UNIVERSIDADE FEDERAL DE PERNAMBUCO

GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO
CENTRO DE INFORMÁTICA

SISTEMA DE RECONHECIMENTO DE MESES MANUSCRITOS

TRABALHO DE GRADUAÇÃO

POR

LEANDRO REINAUX ALMEIDA

Recife, Novembro de 2009

UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

LEANDRO REINAUX ALMEIDA

“SISTEMA DE RECONHECIMENTO DE MESES MANUSCRITOS”

*ESTE TRABALHO FOI APRESENTADO Á GRADUAÇÃO DO
CENTRO DE INFORMÁTICA DA UNIVERSIDADE FEDERAL DE
PERNAMBUCO COMO REQUISITO PARCIAL PARA A
CONCLUSÃO DO CURSO DE CIÊNCIA DA COMPUTAÇÃO.*

ORIENTADOR(A): PROF. GEORGE DARMITON DA CUNHA CAVALCANTI

RECIFE, NOVEMBRO/2009

Agradecimentos

Esta monografia é mais um passo dessa longa caminhada chamada vida. Como em qualquer outra fase, ela é composta de pessoas, que ao meu redor completam essa etapa em conjunto. Portanto, agradeço a todos que hoje estão ao meu redor e também a aqueles já não estão mais, mas que me ajudaram a chegar até aqui e completar essa etapa.

Agradeço, em particular, a algumas pessoas que contribuíram diretamente para este trabalho:

Aos meus pais Paulo Augusto e Rosa Reinaux e familiares, por todo apoio e incentivo recebidos.

A minha namorada Ana Carolina, pelo apoio e paciência ao longo desse e de outros semestres atarefados da faculdade.

Aos professores, em especial ao meu orientador, pela disposição e pelo ensino.

Aos colegas de universidade, pelo convívio e pelo compartilhamento de experiências. Em especial a Everson Veríssimo, por ter contribuído na construção da base de dados do trabalho e aos colegas que participaram comigo no Projeto de Desenvolvimento, que trouxeram grande crescimento pessoal e profissional: Alana Brito, Ana Cecília, Alan Moura, Gustavo Ferraz, Hugo Simões, Luiz Guimarães, Marcela Meira Lins, Marcelo Costa, Márcio de Mendonça, Petrônio Gomes, Rodrigo Mateus, Thiago Monteiro e Victor Hazin.

Aos colegas de trabalho e integrantes do CITi, que me ajudaram a crescer profissionalmente, em especial Cássio Melo e Daniel Arcoverde, pela contribuição após a minha passagem pela empresa júnior e Bruno Marques por ter ingressado junto comigo na atual empresa que trabalho e por passar pelas mesmas dificuldades desse trabalho no semestre corrente.

Resumo

O reconhecimento de palavras manuscritas é o processo de identificar termos escritos manualmente que foram digitalizados. O reconhecimento de palavras pode ser dividido em duas modalidades, reconhecimento *on-line* ou *off-line*. A abordagem *on-line* fornece uma representação espaço-temporal da entrada, enquanto que a *off-line* envolve análise de espaço-luminosidade da imagem.

O estudo das palavras manuscritas está ligado ao desenvolvimento de métodos de reconhecimento voltados para aplicações do mundo real envolvendo palavras manuscritas, tais como: processamento automático de cheques bancários, envelopes postais, formulários, textos manuscritos, entre outros.

No trabalho realizado foram utilizados como objetos de estudo os 12 meses do ano, que apresentam desafios de tratar palavras semelhantes e com sub-strings em comum: **Janeiro**, **Fevereiro**, **Março**, **Abril**, **Maio**, **Junho**, **Julho**, **Agosto**, **Setembro**, **Outubro**, **Novembro** e **Dezembro**.

Esse trabalho busca tratar o problema de reconhecimento de meses manuscritos através da combinação das seguintes técnicas: (i) técnicas de pré-processamento; (ii) técnicas de extração de características e (iii) técnicas de classificação.

A avaliação do desempenho das técnicas foi feita utilizando uma base de dados de textos manuscritos própria. Os resultados mostram as técnicas mais adequadas para tratar o problema proposto juntamente com uma análise comparativa em relação aos trabalhos já desenvolvidos na área.

Palavras-chave: Sistemas de reconhecimento, Reconhecimento de meses, Meses manuscritos.

Abstract

The recognition of handwritten words is the process of identifying written words that were manually scanned. The recognition of words can be divided into two major modalities, recognition online and offline. The online approach provides a representation of space-time entry, while the case involves off-line analysis of space-image brightness.

The study of handwritten words is linked to the development of identification methods aimed at real-world applications involving handwritten words such as: automatic processing of checks, envelopes, postcards, forms, manuscripts, among others.

In the study were used as objects of study to 12 months of the year, which has the challenges of dealing with similar words and sub-strings in common: **Janeiro**, **Fevereiro**, **Março**, **Abril**, **Maio**, **Junho**, **Julho**, **Agosto**, **Setembro**, **Outubro**, **Novembro** e **Dezembro**.

This paper seeks to address the problem of recognizing handwritten months through a combination of the following techniques: (i) techniques for preprocessing, (ii) techniques for feature extraction, (iii) classification techniques.

A performance evaluation technique was made using an own database of manuscripts. The results show the best techniques for dealing with proposed problem in addition to a comparative analysis in relation to the work already undertaken in the area.

Keywords: Recognition systems, Recognition of months, Months manuscripts.

