Reconhecimento Óptico de Caracteres Manuscritos em Documentos Históricos

PONTES, Alisson. S. L., OLIVEIRA, Adriano Lorena Inácio de.
Escola Politécnica de Pernambuco – Universidade de Pernambuco
adriano@dsc.upe.br, aslp@dsc.upe.br

Resumo

Reconhecimento de textos manuscritos é um importante problema com aplicação em diferentes casos, como, por exemplo, leitura automática de cheques; reconhecimento de endereços postais; indexação automática de documentos, entre outros. Esse trabalho aborda a problemática do reconhecimento de caracteres manuscritos em documentos históricos. Iremos tratar aqui dos processos de segmentação de texto, com destaque para o algoritmo de segmentação de caracteres proposto durante esse trabalho, chamado de sliding window; extração de características usando o método de histogramas de projeção; e redes neurais artificiais (RNA) para classificação.

Abstract

There are many applications concerning handwritten text recognition, for example, check automatic reading; postal addresses recognition; documents automatic index; and others. This paper approaches the handwritten characters recognition in historical documents. We will analyze the processes of text segmentation, with prominence for the sliding window, characters segmentation algorithm considered during this work; characteristics extraction, using the method "Histogramas de Projeção"; and artificial neural networks (RNA) for classification.

1. Introdução

O projeto de ProHist [1] se propõe a desenvolver novos algoritmos para processamento de documentos históricos. Atualmente, o projeto envolve a cooperação entre a universidade de Pernambuco e a universidade Rey Juan Carlos, Espanha. Seus principais objetivos são a preservação de documentos históricos e o desenvolvimento de formas mais fáceis de transmissão desses documentos [9][10].

Utilizamos como base de dados o legado do acervo de Joaquim Nabuco. Esse acervo é composto por cartas datilografadas, manuscritas, cartões postais, entre outros. São mais de 6.500 documentos, totalizando mais de 30.000 páginas. Para esse trabalho usamos apenas documentos manuscritos.

Esse artigo analisa, dentre o processo completo para reconhecimento, as etapas de segmentação de texto, extração de características e classificação.

O próximo tópico aborda o problema da segmentação de texto. O terceiro tópico mostra a importância de se usar um método de extração de características e explica de que forma ele foi implementado nesse trabalho. O quarto tópico explica as diferentes redes neurais utilizadas durante os testes de classificação. O item 5 descreve o modo de testes utilizado na nossa análise. No item 6 são mostrados os percentuais de erro obtidos para cada classificador. O item 7 sugere trabalhos que podem ser realizados a partir dos resultados deste artigo. O item 8 conclui o artigo e no último item as referências utilizadas são numeradas.

2. Segmentação de Texto

Na segmentação de texto [6] nós consideramos como entrada para o sistema o resultado obtido de uma segmentação de documento [13]. Do resultado da segmentação de documento espera-se um documento "limpo", ou seja, foi feito um pré-processamento onde são eliminados todo tipo de ruído e possíveis inclinações no texto. Além disso, trabalhamos apenas com imagens binárias. Alguma técnica de processamento de imagem deve ser utilizada para se obter imagens binarizadas.

As etapas que consistem na segmentação de texto podem ser divididas em segmentação de linhas e segmentação de caracteres.

A **figura 1** mostra as etapas da segmentação de texto. Ao final das etapas de segmentação de texto obtém-se cada caractere isolado.

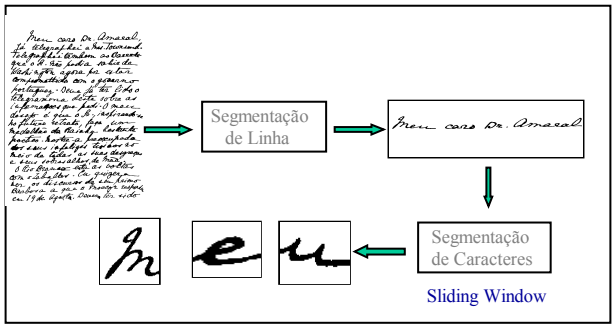

Figura 1. Ilustração das etapas de segmentação de texto.

2.1. Segmentação de Linhas

A segmentação de linhas em documentos manuscritos nem sempre é trivial. Comumente observamos invasão ou até sobreposição de algumas letras sobre letras das linhas adjacentes. Em vista dessa dificuldade, criamos um algoritmo que identifica e segmenta apenas as linhas que podem ser delimitadas por regiões retangulares.

