels cujos valores diferem de um limiar definido a priori (geralmente a cor de fundo). Sobre o vetor resultante é aplicada uma heurística de refinamentos sucessivos, até se obter uma segmentação satisfatória.

O método se divide em 3 etapas: compensação da qualidade da imagem; segmentação inicial e refinamentos sucessivos.

Compensação da qualidade da imagem: Esta etapa busca corrigir problemas na imagem original, reduzindo ou realçando detalhes como ruídos ou contrastes.

Segmentação inicial: Nesta etapa, a imagem sofre uma primeira segmentação, com base nas informações armazenadas na estrutura dos histogramas. Todos os

dígitos visivelmente separados são segmentados com sucesso. Também é possível prever alguns erros, como traços e dígitos conectados. Os pontos ou traços podem ser removidos usando as informações do histograma horizontal. Os dígitos conectados, também previstos com este método, são tomados como entrada para etapas de refinamento.

Refinamento: A primeira etapa de refinamento repete a estratégia do histograma, variando-se apenas os parâmetros de refinamento, sendo possível segmentar dígitos fracamente conectados. As etapas seguintes de refinamento incluem o uso de outras técnicas de segmentação. Em [4], mostra-se o resultado de experimentos onde a segmentação atingiu 95,20% de sucesso, usando apenas o primeiro nível de refinamento.

3. Técnicas de Extração das Características

A representação da imagem contida em cada segmento extraído é crítica para o desempenho da etapa de reconhecimento. No caso deste ser implementado através de modelos neurais, além da representatividade, tem-se a dimensionalidade como outro fator de extrema relevância.

A extração de características tem como objetivo reduzir a dimensionalidade do problema, além de manter, ou até melhorar, sua representatividade.

Investiga-se, neste trabalho, o desempenho absoluto e relativo de um novo método, que se concebeu com vistas a melhorar o desempenho das Redes Neurais. O método se baseia na construção de vetores de características a partir de distâncias tomadas do contorno da imagem até pontos de observação estrategicamente dispostos ao redor da mesma [9].

Como descrito a seguir e exemplificado nas figuras 2 a 5, os pontos de observação objetos deste estudo foram escolhidos como aqueles pertencentes a figura geométrica de um quadrado, de um octógono e de um círculo.



Figura 2: Exemplo de varredura da técnica de extração de características do Quadrado



Figura 3: Exemplo de varredura da técnica de extração de características do Raio



Figura 4: Exemplo de varredura da técnica de extração de características do Diâmetro

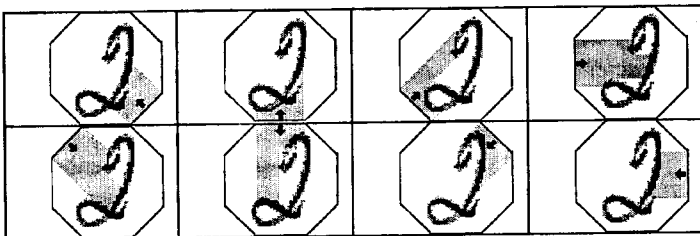


Figura 5: Exemplo de varredura da técnica de extração de características do Octógono

Um estudo comparativo desses métodos de representação e do uso de matriz de bits foi conduzido e, além da superioridade do desempenho, observou-se também que eles oferecem uma solução implícita para problemas importantes como o da escala, da rotação e da translação.

Quadrado: é feita uma varredura da imagem a partir de seus contornos laterais, superior e inferior. Inicialmente, define-se um quadrado que circunscreva toda a figura. A seguir, realiza-se uma varredura, que tem por objetivo calcular a distância, em pixels, de cada ponto da borda do quadrado ao primeiro ponto encontrado na imagem. O processo é repetido da

esquerda para a direita, da direita para a esquerda, de cima para baixo e de baixo para cima. O resultado é armazenado em um vetor de características, que poderá ser interpolado ou não. A implementação é simples e computacionalmente rápida (figura 2).

Octógono: A varredura agora é feita em 8 etapas, formando um octógono (figura 3). O vetor de características compõe-se, assim, de 8 visões distintas da imagem.

