

Extração de Características para o Reconhecimento de Letras Manuscritas

EUGÊNIO SILVA

IME - Instituto Militar de Engenharia, DE9 - Departamento de Engenharia de Sistemas

Praça Gen. Tibúrcio, 80, Praia Vermelha, Rio de Janeiro, RJ, Brasil

eugenio@de9.ime.eb.br

ROBERTO J. RODRIGUES

DCC/NCE - Núcleo de Computação Eletrônica

UFRJ, Caixa Postal 2324, Ilha do Fundão, Rio de Janeiro, RJ, Brasil

rjr@nce.ufrj.br

ANTONIO C. G. THOMÉ

DCC/NCE- Núcleo de Computação Eletrônica

UFRJ, Caixa Postal 2324, Ilha do Fundão, Rio de Janeiro, RJ, Brasil

thome@nce.ufrj.br

RESUMO

Este artigo descreve um conjunto de testes para a avaliação de uma técnica de extração de características proposta em [11] que utiliza como base de dados um conjunto de letras manuscritas. Em [10] foi feita uma avaliação desta técnica, porém a base de dados era composta somente de dígitos. Esta técnica foi desenvolvida principalmente para tratar o problema de reconhecimento de caracteres manuscritos e é baseada na projeção do contorno da imagem sobre os lados de um polígono regular construído em torno de cada caracter. O vetor de características é formado pelas distâncias perpendiculares tomadas a partir de cada lado do polígono até o contorno da imagem. No processo de avaliação compara-se a abordagem proposta usando dois polígonos (quadrado e hexágono) e duas quantidades diferentes de linhas de projeção tomadas de cada lado do polígono, com duas versões de codificação através de mapas de bits (normal e afinado). O poder de discriminação de cada caso é examinado através do uso de um modelo de rede neural MLP (*Multi-Layer-Perceptron*).

Palavras-chave: Extração de Características, Reconhecimento de Caracteres, Reconhecimento de Padrões, Redes Neurais.

1 INTRODUÇÃO

Devido à grande quantidade de variações de estilo e características individuais da escrita de cada pessoa, o reconhecimento de caracteres manuscritos, sejam eles cursivos ou em letra de forma, é uma tarefa bastante complexa e difícil de ser realizada.

O desempenho de um sistema automático de reconhecimento de caracteres manuscritos e até mesmo de caracteres de máquina, depende fortemente da qualidade dos documentos originais, antes e após a sua digitalização. Com o intuito de minimizar os efeitos decorrentes da baixa qualidade das imagens, é comum fazer-se uso de técnicas de compensação destas imagens. Tais técnicas envolvem realce de contornos, remoção de linhas extras, de sublinhados e de ruídos, além de outros. Os problemas mais comuns, relacionados com a qualidade e a dificuldade de

tratamento de uma imagem textual, são os ruídos, distorções, translações, rotações e também variações de estilo, escala, textura e traço [5].

O processo de reconhecimento da escrita manuscrita, 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. Depois do processo de segmentação são extraídas e codificadas as características de cada segmento. Em seguida o reconhecimento 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 manuscritos precisa lidar com problemas como a segmentação de caracteres

inclinados, sobrescritos e com traços conectados. Algumas técnicas de extração de características existentes extraem as características dos caracteres após estes terem sido submetidos a um processo de afinamento da imagem [9]. Existem ainda processos que geram vetores de características baseando-se nas particularidades das curvas que formam os caracteres, tais como a direção dos segmentos, pontos terminais, interseções e ciclos [8].

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.

Realizada a segmentação e a extração das características dos caracteres, o reconhecimento (classificação) pode ser feito utilizando-se de várias ferramentas. Dentre elas destacam-se figuras de mérito estatísticas, algoritmos de clusterização, métodos nebulosos, redes neurais artificiais e ainda métodos híbridos que envolvem duas ou mais técnicas [1 e 2].