Selecionando as melhores amostras dos resultados obtidos na aplicação desse algoritmo, poderemos automatizar os testes para a segmentação de caracteres. Além disso, devido a modularidade possibilitada pelo uso de uma linguagem orientada a objetos, a adição de módulos que identifiquem os casos em que existem linhas tratamento é simples.

2.1. Segmentação de Caracteres

Essa etapa recebe o resultado obtido da segmentação de linhas e retorna cada caractere isoladamente [7][8].

O S*lideing window* é o algoritmo proposto nesse trabalho para segmentação de caracteres. Esse algoritmo foi feito da seguinte forma: primeiro percorre-se a linha horizontalmente, extraindo imagens dela como se estivesse tirando fotografias; depois, para cada imagem, eliminar toda a parte branca que estiver acima ou abaixo da área que representa o caractere. Os resultados obtidos desse algoritmo foram bastantes satisfatórios e podem ser usados como entrada para uma rede neural.

3. Extração de Características

Uma imagem é representada por uma matriz de pixels. As amostras de imagem utilizadas nesse projetos foram normalizadas em matrizes de 32x32 pixels, o que representa 1024 informações.

Como entrada para uma rede neural, 1024 informações seria impraticável na etapa de classificação, por isso utiliza-se técnicas de extração de características [15]. Durante esse projeto foi implementado os Histogramas de projeção.

3.1. Histogramas de Projeção

No Histogramas de Projeção são feitas projeções horizontais e verticais da imagem e colocadas num array. Essa técnica reduz o número de informações descritivas da imagem de 1024 para 64.

A **Figura 2** ilustra o Histograma de Projeção. Veja que para cada linha e cada coluna há um valor do Histograma.

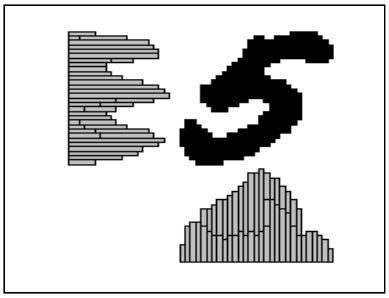

Figura 2. Ilustração do Histogramas de Projeção.

4. Classificação

A última etapa para o reconhecimento de caracteres é a etapa de classificação. Diferentes técnicas de classificação podem ser usadas para reconhecimento de padrões. Todos os métodos de classificação disponíveis devem ser bem estudados e analisados, pois em cada aplicação uma técnica diferente pode se comportar de forma mais satisfatória. Para isso fizemos uma análise comparativa entre algumas redes neurais usadas como classificadores[11]. Usamos regra do vizinho mais próximo (*KNN – k Nearest Neighbors*) [2][3][4]; Redes RBF [2][4]; e máquinas de vetor de suporte (*SVM - Support Vector Machines)* [1][4].

4.1. Regra dos Vizinhos Mais Próximos

A regra dos vizinhos mais próximos (KNN) é um método de classificação supervisionado onde um padrão é dito pertencer a uma classe de acordo com a quantidade de vizinhos que pertençam a essa classe, conforme um critério de distância (distância Euclidiana, geralmente).

A idéia para classificação é bastante simples: na fase de treinamento deve-se armazenar os padrões de treinamento; depois para um novo padrão é calculada a distância euclidiana desse padrão para cada um dos padrões de treinamento; são identificados os k padrões de treinamento mais próximos do novo padrão; a classe que tiver mais representantes dentre esses k padrões será a classe do novo padrão.

Na escolha do parâmetro K deve-se ter cuidado para escolher um número ímpar, pois um número par poderia causar conflito quando um novo padrão tivesse o mesmo número de vizinhos de diferentes classes.

4.2. Máquinas de Vetor de Suporte

Basicamente, o SVM é um algoritmo linear que constrói hiperplanos, com o objetivo de encontrar hiperplanos ótimos, ou seja, hiperplanos que maximizem a margem de separação das classes, para separar os padrões de treinamento em diferentes classes.

Os valores de treinamento são mapeados num espaço de varias dimensões. Depois a SVM encontra um hiperplano linear de separação.