Sumário

CAPÍTULO 1	9
1.1 Motivação.....	9
1.2 Contexto	9
1.3 Objetivo	10
1.4 Estrutura do trabalho	11
CAPÍTULO 2	12
2.1 Binarização	12
2.2 Menor retângulo que contém o componente (<i>Bound Box</i>).....	12
2.3 Correção de inclinação (<i>Skew</i>)	13
2.4 Normalização.....	14
CAPÍTULO 3	15
3.1 Introdução	15
3.2 Características de coluna	15
3.3 Características de contagem de pixel.....	17
3.3.1 Perantonis	17
3.3.2 Vinciarelli	17
3.4 Características de histogramas de gradientes locais	18
CAPÍTULO 4	20
4.1 Introdução	20
4.2 Base de dados	20
4.3 Experimentos	21
4.3.1 Classificadores da ferramenta Weka.....	21
4.3.2 K-NN	21
4.4 Resultados	22
4.4.1 Ferramenta Weka.....	22
4.4.1.1 Extração LGH +SMO	22
4.4.1.1 Extração Vinciarelli +BayesNet.....	23
4.4.1.3 Extração Marti and Bunke +BayesNet.....	24
Taxa de acerto = 71,78%	24
4.4.1.4 Perantonis +BayesNet	24
4.4.2 K-NN	25
4.4.2.1 LGH	25

4.4.2.1 Marti And Bunke	26
4.4.2.1 Vinciarelli	27
4.4.2.1 Perantonis	28
4.4.3 Sumário	29
CAPÍTULO 5	31
5.1 Introdução	31
5.2 Contribuições	31
5.3 Trabalhos Futuros.....	31
Referências Bibliográficas.....	32
Apêndice	33

Lista de Figuras

Figura 1 - Etapas de um sistema de reconhecimento de texto manuscrito	10
Figura 2 - Etapas do pré-processamento.....	12
Figura 3 - bound box aplicado a imagem original.....	13
Figura 4 - Processo de correção de inclinação (Skew)	14
Figura 5 - Ilustração das colunas cujas características estão sendo extraídas. Retirada de [CC08b]	15
Figura 6 - Características extraídas das colunas. Retirada de [CC08b].....	16
Figura 7 - Extração de características baseada em sub-regiões.....	17
Figura 8 - Processo de extração dos histogramas de gradientes locais. Adaptado de [JR08].	19
Figura 9 - Exemplo de data coletada no formulário original.....	20
Figure 10 - Meses extraídos das datas coletadas nos formulários.....	20
Figura 11 - Taxa de erro em relação a variação de K para a extração LGH	25
Figura 12 - Taxa de erro em relação a variação de K para a extração Marti And Bunke	26
Figura 13 - Taxa de erro em relação a variação de K para a extração Vinciarelli	27
Figura 14 - Taxa de erro em relação a variação de K para a extração Perantonis.....	28
Figura 15 - Tela inicial do software.....	33
Figura 16 - Tela de escolha da imagem de entrada	34
Figura 17 - Escolhendo a extração de características	34
Figura 18 - Escolhendo o classificador e os seus parâmetros.....	35
Figura 19 - Resultado do reconhecimento	36
Figura 20 - Reconhecimento da base de dados	37

Lista de Tabelas

Tabela 1 - Matriz de confusão da extração LGH utilizando SMO.....	22
Tabela 2 - Matriz de confusão da extração Vinciarelli utilizando BayesNet	23
Tabela 3 - Matriz de confusão da extração Marti and Bunke utilizando BayesNet	24
Tabela 4 - Matriz de confusão da extração Perantonis utilizando BayesNet	24

Tabela 5 – Matriz de confusão da extração LGH utilizando 3-NN.....	26
Tabela 6 – Matriz de confusão da extração Marti and Bunke utilizando 3-NN.....	26
Tabela 7 – Matriz de confusão da extração Marti and Bunke utilizando 3-NN.....	27
Tabela 8 – Matriz de confusão da extração Perantonis utilizando 3-NN	29
Tabela 9 – Taxa de acerto por mês das melhores combinações de extração e classificador.....	29

CAPÍTULO 1

Introdução

1.1 Motivação

A busca pela automação de atividades repetitivas sempre foi almejada pela sociedade, com a finalidade de aumentar as escalas de trabalho e de ocupar o homem com tarefas que exijam a mente ao invés da repetição. Tratando-se de textos, a primeira grande revolução de automação deu-se no século XV com a invenção da prensa por Gutenberg, com o objetivo de replicar a informação por via de papel. Recentemente vem ocorrendo uma segunda revolução com o objetivo de automatizar a transposição das informações contidas nos papéis para o meio digital, que tem uma capacidade ainda maior de replicação e compartilhamento de informação. Grande parte do trabalho recente se deu no desenvolvimento de sistemas OCR (*Optical character recognition*) [CCD06], os quais são capazes de digitalizar documentos impressos. Porém, o grande desafio atualmente trata-se da digitalização de documentos manuscritos, que possuem alta variabilidade de grafias, devido a forma peculiar que a escrita de cada indivíduo apresenta.

Por todos esses motivos, existem inúmeros trabalhos científicos e comerciais que visam transformar informações manuscritas em representações digitais. Como exemplos comuns de aplicações que se utilizam de mecanismos de reconhecimento de palavras manuscritas, têm-se sistemas bancários para validação automática dos campos dos cheques, aplicações para indexação de documentos históricos e sistemas para leitura de endereço e código postal presentes nos correios.

1.2 Contexto

No contexto de reconhecimento de palavras manuscritas, são desenvolvidos os sistemas de reconhecimento que possuem uma complexidade alta para que seja possível o sistema se adaptar as variações de escritas apresentadas.

