Técnicas em Processamento e Análise de Documentos Manuscritos

Alceu de Souza Britto Jr.[1]

Cinthia Obladen de Almendra Freitas [1]

Edson José Rodrigues Justino [1]

Díbio Leandro Borges [1]

Jacques Facon [1]

Flávio Bortolozzi [1]

Robert Sabourin [2]

Resumo:. Este tutorial tem como objetivo apresentar os principais processos envolvidos no Processamento e Análise de Documentos Manuscritos (PADM): aquisição, pré-processamento, segmentação, extração de características e reconhecimento. Inicialmente, são introduzidos os aspectos relevantes de cada processo de PADM, os diferentes tipos de documentos, as técnicas de filtragem e de segmentação necessárias à extração da informação de interesse. Descreve-se a conceituação e a formulação teórica dos Modelos Escondidos de Markov (MEM) e sua aplicabilidade em sistemas computacionais destinados ao reconhecimento de documentos manuscritos. O tutorial mostra o uso de

1 Programa de Pós Graduação em Informática Aplicada, PUCPR, Rua Imaculada Conceição 1155-Prado Velho – CEP: 80215-901 - {alceu, cinthia, dibio, justino, facon, fborto@ppgia.pucpr.br} – Curitiba (PR)

2 Ecole de Technologie Superieure – ETS – 1100, Rua Notre Dame – H3C1K3 – Montreal (QC) – Canadá – sabourin@livia.etsmtl.ca

MEM no reconhecimento de cadeias numéricas e palavras manuscritas, bem como, na verificação de assinaturas.

Abstract: This work aims to present the main modules of a typical Handwriting Document Processing and Analysis System (HDPAS): data acquisition, preprocessing, segmentation, feature extraction and recognition. First, the relevant aspects of each HDPAS process are given, different types of documents, filtering and segmentation techniques applied to extract the interested data. Then, the main Hidden Markov Models (HMM) concepts are described, as well as, the different types of HMMs and the algorithms required to implement this statistical technique for modeling handwriting text. Finally, some interesting HMM-based methods for handwritten numeral string recognition, handwritten word recognition and signature verification are briefly described.

1 Introdução

Análise e reconhecimento de imagens de documentos é uma área de grande relevância por suas inúmeras aplicações. Por possuir características próprias, cada documento naturalmente exige soluções computacionais específicas. Para uma hieraquização do problema adota-se a seguinte divisão de abordagens: nível inferior, intermediário e superior, conforme Figura 1. Percebe-se que ocorre uma redução progressiva da quantidade de dados manipulados à medida que se passa por níveis crescentes de abstração.

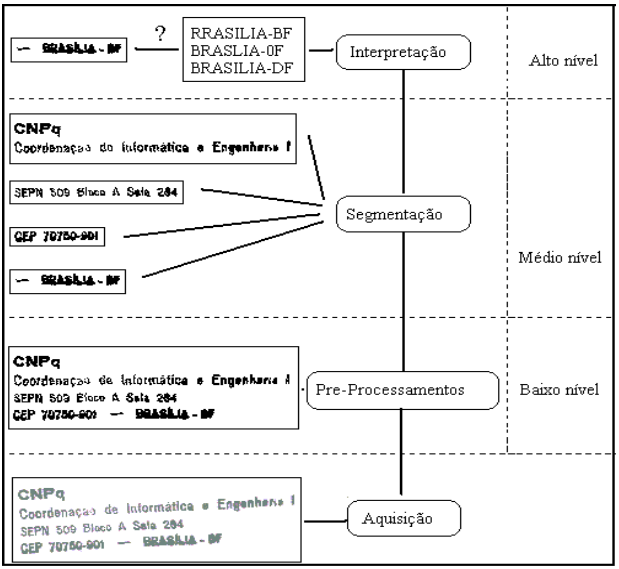


Figura 1. Etapas de processamento e análise de imagens para aplicações em envelopes postais

No nível inferior, os dados de entrada são os *pixels* da imagem original do documento e os dados de saída representam imagens tratadas, na forma de valores numéricos associados a cada *pixel*. Como por exemplo, imagens agora sem sujeiras e ruídos, eliminados por processos de filtragem e realce. No nível intermediário, os valores numéricos obtidos na fase anterior são utilizados na produção de uma lista de características visando a representação adequada de cada um dos campos de um documento. Finalmente, no nível superior produz-se, a partir destas características, uma interpretação do conteúdo de cada um dos campos através de métodos sintáticos, conexionistas e estocásticos.

Um sistema genérico para processamento e análise de imagens de documentos é composto pelos seguintes módulos:

- Aquisição e digitalização de imagens: consiste em transformar documentos em imagens digitais sob a forma de tabelas de valores chamados *pixels*;
- Pré-processamento: deve permitir o tratamento de problemas de inclinação, fundos ruidosos, rabiscos, dados subscritos, dados sublinhados, dentre outros;
- Segmentação: consiste em localizar automaticamente os campos relevantes do documento;
- Interpretação: representa a parte "inteligente" e deve permitir o reconhecimento automático do conteúdo de cada um desses campos.

Este trabalho encontra-se dividido em 7 seções. A Seção 2 apresenta as técnicas de segmentação e de extração de características. A Seção 3 descreve a formulação teórica dos Modelos Escondidos de Markov (MEM). As Seções 4, 5 e 6 apresentam o uso de MEM no reconhecimento de numerais, palavras manuscritas e assinaturas, respectivamente. Finalmente, a Seção 7 conclui o presente trabalho ressaltando os desafios encontrados na área de processamento e análise de documentos manuscritos.

2 Técnicas de segmentação e extração de características

A captura de imagens de documentos, mesmo realizada com o máximo cuidado, é um processo que introduz uma série de degradações provenientes [1]:

- Do próprio processo de digitalização: qualquer que seja a maneira empregada para digitalizar um documento (*scanner*, máquina fotográfica, etc), este sofre um processo de amostragem tanto das coordenadas espaciais quanto da intensidade. A conseqüência disto é que a qualidade da imagem ser sempre pior que a do documento original;
- Da qualidade precária de certos documentos: livros antigos, por exemplo;
- De falhas humanas: a mais comum delas é o posicionamento inadequado do documento no momento de sua digitalização.

Portanto, qualquer sistema de Gerenciamento Eletrônico de Documentos (GED) deve dispor de ferramentas que possam reverter o processo de degradação de maneira a resgatar o máximo de qualidade do documento. As técnicas de melhoria da qualidade de imagens podem ser divididas em duas famílias:

- Técnicas de restauração: a restauração procura obter a imagem original, tendo se possível, um conhecimento a priori da degradação. Sua ênfase está na modelagem de degradações e na recuperação de imagem por inversão do processo de degradação;
- Técnicas de realce: quando não é possível de antemão dispor do processo de degradação ou os padrões de degradação não podem ser avaliados.

