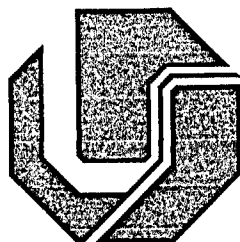


UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE ENGENHARIA ELÉTRICA
PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA



PRÉ-PROCESSAMENTO E SEGMENTAÇÃO DE PALAVRAS
MANUSCRITAS UTILIZANDO TÉCNICAS DE
PROCESSAMENTO DE IMAGENS

Fábio José Parreira

Abril

2002

SISBI/UFU



1000204575

UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE ENGENHARIA ELÉTRICA
PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

MON
621.3
P 253 k
TES/TEM

PRÉ-PROCESSAMENTO E SEGMENTAÇÃO DE PALAVRAS
MANUSCRITAS UTILIZANDO TÉCNICAS DE
PROCESSAMENTO DE IMAGENS

Fábio José Parreira

Texto da dissertação apresentada à Universidade Federal de Uberlândia, perante a banca de examinadores abaixo, como parte dos requisitos necessários à obtenção do título de Mestre em Ciências. Aprovada em 25 de Abril de 2002.

Banca examinadora:

Keiji Yamanaka, PhD, Orientador (UFU)

Rodrigo Pinto Lemos, Dr. (UFG)

Edna Lúcia Flôres, Dr^a. (UFU)

Gilberto Arantes Carrijo, Dr. (UFU)

Pré-Processamento E Segmentação de Palavras Manuscritas

Utilizando Técnicas de Processamento de Imagens

Fábio José Parreira

Texto da dissertação apresentada à Universidade Federal de Uberlândia como parte dos requisitos para obtenção do título de Mestre em Ciências.

Prof. Keiji Yamanaka, PhD

Orientador

Prof. Alcimar Barbosa Soares, PhD

Coordenador do Curso de Pós-Graduação

A Deus,

A minha família,

Em especial à minha mãe,

As minhas irmãs Núbia e Janete,

Pelo apoio e incentivo.

AGRADECIMENTOS

Aos amigos do laboratório “BioLab”, Adriano, Ângela, Eduardo Manco, Eduardo Naves, Kety, Milena, Rogério, Wilton, Gabriel, sempre apoiando nos vários momentos difíceis que tange a caminhada de um mestrado.

A Lidiane, por estar presente e contribuir de certa forma na maior parte da caminhada deste trabalho.

Aos professores da UFU, que sempre colaboraram de alguma forma para que esse trabalho se concretizasse.

Ao professor Keiji Yamanaka, pela oportunidade e orientação.

Ao professor Alcimar Barbosa Soares, pelas sugestões e companheirismo.

A professora Edna L. Flôres, pelas sugestões, que só vieram a enriquecer a dissertação.

Ao Gonçalo Divino Luz e a Joana Maria Proença pela dedicação e companheirismo na secretária da pós-graduação.

RESUMO

Parreira, F. J., & Yamanaka K. Pré-Processamento e Segmentação de Palavras Manuscritas Utilizando Técnicas de Processamento de Imagens, FEELT-UFU, Uberlândia, 2002, 116p.

As etapas de pré-processamento e segmentação são de fundamental importância no processo de reconhecimento de imagens de palavras manuscritas de fôrma ou cursivas, uma vez que têm influência direta no desempenho do sistema reconhecedor. Neste trabalho, um conjunto de técnicas de processamento de imagens é aplicado à imagem da palavra, na fase do pré-processamento, com o objetivo de reduzir as possíveis variações da escrita, tais como a sua inclinação vertical, a inclinação horizontal, e a influência do objeto usado no ato da escrita. Na etapa de segmentação, foram utilizadas as características topológicas e geométricas da imagem para separar individualmente as letras que compõem a palavra, empregando regras de posicionamento dos pontos de segmentação baseadas nas informações da correção vertical e do esqueleto obtidos na fase de pré-processamento. A porcentagem de segmentação obtida utilizando-se estas técnicas nas imagens de 300 palavras manuscritas, foi de 91,25%, independente de quem as escreveu.

Palavras Chave

Binarização, correção horizontal, correção vertical, afilamento, segmentação de palavras manuscritas.

ABSTRACT

Parreira, F. J., & Yamanaka K., Preprocessing and Segmentation of cursive script word Using Techniques of Processing of Images, FEELT-UFU, Uberlândia - Brazil, 2002, 103p.

The stages of preprocessing and segmentation play a fundamental role in the analysis of off-line of image cursive script word or machine word, once they influence the recognition process. In this work, a collection of techniques for processing images are applied to the images obtained from the cursive script word, during the preprocessing phase, with the aim of reducing variations of the writing, such as its vertical inclination, horizontal inclination and inaccuracies due to the use of objects, for example, pen and pencil, that influence the writing. The topologic and geometric characteristics of the image were used to segment the word in its letters, using point-positioning algorithms based on the vertical correction and thinning information obtained in the preprocessing phase. The percentage of segmentation obtained used these techniques in the images of 300 of cursive script word, it was of 91,25%, independent of who wrote them.

Keywords:

Binary image, horizontal correction, vertical correction, thinning, segmentation of cursive script word.

Índice

1 - INTRODUÇÃO	1
1.1 - Motivação	1
1.2 - Objetivos Do Trabalho	3
1.3 - Imagens Manuscritas Utilizadas	4
1.4 - Estrutura Da Dissertação	5
2 - ESTADO DA ARTE EM PRÉ-PROCESSAMENTO E SEGMENTAÇÃO DE IMAGENS	7
2.1 - Introdução	7
2.2 - Histórico De OCR's	7
2.2.1 - O sistema reconhecedor padrão.....	8
2.3 - Algumas Publicações Importantes Sobre Pré-processamento.....	12
2.3.1 - Trabalhos na etapa de pré-processamento	12
2.3.1.1 - binarização	13
2.3.1.1.1 - método de pré-processamento quadrático adaptativo.....	13
2.3.1.1.2 - método de Otsu	14
2.3.1.2 - correção horizontal	15
2.3.1.2.1 - técnica de Kimura	16
2.3.1.2.2 - técnica de Morita.....	17
2.3.1.3 - correção vertical	18
2.3.1.3.1 - técnica de Bozinovic e Shihari	18
2.3.1.3.2 - técnica de Buse	19
2.3.1.4 - afilamento.....	20
2.3.1.4.1 - técnica de Han.....	20
2.3.1.4.2 - técnica de Parker	21

2.3.1.5 - trabalhos desenvolvidos na etapa de segmentação	21
2.3.1.5.1 - segmentação morfológica de palavras cursivas	21
2.3.1.5.2 - segmentação de palavras manuscritas em off-line	23
2.4 - Considerações Finais Do Capítulo	23
3 - TÉCNICAS DE PROCESSAMENTO DIGITAL DE IMAGENS	25
3.1 - Introdução	25
3.1.1 - Processamento digital de imagens	26
3.1.2 - Áreas de aplicação do processamento digital de imagens	26
3.1.3 - Morfologia matemática	27
3.1.3.1 - definições básicas de morfologia matemática	29
3.1.3.2 - operações básicas de morfologia matemática	31
3.2 - Considerações Finais Do Capítulo	33
4 - ESPECIFICAÇÃO DO SISTEMA DESENVOLVIDO NESTE TRABALHO	34
4.1 - Introdução	34
4.2 - Definição Dos Requisitos Do Sistema Desenvolvido Neste Trabalho	35
4.3 - Sistema Proposto Nesta Dissertação	35
4.4 - Considerações Finais Deste Capítulo	37
5 - O SISTEMA DE PRÉ-PROCESSAMENTO UTILIZADO NESTE TRABALHO	38
5.1 - Introdução	38
5.2 - Filtros Empregados	38
5.3 - Binarização	41
5.4 - Correção Horizontal	43
5.4.1 - Determinação das linhas de delimitação da imagem	44
5.4.2 - Localização da linha central horizontal da imagem	47
5.4.3 - Localização da segmentação natural	49

5.4.4 - Detecção dos pontos de mínimos	50
5.4.5 - Eliminação dos mínimos indesejáveis	52
5.4.6 - Cálculo do ângulo de inclinação horizontal	54
5.4.7 - Correção da inclinação horizontal	55
5.5 - Correção Vertical	56
5.5.1 - Cálculo do grau de inclinação vertical da imagem.....	56
5.5.2 - Correção da inclinação vertical	58
5.6 - Afilamento	59
5.6.1 - Verificação do contorno.....	60
5.6.2 - Apagamento do contorno.....	60
5.7 - Resultados Obtidos.....	63
5.8 - Conclusões.....	66
6 - DESCRIÇÃO DO SISTEMA DE SEGMENTAÇÃO DESENVOLVIDO NESTE TRABALHO.....	67
6.1 - Introdução	67
6.2 - Segmentação	67
6.2.1 - Delimitação das zonas de referência	69
6.2.2 - Cálculo da espessura média do traço na imagem.....	71
6.2.3 - Localização do conjunto de pontos dos loops.....	71
6.2.4 - Verificação do conjunto de pontos de máximo na imagem de uma palavra	74
6.2.5 - Verificação do conjunto de pontos de cruzamento na imagem de uma palavra ...	76
6.2.6 - Verificação do conjunto de pontos finais na imagem de uma palavra	78
6.2.7 - Localização dos pontos de segmentação na imagem de uma palavra	78
6.2.7.1 - geração de pontos de corte para letras de fôrma na imagem de uma palavra	80
6.2.7.2 - geração de pontos de corte na imagem da palavra manuscrita	82

6.2.7.2.1 - segmentação natural da imagem da palavra manuscrita	83
6.2.7.2.2 - verificação dos pontos de mínimos da imagem da palavra manuscrita ..	84
6.2.7.2.3 - verificação dos loops na imagem da palavra manuscrita	87
6.2.7.2.4 - verificação dos pontos cruzamento na imagem de uma palavra manuscrita	88
6.2.8 - Corte da imagem da palavra manuscrita nos pontos de segmentação.....	89
6.3 - Resultados Obtidos.....	89
6.4 - Conclusão Deste Capítulo	91
7 - RESULTADOS OBTIDOS.....	93
7.1 - Introdução.....	93
7.2 - Resultados Do Processo De Binarização.....	93
7.3 - Resultados Da Correção Horizontal.....	95
7.4 - Resultados Obtidos Da Segmentação De Palavras Manuscritas	96
7.5 - Critérios De Avaliação	97
7.6 - Resultados Experimentais Utilizando-se O Processo De Segmentação Desenvolvido Neste Trabalho	98
7.6.1 - Grupo 1	98
7.6.2 - Grupo 2.....	100
7.6.3 - Grupo 3.....	103
7.7 - Resultados Obtidos.....	104
7.8 - Conclusão	107
8 - CONCLUSÕES, CONTRIBUIÇÕES DO TRABALHO E SUGESTÕES PARA FUTUROS TRABALHOS.....	108
8.1 - Conclusões.....	108
8.2 - Contribuições Do Trabalho.....	111

8.3 - Sugestões Para Trabalhos Futuros	111
REFERÊNCIAS.....	112
LISTA DE PUBLICAÇÕES	116

Lista de Figuras

Figura 2.1 - Sistema reconhecedor genérico.	9
Figura 2.2 - Imagem em níveis de cinza.	13
Figura 2.3 - Imagem binarizada.	14
Figura 2.4 - Imagem antes de aplicar o método de Otsu.	15
Figura 2.5 - Imagem binarizada após a aplicação do método de Otsu.	15
Figura 2.6 - Imagem apresentando inclinação.	16
Figura 2.7 - Imagem corrigida utilizando a técnica de Kimura.	17
Figura 2.8 - Imagem inclinada.	17
Figura 2.9 - Imagem corrigida utilizando a técnica de Morita.	18
Figura 2.10 - Resultado da Correção vertical após utilizar a técnica de Bozinovic e Shihari. .	19
Figura 2.11 - (a) Imagem original binarizada, (b) Histograma vertical da imagem original, (c) Imagem corrigida, (d) Histograma vertical da imagem corrigida.	20
Figura 2.12 - Imagens de palavras após terem sido segmentadas pelo algoritmo de segmentação morfológica.	22
Figura 2.13 - Segmentação de palavras utilizando o algoritmo de Han.	23
Figura 3.1 - Exemplos de elementos estruturantes binários.	28
Figura 3.2 - Interação do elemento estruturante com a imagem binária.	28
Figura 4.1 - Sistema proposto de pré-processamento e segmentação de caracteres.	36
Figura 5.1 - Máscara para o filtro médio.	39
Figura 5.2 - Aplicação do filtro médio.	39
Figura 5.3 - Máscaras utilizadas no filtro para remoção de picos.	40
Figura 5.4 - Máscaras utilizadas para remoção de vales.	40
Figura 5.5 - (a) Imagem antes de aplicar o filtro de remoção de picos e vales, (b) Imagem antes de aplicar o filtro de remoção de picos e vales ampliada, (c) Imagem após aplicar o filtro de remoção de picos e vales.	41
Figura 5.6 - (a) Imagem antes de utilizar o método de Otsu, (b) Imagem após utilizar o método de Otsu.	43
Figura 5.7 - (a) Imagem binarizada, (b) Histograma horizontal da imagem binarizada.	44
Figura 5.8 - Determinação das linhas de referências.	45
Figura 5.9 - Aplicação da tolerância para o cálculo das novas linhas de referências.	46
Figura 5.10 - Localização da linha central na imagem.	48
Figura 5.11 - Imagem binarizada e seu histograma vertical.	49

Figura 5.12 - Aplicação da segmentação natural na imagem.	50
Figura 5.13 – (a) Máscaras utilizadas na verificação se o contorno aumenta, (b) Máscaras utilizadas na verificação se o contorno estabiliza, (c) Máscaras utilizadas na verificação se o contorno diminui.	50
Figura 5.14 – Eixos de coordenadas.	51
Figura 5.15 - Localização dos pontos de mínimo de uma imagem.	51
Figura 5.16 – (a) Imagem antes de eliminar os mínimos indesejáveis, (b) Imagem após eliminar os mínimos indesejáveis.	54
Figura 5.17 – Imagem da palavra com declive.	54
Figura 5.18 – Imagem da Figura 5.17 após aplicar a correção da inclinação horizontal.	55
Figura 5.19 – (a) Imagem de uma palavra após a correção horizontal, (b) Histograma vertical da palavra.	56
Figura 5.20 – (a) Imagem da palavra corrigida verticalmente, (b) Histograma vertical da imagem corrigida verticalmente.	59
Figura 5.21 - Máscaras utilizadas para a verificação do contorno.	60
Figura 5.22 - Máscaras para verificação da primeira condição do método de apagamento do contorno.	61
Figura 5.23 - Verificação dos pixels vizinhos de um pixel, analisando quando ele possui apenas um pixel vizinho.	62
Figura 5.24 - Verificação dos pixels vizinhos do pixel, que esta sendo analisado, quando ele possui dois ou mais pixels vizinhos.	62
Figura 5.25 – (a) Imagem antes de ser submetida a correção horizontal, (b) Imagem após ser submetida a correção horizontal.	63
Figura 5.26 – (a) Imagem antes de ser submetida a correção horizontal, (b) Imagem após ser submetida a correção horizontal.	63
Figura 5.27 - (a) Imagem antes de ser submetida a correção horizontal, (b) Imagem após ser submetida a correção horizontal.	64
Figura 5.28 – (a) Imagem antes de ser submetida a correção horizontal, (b) Imagem após ser submetida a correção horizontal.	64
Figura 5.29 – (a) Imagem antes de ser submetida ao algoritmo de afilamento, (b) Imagem após ser submetida ao algoritmo de afilamento.	65
Figura 5.30 – (a) Imagem antes de ser submetida ao algoritmo de afilamento, (b) Imagem após ser submetida ao algoritmo de afilamento.	65
Figura 6.1 - Localização das zonas de referência utilizando o histograma horizontal.	69

Figura 6.2 - Imagem de uma palavra que possui as três zonas de delimitação.	70
Figura 6.3 - Imagem de uma palavra que possui apenas uma zona de delimitação.	70
Figura 6.4 - Imagem de uma palavra que possui duas zonas de delimitação.	70
Figura 6.5 - Máscara utilizada para a localização da vizinhança longa.	72
Figura 6.6 – (a) Imagem da palavra antes de aplicar a vizinhança longa, (b) Imagem após apagar os pixels que não pertencem a vizinhança longa.	73
Figura 6.7 - Isolamento dos loops da palavra da imagem da Figura 6.6(a).	73
Figura 6.8 – (a) Máscaras que verificam se o contorno diminui, (b) Máscaras que verificam se o contorno estabiliza, (c) Máscaras que verificam se o contorno aumentam.	74
Figura 6.9 - Localização dos pontos de máximo na imagem de uma palavra.	75
Figura 6.10 - Pontos de máximo localizados na imagem de uma palavra.	75
Figura 6.11 - Máscara utilizada para localizar os pontos de cruzamento na imagem de uma palavra.	76
Figura 6.12 - Pontos de cruzamento em excesso na imagem.	77
Figura 6.13 - Janela utilizada para verificar se existe ponto de cruzamento em excesso na imagem de uma palavra.	77
Figura 6.14 – (a) Imagem da palavra escrita com letra de fôrma, (b) Imagem da palavra escrita com letra cursiva.	79
Figura 6.15 – (a) Imagem da palavra antes de ser submetida a operação morfológica de espessamento, (b) Imagem da palavra após a operação morfológica de espaçamento.	80
Figura 6.16 - Localização das coordenadas de segmentação do loop.....	81
Figura 6.17 - Aplicação do custo na coluna de corte na imagem de uma palavra.....	82
Figura 6.18 - Imagem da palavra manuscrita segmentada naturalmente.	83
Figura 6.19 - Eliminação dos pontos de mínimos nos loops da imagem de uma palavra.....	84
Figura 6.20 - Verificação do custo de corte da imagem de uma palavra manuscrita.	85
Figura 6.21 - Eliminação dos mínimos nos falsos loops de imagem de uma palavra manuscrita.....	85
Figura 6.22 - Eliminação dos mínimos na zona inferior da imagem de uma palavra manuscrita.	86
Figura 6.23 - Verificação das coordenadas dos loops na imagem de uma palavra manuscrita.	87
Figura 6.24 - Verificação dos pontos de cruzamento na zona mediana da imagem de uma palavra manuscrita.....	88

Figura 6.25 – (a) Imagem da palavra manuscrita com tons de níveis de cinza antes de ser segmentada, (b) Imagem da palavra manuscrita após ser binarizada e segmentada.....	89
Figura 6.26 - (a) Imagem da palavra manuscrita com tons de níveis de cinza antes de ser segmentada, (b) Imagem da palavra manuscrita após ser binarizada e segmentada.....	90
Figura 6.27 - (a) Imagem da palavra manuscrita com tons de níveis de cinza antes de ser segmentada, (b) Imagem da palavra manuscrita após ser binarizada e segmentada.....	90
Figura 6.28 - (a) Imagem da palavra manuscrita com tons de níveis de cinza antes de ser segmentada, (b) Imagem da palavra manuscrita após ser binarizada e segmentada.....	91
Figura 7.1 - Imagem de uma palavra manuscrita gravada com tons de níveis de cinza de alta resolução na escala de 300 ppi, com 8 bits.	94
Figura 7.2 - Imagem binarizada utilizando somente o método deOtsu.....	94
Figura 7.3 - Imagem após ser aplicado nela o filtro mediano, o método de Otsu e os filtros de remoção de picos e vales.	95
Figura 7.4 – (a) Imagem com tons de níveis de cinza antes de aplicar o método de correção horizontal desenvolvido neste trabalho, (b) Imagem após aplicar o processo de binarização e o método de correção horizontal desenvolvido neste trabalho.....	96
Figura 7.5 – Imagem da palavra “Troy” inclinada para a direita gravada em tons de níveis de cinza.....	98
Figura 7.6 – Segmentação utilizando o histograma vertical.....	99
Figura 7.7 – Imagem da palavra “Livingston” gravada em tons de níveis de cinza.....	99
Figura 7.8 – Resultado da segmentação natural da palavra “Livingston”.....	100
Figura 7.9 – Imagem da palavra “Wheat” gravada com tons de cinza.....	100
Figura 7.10 – Imagem da palavra “Wheat” segmentada naturalmente.....	100
Figura 7.11- Imagem da palavra “Stuttgart” gravada com tons de níveis de cinza.....	101
Figura 7.12 – Imagem da palavra “Stuttgart” segmentada pelo algoritmo utilizado neste trabalho.....	101
Figura 7.13 – Imagem da palavra “Fairfield” gravada com tons de níveis de cinza.....	102
Figura 7.14 - Imagem da palavra “Fairfield” segmentada pelo algoritmo utilizado neste trabalho.....	102
Figura 7.15 - Imagem da palavra “Tanauaanda” gravada com tons de níveis de cinza.....	103
Figura 7.16 – Imagem da palavra “Tanauaanda” segmentada pelo algoritmo utilizado neste trabalho.....	103
Figura 7.17 - Imagem da palavra “Ardmore” gravada com tons de níveis de cinza.....	104

Figura 7.18 – Imagem da palavra “Ardmore” segmentada pelo algoritmo utilizado neste trabalho..... 104

LISTA DE TABELAS

Tabela 7.1 – Resultados obtidos nos grupos 1, 2, 3 utilizando as técnicas de pré-processamento e segmentação desenvolvidas neste trabalho.....	105
Tabela 7.2 – Resultado final obtido com a média dos grupos 1, 2, 3.....	106
Tabela 7.3 - Resultados obtidos por Xiao e Leedham.....	107

Capítulo 1

INTRODUÇÃO

1.1 - Motivação

A atribuição de tarefas humanas às máquinas, tais como o ato de falar, ler, e caminhar, é algo que tem atraído a atenção de pesquisadores do mundo todo. Este tema tem sido muito pesquisado nas últimas três décadas, e com o desenvolvimento de técnicas de inteligência artificial, muitos trabalhos promissores foram apresentados. Em se tratando, especificamente, da habilidade de leitura de textos, quer sejam eles manuscritos ou impressos, pode-se afirmar que ela tem representado um grande desafio na sua automatização. Embora para os seres humanos esta tarefa seja relativamente simples, pois, desde os primeiros anos de vida, eles são treinados para reconhecer padrões tanto no ato da leitura quanto na escrita, quando se trata de atribuir essa mesma tarefa a uma máquina, o processo torna-se bastante complicado devido, principalmente, à dificuldade no tratamento da infinidade de maneiras da forma de escrever do homem. A automatização da leitura de textos é por si só um atraente desafio científico mas, é também, uma solução para tratar problemas reais de nossa vida moderna. Pode-se verificar que, diariamente, quantidades fabulosas de documentos e papéis são geradas, e devem ser processados com uma velocidade cada vez maior. Além disso, a leitura automatizada de textos pode trazer benefícios sociais significativos para pessoas que tem deficiências visuais. Assim, um sistema de reconhecimento de escrita faz se necessário para a solução de problemas que são vivenciados dia a dia e que, sem dúvida, poderá trazer bem estar social.