É possível dividir o reconhecimento automático de manuscritos em duas grandes vertentes: *on-line* e *off-line* [ACC06]. No caso do reconhecimento *off-line*, a entrada para o sistema é composta de uma imagem de um documento que foi capturada a partir de um *scanner* ou de uma câmera. Já no caso *on-line*, mais informações podem ser capturadas durante o ato da

escrita, tais como: velocidade, pressão, direção. Para tanto, faz-se necessário o uso de um dispositivo especial para capturar a escrita: um *tablet*. Desta forma, sistemas de reconhecimento *on-line* possuem mais recursos para reconhecer o texto de maneira correta, apresentando-se, assim, como um problema mais simples do que o *off-line*.

Sabendo que o objeto de estudo do trabalho é o reconhecimento de palavras manuscritas *off-line*, pode-se dividir a construção dos sistemas propostos nas seguintes etapas:

Figura 1 - Etapas de um sistema de reconhecimento de texto manuscrito

O foco do trabalho a ser desenvolvido, ilustrado na figura acima, será a análise e desenvolvimento de técnicas de extração de características e classificadores, por ser considerado em conjunto com o pré-processamento as etapas mais importantes no processo de reconhecimento. Como imagens de entrada será considerado o domínio dos meses brasileiro por extenso: Janeiro, Fevereiro, Março, Abril, Maio, Junho, Julho, Agosto, Setembro, Outubro, Novembro e Dezembro. Esse domínio além de desafiador, devido a semelhança de algumas palavras, possui diversas aplicações comerciais para bancos, cartórios e repartições públicas.

1.3 Objetivo

O objetivo do trabalho é analisar técnicas de extração de características e classificadores que melhorem a performance de um sistema de reconhecimento de meses manuscritos automático.

1.4 Estrutura do trabalho

Este documento se divide em cinco capítulos: neste primeiro capítulo é apresentada a motivação, o contexto e o objetivo deste trabalho. O capítulo 2 discorre sobre as técnicas de pré-processamento estudadas. No capítulo 3 são descritas as técnicas de extração de características implementadas. Os experimentos e seus resultados estão detalhados no capítulo 4. Por último, o capítulo 5 expõe as considerações finais e as propostas de trabalhos futuros.

CAPÍTULO 2

Pré-processamento

Esse capítulo apresenta as técnicas de pré-processamento estudadas e utilizadas no trabalho desenvolvido.

As etapas utilizadas no pré-processamento seguem o fluxo da Figura 2 (abaixo), na qual cada etapa será descrita nas próximas seções:

Figura 2 - Etapas do pré-processamento

2.1 Binarização

A primeira etapa do pré-processamento é a binarização, que consiste na passagem da imagem de tons de cinza para preto e branco. Existem várias técnicas na literatura que fazem a passagem de maneira local ou adaptativa, de maneira global ou através de um limiar. A técnica escolhida para a binarização foi baseada em limiar, utilizando as funções do Matlab *graythresh*, que determinam o limiar a partir da técnica de Otsu [OT79] e a função *im2bw*, que retorna a imagem binarizada.

2.2 Menor retângulo que contém o componente (*Bound Box*)

A técnica de *bound box* tem como objetivo descartar as partes brancas das imagens, que não trazem informações relevantes para o processo de classificação. Para determinar o bound box a imagem é inspecionada linha a linha nas quatro de direções no sentido de fora para dentro até encontrar a primeira linha que contenha um pixel preto. A figura abaixo ilustra a técnica de *bound box* utilizada:

Figura 3 - bound box aplicado a imagem original

2.3 Correção de inclinação (*Skew*)

Para corrigir a inclinação das palavras é utilizada a técnica de projeções horizontais proposta por [SP06].

Sendo $im(x, y)$ a matriz da representação da palavra, representada por 1's e 0's como fundo de imagem, x_{max} e y_{max} são a largura e a altura da imagem respectivamente. Primeiro são calculadas as projeções da direita (RP) e esquerda (LP) (ver Figura 4) segundo as fórmulas:

$$LP(y) = \sum_{x=0}^{\frac{x_{max}}{2}} im(x, y) \quad RP(y) = \sum_{\frac{x_{max}}{2}}^{x_{max}} im(x, y)$$

Depois é calculada a máxima projeção da esquerda e da direita LP e RP, tendo $y = y_L$ e $y = y_R$. No próximo passo são calculados os valores y_{L1} , y_{L2} , y_{R1} e y_{R2} que correspondem ao valor mais próximo de y_L e y_R , os quais seguem a fórmula $LP(y) < 0,2LP(y_L)$ e $RP(y) < 0,2RP(y_R)$ onde a constante 0,2 é definida por [SP06] como ideal, seguindo as seguintes fórmulas:

$$y_{L1} = y: (LP(Y) < 0.2LP(y_L) \& y = \max(y_i)), y_i \in [0, y_L]$$

$$y_{L2} = y: (LP(Y) < 0.2LP(y_L) \& y = \min(y_i)), y_i \in [y_L, y_{max}]$$

$$y_{R1} = y: (RP(Y) < 0.2RP(y_R) \& y = \max(y_i)), y_i \in [0, y_R]$$

$$y_{R2} = y: (RP(Y) < 0.2RP(y_R) \& y = \min(y_i)), y_i \in [y_R, y_{max}]$$

Figura 4 - Processo de correção de inclinação (Skew)

2.4 Normalização

Por último as imagens são normalizadas através de padronização do seu tamanho. As imagens foram redimensionadas para a altura e a largura média da base de dados. A técnica de redimensionamento foi utilizada através da função *imresize* do Matlab, na qual é utilizada uma técnica de interpolação bicubica, ou seja, é utilizada a média ponderada de uma vizinhança 4x4 para determinar o valor do pixel atual. A altura e a largura média estabelecidas foram respectivamente 74 e 270 pixels. O processo de normalização influencia tanto a extração de características e a classificação, que passarão a trabalhar com uma imagem padronizada, da qual é possível estabelecer uma métrica comparativa de maneira mais simples.