Devido a dificuldade de se conhecer de antemão os processos de degradação e da complexidade computacional da modelagem destas degradações, faz-se pouco uso de técnicas de restauração. As técnicas de realce de imagens são relacionadas com a expansão

de contraste, realce de bordas e suavização. A maioria das técnicas de realce de imagens são heurísticas e orientadas para aplicações específicas. Com o realce de uma imagem, objetiva-se a obtenção de uma imagem apresentando bom contraste, contornos nítidos, riqueza de detalhes e pouco ruído. Devido a grande quantidade de técnicas de realce disponíveis na literatura serão apresentadas aqui as mais utilizadas para documentos.

2.1 Aprimoramento de contraste e detalhes

Um método bastante utilizado no caso de documentos é a equalização de histograma [1] por sua economia em termos de memória usada, por seu tempo de processamento reduzido, e por ser simples e eficiente. O histograma da imagem é reformulado em um histograma diferente, que possui a propriedade de distribuição uniforme objetivando assim a melhoria do contraste da imagem. A Figura 2 exemplifica o processo de equalização de histograma de uma imagem de cheque bancário de baixo contraste.

Figura 2. Exemplo de equalização de histograma

2.2 Redução de ruído

Devido ao aspecto digital das imagens, os filtros não lineares representam as técnicas mais eficientes de suavização de ruído, por exemplo o filtro da mediana [1], o filtro de Nagao-Matsuyama [2].

Filtro da mediana:

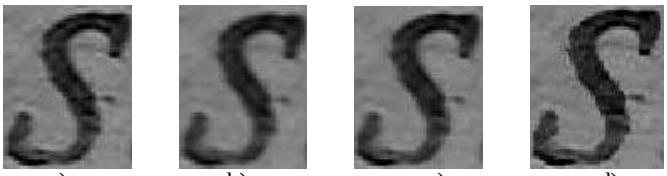
O filtro da mediana faz parte da família dos filtros de ordem e representa na sua categoria o filtro não linear mais fácil de se implementar, permitindo um ganho de qualidade mesmo em casos de documentos bastante ruidosos (Figura 3).

a) b)

Figura 3. Exemplo de filtragem não linear da mediana: a) imagem original; b) resultado

Filtro de Nagao-Matsuyama:

Nagao e Matsuyama [2] propuseram um método de suavização eficiente baseado em 9 máscaras, das quais 8 são assimétricas. Apesar de um esforço computacional elevado, esse filtro realiza o compromisso difícil de suprimir os pontos isolados, reduzir o ruído e ao mesmo tempo realçar as transições e a nitidez da imagem original (Figura 4).



a) b) c) d)

Figura 4. Comparação de filtragem: a) imagem original; b) média; c) mediana; d) Nagao-Matsuyama

2.3 Realce de bordas e detalhes

Qualquer captor introduz uma falta de nitidez mais visível especificamente pelas bordas e detalhes, que representam componentes de altas freqüências. Realçar bordas e detalhes faz-se, portanto, através da amplificação das altas freqüências. Não se deve esquecer que o ruído é também um componente de alta freqüência, um tratamento preliminar é necessário para evitar amplificar ao mesmo tempo ruído e bordas. Dentro das técnicas disponíveis na literatura, pode-se destacar os operadores de diferenciação lineares Gradiente Laplaciano e o Gradiente Morfológico [3] (Figura 5).

a) b) c) d)

Figura 5. Realce de bordas: a) imagem original; b) gradiente; c) gradiente morfológico; d) Laplaciano

2.4 Correção da inclinação

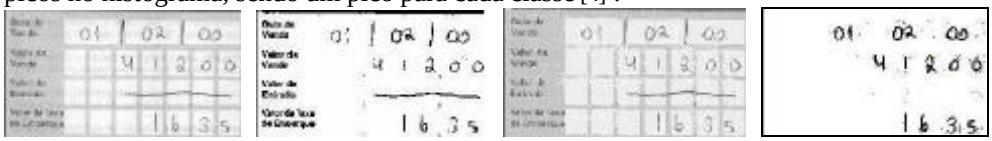
É muito comum, no caso de documentos, ter que corrigir problemas de inclinação provenientes de erros humanos no momento da aquisição, de características próprias de certos documentos como por exemplo livros antigos ou ainda da própria tendência do ser humano em escrever de forma inclinada. Existem muitas abordagens de correção deste tipo de problema como apresentado na Seção 5.

2.5 Segmentação

O objetivo da segmentação é obter, a partir de uma imagem, um conjunto de "primitivas" ou "segmentos significativos" que contém a informação semântica relativa à imagem. A grande dificuldade da segmentação reside no fato de não se conhecer de antemão o número e tipo de estruturas que se encontram na imagem. Essas estruturas são identificadas a partir da geometria, forma, topologia, textura, cor ou brilho sendo escolhidas aquelas que possibilitam a melhor identificação. As abordagens existentes na literatura são múltiplas, dentro das quais a segmentação por região, binarização e multi-binarização, contorno, textura, cor, modelos de Markov, redes neuronais artificiais, algoritmos genéticos, modelos de contornos ativos, morfologia matemática, baseada na teoria dos conjuntos nebulosos, ou ainda híbridas.

Segmentação por binarização e multi-binarização

Transformar uma imagem em níveis de cinza em uma imagem binária, processo chamado de binarização ou ainda de limiarização, é provavelmente no caso de documentos, uma das técnicas mais usadas. Isto porque se faz uso freqüente de processos de reconhecimento de caracteres que necessitam de imagens binárias. Toda técnica de binarização, baseada na noção de histograma, busca particionar uma imagem em duas classes C_0 e C_1, a partir de um limiar L. De forma ideal, o limiar L situa-se no vale entre os dois picos no histograma, sendo um pico para cada classe [4] .



a) b) c) d)

Figura 6. Exemplo de binarização: a) imagem original; b) binarização global; c) multi-binarização; d) extração automática de dígitos por morfologia matemática

Inúmeras técnicas já foram propostas, porém nenhuma delas serve para qualquer tipo de documento [4]. Isto porque, apesar da simplicidade do processo, dependendo da qualidade do original, podendo haver buracos nas linhas, borda rompida na região limítrofe ou região estranha de *pixels*, não se pode nunca esperar resultados perfeitos.

Quando se usa um único valor de limiar para toda a imagem, a binarização é dita global [5]. Quando se busca um limiar analisando intensidades de níveis de cinza dentro de janelas deslizando na imagem, o processo é chamado de binarização adaptativa [6]. Existe ainda uma forma mais completa porém mais complexa que consiste em agrupar os *pixels* da imagem não somente em duas classes, mas em um número maior de classes pertinentes. Este tipo de binarização chama-se de multi-binarização ou ainda de binarização multimodal [7], [8]. As Figuras 6-b) e 6-c) ilustram o exemplo de uma binarização global e de uma multi-binarização de uma imagem complexa de recibo de cartão de crédito. A imagem foi

automaticamente particionada em 5 classes, resultando numa imagem muito parecida com a imagem original.