Basicamente, duas abordagens são adotadas em relação ao tipo de texto: o reconhecimento de palavras escritas a máquina e o reconhecimento de palavras escritas a mão.

Sistemas de reconhecimento de palavras escritas a máquina, os chamados reconhecimento óptico de caracteres (OCRs), foram os primeiros a serem pesquisados e desenvolvidos. Atualmente, soluções eficientes podem ser encontradas a nível comercial. Tecnicamente, as letras impressas apresentam um certo grau de facilidade no momento do reconhecimento, visto que essas palavras, geralmente, se apresentam em tamanhos uniformes, posições definidas das letras, e o mesmo estilo de letra ou fonte. Já, no caso de palavras escritas a mão, ou manuscritas, esta uniformidade não é encontrada. Elas apresentam tamanhos irregulares, posições diversas e, ainda, estilos diferentes na forma de escrever conforme a pessoa que escreve. Essas variações ocorrem com diferentes escritores e, até mesmo, com o mesmo escritor. Devido a estas características típicas da escrita manual, torna-se difícil o desenvolvimento de um sistema automatizado de reconhecimento com uma taxa de reconhecimento satisfatória.

A diversidade na escrita manual ocorre devido aos diferentes hábitos e estilos entre os escritores. Isto tem imposto algumas restrições ao desenvolvimento de sistemas de reconhecimento, tais como, limitação do número de palavras a serem reconhecidos, número limitado de escritores a serem tratados para diminuir a variedade de estilos de escrita, pois, não sendo assim, uma taxa satisfatória de reconhecimento de palavras dificilmente poderá ser alcançada. A grande motivação entre os pesquisadores que trabalham nesta área tem sido a busca de soluções, onde essas limitações possam ser ultrapassadas. Nessa tentativa, uma solução seria a implantação de um método com grande capacidade de generalização das entradas de um sistema reconhecedor, para que possam ser tratadas estas variações que ocorrem nas diversas formas de escrita. Para alcançar este objetivo, foi desenvolvido neste

trabalho de pesquisa, uma etapa de pré-processamento das imagens das palavras. Esta etapa recebe como entrada uma imagem original em nível de cinza e, são aplicados, a ela inicialmente, filtros com o objetivo de remover ruídos indesejáveis no processo de reconhecimento. Em seguida, é realizado um processo de binarização da imagem, ou seja, ela é transformada em pixels brancos e pretos, preservando o máximo de informações sobre os caracteres. Como a escrita manual, geralmente, se apresenta com algum tipo de inclinação, mecanismos de correção das inclinações ocorridas no momento da escrita são aplicados. Estas inclinações podem ser com relação a uma vertical e, também, com relação à horizontal. Finalmente, é realizado um processo de afilamento da imagem para que os traços fiquem na espessura de um pixel. Este afilamento é realizado para que as informações da escrita como, pontos finais, pontos de cruzamento, loops, pontos de máximo, pontos de mínimos possam ser eficientemente determinados. Todas estas informações serão utilizadas na etapa de segmentação.

Para a separação das letras que compõem a palavra manuscrita, tarefa esta que se faz necessária para que, individualmente, cada caractere possa ser introduzido e tratado no sistema reconhecedor, mecanismos de segmentação foram desenvolvidos e aplicados nas imagens das palavras binarizadas.

1.2 - Objetivos Do Trabalho

Em um sistema OCR, a etapa do pré-processamento, juntamente com a etapa de segmentação possui uma grande responsabilidade e influenciam decisivamente no sucesso do sistema reconhecedor automático de palavras. Assim, a presente pesquisa concentrou-se no desenvolvimento e na implementação destas duas etapas. Todo o processo foi desenvolvido baseando-se apenas nas características estáticas das imagens originais, ou seja, o sistema de reconhecimento de palavras é “off-line”, por tanto, estão disponíveis apenas as informações

que podem ser tratadas através da aplicação de técnicas de processamento digital de imagens, tendo como objetivo final, a extração das características essenciais presentes na composição de uma letra, tais como, loops, retas e curvas.

Os objetivos principais desta dissertação são:

- Propor melhoria nos métodos de pré-processamento de imagens de palavras manuscritas.
- Propor um algoritmo de segmentação de imagens de palavras manuscritas, baseado nas características da imagem, tais como: loops, ponto final, ponto de mínimo e ponto de máximo.
- Sugerir estudos futuros baseados no trabalho desenvolvido nesta dissertação.

1.3 - Imagens Manuscritas Utilizadas

As imagens utilizadas nesta pesquisa são oriundas da base de dados do Centro de Excelência para Análise e Reconhecimento de Documentos (CEDAR), da Universidade de Buffalo, Nova Iorque, desenvolvido especialmente para pesquisas de análise e reconhecimento de documentos. O CEDAR, em parceria com os correios norte americanos, desenvolveu esta base de dados que tem sido uma referência para as várias pesquisas de todo o mundo. Esta base de dados é constituída de imagens de manuscritos de correspondências que passaram por postos de correio. São imagens de:

- nomes de ruas, estados e cidades;
- códigos postais;
- dígitos; e
- caracteres alfanuméricos.

O grande desafio no desenvolvimento de pesquisas utilizando esta base de dados é que suas imagens preservem todas as características da imagem original, tendo sido obtidas por um sistema de aquisição de imagens, em níveis de cinza, e na resolução de 300 dpi. Estas imagens representam uma situação real no processamento de palavras manuscritas, devido aos seguintes fatos:

- sobre as imagens não é aplicado qualquer tipo de tratamento gráfico, dando a liberdade para o pesquisador desenvolver e aplicar seu próprio sistema de pré-processamento; e
- existência de uma grande diversidade de escritores, visto que a quantidade de correspondência que circula nos correios diariamente, é muito alta.

1.4 - Estrutura Da Dissertação

A presente dissertação foi dividida nos seguintes capítulos:

- Capítulo 1: Apresentação da motivação, estado da arte dentro do tema, e os objetivos a serem alcançados na presente pesquisa.
- Capítulo 2 Discussão sobre mecanismos de pré-processamento e segmentação de palavras manuscritas, visando obter informações sobre os principais métodos empregados em pesquisas anteriores.
- Capítulo 3: Apresentação do histórico sobre o tratamento de imagens e operações morfológicas.
- Capítulo 4: Especificação do sistema desenvolvido e suas diversas partes.
- Capítulo 5: Descrição das técnicas desenvolvidas e utilizadas na etapa de pré-processamento.
- Capítulo 6: Descrição das técnicas propostas na etapa de segmentação dos caracteres.

- Capítulo 7: Apresentação dos resultados obtidos com o sistema proposto, e uma análise comparativa com outros trabalhos, destacando a eficiência e a consistência dos algoritmos empregados.
- Capítulo 8: São apresentadas as conclusões e realizada uma análise geral de todo o trabalho de pesquisa, e listadas algumas sugestões para serem desenvolvidas como trabalhos futuros.

Capítulo 2

ESTADO DA ARTE EM PRÉ-PROCESSAMENTO E SEGMENTAÇÃO DE IMAGENS

2.1 - Introdução

Neste capítulo, é apresentado um pequeno histórico do sistema automático de reconhecimento de caracteres, e alguns trabalhos que se destacam nas áreas de pré-processamento e de segmentação de caracteres.

2.2 - Histórico De OCR's

O reconhecimento de padrões é um ramo da ciência que trata da classificação de objetos de acordo com as suas características intrínsecas. As suas origens datam da época do surgimento dos computadores, podendo-se encontrar trabalhos de pesquisas relacionadas ao reconhecimento de imagens de dígitos, e imagens de palavras ao longo de todas essas décadas.

Na década de 40, com o surgimento dos computadores digitais, tornou-se possível o desenvolvimento do padrão de OCR, e que se assemelha muito ao atual. Desde então, muitos métodos foram desenvolvidos para satisfazer às exigências cada vez maiores na aplicação de tais sistemas. Com a sua evolução, a sua utilização tem sido verificada em varias aplicações, tais como: leitura automática de endereços postais, leitura de cheques bancários, e sistemas de auxílio aos deficientes visuais [1].

Pode se dizer que, ainda hoje, uma grande parte dos documentos existentes no mundo estão em forma de papéis, ocupando um enorme espaço físico para acomodar os mesmos, exigindo um enorme trabalho manual no seu processamento. Com o surgimento de OCRs, esta tarefa veio a ser facilitada, permitindo tratar estes documento com mais eficiência e rapidez, além da diminuição do espaço físico para armazenamento [2].

Em se tratando de processamento automático de documentos, podendo os mesmos se apresentarem na forma de textos e gráficos, quando convertidos em imagens digitais, o objetivo é a extração das informações desejadas [3], baseando-se nos princípios empregados por seres humanos para o desempenho de tal tarefa. Em muitas aplicações, os conteúdos a serem processados são palavras manuscritas ou impressas à máquina. O seu reconhecimento automático tem alcançado um progresso significativo nos últimos 30 anos, graças ao desenvolvimento de muitas técnicas eficazes para resolver o problema, sendo que os melhores sistemas propostos tem alcançado resultados de reconhecimento correto na faixa de 90% [2].

2.2.1 - O sistema reconhecedor padrão

Os sistemas reconhecedores de padrões, geralmente seguem o modelo mostrado na figura 2.1. Sendo baseado na forma de aquisição das palavras, eles são classificados em duas grandes categorias: sistema on-line e sistema off-line [4].

No sistema **on-line**, os dados são recebidos diretamente de algum tipo de dispositivo de entrada que está conectado diretamente ao computador, sendo possível obter informações dinâmicas como: a ordenação dos traços, a seqüência do traçado da escrita, a sobreposição dos traços, e a posição correta da localização de cada um destes traços. A imagem é processada em tempo real, na medida em que é efetivado o processo de escrita. Estas informações dinâmicas são fundamentais para o processo de reconhecimento [5].

Nos sistemas **off-line**, a entrada de dados ocorre através de dispositivos de aquisição de imagens como os “scanners”, que capturam as imagens da escrita. Ou seja, o processo de escrita já foi realizado em um momento anterior e o que se tem disponível é a imagem que foi escrita. Portanto, os dados dinâmicos que são facilmente obtidos no sistema on-line, não são mais possíveis de serem coletados e utilizados nas etapas posteriores de reconhecimento. Assim, as informações relevantes da escrita devem ser obtidas a partir da imagem estática, o que traz uma dificuldade a mais para os sistemas de reconhecimento off-line [6].

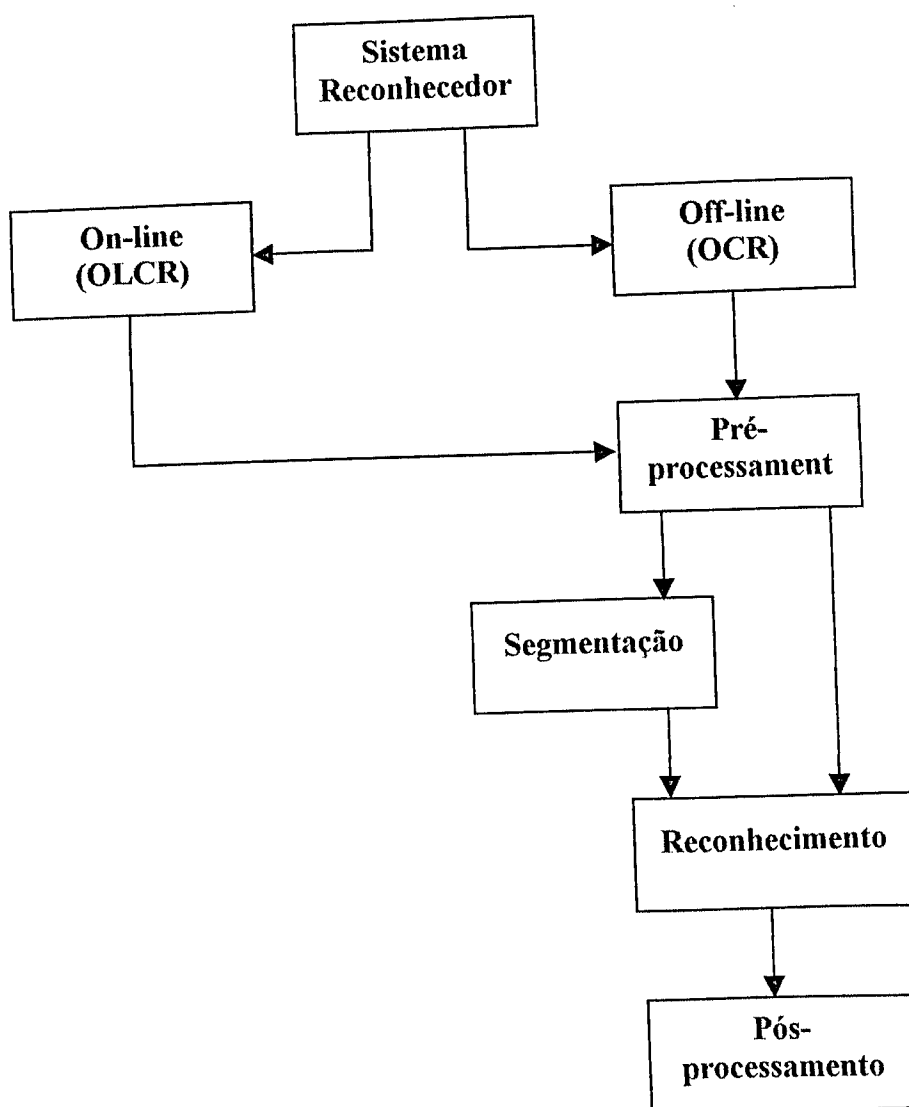


Figura 2.1 - Sistema reconhecedor genérico.

Mas os sistemas on-line também não estão isentos de problemas que dificultam o reconhecimento das palavras. Um exemplo de problema seria a dificuldade em trabalhar os traços que são postergados na escrita da palavra, como o caso do corte da letra t e o ponto sobre a letra i.

Após a distinção quanto à forma de aquisição dos dados, ambos os sistemas passam através da etapa de pré-processamento, onde são aplicados algoritmos que fazem a binarização da imagem, a correção das distorções (inclinações horizontal e vertical) sofridas pela imagem, e a eliminação dos ruídos.

Em seguida, com a finalização da etapa de pré-processamento, um sistema típico pode adotar, na etapa de análise da composição da palavra, uma das três possíveis abordagens, sendo estas baseadas na inclusão ou não do processo de segmentação [7]. Estas abordagens são:

- Abordagem global. Esta abordagem considera a palavra como uma entidade única. As características para a sua classificação são extraídas da imagem como um todo, extraíndo, basicamente, as informações de seu contorno, ou seja, a palavra não é segmentada, e a classificação é feita em conformidade com o sistema classificador de palavras, que verifica a sua base de informações para fornecer uma resposta [6].
- Na abordagem baseada em segmentação, a imagem da palavra é particionada em uma seqüência de segmentos, onde esses segmentos, que podem constituir um caractere, são reconhecidos através do classificador de caracteres. Em seguida, os segmentos reconhecidos são comparados com as possíveis letras, produzindo-se, assim, a sua classificação [8].
- Abordagem híbrida. Essa abordagem combina o método de abordagem global com o método baseado em segmentação, em um único processo de reconhecimento [7].

Sistemas de reconhecimento baseado na abordagem global têm sido utilizado [7] nos casos em que o vocabulário é limitado, por exemplo, no reconhecimento de cheques

bancários. Quando se trata de um vocabulário maior, esta abordagem torna-se ineficiente, pois, neste caso, faz-se necessário manter armazenado um grande número de contornos de palavras para serem comparadas com as imagens de entrada. Logo, o sistema de reconhecimento irá exigir uma grande quantidade de espaço para o armazenamento dessas palavras. Como, geralmente, o reconhecimento é baseado em algum processo de comparação, esta tarefa deverá tomar considerável tempo de execução. Além disso, à medida em que o número de palavras no vocabulário for aumentando, seus contornos irão se assemelhando cada vez mais, dificultando o reconhecimento. Outra desvantagem da abordagem global é que ele não reconhece palavras escritas incorretamente, a menos que ela tenha sido adicionada ao vocabulário da sua base de dados. O sistema global só é vantajoso se no vocabulário das palavras a serem reconhecidas for bastante reduzido.

A abordagem baseada em segmentação de caracteres [6, 7, 8, 9] tem sido bastante empregada no tratamento de grandes vocabulários, porque ela trabalha com as letras e não com a palavra, sendo assim, a quantidade de dados a serem mantidos como vocabulário é bem menor que a abordagem global. Mas, no sistema baseado em segmentação, o grande problema que se apresenta é em relação à localização, com exatidão, do limite entre as letras. Em se tratando de letras cursivas, elas se interconectam na palavra, dificultando a sua segmentação. Casos difíceis de serem tratados são a justaposição das letras como, por exemplo, as combinações **ui**, **vv**, **wi**. Para contornar este problema, uma solução que tem sido proposta é forçar, em determinadas situações, as pessoas a escreverem dentro de caixas com tamanho definido, dividindo, assim, as palavras em seus caracteres constituintes. Esta solução tem sido adotada nos casos de preenchimento de formulários mas, trata-se de uma solução não natural.

Para o tratamento da escrita cursiva, como no caso de nomes e endereços presentes em correspondências, a solução ainda é a segmentação. Faz-se necessário, então, o

desenvolvimento de mecanismos eficientes para que o particionamento da palavra possa ser realizado com sucesso.