CAPÍTULO 3

Extração de características

3.1 Introdução

Este capítulo contém a explicação dos métodos de extração de características utilizados no trabalho. As extrações são divididas de acordo com o seu tipo: Características de colunas, características de contagem de pixel, características de histogramas de gradientes locais

3.2 Características de coluna

A extração de característica de coluna utilizada no trabalho, foi baseada na dissertação de Cesar [CC08b] e no artigo de Marti e Bunke [MB01a], na qual nove características foram extraídas de cada coluna com largura de 1 pixel, no sentido esquerdo-direita.

Figura 5 - Ilustração das colunas cujas características estão sendo extraídas. Retirada de [CC08b]

A descrição de cada característica está a seguir:

1. *Peso da janela*: total de pixels pretos presentes na janela.

$$f_1 = \sum_{y=1}^m p(x,y),$$

x e y representam a coluna e a linha da imagem, respectivamente, p é a função que recupera o valor do pixel (“0” ou “1”) da coordenada da imagem e m é a altura da imagem.

$$f_2 = \frac{1}{m} \sum_{y=1}^m y \cdot p(x,y)$$

2. *Centro de gravidade.*

3. *Momento de segunda ordem.*
$$f_3 = \frac{1}{m^2} \sum_{y=1}^m y^2 \cdot p(x, y)$$

4. *Posição do contorno superior:* coordenada do pixel preto mais alto da janela.

5. *Posição do contorno inferior:* coordenada do pixel preto mais baixo da janela.

6. *Gradiente do contorno superior:* direção (“para cima”, “nulo” ou “para baixo”) obtida pela comparação das posições dos contornos superiores da coluna anterior e a atual.

7. *Gradiente do contorno inferior:* direção (“para cima”, “nulo” ou “para baixo”) obtida pela comparação das posições dos contornos inferiores da coluna anterior e a atual.

8. *Número de transições preto-branco:* número total de transições “preto-branco” observadas no sentido de cima para baixo da janela.

9. *Número de pixels pretos entre o contorno superior e inferior da janela.*

A Figura 6 ilustra duas colunas cujas características estão sendo extraídas e suas vizinhas à esquerda respectivamente.

Figura 6 - Características extraídas das colunas. Retirada de [CC08b]

No exemplo da esquerda, o total de transições preto-branco é igual a três e existe um pixel preto entre os contornos. Pode-se observar que as posições dos contornos superiores e inferiores, tanto da coluna como da vizinha, são distintas; isso permite determinar os seus gradientes como “para cima” e “para baixo”, respectivamente.

No exemplo da direita, o número de transições preto-branco é um e não existe pixel preto entre os contornos. Pode-se observar que as posições dos contornos superiores e inferiores coincidem; Isso significa que o gradiente de ambos os contornos é “nulo”.

Situações especiais foram tratadas da seguinte maneira:

- a. Ausência de pixels pretos na coluna em análise: posição dos contornos igual a zero.
- b. Ausência de pixels pretos na coluna antecedente, ou seja, inexistência de contornos: gradientes são considerados como “nulo”.

3.3 Características de contagem de pixel

3.3.1 Perantonis

A técnica de extração de característica utilizada no trabalho é baseada na proposta por Perantonis [SP06]. As imagens são divididas em sub-regiões e para cada sub-região é calculada a quantidade de pixels pretos.

A figura 7b ilustra com intensidade de cores as regiões que possuem mais pixels pretos:

(a) Imagem dividida em quadrados

(b) Quantidade de pixels em cada quadrado

Figura 7 - Extração de características baseada em sub-regiões

3.3.2 Vinciarelli

A extração de características de contagem de pixel proposta por Vinciarelli [AV04], utiliza uma janela deslizante de largura 16 pixels. Tal janela é dividida em grids 4x4, dos quais é extraída a seguinte característica:

$$f = \sum_{i=1}^4 \frac{p(grid_i)}{p(grid)}$$

Onde p é a função que conta a quantidade de pixels pretos de uma determinada área da imagem.

3.4 Características de histogramas de gradientes locais

A extração de características de histogramas de gradientes locais proposta por Rodríguez [JR08] utiliza pequenas etapas de pré-processamento e depois são extraídos os gradientes locais.

Primeiramente é utilizado um filtro gaussiano de borramento na imagem. Após o borramento da imagem são calculados os componentes verticais e horizontais do gradiente segundo as equações abaixo:

$$G_x = L(x + 1, y) - L(x - 1, y)$$

$$G_y = L(x, y + 1) - L(x, y - 1)$$

Após esse passo, a magnitude m e a direção θ são obtidas para cada coordenada utilizando as equações:

$$m(x, y) = \sqrt{G_x^2 + G_y^2}$$

$$\theta(x, y) = \text{atan2}(G_y, G_x)$$

Onde atan2 é a função que retorna a direção do vetor (G_x, G_y) no range $[-\pi, \pi]$ levando em conta $\tan^{-1}(\frac{G_y}{G_x})$ e o sinal de G_x e G_y . Depois os ângulos dos gradientes são quantizados em um número T de orientações espaciais, no quais as magnitudes que possuem a mesma orientação são acumuladas em um histograma. Na figura 8 é exemplificado um exemplo do fluxo da extração de característica.

Figura 8 - Processo de extração dos histogramas de gradientes locais. Adaptado de [JR08].

CAPÍTULO 4

Experimentos

4.1 Introdução

Este capítulo contém os experimentos realizados neste trabalho. Inicialmente, a base de dados utilizada nos experimentos é apresentada, seguida da descrição do método de avaliação. Por último, serão feitas análises sobre os resultados obtidos de acordo com as métricas estabelecidas.