Segmentação morfológica

Uma outra abordagem de filtragem não linear que está propiciando grandes avanços em termos de segmentação de documentos é a morfologia matemática. A qual consiste em extrair uma informação relativa à geometria e à topologia de um conjunto desconhecido de uma imagem pela transformação a partir de um outro conjunto perfeitamente definido chamado de *elemento estruturante* [3], [9]. Enquanto as ferramentas acima citadas processam unicamente imagens em níveis de cinza, a morfologia matemática processa da mesma maneira imagens em níveis de cinza e binárias. Esta abordagem é uma das poucas ferramentas que tem êxito no processamento de documentos complexos [3], [9]. A Figura 7 ilustra a remoção automática do fundo complexo de um cheque bancário a partir da ferramenta denominada *Fillhole*. A Figura 8 mostra outras aplicações da morfologia matemática na extração e localização de dados em imagens.

a) b)

Figura 7. Exemplo de processo morfológico: a) imagem original; b) resultado do operador *Fillhole*

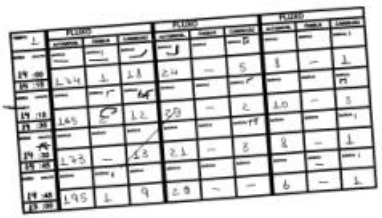
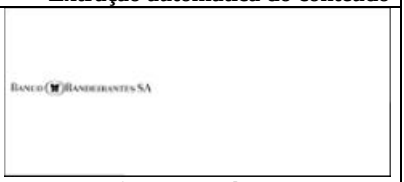
Exemplos	Resultados de Segmentação	
Extração automática de dados manuscritos em formulário [10]	 **Formulário original inclinado**	**Extração automática do conteúdo**
Extração automática do logotipo [11]	**Cheque bancário**	 **Extração automática do logo**

Figura 8. Exemplos de localização e extração de dados em imagens

3 Modelos escondidos de Markov (hidden Markov models) para reconhecimento

Modelos de Markov são representações utilizadas para se modelar um sinal através de uma seqüência de observações. Em uma Cadeia de Markov supõe-se uma fonte gerando tais saídas observáveis, denominada de Fonte de Markov. Os símbolos gerados a partir dessa fonte são dependentes apenas de observações anteriores, as quais foram geradas da mesma forma e assim sucessivamente. O número de seqüências anteriores consideradas para gerar uma saída é conhecido como ordem da Cadeia de Markov. Na maioria das aplicações conhecidas cadeias de primeira e segunda ordem são suficientes, mesmo porque a complexidade computacional cresce exponencialmente a partir de então.

Cada estado de uma Cadeia de Markov representa uma observação/símbolo de um evento físico correspondente, o que proporciona computar a partir de uma dada seqüência de símbolos quais foram os estados que geraram tal seqüência. Contudo, em aplicações reais mais de um símbolo pode ser observado por estado e a origem dessas observações torna-se imprecisa e dependente.

Em um Modelo Escondido de Markov (MEM) cada estado representa uma probabilidade sobre todos os símbolos, por isso a denominação "escondido" pois é o conjunto dos símbolos que está representado. A estrutura restante do modelo é a mesma de uma Cadeia de Markov. Um MEM, portanto, possibilita computar a seqüência de estados com maior probabilidade de ter gerado a seqüência observada de símbolos. Referências detalhadas sobre MEM e algumas aplicações podem ser encontradas em [12].

3.1 Definição de MEM

Um MEM discreto pode ser descrito por $\lambda = (Q, V, A, B, \pi, T)$, onde:

- T : comprimento da seqüência de observações,
- $Q = \{q_1, q_2, ..., q_N\}$: o conjunto de estados do modelo, q_t designará o estado da cadeia no instante t, e N é o número de estados do modelo,
- $V = \{v_1, v_2, ...v_m\}$: conjunto de símbolos de observações possíveis. O_t designará o símbolo da observação ao instante t,
- M = número de símbolos da observação,
- $A = \{a_{ij}\}_{1 \pounds i, j \pounds N}$: matriz de transição de estados. O elemento a_{ij} corresponde a probabilidade de transição do estado q_i para o estado q_j:

$$a_{ij} = pr\left(Q_{t=1} = q_j / Q_t = q_i\right) \qquad \forall i, j \in [1...N], \forall t \in [1, T-1] \qquad \textbf{(1)}$$

com as seguintes restrições:

$$a_{ij} \geq 0 \quad \forall i, j \quad e \quad \sum_{j=1}^{N} a_{ij} = 1 \qquad \textbf{(2)}$$

- B = $\{b_{jk}\}_{1£j£N;\ 1£k£M}$: matriz de probabilidade de observação de símbolos condicionada pelos estados da cadeia. O elemento b_{jk} representa a probabilidade de observar o símbolo V_k quando o modelo se encontra no estado q_j:

$$b_j = pr\left(O_t = v_k / Q_t = q_j\right) \qquad 1 \leq j \leq N \qquad 1 \leq k \leq M \qquad \textbf{(3)}$$

com as seguintes restrições:

$$b_{jk} \geq 0 \quad \forall j, k \quad e \quad \sum_{k=1}^{M} b_{jk} = 1 \quad \forall j \qquad \textbf{(4)}$$

- p = $\{p_i\}_{1£i£N}$: conjunto de densidades de probabilidade inicial, p_i representa a probabilidade que o processo de Markov inicie no estado q_i:

$$p_i = pr\left(Q_1 = q_i\right) \qquad 1 \leq i \leq N \qquad \textbf{(5)}$$

$$p_i \geq 0 \quad \forall i \quad e \quad \sum_{j=1}^{N} p_i = 1 \qquad \textbf{(6)}$$

A avaliação desses processos de Markov é inteiramente determinada pela probabilidade inicial e pela probabilidade das transições sucessivas. O modelo de Markov assim definido é dito de primeira ordem.

3.2 Tipos de MEM

A determinação do MEM ótimo é uma tarefa que depende estritamente da estrutura do modelo e do número de estados escolhidos. De um modo geral existem 3 tipos de estruturas para MEM:

- os modelos sem restrição, ditos ergóticos,
- os modelos seqüenciais e
- os modelos paralelos.

Os exemplos típicos desses modelo estão representados na Figura 9. Nos métodos sem restrição, Figura 9a, todas as transições possíveis entre os estado da cadeia são autorizadas. Isto é possível, se não se fizer restrição a nenhum dos valores a_{ij} igual nulo.

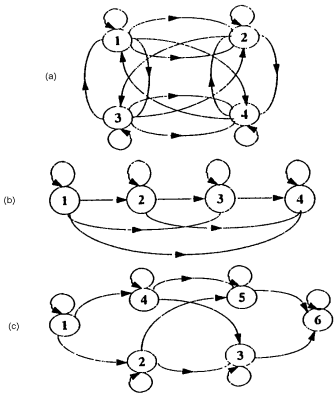


Figura 9. Tipos de estrutura dos modelos de Markov: a) modelo sem restrições; b) modelo seqüencial; c) modelo paralelo [13]

Os modelos seqüenciais e os modelos paralelos fazem parte dos modelos que seguem a *Topologia de Bakis*, ou ainda, denominados de "left-right" ou "esquerda-direita". Para esses modelos, a matriz de transição de estados é triangular superior. Os modelos seqüenciais, Figura 9b, funcionam segundo uma evolução em série do modelo através de seus estados, mesmo que qualquer um desses estados possam ser saltados no curso do processo. Para os modelos paralelos, Figura 9c, muitos caminhos através da rede de Markov são permitidos, sabendo que cada um desses caminhos pode saltar um ou vários estados do modelo.