Círculo: A varredura, neste caso, é feita radialmente (figura 4), após ter-se gerado um círculo em torno da figura. Calculam-se as distâncias, de forma similar às técnicas anteriores, e armazena-se o resultado em um vetor de características. Pode ocorrer ocultação de parte da imagem, quando esta se situa após o centro do círculo. Este problema é resolvido aplicando-se o método do Diâmetro (figuras 5 e 6). Possui implementação simples e de complexidade computacional relativamente baixa.

4. Avaliação Comparativa

Disponha-se, para a execução deste trabalho de pesquisa, de um banco de imagens de dígitos extraídos do CEP, coletadas por alunos da UFRJ. Os dígitos desse banco de dados estão armazenados em formato de matriz de bits, com 16x16 pixels, totalizando um vetor de características de dimensão 256. Por esta razão, optou-se por compatibilizar a dimensão de todos os vetores de características com a dimensão dos dados já existentes, de 256 pontos.

Para avaliar a eficiência dos diferentes tipos de vetores gerados pelos métodos de extração de características propostos neste trabalho foram utilizados diversos modelos de Redes Neurais no processo de reconhecimento.

Além dos modelos neurais, tipicamente lineares, outras figuras de mérito estatísticas, extraídas da literatura [1,2,3,4] foram utilizadas para avaliar a eficiência dos diferentes vetores de dados. Para realizar essa avaliação, fez-se uso de dois critérios de medição: o *Discriminante de Fisher* [3] e o critério definido neste trabalho e denominado de *Critério das Distâncias*. Ao final do trabalho, realiza-se uma comparação entre os resultados dessas duas avaliações com os resultados apresentados pelas Redes Neurais.

4.1 Matriz de Bits

Além das técnicas propostas neste trabalho, a técnica de representação de dados por mapa de bits também foi avaliada pelos critérios citados na sessão anterior, uma vez que a essa pode ser considerada como a forma canônica de representação de imagens.

O mapeamento de bits é uma técnica de representação amplamente utilizada para armazenamento de características de imagens

digitalizadas. É uma forma simples de representação que, a partir da imagem digitalizada do caractere, produz uma matriz binarizada, contendo valores 0 ou 1. As células que representam um ponto pertencente à imagem contém o valor 1 e as outras células, o valor 0.

4.2 Critério das Distâncias

Esse critério realiza a avaliação da qualidade de um conjunto de dados através da definição de uma medida baseada no cálculo das distâncias entre os centros de massa das classes de dados existentes. O cálculo dessa medida, bem como a sua avaliação, será descrita a seguir.

Inicialmente, as amostras de dígitos disponíveis são separadas em 10 grupos. Cada grupo contendo apenas amostras de um dígito, ou seja, o grupo G_0 contém apenas amostras de 0's, o grupo G_1 as amostras de 1's, e assim por diante.

Após esse procedimento, calcula-se os centros de massa de cada grupo, através da fórmula (1), obtendo-se 10 vetores, cada um representante de um dos conjuntos G_i . Cada vetor é denominado de cm_i , $0 \leq i \leq 9$.

$$cm_i = \frac{1}{n} \sum_{j=1}^{256} \sum_{k=1}^n a_{kj} x_j \quad (1)$$

Onde: $0 \leq i \leq 9$ e $n=n^o$ de vetores do conjunto.

A seguir, para cada conjunto, calcula-se as distâncias de cada um de seus elementos a cada um dos centros de massa cm_i , utilizando-se a fórmula (2). As distâncias são então agrupadas em novos conjuntos, onde cada conjunto contém as distâncias de todos os dígitos de determinado tipo para um dos centros de massa.

d_{ij} = conjunto das distâncias entre cada dígitos da classe j ao centro de massa da classe i .

$$d_{ij} = \left[\left(\sum_{k=1}^{256} (cm_{ik} - a_{jk})^2 \right)^{1/2} \right] \quad (2)$$

Onde: $0 \leq i \leq 9$

Ao final, tem-se o conjunto d_{00} , contendo as

distâncias de cada elemento do conjunto G_0 ao centro de massa da classe G_0 .

do Critério das Distâncias ao trabalho e a matriz de bits.