Um kernel é o produto interno em algum espaço de características. Diferentes kernels têm sido propostos na literatura. Alguns exemplos são:

- Linear
- Polinomial
- Sigmóide
- Função de base radial (RBF)

No relatório passado foram utilizados o kernel RBF e o kernel polinomial. SVMs com kernel RBF possuem dois parâmetros, chamados C (o parâmetro de penalidade do termo de erro (C>0)) e gamma (a largura dos kernels RBF), SVMs com kernel polinomial possuem parâmetros C e d. Esses parâmetros têm grande influência na classificação dos dados e na generalização dos mesmos e, portanto, esses valores devem ser cuidadosamente selecionados de acordo com o problema.

4.3. Funções Base Radiais

São utilizadas para aplicações em problemas de aprendizagem supervisionada (regressão, classificação).

A função de ativação aplicada a um nodo de uma RBF utiliza como medida a distância entre os vetores de entrada e de peso. Funções radiais representam uma classe especial de funções cujo valor diminui ou aumenta em relação à distância de um ponto central.

O algoritmo de treinamento usado nas redes RBF é o DDA (Dynamic Decay Adjustment) [12]. Com esse algoritmo, as redes não precisam de uma estrutura física totalmente pré-definida. Não é necessário definir a quantidade de camadas intermediárias, pois o algoritmo mesmo calcula o melhor número de camadas escondidas, assim como o número de nodos em cada camada escondida

5. Análise de dados

Para cada aplicação específica uma combinação diferente entre método de extração de características e de classificação pode comportar-se melhor. Por isso após a implementação dos Histogramas de Projeção fez-se necessária a realização de testes para mensurar os resultados obtidos a partir da utilização desse método de extração de características em conjunto com os classificadores descritos anteriormente [14].

Os testes foram executados utilizando a ferramenta de mineração de dados Weka e validação cruzada 10-fold como modo de teste. Na validação cruzada 10-fold o conjunto das amostras que serão utilizadas como entrada para os classificadores são divididos em dez grupos. Nove desses grupos são utilizados para treinamento das redes e um é usado para testes. Isso é repetido dez vezes, em cada vez utiliza-se um grupo diferente para testes.

6. Resultados Obtidos

Dentre todos os cassificadores, o SVM com kernel RBF obteve os melhores resultados para C = 100 e Gamma = 0,01. A **tabela 1** mostra o desempenho desse classificador.

A **tabela 2,** que mostra o melhor resultado obtido, deixa claro como é de extrema importância a decisão sobre quais parâmetros usar para obtenção dos melhores resultados. A mudança desses parâmetros afeta totalmente os resultados, podendo em alguns conjuntos de parâmetros, encontrarmos valores próximos dos 50% enquanto em outros conjuntos conseguimos até 14%.

Tabela 1. Desempenho do classificador SVM com kernel polinomial como função dos parametros C e d.

C	d = exponent	Histogramas de Projeção
1	1	14.5%
1	3	16.5%
1	9	31%
10	1	16%
10	3	16.5%
10	9	31%
100	1	16%
100	3	16.5%
100	9	31%

Tabela 2. Desempenho do classificador SVM com kernel RBF como função do parametro C e Gamma.

C	Gamma	Histogramas de Projeção
1	0.1	18%
1	0.01	46.5%
1	0.001	56.5%
10	0.1	15.5%
10	0.01	19%
10	0.001	44%
100	0.1	16%
100	0.01	14%
100	0.001	19%

Tabela 3. Desempenho do classificador RBF Neural Network como função do número de unidades ocultas.

RBF units	Histogramas de Projeção
2	21.5%
10	22.5%
20	22%
50	22%
70	22%

Tabela 4. Desempenho do classificador KNN como função do número de vizinhos utilizados na classificação.

K	Histogramas de Projeção
1	18.5%
3	18%
5	20%
7	24.5%
9	24.5%

7. Trabalhos Futuros

Adicionar módulos ao algoritmo de segmentação de linhas, para possibilitar a segmentação de linhas conectadas ou sobrepostas.

A análise de dados descrita nesse artigo foi feita utilizando dígitos que foram extraídos manualmente. Para validar os resultados obtidos dos métodos que foram explanados nesse artigo, seria necessário realizar experimentos de classificação que utilizassem os resultados desses algoritmos.

8. Conclusão

A análise comparativa entre os diferentes métodos de classificação, utilizando-se os Histogramas de Projeção como método de extração de características, é um ótimo parâmetro para os que pretendem classificar caracteres manuscritos. Essa análise completa trabalhos encontrados na literatura ao usar um outro método de extração de características [14].