Na tentativa de melhorar os resultados do sistema reconhecedor, ultimamente os pesquisadores têm incorporado várias experiências com Redes Neurais Artificiais [9] e, também, com morfologia matemática [10, 11] no processo de segmentação.

Após a decisão de incluir ou não a etapa de segmentação, o sistema passa para a etapa de reconhecimento propriamente dito, que tem como objetivo classificar a palavra de entrada gerando palavras ou letras como possíveis candidatas.

Logo em seguida, o sistema passa para a etapa de pós-processamento verificando se a palavra ou a letra candidata, gerada pela etapa de reconhecimento, pode constituir uma palavra que pertence a um vocabulário pré-definido. Se for encontrada a palavra candidata no vocabulário, esta será a saída do sistema de reconhecimento automático. Se a palavra não existe no vocabulário, então a etapa de segmentação deverá ser realimentada para corrigir um possível erro de posicionamento dos pontos de segmentação.

2.3 - Algumas Publicações Importantes Sobre Pré-processamento.

Para o desenvolvimento desta pesquisa, foram analisados vários trabalhos que tratam de sistemas reconhecedores. Como o objetivo do presente trabalho de pesquisa é propor soluções para as etapas de pré-processamento e segmentação, são apresentados a seguir, alguns trabalhos relacionados a estes dois temas.

2.3.1 - Trabalhos na etapa de pré-processamento

A etapa de pré-processamento consiste, basicamente, nas sub-etapas a seguir: binarização, correção horizontal, correção vertical, normalização do tamanho e afilamento da

imagem. Essas sub-etapas buscam eliminar as distorções existentes na escrita manual, com o objetivo de facilitar o processamento da etapa posterior de segmentação. E, contribuir no aumento da taxa de acertos na etapa de reconhecimento.

2.3.1.1 - binarização

O método de binarização consiste em transformar uma imagem com níveis de cinza, em uma imagem preto e branco. O objetivo de realizar esta transformação é facilitar o trabalho. A convenção adotada para os pixels binarizados é o fundo da imagem é representado pelo bit zero, e o objeto é representado pelo bit 1, diminuindo assim, a quantidade de níveis de cinza da imagem a ser analisada e aumentando o desempenho do sistema de processamento da imagem.

2.3.1.1.1 - método de pré-processamento quadrático adaptativo

O método de pré-processamento quadrático adaptativo é similar ao “blind adaptive equalization” dos canais de comunicação binária. Essa similaridade se deve ao fato de que em um sistema de comunicação binária, é recebido um sinal distorcido e ruidoso, da mesma forma em que um processador de imagens é recebida uma imagem distorcida e ruidosa [13].

A Figura 2.2 mostra uma imagem com vários níveis de cinza.

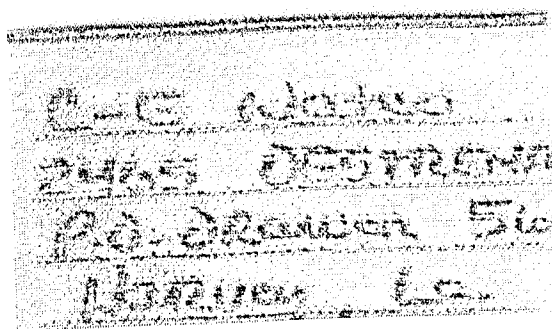


Figura 2.2 - Imagem em níveis de cinza.

Um equalizador é projetado para reduzir os efeitos dos ruídos nos sinais, assim como, a etapa de pré-processamento de imagens deve ser concebida para compensar as distorções no documento original. A Figura 2.3 mostra uma imagem binarizada.

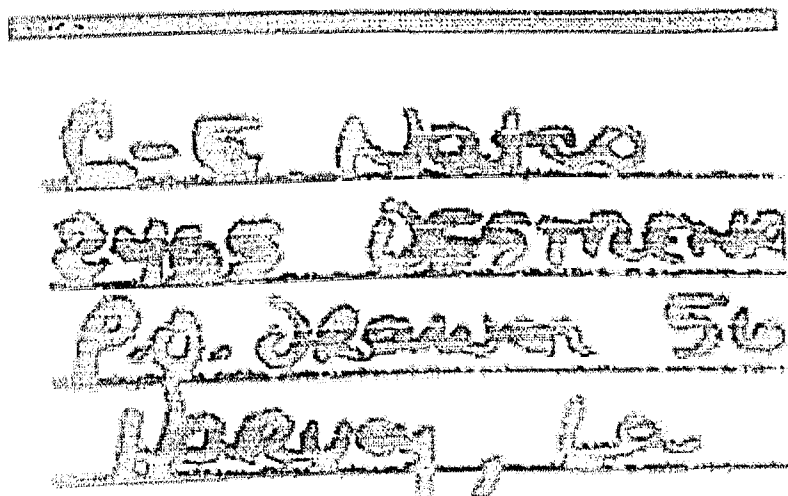


Figura 2.3 - Imagem binarizada.

O método para comparar as distorções no documento original utiliza um filtro quadrático não linear para satisfazer às necessidades de realce de bordas e redução dos ruídos ou distorções presentes nas imagens dos caracteres, para se conseguir uma imagem binarizada de melhor qualidade possível. Como exemplo, pode-se observar a Figura 2.2, onde é mostrado uma imagem com níveis de cinza, e a Figura 2.3 mostra a mesma imagem binarizada.

2.3.1.1.2 - método de Otsu

O método de Otsu [14] emprega procedimentos estatísticos para selecionar o melhor limiar do nível de cinza na imagem que permita uma divisão entre os pixels de fundo e os pixels do objeto da imagem.

Tal procedimento se resume no cálculo da máxima variância σ entre o objeto e o fundo, com limiar k , que faz a delimitação entre eles. A figura 2.4 mostra uma imagem antes de aplicar o método de Otsu.

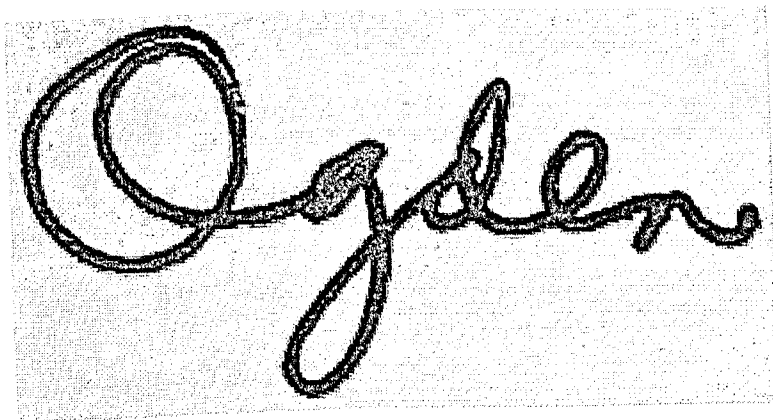


Figura 2.4 - Imagem antes de aplicar o método de Otsu.

O resultado da aplicação do método de Otsu é mostrado na Figura 2.5.

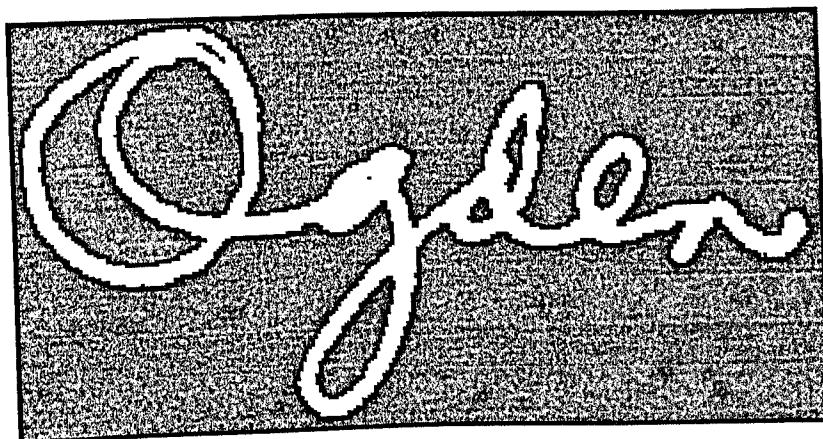


Figura 2.5 - Imagem binarizada após a aplicação do método de Otsu.

2.3.1.2 - correção horizontal

A correção horizontal se faz necessária quando os escritores escrevem as palavras sem seguir uma linha de referência horizontal. Assim, para facilitar as tarefas das etapas

posteriores de reconhecimento, todas as letras são corrigidas para a mesma referência horizontal. A correção a ser realizada é definida baseando-se no cálculo do ângulo entre o eixo horizontal e a direção média de inclinação da palavra. Essa inclinação pode ser um auge ou um declive. O ângulo será zero quando a palavra estiver escrita sem inclinação, não exigindo, neste caso, a realização da correção horizontal.

2.3.1.2.1 - técnica de Kimura

Kimura [15], juntamente com outros pesquisadores, descreveu um sistema de correção horizontal baseado na análise das regiões de fundo da imagem, onde cada pixel pertencente ao fundo da imagem é analisado, verificando se ele pertence a região LCH (lower convex hulls), ou seja, para cada pixel com valor zero, é verificado se ele possui vizinhos nas posições superior, direita e esquerda.

Após a localização da região LCH ocorre a ligação de todas as regiões inferiores na imagem desta região. Os resultados obtidos por Kimura para 439 imagens ficaram em torno de 92,8% de acerto de correção. A Figura 2.6 mostra uma imagem apresentando inclinação e a Figura 2.7 ilustra uma imagem corrigida utilizando a técnica de Kimura.

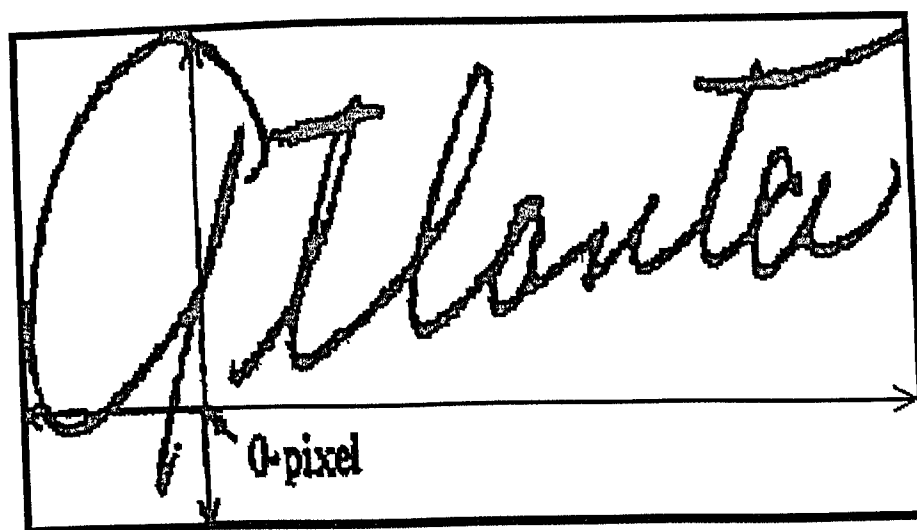


Figura 2.6 - Imagem apresentando inclinação.

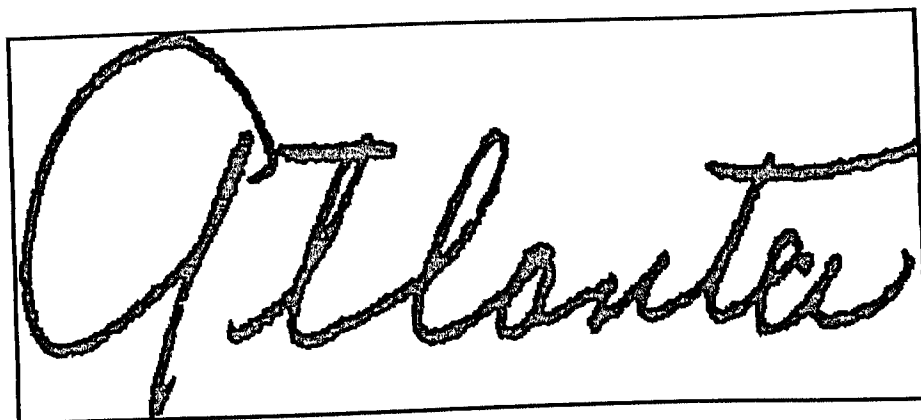


Figura 2.7 - Imagem corrigida utilizando a técnica de Kimura.

2.3.1.2.2 - técnica de Morita

A técnica empregada por Morita [16], utiliza várias ferramentas de morfologia matemática, com o objetivo de preparar a imagem, para serem extraídos os pontos de mínimos, que são obtidos a partir do contorno inferior da palavra: Após a detecção dos mínimos no contorno inferior da palavra, é empregada a técnica dos mínimos quadrados para localizar a reta que melhor se adequa à inclinação da palavra. A Figura 2.8 mostra uma imagem inclinada e a Figura 2.9 mostra uma imagem após utilizar a técnica de Morita.

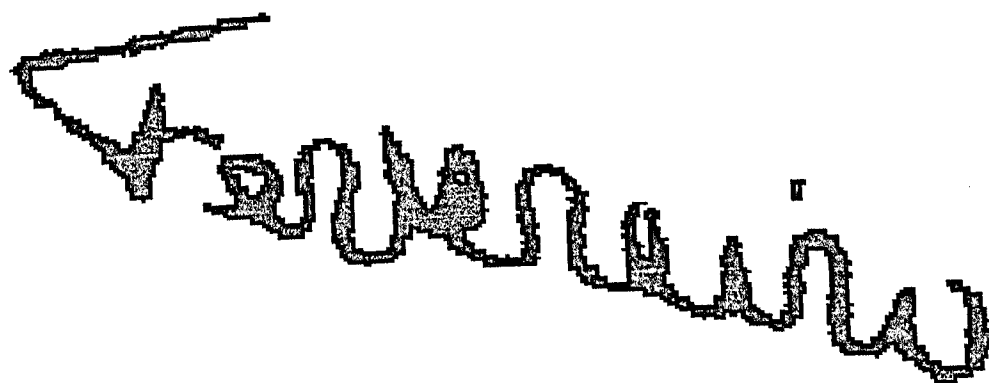


Figura 2.8 - Imagem inclinada.



Figura 2.9 - Imagem corrigida utilizando a técnica de Morita.

2.3.1.3 - correção vertical

A inclinação vertical da palavra é definida como sendo o ângulo entre o eixo vertical e a direção média das inclinações das letras. Assim como na inclinação horizontal, os escritores tendem a escrever suas palavras inclinadas para a direita ou para a esquerda, dificultando o processamento da etapa seguinte de segmentação. Faz-se necessário, então, a sua correção, colocando todas as letras que compõem a palavra na vertical.

2.3.1.3.1 - técnica de Bozinovic e Shihari

A técnica apresentada por Bozinovic e Shihari [4], consiste em delimitar em janelas cada linha vertical da palavra, encontrando os seus respectivos ângulos de inclinação. Para localizar a inclinação resultante é feita uma média entre cada inclinação encontrada. O resultado está mostrados na Figura 2.10.

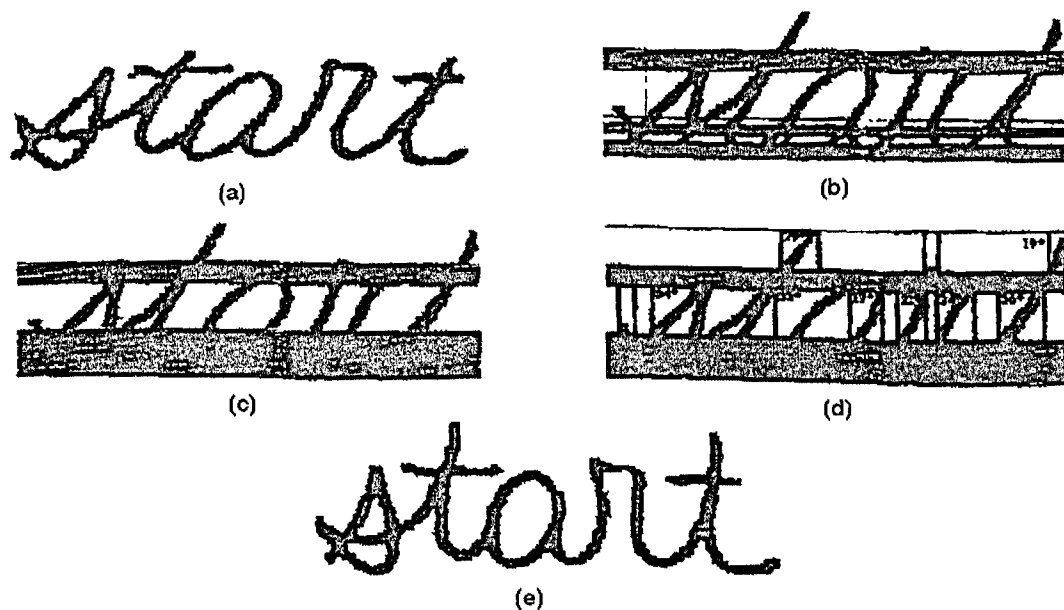


Figura 2.10 - Resultado da Correção vertical após utilizar a técnica de Bozinovic e Shihari.

2.3.1.3.2 - técnica de Buse

A técnica apresentada por Buse, é baseada na entropia mínima do histograma vertical para calcular a inclinação da palavra [17], a idéia básica é calcular o histograma vertical para cada imagem inclinada utilizando ângulos distintos. A inclinação dominante da palavra é determinada pela entropia mínima do histograma vertical. Assim a entropia fornece a média do tamanho da palavra na vertical para cada histograma vertical calculado. Para localizar o ângulo resultante, é localizada no histograma onde a entropia é menor comparada com a entropia dos demais histogramas. A Figura 2.11 mostra a imagem original e a imagem corrigida utilizando a técnica de Buse com seus respectivos histogramas.

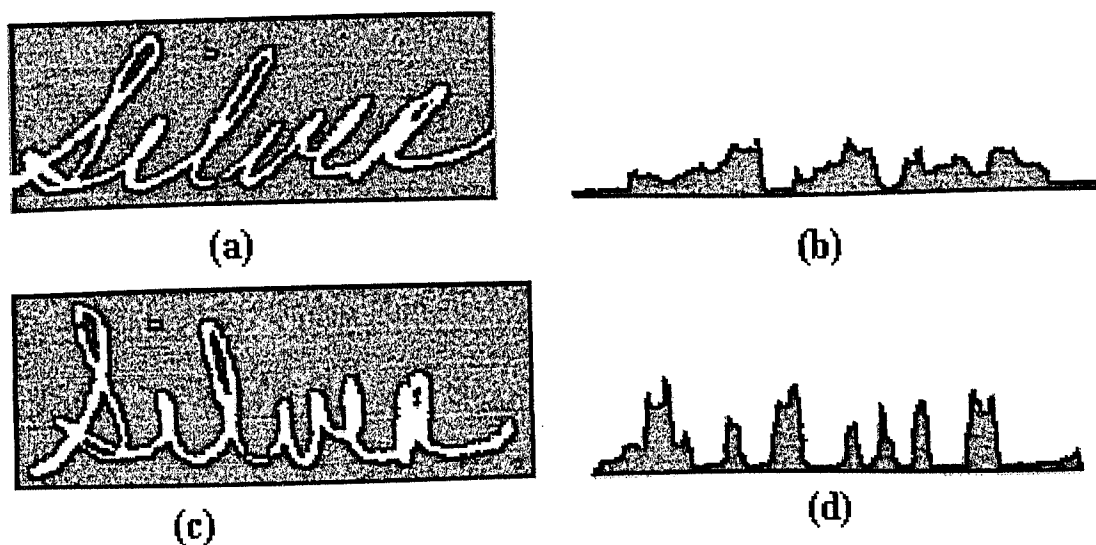


Figura 2.11 - (a) Imagem original binarizada, (b) Histograma vertical da imagem original, (c) Imagem corrigida, (d) Histograma vertical da imagem corrigida.

2.3.1.4 - afilamento

O objetivo de realizar o afilamento, em uma imagem é reduzir as informações ao mínimo possível, conservando as características da mesma. Com isso, são obtidas informações importantes, tais como: ponto final e ponto cruzamento.