A análise das tabelas permite concluir que o método do Quadrado é o que apresenta a melhor potencialidade de discriminação das classes. Já o método da Matriz de Bits gera conjuntos de dígitos cujos centros de massa se localizam muito próximos

fim de obter-se as razões rd_{ij} , fórmula (4), que representam a distância relativa de cada classe ao centro de massa das demais, em relação a distância ao seu próprio centro de massa.

$$md_{ij} = \frac{1}{n_{ij}} \sum_{i,j=0}^9 d_{ij} \quad (3)$$

$$rd_{ij} = \frac{md_{ii}}{\sum_{k=1}^{n_{ij}} md_{ij}(k)} \quad (4)$$

Onde, $0 \leq i,j \leq 9$ e $n_{ij} = n^o$ de elementos do conjunto d_{ij}

O somatório SRD_i , descrito pela fórmula (5), representa o grau de separação da classe i das demais classes. O fator $SDCM$ (Somatório das Distâncias aos Centros de Massa), calculado segundo a fórmula (6), indica o grau de separação total entre as classes. Quanto menor esse fator, melhor o grau de separação e, conseqüentemente, melhor a qualidade esperada dos dados gerados pelo método de extração de características em questão.

$$SRD_i = \sum_{j=0}^9 rd_{ij}, \quad 0 \leq i \leq 9 \quad (5)$$

$$SDCM = \sum_{i=0}^9 SRD_i \quad (6)$$

Para que uma técnica de representação seja eficiente, é necessário ainda que a distância média entre os elementos de um conjunto e seu centro de massa seja menor do que as distâncias médias desses elementos aos outros centros de massa. Quanto menor for a distância ao centro de massa próprio, quando comparada às distâncias aos outros centros de massa, mais eficiente pode ser considerada a separação das classes.

As tabelas 1 e 2 mostram os valores de rd_{ij} e SRD_i para os métodos do Quadrado e Matriz de Bits. As linhas correspondem aos centros de massa e cada coluna representa a rd_{ij} entre o centro de massa correspondente à linha e aos elementos da classe

Obtém-se, no total, 10 conjuntos de distâncias para cada G_i , totalizando 100 conjuntos. São eles:

$d_{00}, d_{01}, d_{02}, \dots, d_{09}, d_{10}, d_{11}, \dots, d_{97}, d_{98}, d_{99}$

Depois, calcula-se as médias de cada conjunto fórmula (3), e divide-se cada md_{ij} pela média n

entre si. Essa característica pode comprometer significativamente o desempenho do processo de reconhecimento dos dígitos, pois é um indício de que as diferentes classes encontram-se bastante sobrepostas no espaço, indicando alto grau de semelhança entre os vetores de características de cada classe.

Tabela 1: Médias das distâncias dos elementos de cada classe aos centros de massa de todas as classes, utilizando-se o método de extração de características do Quadrado.

QUADRO DE DISTÂNCIAS AOS CENTROS DE MASSA		CLASSES									
		0	1	2	3	4	5	6	7	8	9
CENTROS DE MASSA	0	1,00	0,52	0,91	0,73	0,69	0,78	0,73	0,63	0,73	0,69
	1	0,42	1,00	0,60	0,64	0,67	0,66	0,62	0,77	0,70	0,66
	2	0,70	0,64	1,00	0,90	0,81	0,90	0,86	0,78	0,87	0,81
	3	0,60	0,67	0,85	1,00	0,84	0,94	0,84	0,84	0,87	0,89
	4	0,60	0,70	0,82	0,85	1,00	0,85	0,82	0,88	0,85	0,90
	5	0,61	0,66	0,84	0,93	0,82	1,00	0,90	0,81	0,91	0,88
	6	0,64	0,66	0,86	0,87	0,82	0,92	1,00	0,76	0,90	0,79
	7	0,53	0,76	0,75	0,83	0,87	0,81	0,75	1,00	0,85	0,89
	8	0,61	0,70	0,85	0,88	0,85	0,91	0,89	0,85	1,00	0,88
	9	0,60	0,67	0,81	0,90	0,90	0,90	0,78	0,89	0,88	1,00
SRD		5,30	5,98	7,29	7,52	7,27	7,65	7,19	7,21	7,54	7,40

Tabela 2: Médias das distâncias dos elementos de cada classe aos centros de massa de todas as classes, utilizando-se o método de extração de características da Matriz de Bits.