2 POSICIONAMENTO

O trabalho relatado aqui é uma complementação do trabalho realizado em [10]. Naquela ocasião fez-se um estudo comparativo entre os mesmos métodos de extração de características que são avaliados aqui, ou seja, o método de projeção poligonal segundo os polígonos quadrado e hexágono e o método do mapa de bits tanto normal quanto afinado. No entanto, a base de dados utilizada no trabalho anterior era composta apenas de caracteres numéricos ao passo que agora a base é composta das 26 letras do alfabeto inglês. No primeiro experimento, para cada um dos polígonos foram utilizados cinco diferentes tamanhos de vetores de características: 32, 64, 128, 256 e 512. A avaliação foi feita por meio de redes neurais artificiais, mais especificamente o modelo MLP (*Multi-Layer-Perceptron*). Feitos os testes verificou-se que o método de projeção poligonal se mostrou mais eficiente que o método do mapa de bits em ambos os modos. Dentre os modos de projeção poligonal o método do quadrado se mostrou mais eficiente que o método do hexágono. A Tabela 1 mostra os resultados obtidos após a realização de 900 ciclos de treinamento da rede neural.

A tabela mostra para cada método a configuração da rede neural utilizada (entrada – neurônios na camada escondida – neurônios na camada de saída) e o desempenho alcançado. De posse destes resultados constatou-se a superioridade do método do quadrado frente aos outros métodos. Ainda naquela oportunidade verificou-se que, apesar de o método do quadrado 512 ter se mostrado mais eficiente, este não seria a melhor opção.

Tabela 1: resultados obtidos com 900 ciclos de treinamento (*epochs*) para o caso dos dígitos

Método	Config. da Rede	% de Erros	% de Acertos
Quad-32	24 – 17 – 10	45.80	54.20
Quad-64	56 – 23 – 10	10.02	89.98
Quad-128	116 – 47 – 10	6.76	93.24
Quad-256	240 – 87 – 10	6.53	93.47
Quad-512	484 – 173 – 10	5.36	94.64
Hex-32	37 – 17 – 10	53.26	46.74
Hex-64	63 – 23 – 10	32.75	67.25
Hex-128	136 – 47 – 10	12.00	88.00
Hex-256	256 – 87 – 10	7.69	92.31
Hex-512	520 – 173 – 10	7.11	92.89
M. Bits	223 – 85 – 10	11.42	88.58
M. Bits Af	252 – 85 – 10	11.07	88.93

O aumento da eficiência foi muito pequeno frente ao aumento da complexidade no processamento computacional. Concluiu-se então que para os caracteres numéricos o melhor método de extração de características seria o quadrado com vetor de características de tamanho 128 ou 256. Com base nestes resultados decidiu-se concentrar as investigações do presente trabalho considerando apenas vetores de características de tamanho 128 e 256 para o método de projeção poligonal que utiliza o quadrado e o hexágono como polígonos extratores de características.

3 CAPTURA E SEGMENTAÇÃO

A base de dados usada no atual experimento foi obtida a partir da digitalização de formulários de concurso pré-impresos devidamente preenchidos por várias pessoas. A diversidade dos dados é garantida visto que os formulários foram preenchidos por pessoas de ambos os sexos, diferentes idades e diferentes níveis de escolaridade. Os formulários são compostos por campos numéricos, alfabéticos e alfanuméricos como nome, endereço, idade, telefone, CEP e outros.

O dispositivo de captura usado nesta investigação foi um *scanner* simples e a imagem original foi digitalizada segundo uma resolução de 200dpi em modo colorido. O uso de cores no modo de captura tem a finalidade de permitir a remoção das linhas delimitadoras das quadrículas em que são escritos os caracteres no formulário assim como para extrair e armazenar a fotografia e a assinatura do candidato. Após o processo de captura é necessário informar ao sistema de segmentação onde estão as regiões de interesse do formulário, ou seja, deve-se informar onde estão os campos do formulário que devem ser extraídos para posterior segmentação. Feito isso, todo o processo de segmentação transcorre de forma automática para todos os formulários. Finalizados os processos de

captura e segmentação a imagem relativa a cada caracter foi convertida para tons de cinza (*grayscale*) e binarizada e todas as imagens foram armazenadas em disco no formato TIF.