Cada uma das estruturas dos modelos da Figura 9 podem ser generalizadas por incluir um número de estados arbitrário. Entretanto, o número de parâmetros a estimar em um modelo de Markov é da ordem de N^2 (para a matriz A) mais NxM (para a matriz B). Por conseqüência, se N é muito grande, uma determinação precisa das matrizes A e B ótimas vem a ser muito difícil de realizar por uma base de treinamento de tamanho limitado. Não existem meios teóricos pré-determinados de maneira a escolher o número N de estados necessários no modelo, uma vez que os estados nem sempre estão fisicamente ligados aos fenômenos observáveis.

Além da variabilidade do tipo da estrutura do MEM, esses ainda podem ser classificados segundo o endereço onde são colocadas as observações em: a) modelos de estados e b) modelos de arcos.

A Figura 10a mostra um exemplo de modelo de estados, enquanto a Figura 10b mostra um exemplo de modelo de arcos. No caso dos modelos de arcos, as duas probabilidades, a_{ij} e $b_j(k)$ consideradas no modelo de estados são substituídas por $pr_{ij}(k)$ que corresponde a probabilidade de transição do estado q_i ao estado q_j emitindo o símbolo da observação v_k.

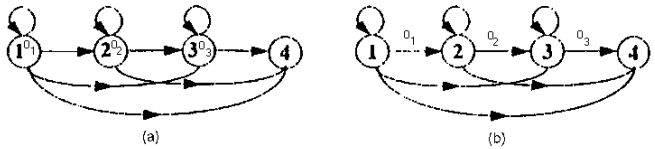


Figura 10. Modelo de estados e modelos de arcos: a) modelo de estados; b) modelo de arcos [13]

3.3 Treinamento: algoritmo *Baum-Welch*

O algoritmo de Baum-Welch busca treinar os modelos por máxima verossimilhança. Devido ao grande número de fatores de variabilidade presentes em textos manuscritos, a vantagem deste algoritmo é que ele considera todos os alinhamentos possíveis e maximiza a verdadeira probabilidade de emissão das observações.

Dada uma estimação qualquer λ do modelo, e uma seqüência de observações O, as melhores estimativas dos parâmetros do novo modelo λ^{NM} são dadas por:

$$a_{ij}^{NM} = \frac{P(\text{transição do estado } q_i \text{ ao estado } q_j / O, l)}{P(\text{transição do estado } q_i \text{ a um estado qualquer} / O, l)} \quad \textbf{(7)}$$

$$b_{jk}^{NM} = \frac{P(\text{emissão do símbolo } v_k \text{ ao estado } q_j / O, l)}{P(\text{emissão de um símbolo qualquer a paritr do estado } q_j / O, l)} \quad \textbf{(8)}$$

$$p_i^{NM} = P(\text{seqüência de observação começando pelo estado } q_i / O, l) \quad \textbf{(9)}$$

O interesse do algoritmo de Baum-Welch reside no fato de que as fórmulas de reestimação de A,B e π acima, garantem o acréscimo de *P(O/l)* até que um ponto de convergência seja atingido. Uma descrição detalhada deste algoritmo pode ser encontrada em [12].

3.4 Reconhecimento

O algoritmo de *Viterbi* permite encontrar a melhor seqüência de estados, $q = (q_1\, q_2 \ldots q_T)$, para um dada seqüência de observações $O = (o_1, o_2, \ldots, o_T)$. Para tal, deve-se definir a quantidade:

$$d_t(i) = \max_{q_1, q_2, \ldots, q_{t-1}} P[q_1 q_2 \ldots q_{t-1}, q_t = i, o_1 o_2 \ldots o_t \mid l] \quad \textbf{(10)}$$

onde $d_t(i)$ é a mais alta probabilidade ao longo de um caminho simples, no tempo *t*, com valores para as primeiras *t* observações terminando no estado *i*. Por indução tem-se que:

$$d_{t+1}(j) = [\max_i d_t(i) a_{ij}] b_j(O_{t+1}) \quad \textbf{(11)}$$

Para realmente recuperar a seqüência de estados, faz-se necessário continuar propagando o argumento maximizado das equações (12) e (13), para cada t e j. Pode-se fazer isso através de um vetor $Y_t(j)$. O procedimento completo para encontrar a melhor seqüência de estados é apresentado a seguir.

Assim, no processo para encontrar o caminho máximo provável para a seqüência de estados, o cálculo de $d_t(j)$ para $1 £ j £ N$, usando a recursão $d_1(j)$, mantendo-se sucessivamente apontado o estado ótimo, dentro do máximo encontrado nas operações.

INICIALIZAÇÃO

$$d_1(i) = p_i b_i(o_1) \qquad 1 £ i £ N \quad \textbf{(12)}$$

$$Y_1(i) = 0 \quad \textbf{(13)}$$

INDUÇÃO

$$d_t(j) = \max_{1 \le i \le N} [d_{t-1}(i) a_{ij} b_j(o_t)], \qquad 2 \le t \le T \qquad 1 \le j \le N \quad \textbf{(14)}$$

$$y_t(j) = \arg\max_{1 \le i \le N} [d_{t-1}(i) a_{ij}], \qquad 2 \le t \le T \qquad 1 \le j \le N \quad \textbf{(15)}$$

TÉRMINO

$$q_T^* = \arg\max_{1 \le i \le N} [d_T(i)] \quad \textbf{(16)}$$

$$P^* = \max_{1 \le i \le N} [d_T(i)] \quad \textbf{(17)}$$

CAMINHO DE RETORNO (BACKTRACKING)

$$q_t^* = y_{t+1}(q_{t+1}^*), \qquad t = T-1, T-2, \ldots, 1 \quad \textbf{(18)}$$

4 Reconhecimento de numerais manuscritos usando MEM

O reconhecimento de cadeias numéricas manuscritas tem recebido atenção especial nos últimos anos graças ao número potencial de aplicações práticas, tais como: a leitura de códigos postais em envelopes, o reconhecimento da quantia numérica em cheques bancários, o processamento de formulários de impostos e recenseamento, dentre outras. Na literatura é possível encontrar diferentes abordagens que tratam desse problema [14], [15],

[16],[20],[17], onde o desafio consiste em reconhecer cadeias numéricas manuscritas de tamanho desconhecido *a priori*.