4.2 Base de dados

Os experimentos realizados utilizaram uma base de dados coletada com 250 usuários. Cada usuário foi pedido que escrevesse, em letra de forma, uma data de cada mês do ano no mesmo formato do cheque, como mostrado no exemplo abaixo:

Figura 9 - Exemplo de data coletada no formulário original

No total foram obtidos 2996 exemplos, tendo em média 250 exemplos por classe. Após a coleta, os formulários foram *scaneados* e os meses foram extraídos de cada exemplo para formar a base de dados, como nos exemplos a seguir:

Figure 10 - Meses extraídos das datas coletadas nos formulários

4.3 Experimentos

Foram realizados dois grupos de experimentos com a base de dados citada na seção anterior: (i) Experimentos com os classificadores disponíveis na ferramenta Weka[WEK] e (ii) Experimentos com o classificador K-NN desenvolvido.

4.3.1 Classificadores da ferramenta Weka

Após a realização de vários testes com os classificadores disponíveis na ferramenta Weka, os melhores resultados obtidos foram com os classificadores BayesNet e SMO.

BayesNet: Rede Bayesiana, que foi utilizada com os parâmetros padrão da ferramenta.

SMO: Implementação da ferramenta para SVM (Support Vector Machine), na qual as entradas são normalizadas de acordo com a base de treinamento. Foram realizados testes tanto com o kernel polinomial quanto com o *RBKernel*, que utiliza *Radial Basis Functions*.

Os experimentos foram realizados com 10 *fold cross validation* (validação cruzada), ou seja, a base de dados foi repartida em 10 partes e foram realizados testes com cada uma das partes enquanto as 9 restantes eram utilizadas para treinamento. Ao final são obtidas as médias da taxa de acerto e a matriz de confusão é acumulada a cada iteração da validação cruzada.

4.3.2 K-NN

Foi implementado o classificador K-NN tradicional utilizando distância euclidiana como métrica de distância, segundo a fórmula abaixo:

$$d = \sqrt{\sum_{i=1}^n (p_i - q_i)^2}$$

Na qual n é a quantidade de características, p é um vetor de treinamento e q é um vetor de teste.

Durante os experimento, o parâmetro K foi variado dentre o intervalo de 1 a 100 com o objetivo de encontrar o valor no qual o erro fosse mínimo. A base de dados foi repartida de maneira que 200 exemplos de cada classe fossem utilizados para treinamento e o restante para teste, o que significa uma taxa média de 80% para treinamento e 20% para teste.

4.4 Resultados

A seção de resultados está dividida nas mesmas etapas explanadas na seção anterior de experimentos. Primeiramente são apresentados os resultados obtidos com a ferramenta Weka, em seguida com o classificador K-NN e ao final é feita uma sumarização dos resultados, que estão expressos em taxa de acerto e matriz de confusão, na qual as colunas representam a resposta dada pelo classificador e as linhas a classificação esperada.

4.4.1 Ferramenta Weka

Para cada extração de características será mostrado apenas o melhor resultado obtido:

4.4.1.1 Extração LGH +SMO

	Jan	Fev	Mar	Abr	Mai	Jun	Jul	Ago	Set	Out	Nov	Dez
Jan	40	20	22	20	33	61	12	17	4	8	9	7
Fev	14	38	22	21	19	54	14	11	10	15	8	19
Mar	12	17	60	8	29	55	12	16	14	9	8	10
Abr	14	22	20	31	32	45	10	26	12	6	16	12
Mai	21	15	28	14	64	41	24	13	5	6	4	12
Jun	20	17	21	15	32	74	29	24	14	8	6	7
Jul	23	16	19	12	43	52	30	12	6	9	6	4
Ago	19	17	20	20	21	50	11	53	9	9	8	11
Set	11	18	16	24	23	46	12	13	35	16	13	21
Out	24	17	23	10	26	40	15	20	12	37	17	7
Nov	16	24	12	23	22	32	6	21	20	22	26	24
Dez	6	25	18	10	20	41	16	10	25	11	28	53

Tabela 1 - Matriz de confusão da extração LGH utilizando SMO

Taxa de acerto = 18,06%

O pior resultado obtido, devido aos resultados da extração de características. Os vetores resultantes da implementação do LGH possuem várias características que possuem apenas zeros, portanto não ajudam na classificação. Os valores que não são zeros também não discriminam bem as classes, pois possuem valores similares para cada uma delas. Portanto é preciso revisar a implementação do LGH para obter melhores resultados.

4.4.1.1 Extração Vinciarelli +BayesNet

	Jan	Fev	Mar	Abr	Mai	Jun	Jul	Ago	Set	Out	Nov	Dez
Jan	183	0	33	5	4	9	6	7	1	1	0	4
Fev	7	187	14	1	8	1	1	0	7	0	13	6
Mar	10	2	220	9	4	2	0	1	0	1	1	0
Abr	11	0	5	190	16	6	5	12	1	0	0	0
Mai	4	0	13	16	194	10	1	8	1	0	0	0
Jun	22	0	19	4	1	179	33	7	1	1	0	0
Jul	1	0	18	1	1	32	173	4	2	0	0	0
Ago	12	6	9	5	2	4	7	199	0	2	1	1
Set	10	7	6	1	2	0	4	0	173	7	12	26
Out	5	4	2	1	6	4	6	1	10	200	6	3
Nov	11	3	6	0	2	0	3	3	11	6	184	19
Dez	12	6	3	1	1	0	5	0	40	3	18	174

Tabela 2 - Matriz de confusão da extração Vinciarelli utilizando BayesNet

Taxa de acerto = 75,32%

A extração Vinciarelli em conjunto com o classificador BayesNet apresentou uma taxa de acerto relativamente boa apesar de cometer alguns erros em determinados meses. A troca entre Junho e Julho foi a que mais ocorreu, devido a diferença de apenas uma letra entre as duas classes. Outro caso de troca ocorreu dentre os meses de Setembro e Dezembro, que possuem a substring “embro” em comum. Além disso, foi classificado erroneamente um número considerável de instâncias como Janeiro e Março.