2.3.1.4.1 - técnica de Han

A algoritmo de Han [18], consiste em analisar os pixel marcados como contorno e verificar se esses pixels podem ou não serem eliminados, de acordo com as máscaras de condições de eliminação, que visam verificar a quantidade de pixels pretos na vizinhança delimitada por uma máscara de 3x3. O algoritmo de Han segue os seguintes passos:

- Calcula a quantidade de pixels pretos na vizinhança da máscara de 3x3.

- Para cada pixel do contorno que possua, uma quantidade de pixel k menor que oito, não verificados os seus vizinhos com as máscaras de eliminação, se as condições de eliminação forem satisfeitas, o pixel é apagado.
- Os passos acima são repetidos, até que nenhuma das condições de eliminação sejam satisfeitas.

2.3.1.4.2 - técnica de Parker

O algoritmo empregado por Parker [19] para obter o afilamento de uma imagem, baseia-se nos vizinhos do pixel em consideração, onde o índice cruzado é calculado assim como transição de estado de cada pixel para o seu vizinho.

2.3.1.5 - trabalhos desenvolvidos na etapa de segmentação

A etapa de segmentação é muito importante quando se trata de reconhecimento de palavras manuscritas empregando um grande vocabulário, pois as palavras são consideradas como uma seqüência de letras, ficando mais fácil para o sistema reconhecedor analisar individualmente cada letra do que analisar a palavra como um todo. A grande dificuldade do processo de segmentação é determinar os pontos de corte que correspondem exatamente a uma letra.

2.3.1.5.1 - segmentação morfológica de palavras cursivas

Este algoritmo de segmentação morfológica de palavras cursivas [10], é baseado em duas hipóteses: Existem pontos de segmentação natural na palavra e os pontos de segmentação que estão localizados entre as ligações das letras.

A localização dos pontos de segmentação natural consiste em analisar a projeção do histograma vertical da imagem em cinco ângulos diferentes, verificando a ausência de pixels.

A localização dos pontos de segmentação entre as ligações das letras é verificada na esperança de encontrar pontos de segmentação, onde essa verificação das ligações é feita através de operações morfológicas de abertura e fechamento. Algoritmos genéticos são utilizados com a finalidade de determinar o melhor elemento estruturante para obter essas ligações.

Alguns resultados obtidos utilizando o algoritmo de segmentação morfológica são mostrados na Figura 2.12, onde algumas letras tais como **u**, **n** e o **m** foram segmentadas incorretamente.

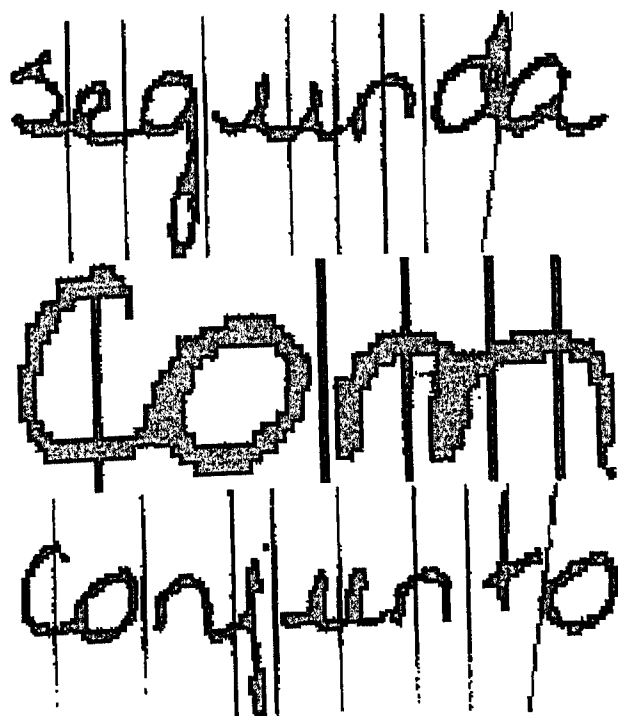


Figura 2.12 – Imagens de palavras após terem sido segmentadas pelo algoritmo de segmentação morfológica.

2.3.1.5.2 - segmentação de palavras manuscritas em off-line

No trabalho de Han [12], a segmentação é obtida através de um conjunto de regras heurísticas que tem por finalidade a localização dos possíveis limites das letras.

As regras heurísticas são baseadas na associação existente entre as características geométricas e topológicas da imagem. O sistema incorpora alguns passos do pré-processamento tais como: normalização e aplicação do algoritmo de esqueletização. A Figura 2.13 mostra a segmentação de palavras utilizando o algoritmo de Han

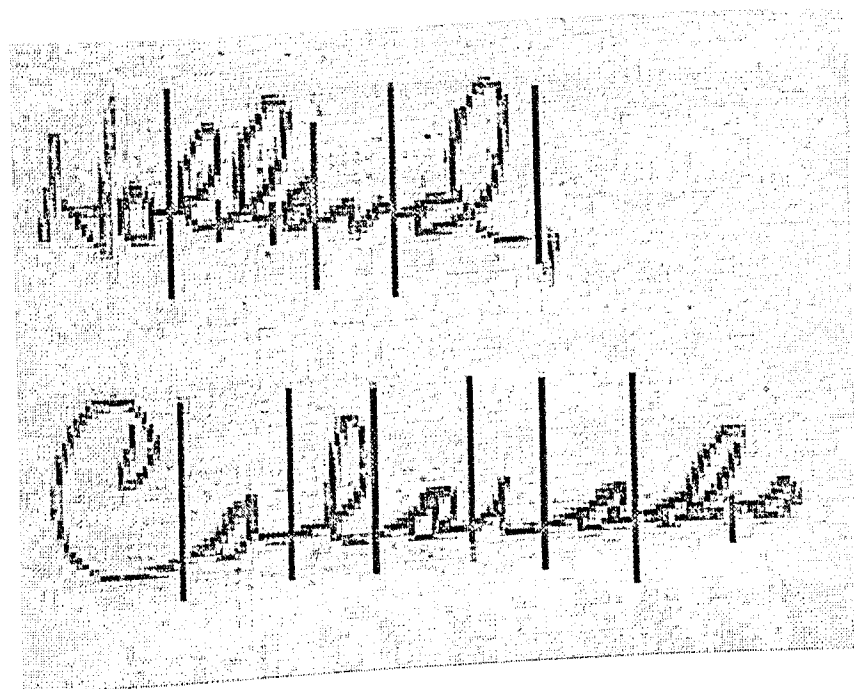


Figura 2.13 - Segmentação de palavras utilizando o algoritmo de Han.

2.4 - Considerações Finais Do Capítulo

Neste capítulo foram pesquisados vários trabalhos considerados como clássicos e contemporâneos, com o objetivo de buscar o que tem sido feito até então, e buscar novas linhas de pesquisas para os problemas relacionados com o tema, pois a tendência é buscar

sempre a melhoria dos resultados dos sistemas de pré-processamento e segmentação para facilitar a etapa de reconhecimento. Seguindo esse objetivo foram apresentadas algumas publicações, mais importantes nesta área.

Capítulo 3

TÉCNICAS DE PROCESSAMENTO DIGITAL DE IMAGENS

3.1 - Introdução

Uma das primeiras aplicações das técnicas de processamento digital de imagens [20] foi com objetivo de melhorar imagens digitalizadas para jornais que eram enviadas por meio de um cabo submarino de Londres para Nova Iorque. O sistema Bartlane de transmissão de imagens via cabo, introduzido no início dos anos 20, reduziu o tempo necessário, que era de mais de uma semana, para menos de 3 horas, para a transmissão de imagens através do oceano Atlântico. Este sistema era capaz de codificar imagens em 5 níveis diferentes de brilho. Em 1929, essa capacidade foi aumentada para 15 níveis. Os melhoramentos continuaram a ser feitos ao longo dos 35 anos que se seguiram. No entanto, foi com o surgimento dos computadores digitais de grande porte e, com o programa espacial americano (quando imagens da Lua foram processadas para corrigir as distorções provocadas pela câmera de televisão a bordo), que as técnicas de processamento de imagens passaram a ganhar maior destaque.

Atualmente, com o desenvolvimento das técnicas de processamento digital de imagens, essa área de pesquisa se dividiu em duas partes: o processamento digital envolvendo a matemática convencional, e processamento de imagens empregando as ferramentas da morfologia matemática.

Este capítulo descreve as operações que são consideradas básicas na morfologia matemática.

3.1.1 - Processamento digital de imagens

Basicamente, o processamento digital de imagens procura adequar as imagens originais para o processamento computacional através da digitalização de sua função $f(x,y)$, tanto espacialmente quanto em amplitude. A digitalização das coordenadas espaciais (x,y) é denominada amostragem da imagem, enquanto que a digitalização da amplitude é denominada quantização em níveis de cinza. Assim, as imagens digitalizadas são representadas na forma de uma matriz de M linhas por N colunas [20]. Cada elemento da matriz representa um pixel da imagem. A função $f(x,y)$ é definida pela equação 3.1.

$$f(x,y) \approx \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & \dots & f(1,N-1) \\ \vdots & \vdots & \dots & \vdots \\ f(M-1,0) & f(M-1,1) & \dots & f(M-1,N-1) \end{bmatrix} \quad (3.1)$$

As técnicas clássicas de processamento de imagens envolvem a manipulação matemática dessas matrizes através de transformadas e filtragem para extração ou realce de características de interesse.

3.1.2 - Áreas de aplicação do processamento digital de imagens

Na área de processamento digital de imagens existem muitos problemas que necessitam de técnicas capazes de melhorar a qualidade das imagens para que possam ser feitas análises e interpretações por profissionais.

Na medicina, o processamento de imagens por computador vem melhorando o contraste ou controlando o nível de intensidade de cores, facilitando a interpretação das

imagens de raios X ou outras imagens biomédicas, para o diagnóstico de enfermidades. Dentre outras pesquisas, destacam-se, atualmente, aquelas para diagnóstico de tumores causados pelo câncer de mama.

O realce e a restauração de imagens podem ser úteis para processar imagens de objetos irrecuperáveis, que possuem grande valor histórico. Esta técnica pode ser utilizada na arqueologia, onde os métodos de processamento de imagens podem ser utilizados para restaurar fotografias degradadas, recuperando registros disponíveis de artefatos raros.

Dessa forma, as aplicações de processamento digital de imagens para interpretação humana podem ser encontradas na área militar, industrial, astronômica, biológica e outras.

3.1.3 - Morfologia matemática

A morfologia matemática [19,20] estuda as estruturas matemáticas que são aplicadas nas imagens. Seu estudo concentra-se na análise da estrutura geométrica das entidades presentes em uma imagem. Entretanto, uma imagem é uma representação gráfica, plástica ou fotográfica de um objeto. Então, ela pode conter qualquer tipo de matéria e, por isso, a morfologia matemática pode ser aplicada a diversas áreas. Portanto, é possível analisar a matéria proveniente da biologia, da medicina, da visão computacional, do controle de qualidade, do reconhecimento de padrões, e das demais áreas.

Em se tratando de processamento de imagens, a morfologia matemática é uma ferramenta muito versátil, pois permite processar imagens com objetivos de realce, de segmentação, de detecção de bordas, de esqueletização, de afinamento, de análise de formas, entre outros.

O princípio básico da morfologia é extrair as informações relacionadas à geometria e à topologia de um conjunto desconhecido X de uma imagem. Esse conjunto é comparado, a partir de uma transformação, com um conjunto E , definido e conhecido em forma e tamanho,

chamado de elemento estruturante. O poder da morfologia matemática aplicado em imagens se encontra nesse elemento estruturante. Através desse processo de comparação entre os dois conjuntos, consegue-se avaliar o conjunto X e extrair as informações necessárias de geometria e topologia. Na Figura 3.1 são mostrados os elementos estruturantes binários básicos :

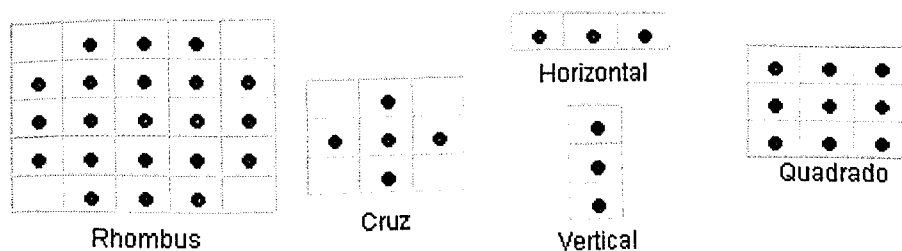


Figura 3.1 - Exemplos de elementos estruturantes binários.

O formato e o tamanho do elemento estruturante servem para testar e quantificar a maneira como o elemento estruturante encontra-se na imagem. A Figura 3.2 mostra como esse elemento estruturante interage com uma imagem binária. Ao marcar a posição do elemento estruturante, pode-se ter uma primeira idéia da estrutura geométrica da imagem analisada.

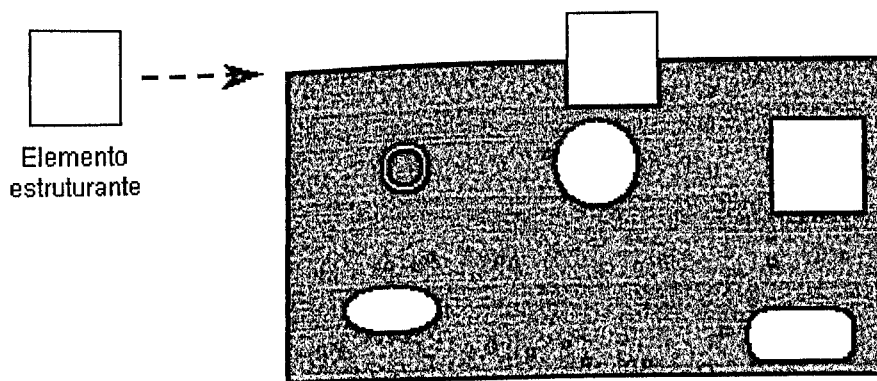


Figura 3.2 - Interação do elemento estruturante com a imagem binária.

Através do exemplo de interação, percebe-se que em algumas situações o elemento estruturante encaixa-se na imagem e, em outras, pode acontecer de não coincidir. Se o

formato ou o tamanho desse elemento for alterado, os resultados da aplicação serão diferentes. Portanto, para cada tipo da informação a ser extraída da imagem, existe um elemento estruturante a ser utilizado.

A morfologia matemática aplicada a imagem binária, consiste de uma operação morfológica binária que é completamente determinada através da vizinhança examinada ao redor do ponto central, da configuração dos pontos pretos (pixels com valor zero), e brancos (pixels com valor um) nessa vizinhança, e do algoritmo morfológico utilizado.

A morfologia matemática não requer cálculos matemáticos complexos. As operações básicas da morfologia matemática são: dilatação, erosão, abertura e fechamento. Essas quatro operações originam vários algoritmos morfológicos como espessamento, afinamento ou esqueletização. Neste trabalho, foi utilizado o algoritmo de espessamento, que faz a dilatação entre as letras até atingir uma espessura de pixel entre elas [19].

3.1.3.1 - definições básicas de morfologia matemática

Segundo Serra [22], existem as seguintes notações e definições na morfologia matemática:

- Reflexão do conjunto X em relação à origem:

$$\overset{v}{X} = \{x : -x \in X\} \quad (3.2)$$

- Complemento do conjunto X :

$$X^c = \{x : x \notin X\} \quad (3.3)$$

- Translação do conjunto X por um vetor h:

$$X_h = \{x : x \pm h \in X\} \quad (3.4)$$

- Conjunto união, ou seja, conjunto de pontos que pertencem a X ou Y:

$$U = X \cup Y \quad (3.5)$$

- Conjunto intersecção, ou seja, conjunto de pontos que pertencem a X e Y:

$$I = X \cap Y \quad (3.6)$$

- Conjunto X contido em Y:

$$X \subset Y \quad (3.7)$$

- Conjunto Y contém X:

$$Y \supset X \quad (3.8)$$

3.1.3.2 - operações básicas de morfologia matemática

- Dilatação

A dilatação é definida segundo Facon[21] pela equação (3.2):

$$A \oplus B = \{x \mid \overset{\vee}{B}_x \cap A \neq \emptyset\} \quad (3.2)$$

onde:

A e B são conjuntos de Z^2 (inteiros);

\emptyset é conjunto vazio.

Assim, o processo de dilatação consiste em obter a reflexão de B em torno de sua origem e, em seguida, a translação do resultado da reflexão por x.

A dilatação de A por B é o conjunto de todos os deslocamentos x tais que $\overset{\vee}{B}$ e A sobreponham-se em pelo menos um elemento não nulo. O conjunto B é considerado o elemento estruturante da dilatação.

- Erosão

A operação de erosão é definida [21], pela equação (3.3):

$$A \ominus B = \{x \mid B_x \subset A\} \quad (3.3)$$

onde: A e B são conjuntos de Z^2 .

A erosão de A por B é o conjunto de todos os pontos x tais que B, quando transladado por x, esteja contido em A

- Abertura

A abertura de um conjunto A por um elemento estruturante B é definida pela equação (3.4):

$$A \circ B = (A \ominus B) \oplus B \quad (3.4)$$

onde:

A e B são conjuntos de Z^2 .

A interpretação da equação (3.4) é que a abertura de A por B é simplesmente a erosão de A por B, seguido da dilatação do resultado por B.

- Fechamento

O fechamento de um conjunto A por um elemento estruturante B é definido pela equação (3.5):

$$A \bullet B = (A \oplus B) \ominus B \quad (3.5)$$

onde:

A e B são conjuntos de Z^2 .

Dessa forma, o fechamento de A por um elemento estruturante B é apenas a dilatação de A por B seguida da erosão do resultado por B.

3.2 - Considerações Finais Do Capítulo

Este capítulo descreveu um apanhado histórico sobre a importância do processamento de imagens e algumas áreas de aplicação, juntamente com as definições de conjuntos sobre o ponto de vista morfológico e as principais operações da morfologia matemática.

Capítulo 4

ESPECIFICAÇÃO DO SISTEMA DESENVOLVIDO NESTE TRABALHO

4.1 - Introdução

As etapas de pré-processamento e segmentação são de fundamental importância para o bom desempenho de um sistema reconhecedor automático de escrita manual. As entradas deste sistema constituem um universo onde seus elementos não são bem comportados, apresentando variações extremamente grandes entre si, tornando-os difíceis de serem tratados. As variações de padrões entre os diversos tipos de escrita ocorrem devido a vários fatores como, por exemplo, à maneira característica de escrever do próprio escritor, do seu estado emocional, da sua escolaridade, do objeto utilizado (lápiz, caneta, pincel, etc.). Em sistemas de reconhecimento off-line, a entrada é uma imagem da escrita capturada por um sistema de aquisição. Isto faz com que a qualidade da imagem capturada seja dependente não apenas do estado do documento original mas, também, das condições em que se apresenta este sistema de aquisição, ou ainda, do posicionamento do papel diante do dispositivo de leitura. No presente trabalho, será proposto um sistema de pré-processamento e segmentação de caracteres, que faça com que um sistema de reconhecimento que o utilize seja o menos dependente possível do estado da imagem de entrada, e consiga produzir resultados significativos em sua saída. O objetivo deste capítulo é mostrar o sistema de pré-processamento e segmentação de caracteres desenvolvido neste trabalho.

4.2 - Definição Dos Requisitos Do Sistema Desenvolvido Neste Trabalho

Os requisitos considerados mais significativos em um sistema de pré-processamento e segmentação são:

- O sistema deve ser independente do tipo de papel empregado para escrever, seja um caderno com linhas, folha em branco, folha verde, e outros.
- O tipo de objeto empregado para escrever, pode ser lápis, lapiseira, caneta ponta fina, caneta ponto grossa ou caneta porosa, não deve influenciar no resultado do sistema.
- As inclinações com a horizontal, sendo elas acline ou declive devem ser irrelevantes para o sistema.
- O sistema deve ser tolerante às inclinações da escrita com relação à linha vertical, que são letras deitadas para a direita ou para a esquerda.
- Capacidade de tratar as distorções ocasionadas no momento da digitalização da imagem. Geralmente, essas distorções são ocasionadas pelo incorreto posicionamento incorreto do documento no aparelho de captura, “scanner”, e a inserção de possíveis degradações na imagem.