Recentemente em [18] e [19] constata-se o benefício do uso da segmentação implícita baseada em MEM no reconhecimento de textos manuscritos. A Figura 11 mostra a arquitetura do método proposto em [14], onde dois conjuntos de modelos de Markov são utilizados com o objetivo de encontrar um melhor compromisso entre os processos de segmentação e reconhecimento. Um primeiro conjunto é utilizado para definir *N* caminhos de segmentação-reconhecimento para uma determina cadeia numérica. Em seguida, um segundo conjunto de MEMs considerando apenas aspectos relacionados ao reconhecimento de dígitos isolados, é usado em um processo de verificação. O método é composto de dois estágios denominados EBCCN (Estágio Baseado no Contexto de Cadeias Numéricas e EV (Estágio de Verificação). No primeiro, um processo baseado em programação dinâmica relaciona MEMs, representado as 10 classes de dígitos, com a cadeia sendo processada. Os MEMs ($l_c^0, l_c^1, ..., l_c^9$) utilizados nesse estágio são treinados com dígitos isolados, porém considerando-se informação de contexto referente à variação de inclinação vertical e tamanho de cadeias numéricas, bem como, ao espaço entre os dígitos na cadeia. As características extraídas a partir do primeiro plano da imagem (*pixels* pretos) contemplam simultaneamente segmentação e reconhecimento.

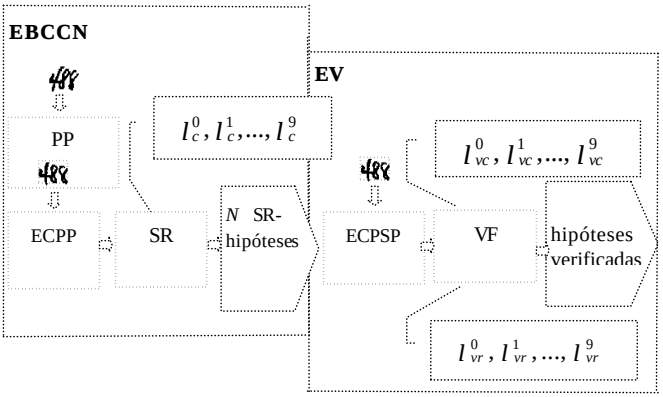


Figura 11. Arquitetura do sistema

O estágio de verificação reordena as *N* melhores hipóteses de segmentação-reconhecimento fornecidas pelo primeiro estágio. Para este propósito, um classificador de dígitos baseado em MEM é treinado com dígitos isolados sem considerar informação de contexto. Um novo conjunto de características baseado em *pixels* do primeiro e segundo plano da imagem é usado para melhorar o desempenho dos MEMs em termos de reconhecimento. Além disso, outros 10 MEMs ($l_{vr}^0, l_{vr}^1, ..., l_{vr}^9$) baseados nas linhas da imagem do dígito são combinados com os MEMs baseados em colunas ($l_{vc}^0, l_{vc}^1, ..., l_{vc}^9$) visando uma melhor representação das classes de dígitos.

A Figura 15 mostra exemplos de cadeias numéricas utilizadas em [14] na avaliação do método descrito anteriormente.

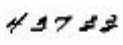
Acertos		Erros		
Amostra	Resp. sistema	Amostra	Resp. correta	Resp. sistema
	25		84297	84187
	43733		90898	90398
	22699		13210	13211
	76540		45537	95537
	6001		60170	68170

Figura 15. Acertos e acertos no reconhecimento de cadeias numéricas

5 Reconhecimento de palavras manuscritas usando MEM

A maioria dos métodos de reconhecimento da escrita cursiva tem como base um léxico de palavras. Portanto, visa-se encontrar uma solução entre uma lista de palavras contidas no léxico ou rejeitar a palavra recebida como entrada. Nestas condições, quanto maior o tamanho do léxico, maior a complexidade e maior a lista de soluções possíveis, uma vez que existe uma probabilidade maior de existirem palavras parecidas em um léxico de grandes dimensões, por exemplo: nomes de ruas e cidades em endereços postais. Os cheques bancários são um exemplo de léxico de pequenas dimensões.

5.1 Etapas necessárias ao reconhecimento de palavras manuscritas

As etapas necessárias ao reconhecimento de palavras manuscritas foram mencionadas anteriormente neste artigo [21],[22],[23]. Descreve-se então as etapas de Pré-processamento, Estratégia de Segmentação, Seleção e Extração de primitivas, e o Reconhecimento propriamente dito para palavras manuscritas.

Pré-processamento

O pré-processamento das imagens é composto, geralmente, pelas seguintes etapas:

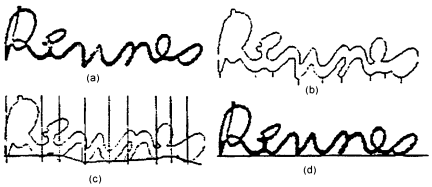
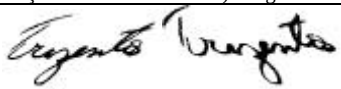
- **Correção da linha de base da escrita (*skew*):** visa detectar e corrigir o ângulo de inclinação com a horizontal de modo que deixar a escrita horizontal. De um modo geral, os métodos de correção de linha de base são globais (considerando que a inclinação é válida para a palavra como um todo [21],[24],[25]) ou locais

(considerando que a inclinação horizontal não é constante e igual para toda a palavra, realizando pequenas correções localizadas [13]).

- **Correção da inclinação vertical dos caracteres (*slant*):** inclinação vertical dos caracteres é o ângulo formado entre o eixo da direção de escrita dos caracteres e o eixo vertical. Já que a maioria das primitivas são sensíveis à inclinação vertical, corrigir este ângulo torna a palavra mais vertical, reduzindo assim a variabilidade da escrita e simplificando a etapa de reconhecimento. As técnicas mais simples supõem que esta inclinação é uniforme para uma mesma palavra, realizando uma correção global [21],[26],[27]. Outras técnicas trabalham isolando segmentos de palavra e efetuando a correção individual de cada segmento [22].

- **Detecção e normalização do corpo das palavras:** corpo das palavras é a parte formada pelas letras minúsculas. As técnicas diferenciam-se no fato de considerar o corpo das palavras uma faixa necessariamente horizontal [22] e [13] ou não [28]. A detecção do corpo da palavra não é somente importante para a etapa de normalização do corpo das palavras, mas também para a determinação das zonas principais de uma palavra. Estas zonas são 3 ou mais áreas, denominadas geralmente de: ascendente, corpo e descendente. Nestas áreas são extraídas as primitivas que representam as palavras propriamente ditas.

A Figura 16 exemplica as etapas de pré-processamento para diferentes técnicas aplicados por diferentes autores.