De maneira geral a extração Vinciarelli possui um bom desempenho global, apesar de necessitar de características mais específicas para resolver o problema das trocas de meses que possuem substring em comum.

4.4.1.3 Extração Marti and Bunke +BayesNet

	Jan	Fev	Mar	Abr	Mai	Jun	Jul	Ago	Set	Out	Nov	Dez
Jan	196	2	16	0	0	6	6	13	4	8	2	0
Fev	8	170	12	0	8	1	2	1	10	6	13	15
Mar	22	3	180	0	10	7	12	9	0	1	6	0
Abr	9	2	11	210	5	0	2	3	0	3	0	1
Mai	9	2	15	3	189	16	4	7	1	0	0	1
Jun	5	1	18	0	1	150	67	9	5	4	4	3
Jul	10	0	23	0	2	28	158	2	3	3	1	2
Ago	6	4	10	0	1	6	11	198	1	4	1	6
Set	6	7	3	0	2	4	4	1	169	18	13	21
Out	3	0	4	0	2	4	9	5	11	204	4	2
Nov	9	5	9	0	0	3	1	2	18	7	158	36
Dez	10	4	4	0	0	2	9	1	17	3	45	168

Tabela 3 - Matriz de confusão da extração Marti and Bunke utilizando BayesNet

Taxa de acerto = 71,78%

A extração Marti And Bunke em conjunto com o classificador BayesNet conseguiu um desempenho similar a extração Vinciarelli porém com problemas um pouco mais grave quanto a troca Junho e Julho. Também houve problemas de troca entre Setembro e Dezembro, no entanto menos graves que a troca entre Novembro e Dezembro.

4.4.1.4 Perantonis +BayesNet

	Jan	Fev	Mar	Abr	Mai	Jun	Jul	Ago	Set	Out	Nov	Dez
Jan	198	1	28	5	1	10	2	5	3	0	0	0
Fev	6	192	12	0	0	0	4	6	6	2	13	4
Mar	9	0	226	6	2	3	0	2	0	2	0	0
Abr	12	2	8	197	15	1	4	6	1	0	0	0
Mai	6	0	10	14	191	13	2	10	0	1	0	0
Jun	12	0	19	1	4	183	46	0	1	0	0	1
Jul	3	0	20	1	2	36	168	2	0	0	0	0
Ago	11	4	11	9	3	3	7	195	1	3	1	0
Set	9	5	3	1	0	3	4	0	183	8	6	26
Out	2	5	4	2	1	5	5	1	7	209	6	1
Nov	4	8	6	0	0	1	2	5	11	5	198	8
Dez	6	4	4	3	0	0	3	1	45	6	13	178

Tabela 4 - Matriz de confusão da extração Perantonis utilizando BayesNet

Taxa de acerto = 77,39%

Dentre as extrações em conjunto com os classificadores do Weka, Perantonis foi a que apresentou o melhor desempenho. A extração teve o mesmo problema apresentado pelas extrações Marti and Bunke e Vinciarelli, que cometeram trocas em relação a Junho e Julho e Setembro por Dezembro.

4.4.2 K-NN

4.4.2.1 LGH

Figura 11 - Taxa de erro em relação a variação de K para a extração LGH

Taxa de acerto = 27,39 para K = 3

	Jan	Fev	Mar	Abr	Mai	Jun	Jul	Ago	Set	Out	Nov	Dez
Jan	15	6	14	4	8	3	1	1	0	0	0	1
Fev	11	9	5	3	3	2	2	2	4	0	0	4
Mar	10	6	14	2	3	5	3	3	1	2	1	0
Abr	8	11	5	6	7	1	2	3	0	2	1	0
Mai	9	5	9	2	10	1	4	3	1	1	2	0
Jun	11	13	15	5	8	5	3	2	1	0	3	1
Jul	12	3	5	3	2	2	2	2	0	1	0	0
Ago	4	8	5	8	8	4	4	3	1	2	1	0
Set	7	9	8	4	4	4	3	1	1	2	3	2
Out	14	6	6	4	5	1	4	2	4	0	1	1
Nov	9	13	10	3	5	2	2	0	1	1	0	2
Dez	13	5	9	6	4	8	4	1	6	1	2	4

Tabela 5 – Matriz de confusão da extração LGH utilizando 3-NN

A extração LGH novamente não apresentou uma boa performance, da mesma maneira que aconteceu utilizando os classificadores da ferramenta Weka. A maioria dos casos foram classificados como Janeiro, Fevereiro e Março.

4.4.2.1 Marti And Bunke

Figura 12 - Taxa de erro em relação a variação de K para a extração Marti And Bunke

Taxa de acerto = 67,23 para K = 3

	Jan	Fev	Mar	Abr	Mai	Jun	Jul	Ago	Set	Out	Nov	Dez
Jan	33	5	11	0	1	2	0	1	0	0	0	0
Fev	4	35	5	0	0	0	0	0	0	0	0	1
Mar	1	0	48	0	0	1	0	0	0	0	0	0
Abr	0	4	18	24	0	0	0	0	0	0	0	0
Mai	9	5	11	1	18	0	1	1	0	0	1	0
Jun	16	6	27	0	0	15	2	1	0	0	0	0
Jul	7	3	13	0	0	2	6	1	0	0	0	0
Ago	9	4	12	0	1	0	0	21	0	1	0	0
Set	3	6	14	0	0	0	0	1	17	0	1	6
Out	7	8	8	0	0	0	0	0	3	22	0	0
Nov	7	5	16	0	0	0	0	2	8	0	5	5
Dez	5	17	13	0	0	1	0	1	12	0	3	11

Tabela 6 – Matriz de confusão da extração Marti and Bunke utilizando 3-NN

A extração Marti And Bunke caiu de desempenho em relação ao obtido com a BayesNet, com uma taxa de acerto de 67,23% contra 71,78%. O maior problema enfrentado utilizando 3-NN foi quanto ao mês de março, apesar de ter classificado corretamente 48 instâncias das 50, foram classificadas como Março 196 instâncias no total.