4.3 - Sistema Proposto Nesta Dissertação

A Figura 4.1 mostra os blocos que compõem o sistema proposto neste trabalho.

- Filtro médio: o sistema recebe como entrada a imagem da escrita em tons de cinza. Nesta etapa, são aplicados os filtros para a remoção dos ruídos presentes na imagem, sem que haja perda de informações importantes para o processo de reconhecimento.
- Binarização: neste bloco, é realizada a transformação da imagem com tons de níveis de cinza, para apenas dois tons, branco e preto, utilizando o algoritmo de determinação de um

valor de limiar para que seja definido se um pixel deve ser de um tom ou de outro tom. Assim, a palavra escrita fica caracterizada em relação ao fundo.

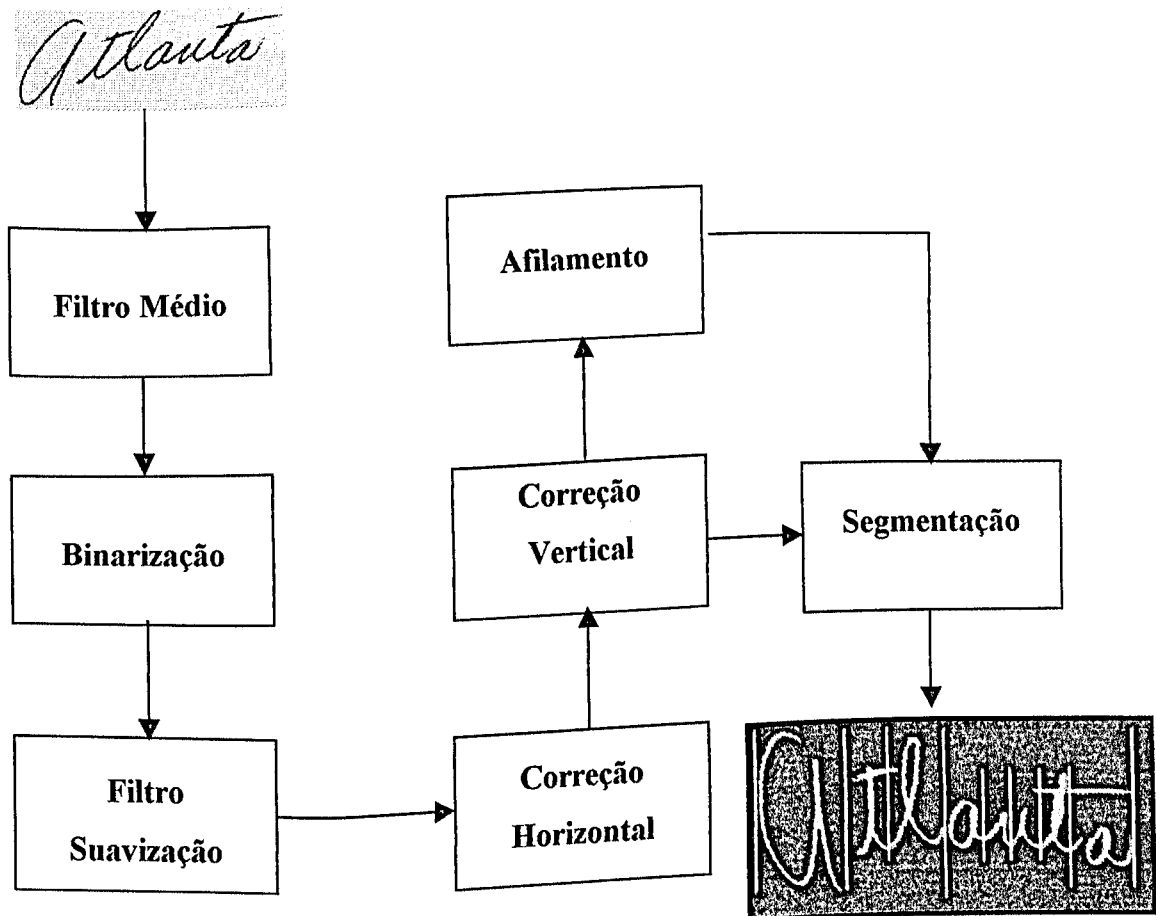


Figura 4.1 - Sistema proposto de pré-processamento e segmentação de caracteres.

- Filtro de suavização: a aplicação deste filtro tem por finalidade a remoção do serrilhamento que surge após a imagem ter sido binarizada, tornando os traços mais arredondados.
- Correção horizontal: este bloco tem como objetivo verificar a existência ou não de inclinação da palavra em relação a uma linha de referência horizontal. Caso exista, um algoritmo de determinação do ângulo de inclinação da palavra é aplicado e, toda a palavra é corrigida por este ângulo.

- Correção vertical: tem como objetivo verificar a existência ou não de inclinação em relação a uma linha de referência vertical. Caso exista, os traços que compõem a palavra serão ajustados na vertical, com a eliminação desta inclinação.
- Afilamento: este processo faz a redução dos traços da imagem à espessura de um pixel.
- Segmentação: nesta etapa, a palavra deverá ser particionada em seus letras constituintes. Os dados obtidos na etapa da correção vertical e na etapa de afilamento, são utilizados para a determinação dos pontos de separação de uma letra e outra.

4.4 - Considerações Finais Deste Capítulo

Neste capítulo, foram apresentadas as especificações a serem atendidas na construção do sistema proposto, as suas etapas componentes, juntamente, com uma breve explicação da finalidade de cada uma delas.

Capítulo 5

O SISTEMA DE PRÉ-PROCESSAMENTO UTILIZADO NESTE TRABALHO

5.1 - Introdução

Como foi observado nos capítulos anteriores, deste trabalho, o maior desafio que se apresenta no desenvolvimento de sistemas de reconhecimento automático de palavras manuscritas, que sejam independentes do autor da escrita, está na variação dos estilos de escrita de cada pessoa. Cada um apresenta sua maneira própria de escrever, individualizando o traçado, provocando, assim, um universo extremamente grande de formas de escrita. E, o seu tratamento pelo computador constitui uma tarefa não trivial. O método proposto neste trabalho manipula as desigualdades ocorridas entre as palavras através de uma seqüência de procedimentos aplicados à imagem de entrada. Estes procedimentos são: **aplicação de filtros, binarização, correção horizontal, correção vertical e afilamento**, constituindo o sistema de pré-processamento. Abaixo, seguem as explicações de cada etapa do pré-processamento, juntamente com os respectivos algoritmos que foram empregados.

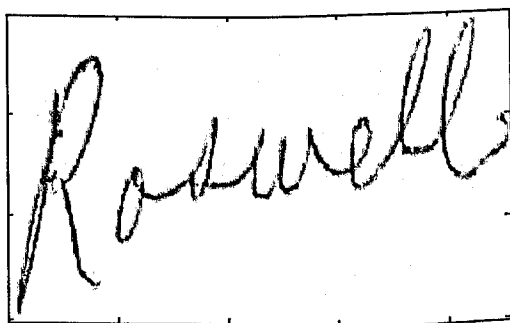
5.2 - Filtros Empregados

Os filtros utilizados neste trabalho são:

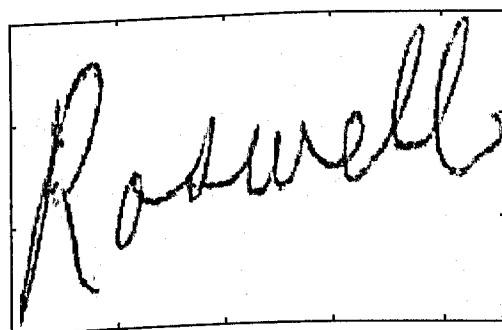
Filtro médio _ a sua aplicação tem por finalidade minimizar as diferenças entre os níveis de cinza na imagem, diminuindo , assim, os pontos isolados na palavra. A máscara utilizada para aplicar o filtro médio está mostrada na Figura 5.1, propiciando a realização da média entre os níveis de cinza em uma janela de 3x3, atribuindo essa média ao pixel analisado. A Figura 5.2 mostra a imagem após a aplicação deste filtro.

$$\frac{1}{9}x \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

Figura 5.1 - Máscara para o filtro médio.



(a) Imagem antes de aplicar o filtro médio.



(b) Imagem após aplicar o filtro médio.

Figura 5.2 - Aplicação do filtro médio.

Filtro para remoção de picos _ este filtro remove os picos das imagens utilizando-se as máscaras mostradas na Figura 5.3, se o pixel analisado pertencer a uma dessas máscaras ele será apagado, passando de um para zero, eliminando assim os picos da imagem.

Todas as imagens das palavras antes de serem aplicadas nesse filtro são binarizadas para facilitar a localização do contorno da palavra.

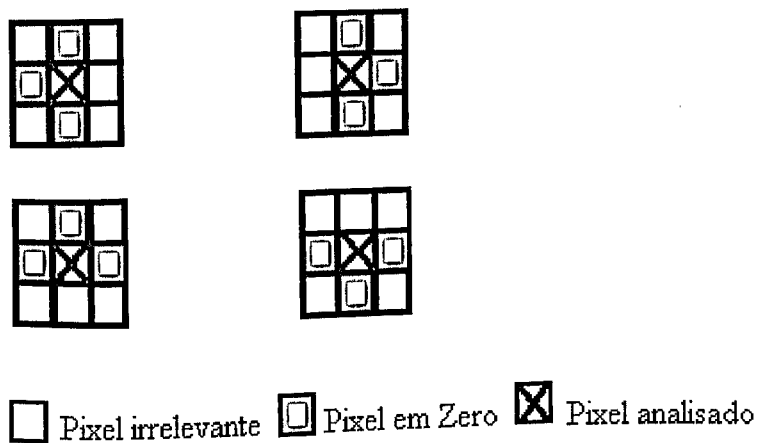


Figura 5.3 – Máscaras utilizadas no filtro para remoção de picos.

Filtro para remoção de vales _ com este filtro, é realizada a remoção dos vales existentes nas imagens já binarizadas, com a finalidade de facilitar a verificação do contorno da imagem. A Figura 5.4 mostra as máscaras utilizadas para remoção de vales, se o pixel analisado pertencer a uma dessas máscaras, o pixel em zero vai ter o seu valor alterado para um, eliminando os vales da imagem. A Figura 5.5 mostra as imagens antes e depois da aplicação do filtro para remoção de picos e vales.

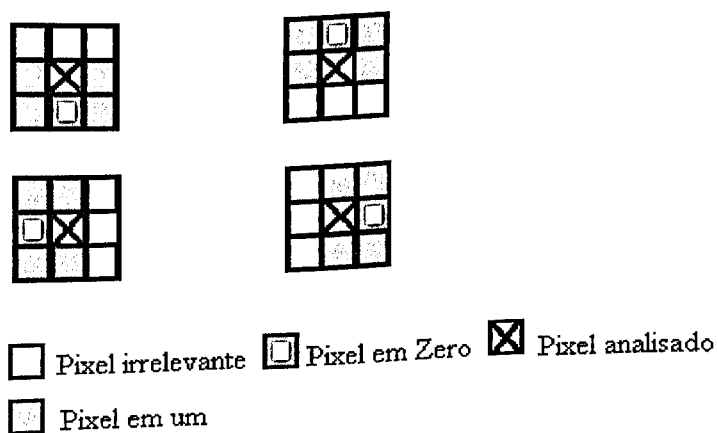


Figura 5.4 - Máscaras utilizadas para remoção de vales.

Na Figura 5.5(b), a seta vermelha indica a área de pico na imagem, devendo remover esses pixels, atribuindo o valor zero para cada um deles. As duas setas azul mostram as áreas de vales na imagem, devendo ser preenchidas com valor um.

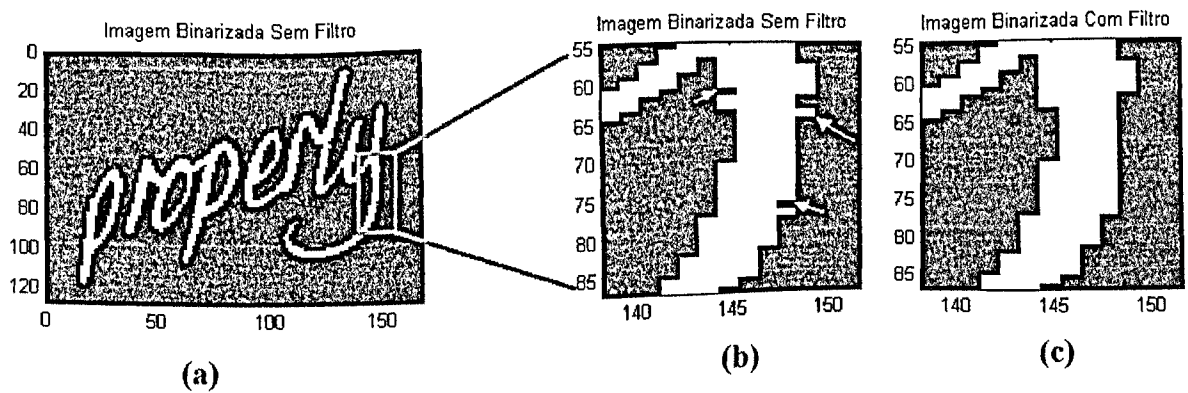


Figura 5.5 – (a) Imagem antes de aplicar o filtro de remoção de picos e vales, (b) Imagem antes de aplicar o filtro de remoção de picos e vales ampliada, (c) Imagem após aplicar o filtro de remoção de picos e vales.

5.3 - Binarização

As imagens utilizadas neste trabalho estão em tons de cinza. Os traços das palavras estão apresentados em um tom de cinza mais escuro, e o fundo (*Background*) em um tom de cinza mais claro. O tom de cinza da imagem pode variar entre valores de 0 a 255. A ação de binarizar uma imagem consiste em transformá-la em uma representação de somente dois níveis de cinza, o preto e o branco, indicados por **zero** e **um**, respectivamente. O zero representa o fundo da imagem e o um representa o próprio objeto contido na imagem. Um problema que deve ser tratado é com relação à determinação do limiar ideal para proceder a separação entre os pixels de fundo dos pixels que fazem parte da palavra, visto que estes elementos da imagem podem estar bem próximos em níveis de cinza dificultando a sua caracterização.

Para a determinação do valor ideal de limiar, foi empregado o método de Otsu [14], que utiliza princípios estatísticos de análise da variância dos níveis de cinza presentes na imagem. Este algoritmo é descrito abaixo:

Considere a imagem a ser binarizada constituída de duas classes de pixels:

C0, constitui o fundo da imagem (background) e,

C1, constitui o objeto.

Essas duas classes serão delimitadas por um limiar (threshold) denominado K. E serão divididas em:

$$C0 = [1, \dots, k]$$

$$C1 = [k+1, \dots, L]$$

$$\mu_T = \sum_{i=1}^{Nc} i * P_i$$

Repita para K = 1 até Nc

Se (P(k) > 0)

$$\omega(k) = \sum_{i=1}^k P_i$$

$$\mu(k) = \sum_{i=1}^k i * P_i$$

*Se ($\omega(k) * [1 - \omega(k)] > 0$)*

$$\sigma_B^2(k) = \frac{[\mu_T * \omega(k) - \mu(k)]^2}{\omega(k)[1 - \omega(k)]}$$

Se ($\sigma_B^2(k) > \text{Max} \sigma_B^2$)

$$\text{Max} \sigma_B^2 = \sigma_B^2(k)$$

$$\text{Limiar} = k$$

Fim

Fim

Fim

Fim

L: níveis de cinza da imagem.

n(i): quantidade de pixels no nível i.

N: número total de pixels na imagem.

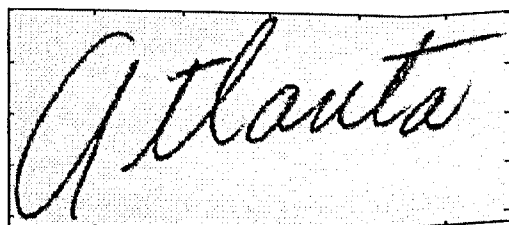
Nc: nível de cinza máximo.

$N = n1 + n2 + \dots + nL.$

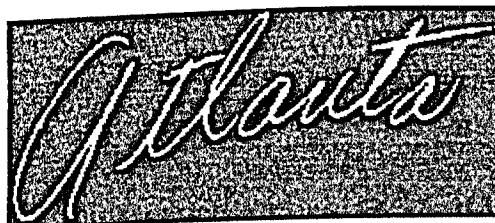
$P(i) = n(i) / N$ para $P(i) \geq 0$

O limiar ótimo k , é aquele em que o valor da variância entre as classes é máxima. A

Figura 5.6, mostra alguns exemplos de imagens binarizadas pelo método de Otsu.



(a)



(b)

Figura 5.6 – (a) Imagem antes de utilizar o método de Otsu, (b) Imagem após utilizar o método de Otsu.

5.4 - Correção Horizontal

Para a correção da inclinação da palavra, em relação a uma linha de referência horizontal, foi desenvolvido neste trabalho um algoritmo baseado em Morita [16] com algumas mudanças, na seguinte seqüência de passos.

Determinação das linhas de delimitação da imagem,

Localização da linha central horizontal,

Localização da segmentação natural,

Detecção dos pontos de mínimos,

Eliminação dos mínimos indesejáveis,

Cálculo do grau de inclinação,

Correção da inclinação.

5.4.1 - Determinação das linhas de delimitação da imagem

As linhas de delimitação, linha de base superior (LBS) e linha de base inferior (LBI), são obtidas baseadas no histograma horizontal da imagem. Este histograma é determinado realizando o levantamento do número de pixels brancos em cada linha horizontal da imagem. A determinação das posições das linhas de delimitação tem como objetivo estabelecer três regiões na imagem: Superior, mediana, e inferior. A região superior contém os traços superiores que aparecem nas letras como o **t**, **l**, e **d**; na região mediana está o corpo da palavra, e a região inferior contém os traços inferiores que compõem as letras do tipo **f**, **g** e **p**. A Figura 5.7, mostra uma imagem binarizada e seu histograma horizontal correspondente, representado na cor azul.

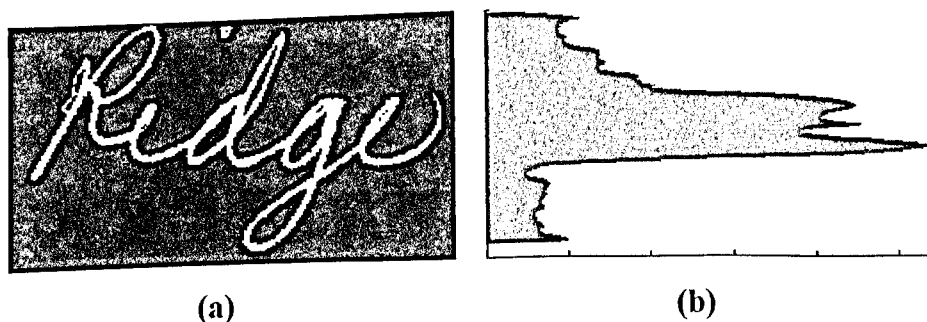


Figura 5.7 – (a) Imagem binarizada, (b) Histograma horizontal da imagem binarizada.

Para a determinação das posições das linhas de delimitação LBS e LBI, inicialmente, é levantado o valor de pico máximo ($\max(\text{HistH})$) do histograma horizontal. Este valor, multiplicado por uma constante, 0,55, como mostra na equação 5.1, produzirá um valor de concentração de pixels (ConPixel), que, aplicado ao histograma, fornecerá as coordenadas verticais de LBS e LBI.

$$\text{ConPixel} = \max(\text{HistH}) * 0,55 \quad (5.1)$$

A constante 0,55 foi determinada heurísticamente, através de vários testes com diferentes imagens, sendo este, o valor mais adequado para o conjunto de imagens utilizadas neste trabalho. A região mediana é delimitada por LBS e LBI. As linhas superior e inferior são posicionadas nos extremos do histograma.

A Figura 5.8 mostra, as linhas: Linha superior (LS), Linha de base superior (LBS), Linha de base inferior (LBI), Linha inferior (LI).