O método de segmentação usado nesta investigação é baseado em histogramas de projeção conforme descrito em [12]. O histograma de projeção representa uma estrutura que armazena o resultado da projeção da imagem sobre cada uma das duas dimensões existentes. A cada dimensão é associado um vetor onde é armazenado o número de pixels que representa a energia acima de um certo limite (geralmente definido como a cor de fundo). A segmentação é então realizada aplicando-se uma série de refinamentos sucessivos até se chegar a um resultado satisfatório. O método pode ser visto como uma seqüência de três etapas distintas: inicia-se com uma compensação da qualidade da imagem seguida por uma segmentação inicial e então se necessário, por um conjunto de sucessivos refinamentos [12].

O passo inicial é usado para aprimorar a qualidade da imagem digitalizada, reduzindo ou realçando determinados detalhes como ruídos ou contrastes. A imagem é inicialmente segmentada com base nas informações armazenadas nos vetores do histograma de projeção. O algoritmo é bastante rápido e todos os caracteres desconectados são segmentados satisfatoriamente neste estágio. Pontos e linhas esparsos podem ser removidos usando as informações retiradas do histograma obtido com a projeção horizontal. Caracteres levemente conectados são facilmente destacadas neste estágio, no entanto se o traço for mais denso, a segmentação só será possível no próximo estágio de refinamento.

O estágio de refinamento pode ter uma ou mais fases, na primeira fase porém se reaplica a estratégia da segmentação inicial alterando-se os parâmetros do algoritmo. Este procedimento permite a segmentação de caracteres fracamente conectados. O passo seguinte inclui a utilização de outras técnicas de segmentação. O resultado alcançado com os experimentos conduzidos mostraram que é possível alcançar uma precisão de cerca de 95,20% utilizando apenas o primeiro nível de refinamento. Deve-se ressaltar que este método foi escolhido por ser apropriado para segmentar caracteres escritos à mão, porém não cursivos.

4 EXTRAÇÃO E CODIFICAÇÃO DE CARACTERÍSTICAS

O método de extração de características aqui avaliado representa uma nova abordagem, derivada dos algoritmos conhecidos de detecção de contorno [11 e 12]. Os vetores de características são baseados na computação de um conjunto de distâncias tiradas a partir do contorno da imagem até um polígono de referência.

O polígono de referência deve ser regular porém pode ter qualquer número de lados. Este deverá ser posicionado em torno da imagem-alvo como mostrado na Figura 1. Qualquer polígono pode ser construído com base nas Equações (1, 2, 3 e 4).

O processo básico toma a distância de cada lado do polígono até o contorno da imagem e armazena esta distância em um vetor que também contém o número de lados e o número de pontos (tamanho do lado) para cada lado. O número de características extraídas pode não ser o mesmo para diferentes tipos de polígonos, ainda que o número de características desejadas seja o mesmo. O número de características computado é sempre muito próximo do desejado e isto acontece devido à diferença na geometria e inclinação dos lados para os polígonos escolhidos. Esta diferença é sempre pequena e portanto, não é relevante para o problema em si.

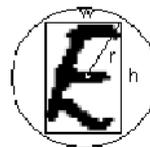


Figura 1a: círculo circunscrito à imagem

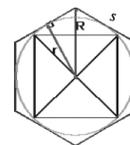


Figura 1b: polígono circunscrito ao círculo

$$r = \frac{1}{2} \sqrt{h^2 + w^2} \quad (1)$$

$$R = \frac{r}{\cos\left(\frac{\pi}{n}\right)} \quad (2)$$

$$R = \frac{1}{2} \sqrt{h^2 + w^2} * \cos\left(\frac{\pi}{n}\right)^{-1} \quad (3)$$

$$s = \sqrt{h^2 + w^2} * \tan\left(\frac{\pi}{n}\right) \quad (4)$$

onde,

r = raio do círculo circunscrito (metade da diagonal)

h = altura da imagem

w = comprimento da imagem

n = número de lados do polígono

R = raio do círculo circunscrito (em relação ao hexágono) – lado do hexágono

s = tamanho do lado do hexágono.

Usando um algoritmo de reconstrução da imagem do caracter a partir do vetor de características gerado foi possível perceber que a fidelidade resultante não se mostrou favorável com polígonos de número ímpar de lados. Desta forma, ficou decidido concentrar a investigação em cima de polígonos com número par de lados, mais especificamente o quadrado e o hexágono. A Figura 2 mostra a seqüência das projeções para o hexágono. A investigação inclui também a comparação do desempenho dos mesmos em relação a dois outros

tipos de métodos utilizando mapas de bits, no modo normal e no modo afinado [9].