4.4.2.1 Vinciarelli

Figura 13 - Taxa de erro em relação a variação de K para a extração Vinciarelli

Taxa de acerto = 27,90 para K = 3

	Jan	Fev	Mar	Abr	Mai	Jun	Jul	Ago	Set	Out	Nov	Dez
Jan	28	9	0	0	0	10	6	0	0	0	0	0
Fev	21	9	0	0	0	6	9	0	0	0	0	0
Mar	15	7	3	0	0	10	14	0	0	1	0	0
Abr	9	8	3	0	1	10	15	0	0	0	0	0
Mai	22	12	1	0	4	5	3	0	0	0	0	0
Jun	38	11	0	0	0	10	7	1	0	0	0	0
Jul	14	6	0	0	0	6	5	0	0	1	0	0
Ago	27	4	0	0	0	14	2	1	0	0	0	0
Set	25	6	0	0	0	7	8	1	0	1	0	0
Out	30	4	0	0	0	5	6	0	0	3	0	0
Nov	25	9	0	0	0	10	3	0	0	1	0	0
Dez	29	12	0	0	0	10	9	0	1	2	0	0

Tabela 7 – Matriz de confusão da extração Marti and Bunke utilizando 3-NN

A técnica Vinciarelli teve uma queda alta de desempenho, sendo 75,32% a taxa de acerto obtida com o classificador BayesNet contra 27,90% do 3-NN. A maioria das instâncias foram classificadas como Janeiro, enquanto que nenhuma instância foi classificado como Novembro e Dezembro.

4.4.2.1 Perantonis

Figura 14 - Taxa de erro em relação a variação de K para a extração Perantonis

Taxa de acerto = 91,26% para K = 3

	Jan	Fev	Mar	Abr	Mai	Jun	Jul	Ago	Set	Out	Nov	Dez
Jan	48	2	0	0	0	0	0	1	0	1	1	0
Fev	0	44	0	0	0	0	0	0	0	0	1	0
Mar	1	2	46	0	0	0	0	0	0	0	0	1
Abr	0	3	0	37	0	0	0	2	1	0	2	1
Mai	2	5	2	1	29	1	0	1	2	0	1	3
Jun	13	3	0	0	0	37	10	0	3	1	0	0
Jul	3	0	0	0	0	3	23	1	0	0	0	2
Ago	2	3	1	0	0	0	0	41	0	0	1	0
Set	1	2	0	0	0	0	0	0	23	1	8	13
Out	0	2	0	0	0	0	0	0	0	45	1	0
Nov	0	4	0	0	0	0	0	0	4	1	36	3
Dez	0	1	0	0	0	0	0	0	14	1	6	41

Tabela 8 – Matriz de confusão da extração Perantonis utilizando 3-NN

A extração Perantonis utilizando o 3-NN foi a técnica que obteve melhor desempenho. A melhoria foi de 77,39% para 91,26% quanto a taxa de acerto em relação ao classificador BayesNet. O problema da classificação errônea do mês Março foi corrigido, passando apenas para 3 instâncias classificadas incorretamente. Porém, o problema da troca entre Setembro e Dezembro persistiu.

4.4.3 Sumário

A tabela abaixo é o sumário das melhores combinações de classificador e extração de características, tendo como métrica a taxa de acerto.

	Perantonis(3-NN)	Vinciarelli(Bayes Net)	Marti And Bunke(BayesNet)	LGH + 3-NN
Janeiro	90,57%	72,30%	77,50%	28,30%
Fevereiro	97,78%	76,30%	69,40%	20,00%
Março	92,00%	88,00%	72,00%	28,00%
Abril	80,43%	77,20%	85,40%	13,04%
Março	61,70%	78,50%	76,50%	21,28%
Junho	55,22%	67,00%	56,20%	7,46%
Julho	71,88%	74,60%	68,10%	6,25%
Agosto	85,42%	80,20%	79,80%	6,25%
Setembro	47,92%	69,80%	68,10%	2,08%
Outubro	93,75%	80,60%	82,30%	0,00%
Novembro	75,00%	74,20%	63,70%	0,00%
Dezembro	65,08%	66,20%	63,90%	6,35%

Tabela 9 – Taxa de acerto por mês das melhores combinações de extração e classificador

Observamos que as melhores extrações para o problema são as de contagem de pixel, que no geral obtiveram as melhores taxas de acerto. Os maiores problemas enfrentados pela maioria dos classificadores e técnicas de extração foi a troca de Junho por Julho e Setembro por Dezembro, dado que em ambos os casos a quantidade de letras em comum é alta.

CAPÍTULO 5

Conclusões e trabalhos futuros

5.1 Introdução

Este trabalho abordou o problema de reconhecimento de meses manuscritos brasileiros. Foram discutidos problemas presentes nos variados estilos de escrita, que mostram que a tarefa de reconhecimento não é fácil. Quatro técnicas de extração de características foram implementadas com a finalidade de tentar maximizar o acerto da tarefa de classificação.