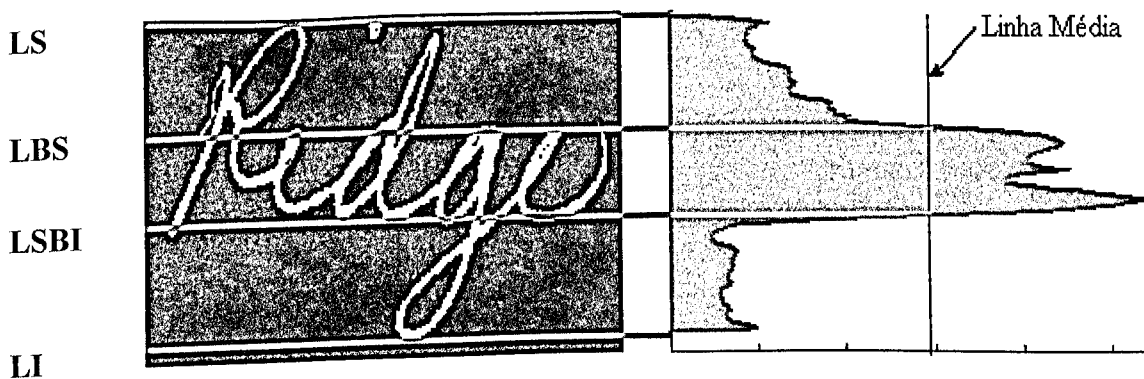


Figura 5.8 - Determinação das linhas de referências.

Na Figura 5.8, pode-se observar que, em uma primeira tentativa de posicionamento, tanto a linha de base superior quanto a linha de base inferior, não atingiram os seus objetivos corretamente, que é buscar a maior abrangência possível do corpo da palavra dentro destas duas linhas. Para alcançar um melhor posicionamento destas linhas, foi estabelecido uma tolerância de deslocamento das mesmas. Para o cálculo de suas novas posições, são aplicadas as equações, (5.2), (5.3), (5.4) e (5.5), onde os valores 0.3 e 0.2 foram obtidos empiricamente.

$$\text{Tolerância Superior} = ((\text{Linha Base Superior} - \text{Linha Superior}) * 0.3) \quad (5.2)$$

$$\text{Linha Base Superior} = \text{Linha Base Superior} - \text{Tolerância Superior} \quad (5.3)$$

$$\text{Tolerância Inferior} = ((\text{Linha Inferior} - \text{Linha Base Inferior}) * 0.2) \quad (5.4)$$

$$\text{Linha Inferior} = \text{Linha Inferior} + \text{Tolerância Inferior} \quad (5.5)$$

As novas posições das linhas LBS e LBI, calculadas nas equações (5.3) e (5.5) são mostradas na Figura 5.9. Pode-se observar que, agora, as novas linhas estabelecem com precisão a região mediana, onde está presente o corpo da palavra.

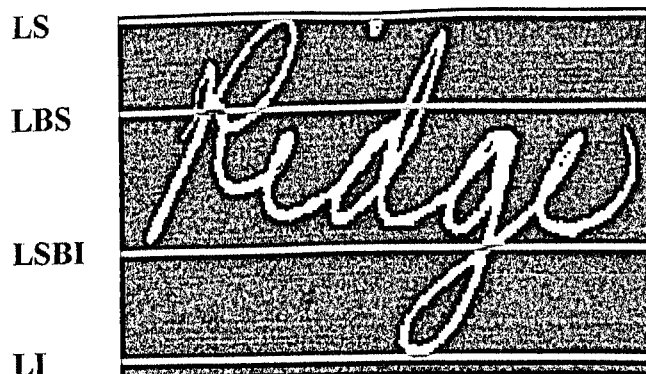


Figura 5.9 - Aplicação da tolerância para o cálculo das novas linhas de referências.

5.4.2 - Localização da linha central horizontal da imagem

A linha central horizontal (LCH), é determinada na imagem com o objetivo de facilitar a ligação entre as letras que constituem a palavra, procurando, assim, corrigir a desconectividade que pode estar presente na mesma. Pois, quando se trata de palavras manuscritas, cursivas, geralmente, elas estão todas interligadas. Porém, no caso de palavras escritas à mão, em letras de fôrma, elas são, geralmente, desconectadas. O tratamento é mais problemático ainda, quando as letras são mistas, ou seja, letras de fôrma misturadas com letras cursivas.

O algoritmo desenvolvido neste trabalho para a determinação da linha central, considera a concentração de pixels que a imagem possui, encontrada pelo seu histograma vertical. O objetivo é encontrar a linha que caracteriza, horizontalmente, o centro da palavra. Para tanto, as posições do histograma que apresentam uma concentração pequena de pixels são eliminadas para não interferir no cálculo da posição da LCH. O histograma é ordenado, primeiramente, em ordem crescente de número de pixels, e o número de linhas horizontais ($NumLinEL$), a serem eliminadas é calculado pela equação (5.6).

$$NumLinEl = (NúmeroLinhasImagem - (max(HistH) * 0.2)) \quad (5.6)$$

Onde:

NúmeroLinhasImagem _ número de linhas horizontais da imagem.

A constante 0,2 foi obtida heurísticamente, através de experimentos com as imagens de entrada.

A eliminação das linhas no histograma começa com a primeira linha que apresenta menor concentração de pixels, e termina na linha com valor NumLinEl, encontrado utilizando a equação (5.6). Com o histograma resultante, é feito o cálculo da média de pixels por linha, MedPixLin, utilizando a equação 5.7.

$$MedPixLin = (TotalPixels / TotalLinhas); \quad (5.7)$$

Onde:

TotalPixels _ soma de todas as concentrações de pixels do histograma resultante;

TotalLinhas _ número de linhas restante no histograma.

Calculado o valor de MedPexLin utilizando-se a equação (5.7) é obtido as linhas: Linha Superior, Linha Base Inferior, Linha Base Superior e Linha Inferior, é calculada a linha central horizontal utilizando-se a equação (5.8).

$$LinhaCentral = (LinhaBaseInferior - LinhaBaseSuperior)/2 + LinhaBaseSuperior). \quad (5.8)$$

A Figura 5.10 mostra a localização da linha central em uma imagem.

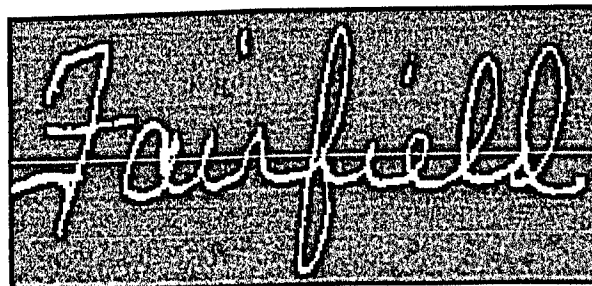


Figura 5.10 - Localização da linha central na imagem.

5.4.3 - Localização da segmentação natural

A segmentação natural é realizada utilizando-se o histograma vertical da imagem como mostrado na Figura 5.7. Onde a quantidade de pixels igual a zero, é a condição de descontinuidade do traço na escrita manual. Os pontos de segmentação natural, geralmente, são devido as palavras escritas com letras de fôrma, ou ao estilo da pessoa que escreve, ou a falha do objeto utilizado na escrita ou, ainda devido à má qualidade da imagem. Estes parâmetros são importantes para definição dos pontos de separação entre as letras.

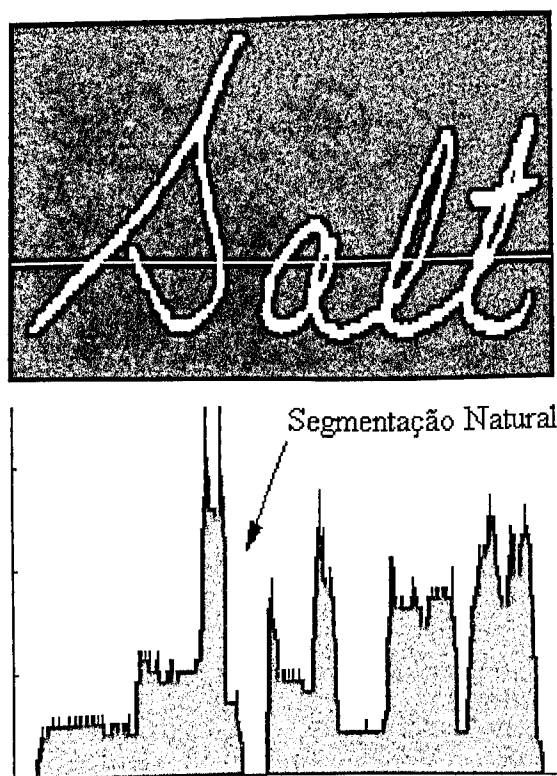


Figura 5.11 - Imagem binarizada e seu histograma vertical.

A Figura 5.12 mostra a separação da letra S pela determinação do ponto de segmentação natural. Porém, estes pontos devem ser tratados para que a inclinação da palavra, no processo de correção horizontal, possa ser obtida corretamente.

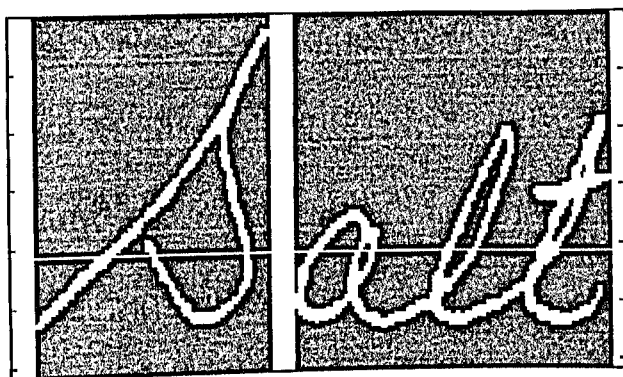


Figura 5.12 - Aplicação da segmentação natural na imagem.

5.4.4 - Detecção dos pontos de mínimos

Os candidatos a pontos de mínimos são aqueles que localizam-se na zona inferior ou na zona mediana, e devem satisfazer à aplicação das máscaras de verificação. Estas máscaras vão seguir o contorno dos traços, pixel a pixel, verificando se houve mudança da coordenada correspondente no eixo vertical (eixo L), obedecendo à seguinte ordem: o eixo aumenta, estabiliza e diminui. Satisfeitas essas condições, o ponto em questão pertence ao conjunto dos mínimos. A Figura 5.9 mostra as máscaras para localizar os pontos de mínimos.

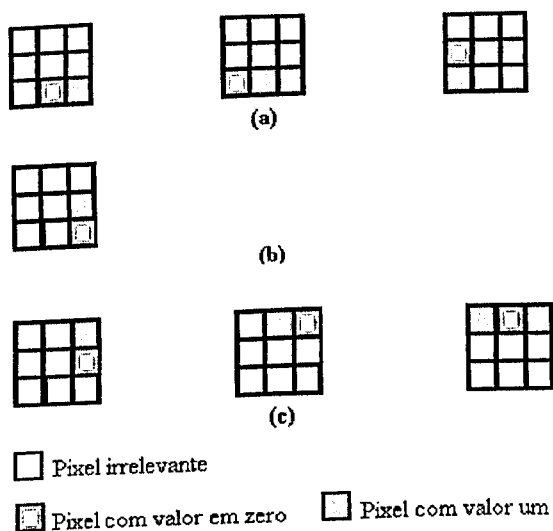


Figura 5.13 – (a) Máscaras utilizadas na verificação se o contorno aumenta, (b) Máscaras utilizadas na verificação se o contorno estabiliza, (c) Máscaras utilizadas na verificação se o contorno diminui.

A Figura 5.11 mostra os eixos de coordenadas.

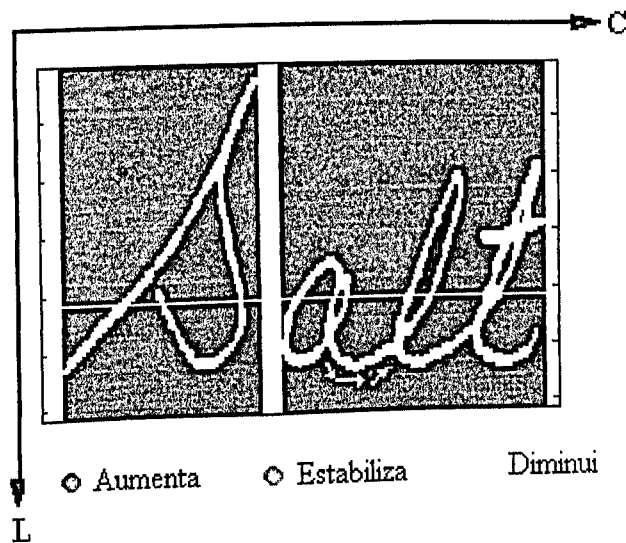


Figura 5.14 – Eixos de coordenadas.

A Figura 5.12 mostra os pontos de mínimo detectados na imagem pela aplicação das máscaras de verificação, inicialmente o contorno inferior da palavra deve aumentar, depois estabilizar e logo em seguida diminuir. Se essas condições forem satisfeitas, o pixel que se encontra na posição estabiliza é considerado como sendo um ponto de mínimo.



Figura 5.15 - Localização dos pontos de mínimo de uma imagem.

5.4.5 - Eliminação dos mínimos indesejáveis

É necessário realizar a eliminação de alguns pontos de mínimos para que os mesmos não interfiram na determinação da reta cuja inclinação represente melhor a inclinação horizontal da palavra. Para isso, foi adotado o procedimento estatístico, denominado de método dos mínimos quadrados [23], que procura a melhor reta, minimizando a soma dos quadrados dos desvios entre os valores observados e os propostos pela reta calculada. São calculadas as distâncias entre a reta impressa e os pontos de mínimos e, o desvio padrão destas distâncias para se obter um parâmetro de eliminação dos mínimos indesejáveis. Este processo é repetido até que não exista nenhum ponto que deva ser eliminado. O algoritmo de eliminação dos mínimos indesejáveis é mostrado a seguir.

Repita Enquanto (houver mínimo a ser apagado)

$$Sl = \sum_{i=1}^n Li$$

$$Sc = \sum_{i=1}^n Ci$$

$$Slc = \sum_{i=1}^n Li * Ci$$

$$SQc = \sum_{i=1}^n Ci^2$$

$$SQL = \sum_{i=1}^n Li^2$$

$$SScl = (Slc - ((Sc * Sl) / n))$$

$$SSc = (SQc - ((Sc^2) / n))$$

$$A = SScl / SSc$$

$$B = ((Sl / n) - (A * (Sc / n)))$$

Repita para i = 1 até n

$$Distância(i) = (Li - (Ci * A + B))$$

Fim

$$Soma = \sum_{i=1}^n Distância(i)$$

$$MédiaAritmética = Soma / n$$

$$DesvioPadrão = \sqrt{\frac{\sum_{i=1}^n (Distância(i) - MédiaAritmética)^2}{n-1}}$$

$$Limite = MédiaAritmética + DesvioPadrão$$

Repita para $i = 1$ até n

Se ($Distância(i) > Limite$)

Apaga o Mínimos(i)

$$n = n - 1$$

Fim

Fim

Fim

Onde:

n _ número total de mínimos locais existentes na palavra.

L e C _ coordenadas dos pontos de mínimos (Mínimos(L,C)).

A Figura 5.13 mostra a imagem antes e depois após a aplicação do algoritmo de eliminação dos mínimos indesejáveis.



Figura 5.16 – (a) Imagem antes de eliminar os mínimos indesejáveis, (b) Imagem após eliminar os mínimos indesejáveis.

5.4.6 - Cálculo do ângulo de inclinação horizontal

Obtida a reta que melhor representa a inclinação da palavra, é determinado o ângulo de inclinação θ da palavra. A Figura 5.16 mostra a imagem de uma palavra com um acento e a Figura 5.17 ilustra a imagem de uma palavra com declive.

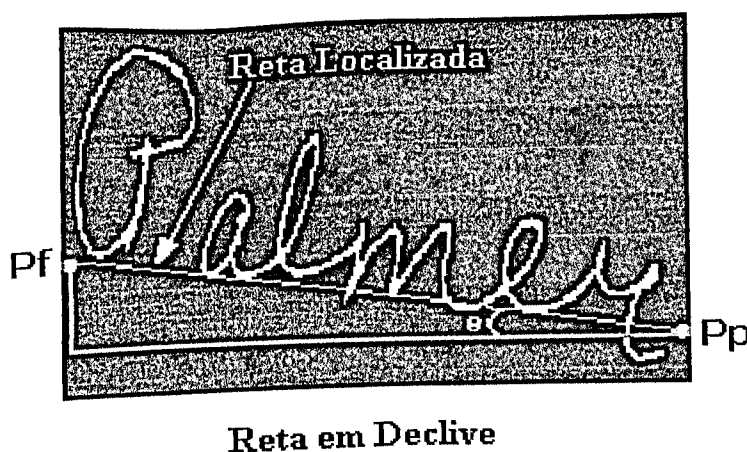


Figura 5.17 – Imagem da palavra com declive.

A Figura 5.16 mostra a imagem de uma palavra com um acento e a Figura (5.17) ilustra a imagem de uma palavra com declive.

Como as coordenadas dos pontos **Pf** e **Pp** da Figura 5.17 que caracterizam a reta de inclinação da palavra são de determinação imediata, o ângulo θ é obtido utilizando a equação (5.9).

$$\hat{\text{Angulo}}\theta = \frac{\text{cateto oposto}}{\text{cateto adjacente}} \quad (5.9)$$

5.4.7 - Correção da inclinação horizontal

Para corrigir a inclinação da imagem, devido ao seu posicionamento no “scanner”, é aplicado um processo de rotação na imagem da palavra utilizando a Transformada de Hook [24]. A equação (5.10) calcula a transformada de Hook, após a determinação do ângulo θ de inclinação, cada coordenada (L, C) é transformada em coordenadas (L', C') dada por:

$$\text{Im } g(L', C') = \begin{bmatrix} \text{Cos}\theta & -\text{Sen}\theta \\ \text{Sen}\theta & \text{Cos}\theta \end{bmatrix} * \begin{bmatrix} L \\ C \end{bmatrix} \quad (5.10)$$

Onde :

$$L' = L * \text{Cos}\theta - C * \text{Sen}\theta$$

$$C' = L * \text{Sen}\theta + C * \text{Cos}\theta$$

A Figura 5.18 mostra a imagem da Figura 5.17 após aplicar a correção da inclinação horizontal.

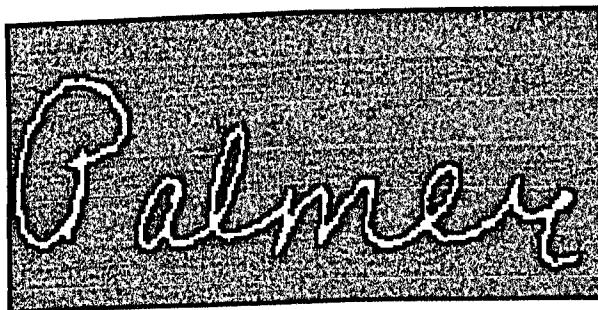


Figura 5.18 – Imagem da Figura 5.17 após aplicar a correção da inclinação horizontal.

5.5 - Correção Vertical

Para realizar a correção vertical, da imagem da palavra é necessário calcular o grau de inclinação vertical dos caracteres, ou seja, o ângulo entre o eixo médio das letras e o eixo vertical. Para tanto, inicialmente, é obtido histograma vertical da imagem. A Figura 5.19 mostra a imagem de uma palavra e seu histograma vertical.

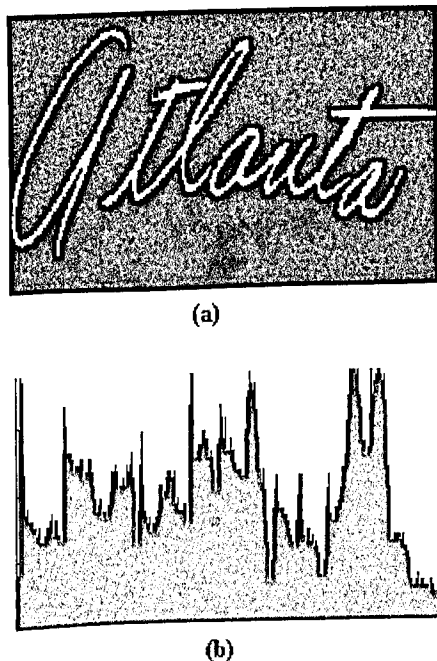


Figura 5.19 – (a) Imagem de uma palavra após a correção horizontal, (b) Histograma vertical da palavra.