Exemplos	Resultados de Pré-Processamento de Palavras Manuscritas
Correção da Linha de Base [13]	 a) imagem original, b) detecção dos mínimos locais do contorno inferior e filtragem dos mínimos, c) aproximação da linha de base e d) imagem corrigida
Correção da Inclinação Vertical dos Caracteres [29]	

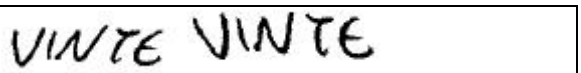
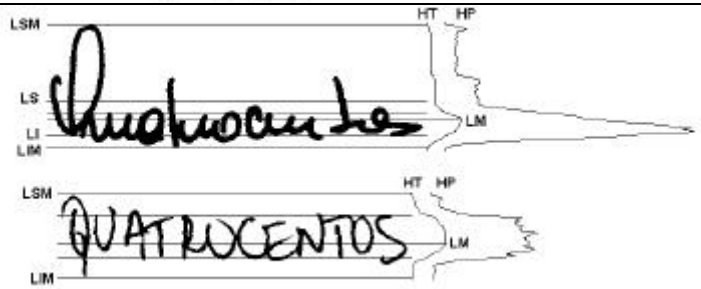
	
Detecção do Corpo das Palavras [29]	 HT = Histograma de Projeção Horizontal das Transições B&P HP = Histograma de Projeção Horizontal de Densidade de *Pixels* LM = Linha Média LSM, LS, LI, LIM = Limite Superior Máximo da palavra, Limite Superior e Limite Inferior do corpo da palavra e Limite Inferior Mínimo da palavra

Figura 16. Exemplos das etapas de pré-processamento de palavras manuscritas.

As aplicações na área de PADM combinam os diferentes tipos de primitivas visando obter métodos mais robustos através de um conjunto pequeno de primitivas porém, com alta representatividade do problema [22],[30]. Portanto, diferentes tipos de formas precisam de diferentes tipos de primitivas, que por sua vez precisam de diferentes tipos de classificadores.

A Figura 18 exemplifica a extração de primitivas para palavras do extenso de cheques bancários para uma abordagem global, sem segmentação explícita, ou seja, trata cada palavra como uma unidade para o reconhecimento [29]. O alfabeto de símbolos básicos é composto por 29 símbolos utilizados na composição dos grafemas. O caracter # representa um separador entre grafemas [29].

Figura 18. Extração de primitivas: a) primitivas perceptivas (PP), b) primitivas perceptivas, concavidades e convexidades rotuladas(PPCCR) [29]

5.2 Reconhecimento de palavras manuscritas usando MEM

Os Modelos Escondidos de Markov – MEM oferecem um modelo formado pela utilização conjunta de métodos estruturais de representação das formas a reconhecer e de um

método estatístico para treinamento dos modelos e reconhecimento das palavras propriamente ditas.

O interesse dos MEM reside na sua possibilidade de modelar eficazmente diferentes fontes de conhecimento. Seu ponto forte está primeiro no fato, da integração coerente dos diferentes níveis de modelagem: morfológico ou traçado da palavra, léxico ou vocabulário de palavras estudadas e sintático ou o modelo da palavra propriamente dita. E segundo, na existência de algoritmos eficazes para determinar o valor dos parâmetros fornecendo a melhor adequação entre o modelo e a base de dados (conhecimento), chamada de base de treinamento.

Topologia e algoritmos empregados

Os MEM utilizados são do tipo discreto, de topologia esquerda-direita (*left-rigth* ou *Bakis Topology)*, com passo igual a 2. Isto, deve-se ao fato de que as palavras são assim melhor modeladas e não se caracterizam, por exemplo, como modelos ergóticos. Outro detalhe da topologia é que o último estado não permite o auto-ciclo, ou seja, não se permite a permanência neste estado. Assim, o modelo não converge rapidamente para o último estado permanecendo aí até o fim das observações. Evita-se que o último estado interfira de maneira prejudicial no treinamento do modelo.

O treinamento dos modelos considera o algoritmo de Baum-Welch [19], realizando o processo de *Cross-Validation* [29]. A partir das seqüências de observações obtidas para cada uma das palavras ou letras das bases de dados de treinamento e validação, desenvolve-se um modelo para cada classe de palavra ou letra. O reconhecimento utiliza o algoritmo *Forward* [19] sobre a base de dados designada para teste.

A Figura 19 mostra alguns exemplos de confusão no reconhecimento de palavras do extenso manuscrito de cheques bancários brasileiros pelo método desenvolvido por [29],[31].

Figura 19. Exemplos de confusão no reconhecimento do extenso de cheques bancários [29]

6 Verificação de assinaturas usando MEM

Nos métodos de verificação de assinaturas manuscritas estáticas, a assinatura é adquirida através de uma imagem [32]. Esta imagem representa o estilo de escrita do autor, extensivamente descrita pela grafoscopia [33]. O objetivo desses métodos é determinar três tipos básicos de falsificações relacionadas com as variabilidades intra e interpessoais. O primeiro tipo, chamado de falsificação aleatória, é usualmente representado por um espécime de assinatura de um autor diferente do modelo, Figura 20b. O segundo chamado de falsificação simples, é representado por um espécime de assinatura com a mesma forma do

nome do autor do modelo, Figura 20c. O último tipo é chamado de falsificação servil, representado por um espécime similar à assinatura do modelo, Figura 20d.

(a) (b)

(c) (d)

Figura 20. a) assinatura verdadeira; b) falsificação aleatória; c) falsificação simples; d) falsificação servil

Cada tipo de falsificação requer o uso de diferentes características para a verificação. Os métodos baseados na abordagem estática são usualmente utilizados para identificar falsificações aleatórias e simples. Esses métodos mostram-se mais adequados para a descrição de características relacionadas como a forma geral das assinaturas. Para esse fim, a abordagem grafoscópica apresenta um conjunto relevante de características que podem ser utilizadas [33].

6.1 Métodos de segmentação de assinaturas

São duas as abordagens principais adotadas na segmentação de assinaturas. A primeira é a abordagem contextual, que consiste na identificação e utilização das letras que constituem o nome do autor do modelo da assinatura. A segunda é a abordagem não contextual, que utiliza características relacionadas com a forma dos traços das mesmas, levando-se em consideração aspectos geométricos e estatísticos desses traços. A vantagem da segmentação não contextual em relação à contextual reside na sua adaptação no tratamento das rubricas, um dos tipos de assinatura freqüentemente utilizado. A segmentação não contextura pode ser efetuada utilizando diferentes abordagens [35], entre elas encontra-se a segmentação por grade ou *grid*. Esse tipo de abordagem sido freqüentemente utilizada [32],[33],[34],[36]. Esse trabalho demonstra como a abordagem usando *grid* é adaptada à estabilidade das partes das assinaturas, Figura 21. Com a utilização do *grid*, é possível incorporar um subconjunto de características estáticas do grafismo (grafologia), através da análise da ocupação do espaço gráfico, tais como:

- Calibre: O calibre introduz duas importantes características de avaliação global da assinatura, o comprimento e a altura da assinatura, figura 22a;
- Proporção: A proporção permite avaliar a regularidade geométrica na forma geral da assinatura, Figura 22b;

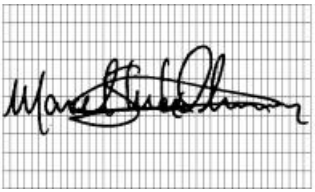


Figura 21. Segmentação por grid

Figura 22. a) calibre; b) proporção; c) espaçamento; d) comportamento base

- Espaço: O espaço revela o tamanho e quantidade de espaços em branco entre os blocos da assinatura, Figura 22c;
- Inclinação: A inclinação descreve o ângulo de inclinação da assinatura em relação à linha de base, Figura 22d.