As taxas de acerto obtidas foram melhores que as obtidas na literatura, como a taxa de 81,75% obtida por [MK04], na qual foi utilizada Redes Neurais Modulares. A diferença para o trabalho de [MK04] é que os meses estudados foram coletados com letra cursiva, enquanto que no trabalho desenvolvido todos os exemplos foram escritos com letra de forma. Além disso, ficou comprovado que as sub-strings em comum dificultam o problema, como nos casos de **Junho** e **Julho** e **Setembro** e **Dezembro**.

Este capítulo apresenta as principais contribuições desse trabalho e algumas propostas para trabalhos futuros.

5.2 Contribuições

- Criação da base de dados de meses manuscritos
- Análise das quatro técnicas de extração de características propostas na literatura
- Desenvolvimento de uma ferramenta de visualização da classificação de uma determinada imagem de entrada(ver Apêndice).

5.3 Trabalhos Futuros

Como trabalhos futuros existem as seguintes sugestões:

- Testar as técnicas de extração de características com Redes Neurais Modulares proposto por [MK04].
- Testar a combinação de técnicas de extração de características.
- Desenvolver uma técnica de extração de características específica para o problema de reconhecimento de meses manuscrito em letra de forma.
- Desenvolver uma abordagem para tratar o problema das trocas entre Junho e Julho e Dezembro e Setembro.

Referências Bibliográficas

- [ACC06] Araújo, Rodrigo Sineco A; Cavalcanti, George Darmiton da Cunha; Carvalho Filho, Edson Costa de Barros. "On-line Verification for Signatures of Different Sizes". In: International Workshop on Frontiers in Handwriting Recognition, 2006. pp. 539-544.
- [AV04] Alessandro Vinciarelli , Samy Bengio , Horst Bunke, "Offline Recognition of Unconstrained Handwritten Texts Using HMMs and Statistical Language Models", IEEE Transactions on Pattern Analysis and Machine Intelligence, v.26 n.6, p.709-720, June 2004.
- [CC08a] C. A. M. Carvalho and G. D. C. Cavalcanti, "An Artificial Neural Network Approach for User Class Dependent Off-line Sentence Segmentation", IEEE International Joint Conference on Neural Networks, Hong Kong, 2008.
- [CC08b] C. A. M. Carvalho and G. D. C. Cavalcanti, "Segmentação de Sentenças Manuscritas através de Redes Neurais Artificiais", Dissertação de Mestrado, Recife, 2008.
- [CCD06] Zanchettin, Cleber; Cavalcanti, George Darmiton da Cunha; Dória, Rodrigo Calheiros; Silva, Eduardo F A; Rabelo, Juliano C B; Bezerra, Byron L D. "A Neural Architecture to Identify Courtesy Amount Delimiters". In: International Joint Conference on Neural Networks, 2006. pp. 5849-5856.
- [GW07] R.C. Gonzalez and R.E. Woods, "Digital Image Processing", Pearson Prentice Hall, 3. ed., 2007.
- [JR08] Rodríguez, J.A. and Perronnin, F., "Local gradient histogram features for word spotting in unconstrained handwritten documents". In: International Conference on Frontiers in Handwriting Recognition, 2008.
- [MB01a] U. V. Marti and H. Bunke. "Using a statistical language model to improve the performance of an hmm-based cursive handwriting recognition system", International Journal of Pattern Recognition and Artificial Intelligence, 15:65–90, 2001.
- [MB01b] U.V. Marti and H. Bunke, "Text Line Segmentation and Word Recognition in a System for General Writer Independent Handwriting Recognition", International Conference on Document Analysis and Recognition, pp. 159-163, 2001.
- [MK04] Kapp, M.N., Freitas, C.O.A., Sabourin, R. (2004) "Handwritten brazilian month recognition: an analysis of two NN architectures and a rejection mechanism", 9th International Workshop on Frontiers in Handwriting Recognition, p. 209-214, 2004.
- [OT79] Otsu, N., "A Threshold Selection Method from Gray-Level Histograms" *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 9, No. 1, 1979, pp. 62-66.

[SP06] B. Gatos, I. Pratikakis, A.L. Kesidis, S.J. Perantonis “Efficient Off-Line Cursive Handwriting Word Recognition”.

[WEK] WEKA 3.6.0. Disponível em: {<http://www.cs.waikato.ac.nz/~ml/weka>}

Apêndice

Funcionamento da ferramenta de auxílio a visualização do reconhecimento de meses manuscritos:

Figura 15 - Tela inicial do software

Na tela inicial é possível ver duas grandes áreas, na área da esquerda é possível reconhecer uma palavra escolhida, enquanto que na área da esquerda é possível rodar os experimentos para toda a base. Clicando no botão Carregar imagem será aberta a seguinte tela:

Figura 16 - Tela de escolha da imagem de entrada

Após selecionar a imagem é possível selecionar a técnica de extração de características, dentre as opções que foram implementadas no trabalho:

Figura 17 - Escolhendo a extração de características

Após selecionar a técnica de extração de características é possível selecionar o classificador, dentre as opções que foram implementadas no trabalho:

Figura 18 - Escolhendo o classificador e os seus parâmetros

Após selecionar a técnica de extração de característica é possível clicar no botão Reconhecer, que irá mostrar a seguinte tela com os resultados:

Figura 19 - Resultado do reconhecimento

Os resultados são mostrados em termos de probabilidade de pertencer a um determinado mês. No exemplo acima o mês foi classificado corretamente como Junho com 100% de probabilidade.

Figura 20 - Reconhecimento da base de dados

Utilizando a área da direita é possível rodar o reconhecimento de toda a base de dados utilizando a técnica de extração de característica e classificador escolhido.