MATRIZ DE BITS - DISTÂNCIAS AOS CENTROS DE MASSA		CLASSES									
		0	1	2	3	4	5	6	7	8	9
CENTROS DE MASSA	0	1,00	0,77	0,90	0,85	0,87	0,91	0,89	0,84	0,88	0,90
	1	0,86	1,00	0,94	0,91	0,92	0,92	0,91	0,95	0,92	0,92
	2	0,92	0,88	1,00	0,91	0,92	0,93	0,94	0,94	0,95	0,94
	3	0,87	0,89	0,93	1,00	0,89	0,96	0,90	0,89	0,92	0,94
	4	0,89	0,87	0,93	0,87	1,00	0,92	0,93	0,94	0,92	0,94
	5	0,90	0,89	0,94	0,94	0,93	1,00	0,94	0,91	0,94	0,95
	6	0,90	0,85	0,94	0,88	0,92	0,94	1,00	0,89	0,93	0,91
	7	0,87	0,92	0,95	0,88	0,95	0,92	0,90	1,00	0,94	0,94
	8	0,88	0,86	0,96	0,89	0,91	0,93	0,93	0,92	1,00	0,94
	9	0,89	0,87	0,94	0,92	0,93	0,95	0,91	0,93	0,94	1,00
SRD		7,98	7,80	8,44	8,06	8,24	8,38	8,24	8,22	8,34	8,37

Tabela 3: Valores encontrados de SDCM, para cada um dos métodos avaliados.

MÉTODO	SDCM
Quadrado	70,34
Octógono	73,43
Diâmetro	72,37
Raio	72,02
Matriz de Bits	82,07

4.3 Discriminante de Fisher

O discriminante de Fisher foi utilizado como uma segunda medida para avaliação do poder que cada técnica de representação possui para separar as classes de dígitos existentes. Pode-se entender essa função discriminante como sendo a diferença máxima entre as médias de cada classe, normalizadas pelo espalhamento total do conjunto de amostras.

Para o cálculo do discriminante, os dados devem estar representados na forma matricial, onde cada linha representa uma amostra e cada coluna uma característica (ou variável) desta amostra. Foram utilizadas 10 matrizes de classes: M_0, M_1, \dots, M_9 ; onde cada matriz M_i contém todas as amostras da classe i correspondente. Além dessas, utilizou-se também uma matriz M , que contém o conjunto total de amostras disponíveis.

Inicialmente, calcula-se as seguintes covariâncias:

$$S_{w_i} = \sum_{n \in C_i} (x_n - m_i)(x_n - m_i)^T \quad (7)$$

= matriz de covariâncias da classe i

$$S_w = \sum_{i=0}^9 S_{w_i} \quad (8)$$

$$m_i = \frac{1}{N_i} \sum_{n \in C_i} x_n \quad (9)$$

$$S_T = \sum_{n=1}^N (x_n - m)(x_n - m)^T \quad (10)$$

= matriz de covariâncias de M

$$m = \frac{1}{N} \sum_{n=1}^N x_n \quad (11)$$

A covariância total das amostras, S_T , equivale à soma das covariâncias internas de cada classe, S_w , com a covariância entre as classes, S_B , definida abaixo. Essa matriz pode ser entendida como uma medida da covariância entre as classes.

$$S_B = \sum_{i=1}^c N_i (m_i - m)(m_i - m)^T \quad (12)$$

$$S_T = S_w + S_B \quad (13)$$

Quando o formato de representação de um conjunto de dados é definido, espera-se que ele seja capaz de maximizar a separação entre as classes de dados. Um formato eficiente de representação deverá minimizar o valor de S_w , ao mesmo tempo em que maximiza o

valor de S_B , significando um espaçamento maior entre as classes do que dentro de cada uma delas. Logo:

$$\frac{S_W}{S_B} \ll 1 \quad (14)$$

Neste trabalho, utilizou-se como discriminante a razão (11) abaixo. Então, quanto menor for a razão DF , melhor será o método de representação.

$$DF = \frac{\text{traço}(S_W)}{\text{traço}(S_B)} \quad (15)$$

Os resultados obtidos estão na *figura 4*. Da análise dos resultados pode-se concluir que o método de representação que possibilita a melhor separação entre as classes de dígitos é o método do quadrado. Depois dele, os melhores são o método do octógono, raio, diâmetro e matriz de bits, nesta ordem.

Tabela 4: Discriminantes de Fisher

MÉTODO	DF
Quadrado	-1.1654
Octógono	-1.1541
Diâmetro	-1.1424
Raio	-1.1451
Matriz de Bits	-1.1248

5. Modelos Neurais

Para avaliar o desempenho dos diferentes formatos de representação e extração das características no reconhecimento de dígitos cursivos foram utilizados dois tipos de sistemas de Redes Neurais, denominados *tipo I* e *tipo II*.