5.5.1 - Cálculo do grau de inclinação vertical da imagem

A idéia básica do algoritmo do cálculo do grau de inclinação da imagem consiste no levantamento de histogramas das imagens obtidas pelo rotacionamento da imagem original, a partir de 30° à esquerda até 30° à direita, em relação ao eixo vertical, com incrementos de $5,5^\circ$ graus, sendo gerado um total de 21 histogramas. Após o processo de geração destes histogramas, é localizado o maior pico de concentração de pixels em cada um destes histogramas, e são selecionados três histogramas que possuem os maiores picos. O valor da

inclinação vertical da imagem original da palavra será o valor médio entre os três ângulos correspondentes aos histogramas selecionados. O algoritmo desenvolvido para a determinação dos três histogramas que possuem os maiores picos e seus ângulos correspondentes, é apresentado a seguir.

Repita para Ângulo = 120° até 60°

Repita para C = 1 até Número Máximo de Colunas

Repita para L = 1 até Número Máximo de Linhas

Se (Imagem(L,C) > 0)

*CI = (C - ((L * (tan(Ângulo)))))*

Se(Cinicial == 0)

PP = C;

Cinicial = CI

Fim

NovaColuna = ((CI - Ci) + PP)

NovaLinha = L

Se ((NovaColuna > 0) and (NovaColuna < Número Máximo de Colunas))

NovaImagem(NovaLinha, NovaColuna) = 1

HistogramaVertical(NovaColuna) = HistogramaVertical(NovaColuna) + 1

Fim

Fim

Fim

Fim

Ordena o histogramaVertical

Maior = Pega o maior valor do histogramaVertical

$$Im = Im + I;$$

$$\hat{AnguloEncontrado}(Im,1) = Angulo;$$

$$AnguloEncontrad (Im,2) = Maior;$$

Fim

Para o cálculo do ângulo de inclinação vertical da palavra original é utilizada a equação 5.10.

$$\hat{AnguloM\u00e9dio} = ((AnguloEncontrado(Max,1) + AnguloEncontrado(Max - 1,1) + AnguloEncontrado(Max-2,1)) / 3) \quad (5.10)$$

5.5.2 - Correção da inclinação vertical

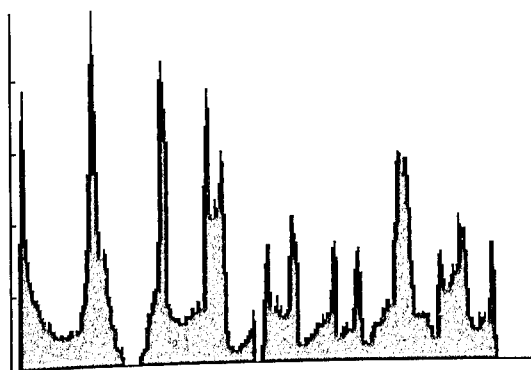
A correção da inclinação vertical é baseada no ângulo de inclinação vertical da palavra original, aplicando-se a transformada Shear [25]. Com a utilização da transformada Shear, as coordenada das linhas (L') são mantidas e as coordenadas das colunas são alteradas de acordo com a equação (5.11).

$$\begin{aligned} L' &= L \\ C' &= (C - (L * (Tan(Angulo\theta)))) \end{aligned} \quad 5.11$$

A Figura 5.19 mostra a imagem da palavra após aplicar a transformada Shear e seu histograma vertical.



(a)



(b)

Figura 5.20 – (a) Imagem da palavra corrigida verticalmente, (b) Histograma vertical da imagem corrigida verticalmente.

5.6 - Afilamento

O afilamento é um processo utilizado na redução dos traços que constituem a palavra a uma espessura correspondente a um pixel, mantendo as características essenciais da imagem original. Este processo é necessário para a localização dos loops, pontos de cruzamento, e ponto final da palavra, que são elementos importantes para a execução do processo de segmentação.

O método empregado por Flôres [1] para realizar o afilamento da imagem, possui basicamente dois passos: o primeiro consiste em marcar o contorno da imagem, e o segundo verifica cada pixel do contorno, analisando se ele pode ser apagado ou não.

5.6.1 - Verificação do contorno

O algoritmo desenvolvido por Flôres considera como contorno as partes externas e internas da imagem. Para fazer esta verificação, foram empregadas as máscaras mostradas na Figura 5.21.

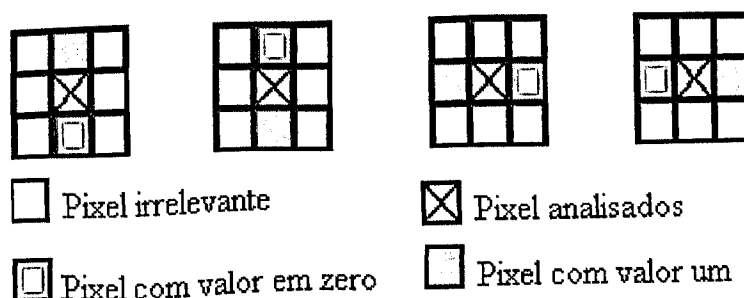


Figura 5.21 - Máscaras utilizadas para a verificação do contorno.

Se qualquer uma das máscaras mostradas na Figura 5.21 for satisfeita, o pixel central da máscara é marcado com o valor três, indicando que ele pertence ao contorno.

5.6.2 - Apagamento do contorno

Esta parte do algoritmo de Flôres [1] é muito importante, pois, é ela que evita o apagamento dos pixels que constituem o afilamento da imagem, e o excesso de erosão. É importante lembrar que no processo de apagamento do contorno são analisados apenas os pixels que foram marcados como pertencentes ao contorno.

O método de apagamento do contorno é dividido, basicamente, em quatro testes:

- **Primeiro teste:** as máscaras mostradas na Figura 5.22 são aplicadas a todo pixel do contorno.

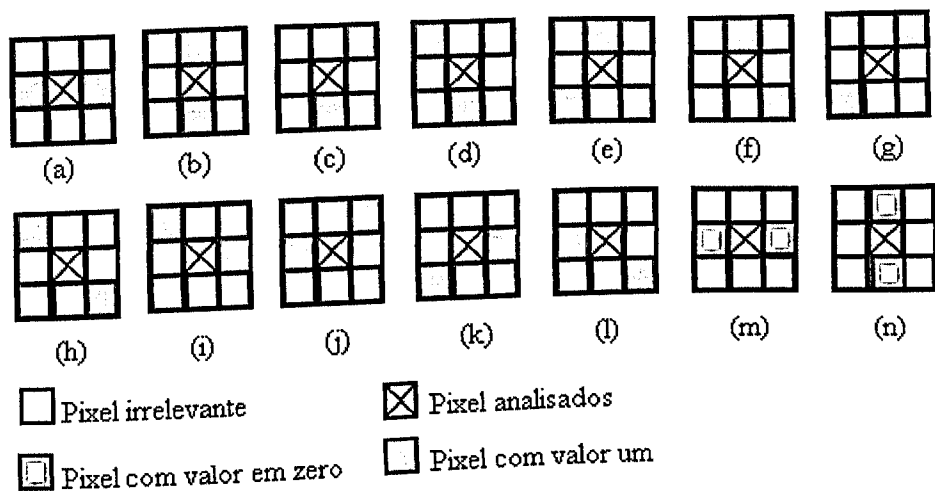


Figura 5.22 - Máscaras para verificação da primeira condição do método de apagamento do contorno.

Se algumas das máscaras, de *a* a *l*, da Figura 5.22 forem satisfeitas, e se, pelo menos uma das máscaras, *m* ou *n*, for também satisfeita, então o pixel considerado não poderá ser apagado.

- **Segundo teste:** este teste verifica a quantidade de vizinhos do pixel que está sendo analisado.

Se o pixel que está sendo analisado possuir apenas um pixel vizinho, indicado pelo número 1 na Figura 5.23, e este vizinho possui dois ou mais pixels como seus pixels vizinhos, e, o pixel 1 deve ser apagado.

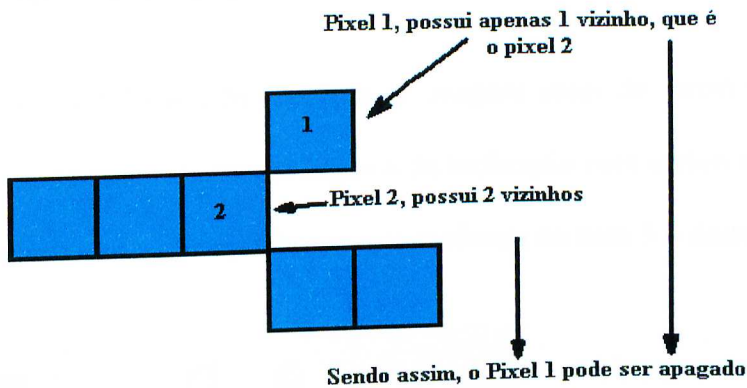


Figura 5.23 - Verificação dos pixels vizinhos de um pixel, analisando quando ele possui apenas um pixel vizinho.

- **Terceiro teste:** se o pixel analisado não possui pixels vizinhos, ele é considerado como sendo um ruído, e, deve ser apagado.
- **Quarto teste:** se o pixel que esta sendo analisado possui dois ou mais pixels vizinhos, ele terá que ser analisado juntamente com estes pixels vizinhos, um de cada vez, aplicando-se as máscaras mostradas na Figura 5.24

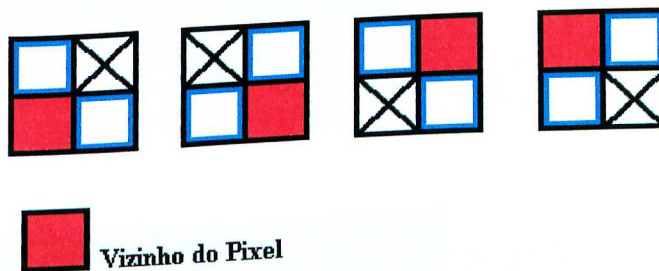
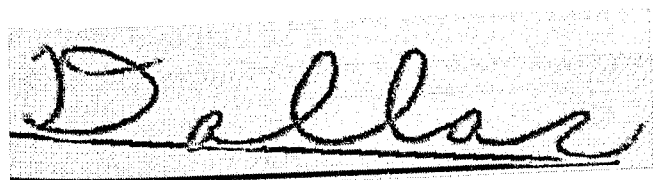


Figura 5.24 - Verificação dos pixels vizinhos do pixel, que esta sendo analisado, quando ele possui dois ou mais pixels vizinhos.

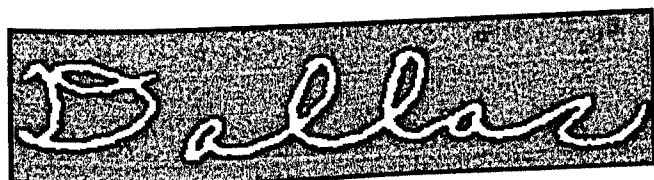
Se alguma das máscaras mostradas na Figura 5.24 for satisfeita para um dos pixels vizinhos ao pixel, que esta sendo analisado ele não é apagado da imagem, pois, se ele for apagado, é provocado uma desconectividade no afilamento da imagem.

5.7 - Resultados Obtidos

As figuras 5.25 e 5.26 mostram as imagens antes de serem submetidas a correção horizontal, com os seus respectivos ângulos de inclinação com o eixo horizontal e as imagens após serem submetidas a correção horizontal explicada no item 5.4 deste capítulo.

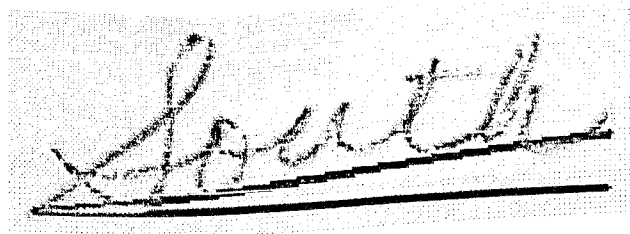


(a)

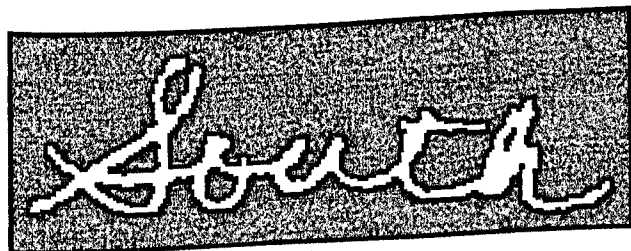


(b)

Figura 5.25 – (a) Imagem antes de ser submetida a correção horizontal, (b) Imagem após ser submetida a correção horizontal.



(a)

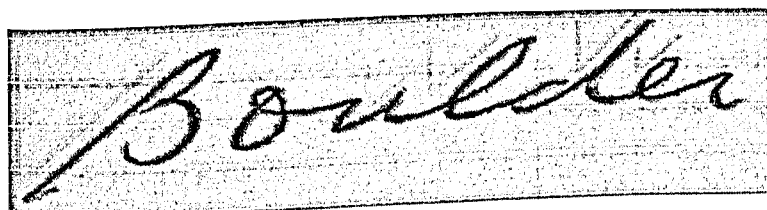


(b)

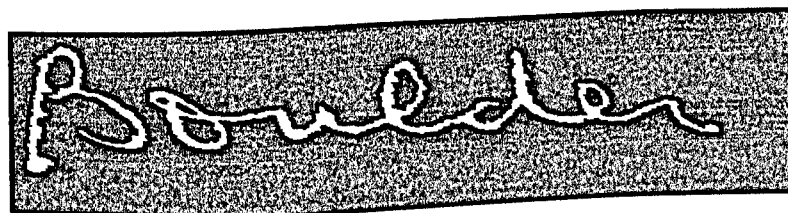
Figura 5.26 – (a) Imagem antes de ser submetida a correção horizontal, (b) Imagem após ser submetida a correção horizontal.

As imagens mostradas nas Figuras 5.26 e 5.26 mostram a eficiência algoritmo desenvolvido neste trabalho para a correção do ângulo com a horizontal tanto em aclone quanto em declive.

As Figuras 5.27 e 5.28 mostram as imagens antes e depois de serem submetidas a correção vertical, explicado no item 5.5 deste capítulo.

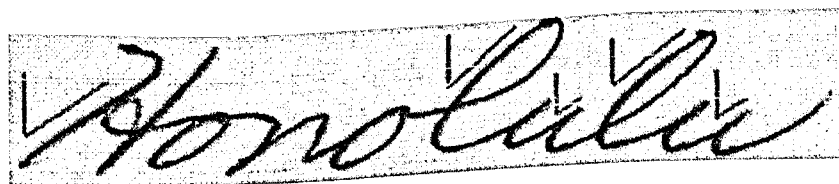


(a)

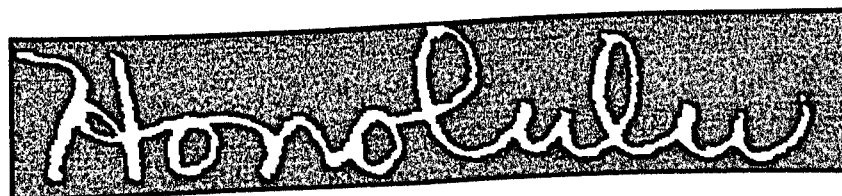


(b)

Figura 5.27 - (a) Imagem antes de ser submetida a correção horizontal, (b) Imagem após ser submetida a correção horizontal.



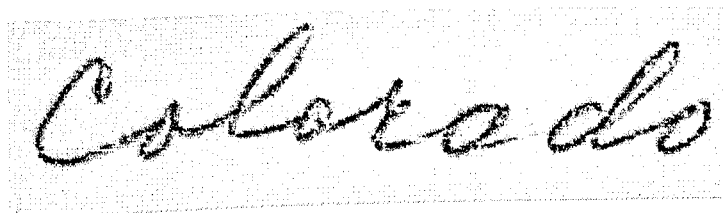
(a)



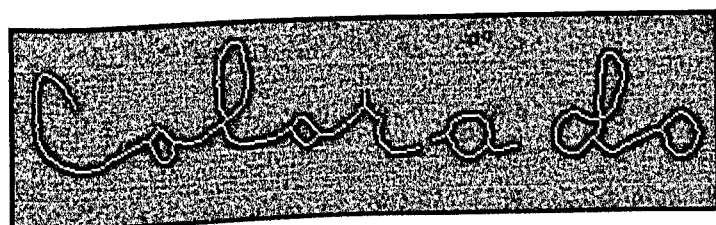
(b)

Figura 5.28 - (a) Imagem antes de ser submetida a correção horizontal, (b) Imagem após ser submetida a correção horizontal.

As Figuras 5.29 e 5.30 mostram as imagens antes e depois de serem submetidas ao algoritmo de afilamento explicado na seção 5.6 deste capítulo.

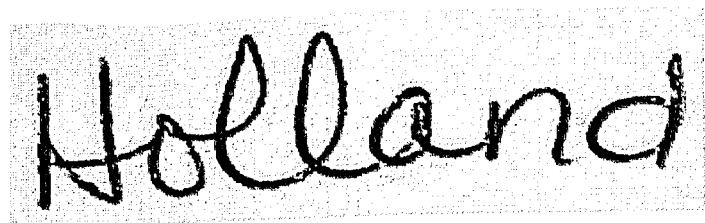


(a)

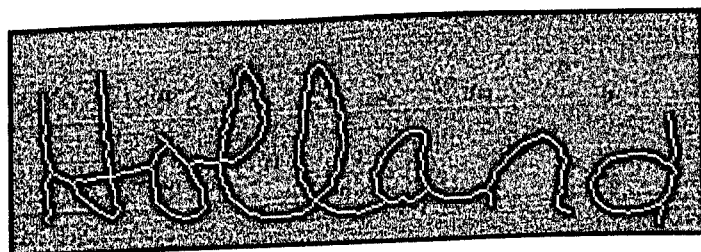


(b)

Figura 5.29 – (a) Imagem antes de ser submetida ao algoritmo de afilamento, (b) Imagem após ser submetida ao algoritmo de afilamento.



(a)



(b)

Figura 5.30 – (a) Imagem antes de ser submetida ao algoritmo de afilamento, (b) Imagem após ser submetida ao algoritmo de afilamento.

5.8 - Conclusões

O sistema de pré-processamento desenvolvido, neste trabalho possui as etapas de binarização, correção horizontal, correção vertical e afilamento. Os resultados obtidos neste sistema mostram a sua eficiência no tratamento de imagens das palavras manuscritas mesmo quando as mesmas não se apresentam bem comportadas. As inclinações em relação à horizontal e a vertical são comuns na escrita manual constituindo um problema para os sistemas automáticos de reconhecimento de manuscritos. Dentro do conjunto de imagens utilizadas neste trabalho, os mecanismos de correção das inclinações desenvolvidos produziram os efeitos desejados. Em particular, o algoritmo de correção vertical, baseado nos histogramas da imagem rotacionada de 30° à esquerda até 30° direita do eixo vertical tornou possível a obtenção do ângulo de correção mais adequado para a imagem de entrada do sistema de pré-processamento, como pode ser comprovado nos resultados apresentados.

Capítulo 6

DESCRIÇÃO DO SISTEMA DE SEGMENTAÇÃO DESENVOLVIDO NESTE TRABALHO

6.1 - Introdução

Neste capítulo, é apresentado o sistema de segmentação de caracteres desenvolvido neste trabalho. O sistema recebe como entrada a imagem da palavra pré-processada, juntamente com as informações relevantes para a segmentação, tais como: a presença de loops, pontos de cruzamento, e ponto final da imagem original.

6.2 - Segmentação

No método desenvolvido para a segmentação de caracteres, são analisados dois grupos de letras manuscritas:

- letras manuscritas cursivas que, geralmente, são conectadas umas as outras.
- letras manuscritas de fôrma, que sempre apresentam uma separação visível umas das outras na palavra.