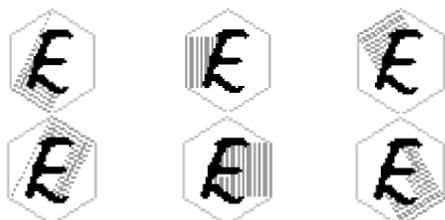


Figura 2: seqüência de projeções para o hexágono

Como descrito anteriormente o vetor de entrada possui uma dimensão predefinida e as características são as distâncias computadas em número de pixels e tomadas a partir de cada lado do polígono até o contorno da imagem. Este método não só provou ter um bom poder de discriminação como também resolveu problemas relacionados com escala, rotação, translação e traço da imagem original. Uma vez que o *bitmap* esteja inicialmente centralizado dentro de um retângulo antes de ser circundado pelo polígono desejado, o problema da translação é automaticamente eliminado. A aplicação de um esquema de interpolação na tomada dos pontos para o cômputo da distância ao contorno resolve o problema de escala e, o problema da rotação, pode ser solucionado aplicando-se uma rotação circular no vetor resultante para a direita ou para esquerda, conforme for a inclinação do dígito e, finalmente, a espessura do traço perde a importância uma vez que as distâncias consideram apenas o contorno da imagem.

5 AMBIENTE DE AVALIAÇÃO

Diferentemente do trabalho realizado em [10], os experimentos relatados neste artigo se restringem ao uso de caracteres alfabéticos. Após o processo de captura e segmentação foi feita a separação dos caracteres em dois grupos, dígitos e letras. Posteriormente foi feita uma filtragem dos caracteres alfabéticos segmentados. Da quantidade total das letras existentes verificou-se que uma pequena quantidade destas se mostraram ininteligíveis, ambíguas ou não foram corretamente segmentadas. Cabe ressaltar que o motivo da segmentação incorreta de alguns caracteres é devido à extrapolação do traço da letra na quadrícula por parte de quem preencheu o formulário. Todas as letras que apresentaram algum dos problemas mencionados foram eliminadas da base de dados. O total de exemplares de cada padrão bem como as quantidades deles utilizadas para treinamento e teste da rede neural são mostrados na Tabela 2.

Tabela 2: distribuição dos exemplares disponíveis de cada padrão

Letras	1	2	3
A	1220	400	820
B	663	400	263
C	607	400	207
D	680	400	280
E	723	400	323
F	489	400	89
G	658	400	258
H	673	400	273
I	732	400	332
J	694	400	294
K	727	400	327
L	741	400	341
M	748	400	348
N	820	400	420
O	863	400	463
P	707	400	307
Q	669	400	269
R	973	400	573
S	788	400	388
T	491	400	91
U	622	400	222
V	703	400	303
W	762	400	362
X	750	400	350
Y	783	400	383
Z	957	400	557
Total	19243	10400	8843

Legenda:

1: total de padrões em cada classe;

2: padrões de treinamento;

3: padrões de teste;

Como mencionado anteriormente, o objetivo maior da avaliação foi verificar se a abordagem proposta em [12] era capaz ou não de garantir um bom desempenho discriminatório para todo o conjunto de letras. Dentre as várias formas de avaliação possíveis decidiu-se pelo uso de uma rede neural artificial que é detalhada mais adiante no item 6. As análises comparativas consideraram características extraídas na forma de mapas de bits (normal e afinado) e dos polígonos quadrado e hexágono usando a técnica aqui descrita.

A matriz do mapa de bits foi obtida convertendo a imagem original para níveis de cinza e posterior binarização. Após estes processos a imagem resultante foi reescalada linearmente (sem distorções) para o tamanho final de 16x16 pixels. A imagem afinada foi obtida através da aplicação de um algoritmo de afinamento [9].

Os vetores de características poligonais foram montados através de uma sucessão de extrações utilizando duas dimensões diferentes para cada tipo de

polígono cada um com 128 e 256 características. Estes vetores foram construídos através de um sistema especialmente desenvolvido para tal.

A formatação do arquivo de saída gerado pelo processo de extração de características segue o padrão ASCII e este arquivo é utilizado como entrada para a rede neural.