6.2 Extração de características usando *grid*

Cada coluna de células do *grid* é convertida em um vetor de características, onde cada elemento possui um ou mais valores numéricos representativo, dependendo da característica usada. Para esse exemplo foram usadas duas características estáticas e uma pseudodinâmica. Na a extração das características estáticas, foi usada uma imagem binária e na pseudodinâmica foi usado o núcleo da imagem da assinatura, obtido através de um processo morfológico de afinamento [35]. O conjunto de características estáticas é formado pela densidade de *pixels*, que representa a quantidade de *pixels* pretos contida em cada célula do *grid* e pela distribuição de *pixels*, que representa a distribuição geométrica dos *pixels* em cada célula do *grid*. Essa última deriva da técnica chamada *Extended-Shadow-Code* (*ESC*) [34] que para esse propósito, utiliza os *pixels* pretos projetados em quatro sensores localizados nas faces de cada célula, a partir de um sistema de eixos central. Cada sensor produz um valor numérico que corresponde ao total de *pixels* projetados. Esse valor numérico e posteriormente normalizado pelo comprimento do sensor, Figura 23.

Como característica pseudodinâmica foi usada a inclinação axial, representada pela inclinação predominante do segmento mais significativo de cada célula, determinada através contagem do número de ocorrências de cada um dos elementos estruturantes que descrevem os tipos básicos de inclinação, Figura 24 [33].

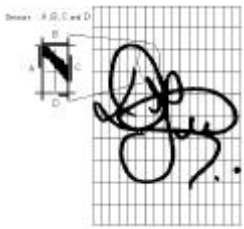


Figura 23. A distribuição de *pixels*

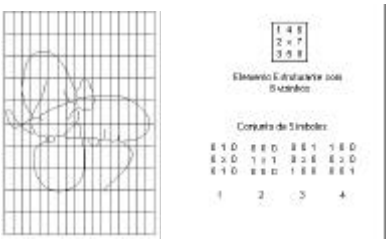


Figura 24. A inclinação axial

6.3 Geração das seqüências de observações/aprendizado/verificação

Após o processo de extração de características, é gerado um conjunto de *codebooks* para cada característica. Para esse fim, é usado um processo de Quantização Vetorial (QV) [32],[33],[20], baseado no algoritmo *K-means*. O resultado é uma seqüência de códigos, que correspondem à seqüência de observações, Figura 25.

Na fase de aprendizado, é gerado um modelo de MEM para assinatura, adaptado a cada autor. Posteriormente, é então utilizada a validação cruzada, para definir dinamicamente o melhor número de estados para cada autor (modelo $\lambda = \{A,B,\pi\}$). A topologia selecionada foi a Bakis, pois ela representa adequadamente as características da escrita Latina. A melhor probabilidade de validação $p_{cv}\ (O/l)$ é usada para definir os parâmetro do limiar, Figura 26. O objetivo é determinar o limiar de aceitação e rejeição levando em consideração um autor específico [33].

Figura 25. Uma seqüência de observações

O limiar médio na apresentado na Equação (33), definido por p_{tn}, representa o logaritmo da probabilidade de aprendizado, normalizado pelo número de seqüências de observações L.

No procedimento de verificação de assinatura, é usado o algoritmo *Forward* [19],[33], onde o logaritmo da probabilidade p_v,é também normalizado por L. A aceitação ou rejeição de uma assinatura é definida pela Equação (37), Figura 27.

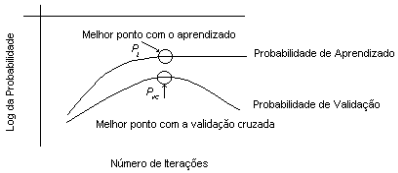


Figura 26. Curvas de aprendizado e validação

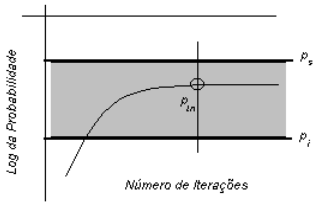


Figura 27. Limiares de aceitação e rejeição

$$p_{tn}(O/l) = \frac{\log p_t(O/l)}{L} \quad \textbf{(18)}$$

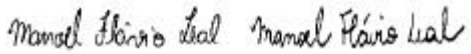
$$p_i = p_{tn} - (p_{tn}.\alpha_1) \quad \textbf{(19)}$$

$$p_s = p_{tn} + (p_{tn}.\alpha_2) \quad \textbf{(20)}$$

$$p_{vn}(O/l) = \frac{\log p_v(O/l)}{L} \quad \textbf{(21)}$$

$$p_s \pounds p_{vn} \pounds p_i \quad \textbf{(22)}$$

O tipo de falsificação de maior ocorrência em cheques bancários, por exemplo, são as falsificações simples [35]. A Figura 28 apresenta um exemplo de falsificação simples detectada pelo método.



(a) (b)

Figura 28. a) Assinatura genuína; b) Falsificação simples

7 Conclusões

Este tutorial apresenta metodologias de processamento e análise de documentos manuscritos, desde o pré-processamento, a segmentação, a extração de características até o

reconhecimento. Através de aplicações reais como o reconhecimento de cadeias numéricas, palavras manuscritas, bem como, a verificação de assinaturas, pode-se perceber a dificuldade da área e também conhecer algumas das soluções existentes.

A complexidade das soluções apresentadas no processo de reconhecimento de documentos manuscritos demonstram que a variabilidade da escrita humana é muito maior que se imagina e que mais estudos são e serão necessários para resolver os desafios do processamento e análise de documentos manuscritos.

Referências

[1] R.C. Gonzales & P. Winz, *"Digital Image Processing. 2nd Edition"*,Addison-Wesley Publishing Company, 1987.

[2] M. Nagao & T. Matsuyama", "Edge preserving smoothing", *Computer Graphics and Image Processing*", Vol. 13, 1979, pp. 394-407.

[3], J.Facon, *"Morfologia Matemática: teoria e exemplos"*, Editora Champagnat, Curitiba, 1996.

[4] L. O´Gorman & R. Kasturi,*"Pixel-level processing in Document Image Analysis*",IEEE Computer Society Press, 1995.

[5] N. Otsu, "A Threshold Selection Method from Gray-level Histograms",*IEEE Transactions on Systems, Man and Cybernetics"*, Vol. 9, No. 1, Abril de 1979, pp. 62-66.

[6], L. Eikvil & T. Taxt & K. Moen, "A Fast Adaptive Method for Binarization of Document Images", *Proc First Int´l Conf Document Analysis and Recognition*, France, 1991, pp. 435-443.