O resultado do sliding window pode ser perfeitamente testado por uma rede neural adaptada para que os resultados que não delimitam caracteres sejam descartados. Em vista dos bons resultados obtidos desse algoritmo, é provável que bons resultados sejam obtidos da etapa de classificação utilizando esse método.

Além disso, em consonância com a metodologia e o espírito científicos, é esperado que os algoritmos aqui criados sejam utilizados por outros que tenham o interesse em solucionar o problema do reconhecimento de caracteres manuscritos, gerando avanços no estado da arte.

9. Referências

[1]PROHIST: http://recpad.dsc.upe.br/site_hist

[2] Cristianini, N. and Shawe-taylor, J. An introduction to Support Vector Machines, Cambridge University Press, 2000

[3] Webb, A. Statistical Pattern Recognition. John Wiley & Sons, second edition, 2002.

[4] Witten, I. H.; Frank, Eibe. Data Mining: Pratical Machine Learning Tools and Techniques. 2nd ed. San Francisco: Morgan Kaufmann, 2005.

[5] BALDISSEROTTO, Carolina. Técnicas de aprendizagem de máquina para previsão de sucesso em implantes dentários. 2005. 80 f. Trabalho de Conclusão de Curso (Graduação) – Engenharia da Computação, Departamento de Sistemas Computacionais, Recife 2005.

[6] S.Basu, C.Chaudhuri, M. Kundu, M. Nasipuri and D.K. Basu. Text line extraction from multi-skewed handwritten documents. Pattern Recognition, Volume 40, Issue 6, June 2007, Pages 1825- 1839.

[7] Brijesh Verma; Michael Blumenstein; Moumita Ghosh. A novel approach for structural feature extraction: Contour vs. direction. Pattern Recognition Letters, Volume 25, Issue 9, 2 July 2004, Pages 975-988

[8] Weliwitage, C; Harvey, A.L.; Jennings, B. Handwritten Document Offline Text Line Segmentation. Digital Image Computing: Tecniques and Applications, 2005. DICTA ‘ 05. Proceedings. Dec. 2005 Pages: 184 – 187

[9] OLIVEIRA, Adriano Lorena Inácio de. *Estimation of Software Project Effort with Support Vector Regression*. Neurocomputing, v. 69, n. 13-15, p. 1749-1753, 2006. http://dx.doi.org/10.1016/j.neucom.2005.12.119

[10] OLIVEIRA, Adriano Lorena Inácio de ; MEDEIROS, E. A.; ROCHA, T. A. B. V.; BEZERRA, M. E. R.; VERAS, R. C. *On the Influence of Parameter theta- on Performance of RBF Neural Networks Trained with the Dynamic Decay Adjustment Algorithm*. International Journal of Neural Systems, v. 16, p. 271-281, 2006. http://dx.doi.org/10.1142/s0129065706000676

[11] OLIVEIRA, Adriano Lorena Inácio de ; MEIRA, Silvio Romero de Lemos . *Detecting Novelties in Time Series through Neural Networks Forecasting with Robust Confidence Intervals*. Neurocomputing , In Press, 2006. http://dx.doi.org/10.1016/j.neucom.2006.05.008

[12] OLIVEIRA, Adriano Lorena Inácio de ; MEIRA, Silvio Romero de Lemos . *Improving RBF-DDA Performance on Optical Character Recognition through Weights Adjustment*. In: IEEE International Joint Conference on Neural Networks (IJCNN´2006), 2006, Vancouver - Canadá.

[13] MELLO, C. A. B.; SANCHEZ, A.; OLIVEIRA, Adriano Lorena Inácio de. *Image Thresholding of Historical Documents: Application to the Joaquim Nabuco's File*. In: 1st EVA conference in Vienna - EVA Vienna 2006, 2006, Vienna. proceedings of Proceedings of the Digital Cultural Heritage Conference, 2006.

[14] OLIVEIRA, Adriano Lorena Inácio de; MELLO, C. A. B. ; SILVA JR, E. ; ALVES, V. M. O. *Optical Digit Recognition for Images of Optical Digit Recognition for Images of Handwritten Historical Documents*. In: Simpósio Brasileiro de Redes Neurais (SBRN'2006), 2006, Ribeirão Preto - SP. Proc. of Brazilian Symposium on Neural Networks (SBRN'2006), 2006, IEEE Computer Society Press.

[15] Øivind Due Trier , Anil K. Jain, Torfinn Taxt. *Feature extraction methods for character recognition-A survey,* Pattern Recognition, Vol. 29, N. 4, pp. 641-662, 1996.