O sistema de tipo I consiste de uma Rede Neural com 256 neurônios de entrada e 10 neurônios de saída, com uma ou duas camadas escondidas. As saídas foram codificadas conforme a *tabela 5*. As redes utilizadas são do tipo Multilayer-Perceptron, treinadas com uma e duas camadas escondidas, com funções de ativação lineares e sigmóides. Várias redes diferentes foram avaliadas e os resultados de treinamento estão nas *tabelas 6 a 8*.

Tabela 5: Representação das saídas das Redes Neurais do tipo I.

CARACTERE	SAÍDA
0	0000000001
1	0000000010
...	...
9	1000000000

O tipo II é composto por dez Redes Neurais, cada uma treinada de forma a reconhecer apenas um dentre todos os dígitos existentes. Todas as dez redes possuem 256 neurônios de entrada e 1 neurônio de saída, que assumirá os valores 0 ou 1. Têm-se então 10 redes, R_i , onde a rede R_i é responsável pela identificação do dígito i , $0 \leq i \leq 9$.

O conjunto de entradas é único, porém as saídas correspondentes irão variar conforme a rede a qual serão aplicadas. Assim, se a rede R_0 estiver sendo treinada, o conjunto de saídas correspondente terá saídas iguais a 1 sempre que a entrada correspondente for a representação de um dígito 0, e terá saída igual a 0 caso contrário. As outras redes serão treinadas de forma análoga.

Na fase de testes, cada amostra é apresentada a todas as redes e os resultados de todas elas são analisados conjuntamente. A rede que fornece o maior valor de saída é considerada a vencedora e o dígito identificado como pertencente à classe correspondente.

6. Desempenho do Reconhecimento

O método de representação de caracteres manuscritos do Quadrado é o que apresentou melhor desempenho no reconhecimento na maioria dos modelos neurais testados.

No modelo linear com uma camada de neurônios escondidos (*tabela 6*), o método do Quadrado, além de obter o melhor desempenho, possui o modelo neural mais simples, quando comparado aos métodos que obtiveram desempenho parecido ao deste (Diâmetro e Raio). Resultado parecido ocorreu com o modelo linear com duas camadas de neurônios escondidos (*tabela 7*), onde o método do Quadrado, apesar de ter obtido o melhor desempenho, alcançou um resultado bastante próximo ao do método do Diâmetro, porém com um modelo neural mais simples.

No modelo do tipo I com funções de propagação sigmóide na camada escondida (*tabela 8*), o método do Quadrado foi novamente superior. No outro modelo do tipo I, com funções de propagação sigmóide camada escondida e linear na camada de saída (*tabela 8*), o método da Matriz de Bits foi melhor, porém o Quadrado novamente apresenta um desempenho próximo e utilizando um modelo neural mais simples.

Nos últimos modelos avaliados, do tipo II, o método do Quadrado mais uma vez apresentou um desempenho melhor que os demais. Os desempenhos de cada método de extração de características, em cada modelo neural implementado, estão detalhados nas *tabelas 6 a 9*.

Tabela 6: Desempenho dos sistemas neurais do tipo I, com uma camada intermediária, função de propagação linear e objetivo 0,1. As células em destaque indicam o melhor desempenho em cada caso.

1 REDE NEURAL - 1 CAMADA ESCONDIDA					
#N R	QUAD.	OCTÓG.	DIÂM.	RAIO	MB
2	0.2176	0.1472	0.2424	0.2301	0.2243
4	0.2399	0.1855	0.2957	0.2439	0.2210
8	0.3453	0.2185	0.3831	0.2934	0.2597
10	0.3905	0.1373	0.3442	0.3304	0.2519
16	0.4446	0.1512	0.4126	0.3875	0.2740
20	0.4655	0.1333	0.4076	0.3843	0.3249
30	0.4351	0.1386	0.4550	0.4401	0.3790
32	0.4378	0.1307	0.4511	0.4320	0.3591
40	0.3912	0.1294	0.4464	0.4345	0.4055

Tabela 7: Desempenho dos sistemas neurais do tipo I, com 2 camadas intermediárias, função de propagação linear em todas as camadas, objetivo de 0,1.