O tamanho real do vetor de características referente ao método do quadrado é exatamente igual ao valor desejado, entretanto, para o método do hexágono existe uma particularidade. No caso do hexágono, o valor adotado foi aumentado para o inteiro mais próximo para cima com o intuito de ser compatível com a proporção real das medidas. Por exemplo: $256 / 4 = 64$ (para o quadrado), mas para o hexágono, a divisão $256 / 6 = 42,666$, não é exata. O valor mais próximo (para mais) é de $258 / 6 = 43$. Esta diferença sempre será pequena o suficiente para não interferir no resultado final.

6 AVALIAÇÃO ATRAVÉS DE REDES NEURAIS

O modelo de redes neurais adotado para avaliar a capacidade de discriminação das técnicas de extração de características foi o MLP, treinado com o algoritmo de retropropagação do erro (*error backpropagation*). A rede neural MLP foi criada valendo-se do software MATLAB, juntamente com a ferramenta *Neural Networks Toolbox* [13]. A arquitetura da rede possui apenas uma camada escondida de tamanho variável, com uma camada de saída com 26 neurônios, ou seja, a rede foi treinada para aprender todas as 26 letras do alfabeto. A codificação das saídas da rede neural segue o esquema mostrado na Tabela 3. O vetor de saída foi escolhido como sendo ortogonal para facilitar o processo de treinamento. O tamanho da camada escondida foi escolhido empiricamente baseado no tamanho do vetor de características de entrada. A função de ativação adotada utiliza um somatório simples das entradas ajustadas pelos seus respectivos pesos. Como função de propagação utilizou-se a função logística sigmoideal (*logsig* no MATLAB) tanto na camada escondida quanto na camada de saída. O treinamento foi realizado com *momentum* e taxa de aprendizado adaptativa através da função *traindxd* do MATLAB. Como o objetivo deste estudo é avaliar o poder de discriminação das diferentes formas de extração do vetor de características, não se dedicou muito tempo na busca da melhor configuração ou modelo de rede neural. Definiu-se arbitrariamente o modelo, a configuração e a estratégia de treinamento, que foram mantidos para as diferentes características para que os métodos fossem avaliados em condições de igualdade.

O conjunto de dados foi dividido em dois subconjuntos distintos: um para treinamento e outro para teste de acordo com a quantidade de dados mostrada na Tabela 2.

Tabela 3: codificação das saídas da rede neural.

Letra	Saída
A	10000000000000000000000000000000
B	01000000000000000000000000000000
...	...
Z	00000000000000000000000000000001

Exceto no caso da matriz do mapa de bits (normal e afinado), todas as informações da entrada (treinamento e teste) foram normalizadas por um fator, onde este fator representa a distância entre os lados opostos do polígono utilizado. Desta forma tem-se a certeza de que todos os valores estão compreendidos entre 0 e 1.

Um outro procedimento adotado para a preparação do conjunto de dados foi a eliminação de todas as colunas com variância nula. Este procedimento bem simples eliminou algumas variáveis do espaço de entrada, e desta forma, reduziu a sua dimensionalidade. Por exemplo: o quadrado de lado 128 foi reduzido para 116. Cogitou-se o uso de outros métodos de redução de dimensionalidade mais complexos, porém constatou-se que estes provocariam perda de informação presente nos vetores de características.

Lembrando mais uma vez que o objetivo maior é a comparação de desempenho entre as diferentes estratégias, definiu-se um simples critério de seleção para a validação da saída da rede neural. A saída foi modificada de acordo com a estratégia “the winner takes all”, onde o maior valor entre todas as 26 saídas foi alterado para um (1) e o restante dos valores foram alterados para zero (0). Dessa forma tem-se índice zero (0) de rejeição.

O treinamento foi realizado 1000 ciclos de treinamento (*epochs*) e os resultados obtidos são mostrados nas Tabelas 4. A representação A-B-C para a configuração da rede neural significa: A = número de entradas; B = número de neurônios na camada escondida e C = número de neurônios na camada de saída.