[7] P.R. Martins, *"Segmentação de Histogramas Multimodias: Simulações, Implementações e Aplicação em Cheques Bancários Brasileiros*", Dissertação de Mestrado, PUCPR-Curitiba, novembro de 2000.

[8], J.C. Yen & F.J. Chang & S. Chang, "A New Criterion for Automatica Multilevel Thresholding", *IEEE Transactions on Image Processing*, Vol.4 No. 4, março 1995, pp. 370-378.

[9] S. Beucher & F. Meyer, "The Morphological Approach to Segmentation:The Watershed Transformation", *Mathematical Morphology in Image Processing*, E. Dougerthy, Rochester Institute of Technology, 1993.

[10] L. A Neves, *"Extração De Células De Dados Manuscritos Em Tabelas*", Dissertação de Mestrado, PUCPR-Curitiba, 1999.

[11] A de Jesus, *"Uma Abordagem Morfológica Para a Segmentação se Logotipos em Cheques Bancários Brasileiros sem Conhecimento a Priori"*. Dissertação de Mestrado, PUCPR-Curitiba, 1999.

[12] Rabiner, L. R. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE*, Vol. 77, No. 2, pp.257-286, Feb. 1989.

[13] A.E. Yacoubi, *"Modélisation markovienne de l'écriture manuscrite. Application à la reconnaissance des adresses postales"*, Thèse de doctorat, Universite de Rennes, France,(1996). 307p.

[14] Britto A.S., Sabourin R., Bortolozzi F., and Suen C.Y. A two-stage HMM-based system for recognizing handwritten numeral strings. *Proceedings of the International Conference in Document Analysis and Recognition (ICDAR´2001)*, Seattle, USA, 11-13 September 2001.

[15] Shi Z., Srihari N. Shin Y.C. and Ramanaprasad V. A System for Segmentation and Recognition of Totally Unconstrained Handwritten Numeral Strings. *ICDAR'97*, Ulm, Germany, 18-20 August 1997, Vol 2, pp. 455-458.

[16] Ha T.M., Zimmermann M. and Bunke H. Off-Line Handwritten Numeral String Recognition by Combining Segmentation-Based and Segmentation-Free Methods. *Pattern Recognition*, Vol. 31, No. 3, 257-272, 1998.

[17] Gader P.D., Keller J.M., Krishnapuram R., Chiang J. and Mohamed M.A. Neural and Fuzzy Methods in Handwriting Recognition. *Computer Magazine*, pp. 79-85, February, 1997.

[18] Bose C.B. and Kuo S-S. Connected and Degraded Text Recognition using Hidden Markov Model. Pattern Recognition, Vol. 27, No. 10, pp. 1345-1363, 1994.

[19] Britto A.S., Sabourin R., Lethelier E., Bortolozzi F., and Suen C.Y. Improvement in Handwritten Numeral String Recognition by Slant Normalization and Contextual Information. *Proceedings of the International Workshop on Frontiers in Handwriting Recognition (IWFRH'2000*), Amsterdam, The Netherlands, 11-13 Septembre 2000, pp. 323-332.

[20] Elms A.J. *The representation and recognition of text using hidden Markov models* (PhD thesis). Department of Eletronic and Electrical Engineering, University of Surrey, April 1996.

[21] M. A. Mohamed, *"Handwritten Word Recognition Using Generalized Hidden Markov Models"*. PhD Thesis, University of Missouri-Columbia, USA (1995).

[22] R.M. Bozinovic, & S.N. Srihari, "Off-line cursive script recognition*", IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol.11, No. 1, (1989), pp68--83.

[23] L. Heutte, *"Reconnaissance de caractères manuscrits: application à la lecture automatique des chèques et des enveloppes postales"*, Docteur Thèse de L'Université de Rouen. Rouen, France, (1994), 239p.

[24] D. Guillevic, *"Unconstrained handwriting recognition applied to the processing of bank cheques"*, Thesis of Doctor's Degree in the Department of Computer Science at Concordia Uiversity, Canada, (1995), 183p.

[25] M. E. Morita, *"Estudo para a melhoria da correção da inclinação da linha de base de palavras manuscritas"*, Dissertação de Mestrado, Curitiba: CPGEI/CEFET-PR, (1998), 137p.

[26] F. Kimura; M. Shridhar; Z. Chen, "Improvements of a lexicon directed algorithm for recognition of unconstrained handwritten words*", International Conference on Document Analysis and Recognition. ICDAR'95*. vol 1, (1995), pp24--27.

[27] A. E. Yacoubi; M. Gilloux; R. Sabourin, C.Y. Suen, "Unconstrained handwritten word recognition using hidden markov models", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.2, no.8, (1999), pp752--760.

[28] M. Côte, *"Utilisation d'un modèle d'accès lexical et de concepts perceptifs pour la reconnaissance d'images de mots cursifs*", Thèse de doctorat, École Nationale Supérieure des Télécommunications, France,(1997). 171p.

[29] C. O. A. Freitas, *"O Uso de Modelos Escondidos de Markov para Reconhecimento de Palavras Manuscritas"*, Tese de Doutorado, PUCPR Pontifícia Universidade Católica do Paraná, junho, (2001), 189p.

[30] S. Madhavanath & V. Govindaraju, "Preceptual features for off-line handwritten word recognition: a framework for prediction, representation and matching", *Advances in Pattern Recognition*, (1998), pp524--531.

[31] C. O. A. Freitas; F. Bortolozzi; R. Sabourin, "Handwritten word recognition: an approach based on mutual information for feature set validation", *Proc. Sixth International Conference on Document Analysis and Recognition, ICDAR'2001*, september, Seattle, USA, (2001).

[32] G. Rigoll, A. Kosmala, "A Systematic Comparison Between On-Line and Off-Line Methods for Signature Verification with Hidden Markov Models", *14th International Conference on Pattern Recognition - vol. II*, Australia (1998), 1755--1757.

[33] E. J. R. Justino, A. El Yacoubi, F. Bortolozzi and R. Sabourin, "An Off-Line Signature Verification System Using HMM and Graphometric Features", DAS 2000, *4th IAPR International Workshop on Document Analysis Systems, Rio de Janeiro, Brazil, (2000),* pp 211--222.

[34] R. Sabourin and Genest, G., "An Extended -Shadow-Code Based Approach for Off-Line Signature Verification: Part -I – Evaluation of the Bar Mask Definition", *12th IAPR International Conference on Pattern Recognition* - Jerusalem, Israel, October 9-13 - IEEE 1051-4651/94, (1994), 450--455.

[35] E. J. R. Justino, *"O Grafismo e os Modelos Escondidos de Markov na Verificação Automática de Assinaturas"*, Tese de Doutorado, PUCPR Pontifícia Universidade Católica do Paraná, junho (2001), 132p.

[36] Y. Qi, Bobby R Hunt, "A Multiresolution Approach to Computer Verification of Handwritten Signatures", *IEEE Transactions on Image Processing*, vol. 4, no. 6, june (1995), pp 870--74.