1 REDE NEURAL - 2 CAMADAS ESCONDIDAS					
# NR	QUAD.	OCTÓG.	DIÂM.	RAIO	MB
2-2	0.2081	0.1927	0.2788	0.1737	0.1779
4-2	0.1851	0.1974	0.2252	0.1950	0.2475
4-4	0.2419	0.2627	0.3043	0.2677	0.1757
8-2	0.2155	0.2172	0.2306	0.2690	0.2133
8-4	0.2453	0.2785	0.2655	0.2878	0.2343
8-8	0.3034	0.3168	0.3327	0.3285	0.2475
10-10	0.3682	0.3149	0.3694	0.3530	0.2740
16-2	0.2297	0.2106	0.2881	0.2483	0.2475
16-4	0.2446	0.2673	0.3536	0.3191	0.2343
16-8	0.3804	0.3102	0.4126	0.3197	0.2751
16-16	0.3824	0.3545	0.3838	0.3680	0.3337
20-10	0.3541	0.3439	0.3788	0.3348	0.3006
20-20	0.4473	0.2878	0.3978	0.3749	0.3425
30-10	0.3757	0.3432	0.3608	0.3549	0.3260
30-20	0.2723	0.2284	0.4550	0.4464	0.3624
30-30	0.2486	0.1743	0.3719	0.4063	0.3381
32-2	0.1426	0.1987	0.2770	0.2069	0.2619
32-4	0.2784	0.2482	0.2953	0.2771	0.2177
32-8	0.3696	0.3254	0.3493	0.3417	0.2818
32-16	0.3034	0.2165	0.4263	0.4019	0.3503
32-32	0.2365	0.1479	0.4331	0.3592	0.4022
40-10	0.3311	0.2785	0.3741	0.3179	0.2972
40-20	0.2081	0.1670	0.3317	0.3223	0.3680
40-30	0.1622	0.1762	0.4007	0.2777	0.3967
40-40	0.1642	0.1432	0.3558	0.3292	0.4011

Tabela 8: Desempenho dos sistemas de tipo I, com uma camada intermediária, funções de propagação sigmóide na camada escondida e sigmóide ou linear na camada de saída, objetivo de 0,01 e 0,1, respectivamente. As células em destaque indicam o melhor desempenho em cada caso. FPE= Função de propagação da camada escondida. FPS = Função de propagação da camada de saída.

1 REDE NEURAL - 1 CAMADA ESCONDIDA										
# NEUR	QUADRADO		OCTÓGONO		DIÂMETRO		RAIO		MATRIZ BITS	
FPE	logsig	logsig	logsig	logsig	logsig	logsig	logsig	logsig	logsig	logsig
FPS	purelin	logsig	purelin	logsig	purelin	logsig	purelin	logsig	purelin	logsig
2	0.4932	0.1453	0.3822	0.1657	0.4817	0.2201	0.4614	0.1348	0.5492	0.3448
4	0.6142	0.1676	0.5017	0.3505	0.5712	0.3935	0.5680	0.2627	0.6530	0.5713
8	0.6378	0.6926	0.4792	0.5069	0.5993	0.5453	0.5912	0.5699	0.6917	0.7017
10	0.6642	0.7277	0.5188	0.5756	0.6018	0.5601	0.6063	0.5824	0.6873	0.6917
16	0.6899	0.7561	0.5188	0.5835	0.6068	0.6216	0.6038	0.6119	0.6972	0.7558
20	0.6649	0.7500	0.5254	0.5934	0.6108	0.6392	0.6150	0.6395	0.6663	0.7591
30	0.6554	0.8054	0.5175	0.5947	0.6072	0.6486	0.6150	0.6702	0.6983	0.7514
32	0.6554	0.8014	0.5333	0.6370	0.6137	0.6489	0.6094	0.6671	0.6707	0.7779
40	0.6412	0.8358	0.5195	0.6475	0.6165	0.6640	0.6038	0.6702	0.6630	0.7801
64	0.6311	0.8554	0.5241	0.6475	0.6108	0.6683	0.6006	0.6865	0.6398	0.7812

Tabela 9: Desempenho dos sistemas do tipo II, com uma camada intermediária, funções de propagação logarítmica e/ou linear e objetivo de 0,01 e 0,1, respectivamente. As células em destaque indicam o melhor desempenho em cada caso. FPE= Função de propagação da camada escondida. FPS = Função de propagação da camada de saída.