Tabela 4: resultados obtidos com 1000 ciclos de treinamento (*epochs*)

Método	Configuração da Rede	Epochs	% Erros	% Acertos
Quad 128	116-47-26	1000	15,15	84,85
Quad 256	240-79-26	1000	13,97	86,03
Hex 128	136-59-26	1000	19,26	80,74
Hex 256	256-81-26	1000	15,67	84,33
M Bits	251-83-26	*401	21,72	78,28
M Bits Af	248-83-26	*570	31,56	68,44

(*) convergiu para o erro mínimo antes de se esgotarem todos os ciclos de treinamento

Nota-se que o polígono quadrado com o vetor de características de tamanho 256 mostrou o melhor resultado. Nota-se também que o método do mapa de bits, tanto normal quanto afinado, convergiu para o erro mínimo antes de serem completados todos os ciclos de treinamento, porém não obteve resultados tão bons quanto os métodos do quadrado e do hexágono.

7 CONCLUSÕES

Os resultados obtidos aqui reforçam os resultados alcançados em [10]. Assim como para a base de dados composta de dígitos, percebeu-se aqui que o método de projeção poligonal segundo o quadrado, com vetor de características de tamanho 256, apresentou o maior poder de extração das características também para o caso das letras.

Foi observada, no entanto, uma tendência de erro entre as letras B e D; H e N e O e Q. Tal fato mostra uma deficiência do método em discriminar padrões cuja diferença esteja localizada na parte interna dos mesmos. Alternativas de solução do problema já estão sendo levantadas e serão avaliadas em breve.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] Bishop, C. M.; "Neural Networks for Pattern Recognition", Oxford University Press, 1995.
- [2] Duda, R. O; Hart, P. E. "Pattern Classification and Scene Analysis", John Wiley & Sons, 1973.
- [3] Kupac, G. V.; Rodrigues, R. J.; Thomé, A. C. G.; "Extração de características para reconhecimento de dígitos cursivos", VIth Brazilian Symposium on Neural Networks - SBRN2000, Rio de Janeiro, november 2000.
- [4] 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.
- [5] 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, pp. 469-487, April 1980.
- [6] 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.
- [7] Verschueren, W.; Schaeken, B.; Cotret, Y. R.; Hermanne, A.; "Structural Recognition of Handwritten Numerals", 7ICPR(84), pp. 760-762.
- [8] 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.
- [9] Pavlidis. T.; "Algorithms for Graphics and Image Processing", Computer Science Press, Inc., 1982.
- [10] Rodrigues, R. J.; Silva, Eugênio; Thomé, A. C. G.; "Feature Extraction Using Contour Projection"; 5th World Multiconference on Systemics, Cybernetics and Informatics SCI2001 – Orlando – USA, July 22 – 25.
- [11] Rodrigues, R. J.; Kupac, G. V.; Thomé, A. C. G.; "Character Feature Extraction Using Polygonal Projection Sweep (Contour Detection)", IWANN2001, Granada - Spain, June 13 a 15, LINCOS, 2001, pp 687-695, 2001.
- [12] Rodrigues, R. J.; Thomé, A. C. G.; "Cursive character recognition – a character segmentation method using projection profile-based technique", The 6th International Conference on Information Systems, Analysis and Synthesis ISAS 2000 – Orlando – USA – August 2000, Proceedings Volume V, pp. 274-279.
- [13] Demuth, H.; Beale, M.; "Neural Network Toolbox User's Guide Version 3.0", Copyright 1992 - 1997 by The MathWorks, Inc.
- [14] L. S. Oliveira, E. Lethelier, F. Bortolozzi, R. Sabourin L.R.B. Schomaker and L.G. Vuurpijl (Eds.), "A New Approach to Segment Handwritten Digits", Proceedings of the Seventh International Workshop on Frontiers in Handwriting Recognition, September 11-13 2000, Amsterdam, Nijmegen: International Unipen Foundation, pp 577-582
- [15] A.S. Britto, E. Lethelier, R. Sabourin, Ching Y. Suen L.R.B. Schomaker and L.G. Vuurpijl (Eds.), "Improvement in Handwritten Numeral String Recognition by Slant Normalization and Contextual Information", Proceedings of the Seventh International Workshop on Frontiers in Handwriting Recognition, September 11-13 2000, Amsterdam, Nijmegen: International Unipen Foundation, pp 323-332.