10 REDES NEURAIS PARALELAS - 1 CAMADA ESCONDIDA										
# NEUR	QUADRADO		OCTÓGONO		DIÂMETRO		RAIO		MATRIZ BITS	
FPE	purelin	logsig	Purelin	logsig	purelin	logsig	purelin	logsig	purelin	logsig
FPS	purelin	logsig	Purelin	logsig	purelin	logsig	purelin	logsig	purelin	logsig
2	0.3514	0.8581	0.3201	0.6172	0.3545	0.5909	0.3574	0.5956	0.3702	0.6685
4	0.3919	0.8682	0.3894	0.6238	0.3939	0.6061	0.3793	0.6708	0.2873	0.7680
8	0.3480	0.8682	0.3300	0.6634	0.4061	0.6121	0.4295	0.6646	0.2928	0.7348
10	0.4122	0.8716	0.3927	0.6700	0.4576	0.6212	0.4545	0.6771	0.2818	0.7348
16	0.4459	0.8649	0.3003	0.6832	0.4273	0.6576	0.4138	0.7147	0.4088	0.7680

7. Conclusões

Os melhores desempenhos ocorreram nos modelos neurais que utilizam uma camada escondida, com funções de propagação sigmóide nesta camada. Nestes modelos (*tabelas 8 e 9*), o método do Quadrado foi sempre superior aos outros modelos. Na comparação direta com o método de Matriz de Bits, em apenas um modelo neural ele se mostra superior aos métodos propostos e, ainda assim, o desempenho é praticamente o mesmo alcançado pelo método do Quadrado. Porém, o modelo utilizado pelo método do Quadrado é bem mais simples, pois o número de neurônios é aproximadamente a metade do que o utilizado pelo outro método.

Nos modelos que utilizam função de propagação linear, o desempenho foi sempre muito deficiente. Isso é razoável, já que o problema envolve vetores de dimensão 256, com sobreposição de classes, e um modelo de separação mais simples dificilmente poderia obter um alto desempenho.

Verificou-se que o modelo neural do tipo II, com dez redes trabalhando em paralelo, é o que fornece o melhor desempenho. Além disso, o desempenho superior é alcançado com modelos bem menores do que os do tipo I. O melhor resultado de todos é obtido com a utilização do método do Quadrado, aliado a um modelo neural de tipo II, com funções de propagação sigmóide na camada escondida e na camada de saída (*tabela 9*). Este modelo consiste de dez Redes Neurais idênticas, em paralelo, cada uma contendo 10 neurônios.

8. Referências Bibliográficas

- [1] Afifi, A. A. ; Clark, V.; "Computer-Aided Multivariate Analysis – 2nd Edition", Chapman & Hall, NY, 1990, pp.
- [2] Bishop, C. M.; "Neural Networks for Pattern Recognition", Oxford University Press, 1995, pp.
- [3] Duda, R. O; Hart, P. E. "Pattern Classification and Scene Analysis", John Wiley & Sons, 1973, pp.
- [4] Rodrigues, R. J.; Thomé, A. C. G.; "Reconhecimento de dígitos cursivos – um método de segmentação por histogramas", a ser publicado no SCI
- [5] Srikantan, G.; Lam, S. W.; Srihari, S. N; *Gradient-Based Contour Encoding For Character Recognition*, Pattern Recognition, Vol. 29, No. 7, pp. 1147-1160, 1996
- [6] Suen, C.Y.; Berthold, M.; Mori, S.; *Automatic Recognition of Handprinted Characters – The State of The Art*, Proceedings of the IEEE, Vol. 68, No. 4, April 1980
- [7] Trier, O. D.; Jain, A.K.; Taxt, T.; "Feature Extraction Methods for Character Recognition – A Survey"; Pattern Recognition, Vol. 29, No. 4, pp. 641-662, 1996
- [8] Verschueren, W.; Schaeken, B.; Cotret, Y. R.; Hermanne, A.; *Structural Recognition of Handwritten Numerals*, CH2046-1/84/0000/0760\$01.00@1984 IEEE
- [9] Yang, L.; Prasad, R.; "Online Recognition of Handwritten Characters Using Differential Angles and Structural Descriptors", Pattern Recognition Letters, nº 14, Dec-93, North-Holland, pp: 1019-1024.