Em ambos os grupos de letras manuscritas cursivas e de fôrma, a segmentação de caracteres torna-se difícil, devido ao formato da escrita de cada pessoa. A maneira de escrever, de cada pessoa, proporciona grandes variação na formação dos caracteres e, algumas vezes, até mesmo a sobreposição dos mesmos, impedindo a segmentação perfeita das

palavras. Dependendo do escritor, a legibilidade da escrita torna-se muito difícil até mesmo para os seres humanos, sendo possível o seu entendimento devido a leituras anteriores dessa palavra. Se o leitor não tem conhecimento dessa palavra, provavelmente, não vai conseguir interpretá-la. Para o computador, é muito difícil posicionar corretamente o corte na segmentação dessa palavra somente com as características contidas na imagem. Assim como o leitor, ele necessita de informações adicionais, tal como um sistema de reconhecimento e pós-processamento. O método de segmentação desenvolvido neste trabalho utiliza informações que são extraídas somente da imagem da palavra manuscrita, estas informações são: os pontos de mínimos, pontos finais, pontos de cruzamentos, e loops encontrados nas letras **a, o, b, d**.

As etapas do método de segmentação desenvolvidos neste trabalho são:

- delimitação das linhas de base;
- delimitação das zonas de referência;
- cálculo da espessura do traço na imagem;
- localização da segmentação natural;
- localização dos mínimos;
- localização do conjunto de pontos dos loops;
- verificação do conjunto de pontos de máximo;
- verificação do conjunto de pontos de cruzamento;
- verificação do conjunto de pontos de finais;
- localização dos pontos de segmentação da palavra; e
- corta a palavra nos pontos de segmentação.

6.2.1 - Delimitação das zonas de referência

As zonas de referências são fundamentais quando se trata da distinção de traços das letras maiúsculas e minúsculas, separando os traços superiores, o corpo da palavra e os traços inferiores, por isso elas são divididas em zonas: superior, mediana e inferior. Nem sempre uma palavra possui as três zonas. Basicamente, a sua localização está relacionada com o histograma horizontal, e com as linhas de referências, como mostrado na Figura 6.1.

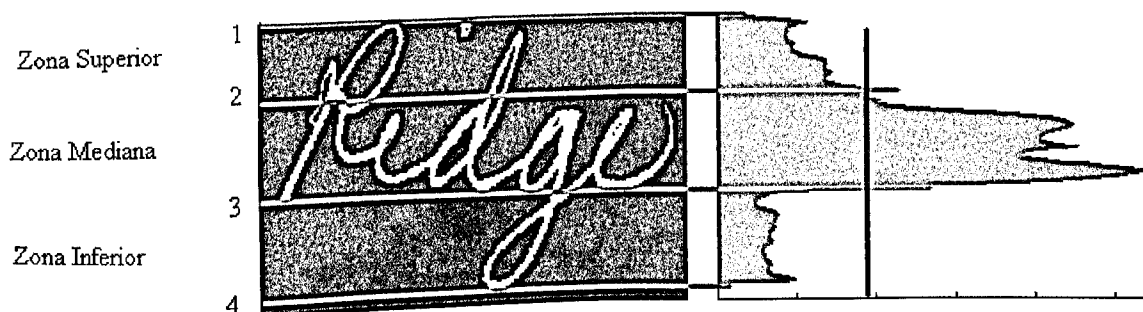


Figura 6.1 - Localização das zonas de referência utilizando o histograma horizontal.

Onde ocorre uma maior concentração de pixels no histograma horizontal, está o corpo da palavra, ou seja, as letras minúsculas.

A zona superior é o espaço entre a linha superior e a nova linha base superior, a zona mediana é o espaço entre a nova linha base superior e a nova linha base inferior, e a zona inferior é o espaço entre a nova linha base inferior e a linha inferior. A Figura 6.1 indica as três zonas de uma imagem.

Existem palavras que são escritas todas em letras de fôrma, elas não apresentam zona superior nem zona inferior, assim como, podem existir palavras que possuem apenas a zona mediana, e a zona inferior. A existência dessas zonas depende da composição da palavra e da caligrafia do escritor. A Figura 6.2 mostra a imagem de uma palavra que possui as três zonas de delimitação.

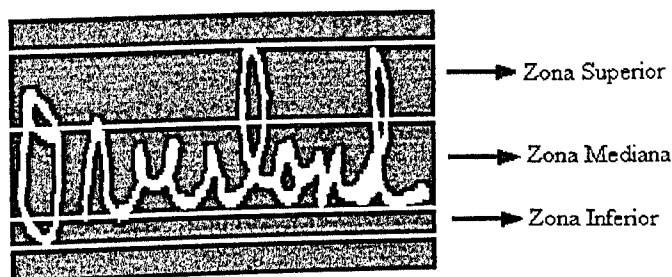


Figura 6.2 - Imagem de uma palavra que possui as três zonas de delimitação.

A Figura 6.3 mostra a imagem de uma palavra que possui apenas uma zona de delimitação.

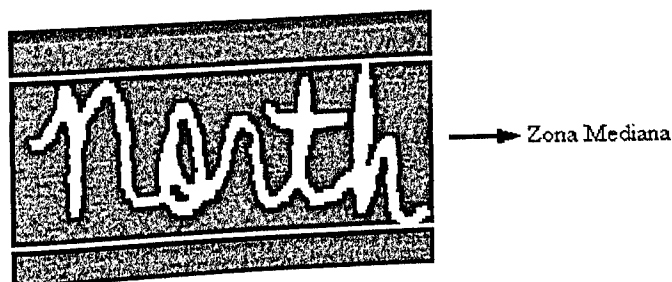


Figura 6.3 - Imagem de uma palavra que possui apenas uma zona de delimitação.

A Figura 6.4 mostra a imagem de uma palavra que possui duas zonas de delimitação.

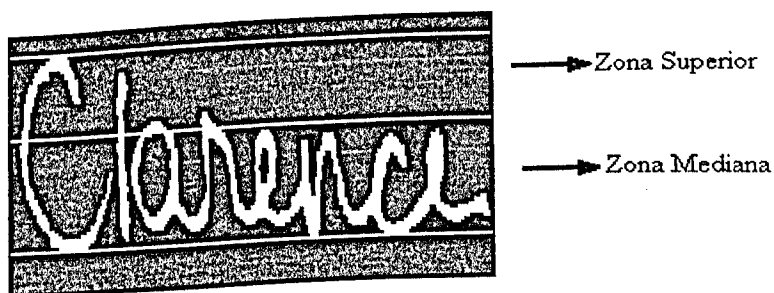


Figura 6.4 - Imagem de uma palavra que possui duas zonas de delimitação.

6.2.2 - Cálculo da espessura média do traço na imagem

A espessura média do traço é baseada nas coordenadas da localização da linha central horizontal da imagem, descrita no capítulo 5 deste trabalho. Após a obtenção dessas coordenadas, a imagem é varrida da esquerda para direita linha por linha, verificando a quantidade de pixel com valor igual a 1, e a quantidade de traços que existem no percurso. Este processo é realizado conforme o seguinte:

LinhaCentral, (descrita no capítulo 5 deste trabalho)

Repita para $C = 1$ até Número Máximo de Colunas

Se($Imagem(LinhaCentral, C) > 0$) and ($Imagem(LinhaCentral, C + 1) > 0$)

$SomaPixelTraco = SomaPixelTraco + 1$;

$Inicio = 1$

Senão Se($Imagem(LinhaCentral, C) > 0$) and ($Imagem(LinhaCentral, C + 1) == 0$)

and ($Inicio > 0$))

$TotalTraço = TotalTraço + 1$;

Fim

Fim

$EspessuraDoTraço = (SomaPixelTraco / TotalTraço)$

6.2.3 - Localização do conjunto de pontos dos loops

Os conjuntos dos pontos dos loops são encontrados utilizando-se a imagem que foi afilada. Eles são muito importantes, pois, nas coordenadas dos loops não pode ocorrer segmentação das palavras, Essas coordenadas representam as letras, tais como: **a, e, l, o, e p**.

O método que localiza essas coordenadas baseia-se na vizinhança longa da imagem e na quantidade de pixels vizinhos próximos ao pixel analisado. Parte do método usado para eliminar a vizinhança longa da imagem é baseado na aplicação da máscara mostradas na Figura 6.5.

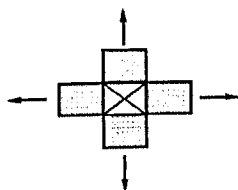
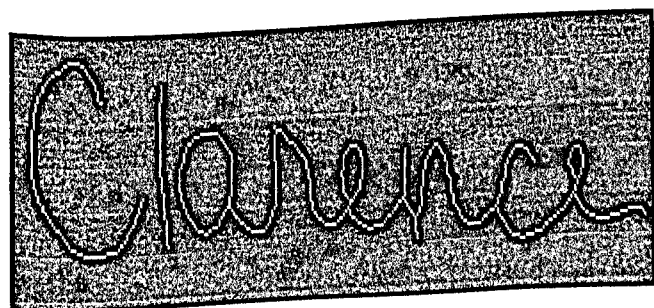


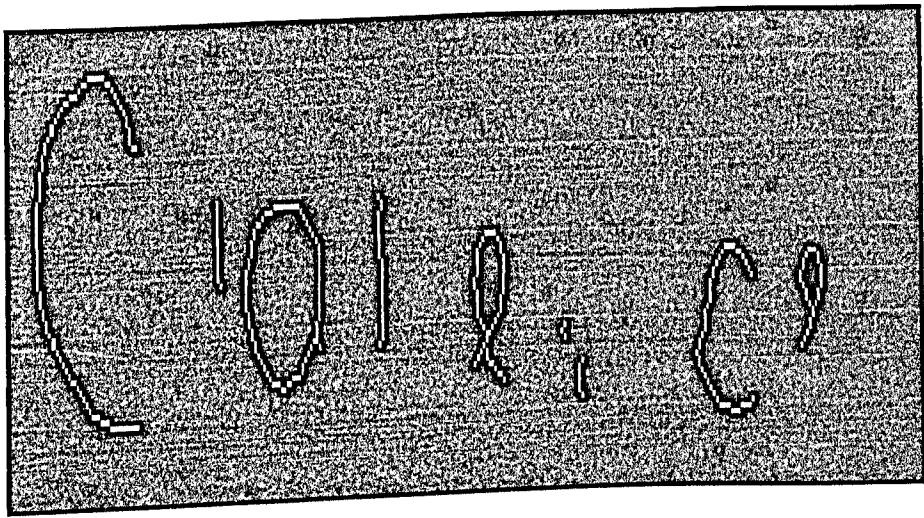
Figura 6.5 - Máscara utilizada para a localização da vizinhança longa.

A vizinhança longa de uma imagem consiste em verificar se cada pixel vizinho do pixel que está sendo analisado, possui menos de quatro pixels vizinhos. Esta verificação é feita até o final da imagem. Se esta condição for verdadeira, o pixel analisado é apagado por ser considerado como não pertencente ao contorno do loop.

Para isolar completamente os loops, a imagem é varrida novamente da direita para a esquerda, verificando os pixels pertencentes ao objeto, ou seja, os pixels com valor igual a 1. Se ele possui apenas um pixel vizinho, esse pixel não pertence ao loop e deve ser apagado. Se ele possui mais de um pixel vizinho, ele deve permanecer na imagem. A Figura 6.6 mostra a imagem da palavra antes de aplicar a vizinhança longa e a imagem após apagar os pixels que não pertencem à vizinhança longa.



(a)



(b)

Figura 6.6 – (a) Imagem da palavra antes de aplicar a vizinhança longa, (b) Imagem após apagar os pixels que não pertencem a vizinhança longa.

A Figura 6.7 mostra apenas os loops da imagem original da Figura 6.6(a) que foram preservados. Nas coordenadas pertencentes aos loops não pode haver segmentação da imagem.

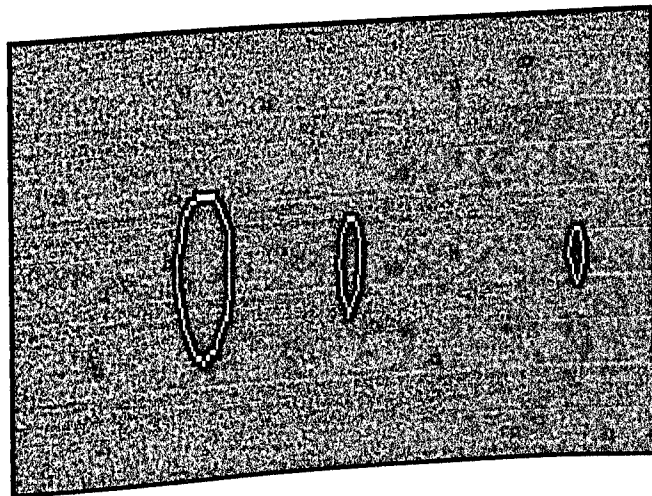


Figura 6.7 - Isolamento dos loops na palavra da imagem da Figura 6.6(a).

6.2.4 - Verificação do conjunto de pontos de máximo na imagem de uma palavra

Os pontos de máximo na imagem de uma palavra são aqueles que se localizam na zona superior, e podem estar, também, na zona mediana, ou seja, são os pontos que estão localizados acima da linha central horizontal da imagem. A finalidade da obtenção das coordenadas desses pontos é a verificação de letras que apresentam traços na direção da zona superior, tais como: a letra **t**, **u**, e outras. Considerando que a imagem já passou pela etapa da segmentação natural, ela já foi dividida em grupos de acordo com a segmentação natural existente. Os pontos de máximo têm que satisfazer a algumas máscaras que seguem o contorno, verificando se houve alterações no eixo vertical (eixo L), e se essas alterações obedecem à seguinte ordem: a linha diminui, estabiliza e aumenta. Satisfazendo essas condições, o pixel em questão pertence ao conjunto dos pontos de máximo.

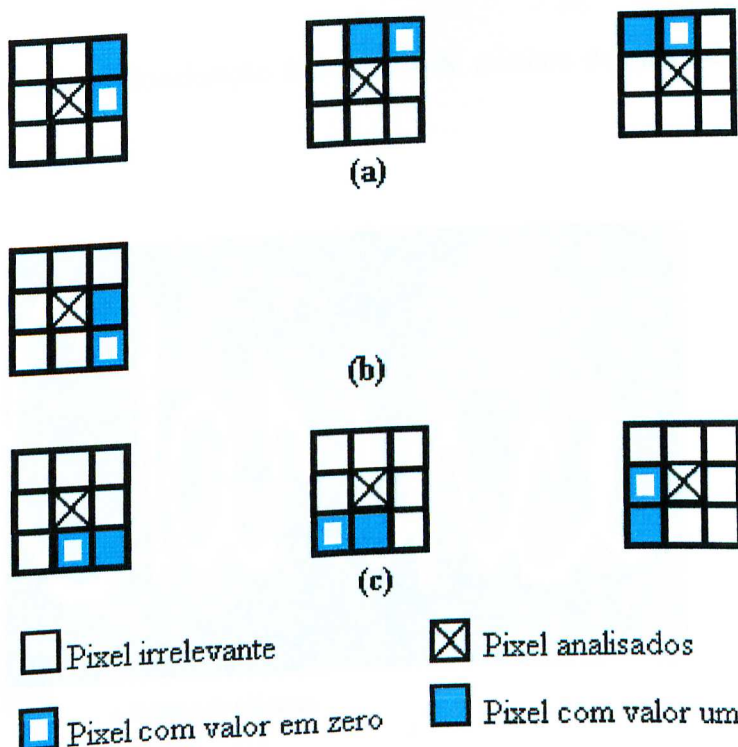


Figura 6.8 – (a) Máscaras que verificam se o contorno diminui, (b) Máscaras que verificam se o contorno estabiliza, (c) Máscaras que verificam se o contorno aumenta.

A Figura 6.8 mostra as máscaras utilizadas para a verificação dos pontos de máximo. Estas máscaras seguem o contorno, com o objetivo de localizar os pontos de máximo.

A Figura 6.9 mostra a localização dos pontos de máximo na imagem de uma palavra.

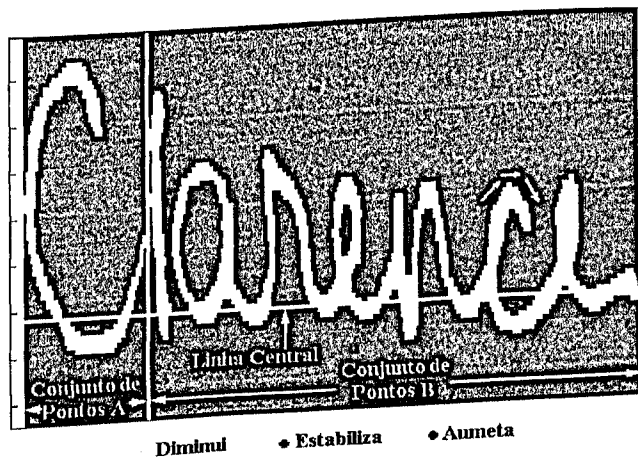


Figura 6.9 - Localização dos pontos de máximo na imagem de uma palavra.

O resultado da localização dos pontos de máximo de uma imagem de uma palavra e mostrado na Figura 6.10.

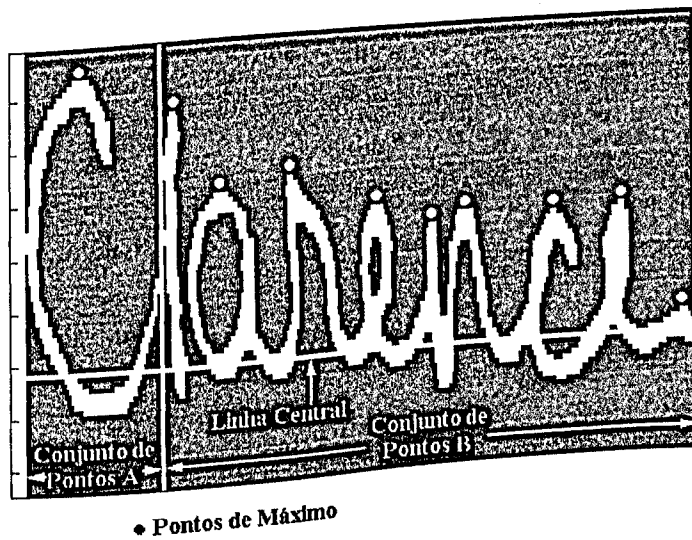


Figura 6.10 - Pontos de máximo localizados na imagem de uma palavra.

6.2.5 - Verificação do conjunto de pontos de cruzamento na imagem de uma palavra

Os pontos de cruzamento na imagem de uma palavra são empregados na verificação da existência de segmentação excessiva entre eles. Geralmente, eles tendem a separar as ligações entre as letras, com exceção das letras **m** e **n**.

Esses pontos são obtidos através do afilamento da imagem, baseado na vizinhança do pixel que está sendo analisado. A máscara usada para encontrar a localização dos pontos de cruzamento é mostrada na Figura 6.11. Esta máscara utiliza o código de cadeia de oito direções [20].

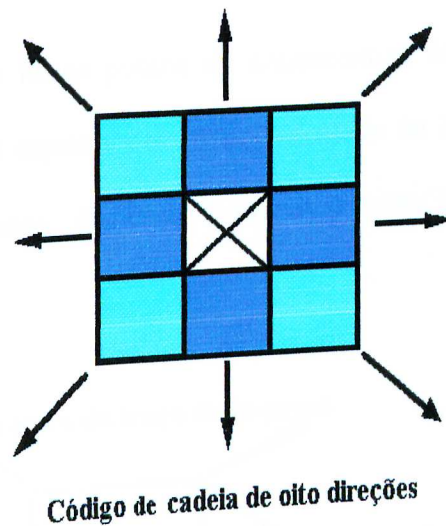


Figura 6.11 - Máscara utilizada para localizar os pontos de cruzamento na imagem de uma palavra.

Se o pixel analisado no afilamento da imagem possuir mais de dois pixels vizinhos, então, ele pode ser considerado pertencente ao conjunto de pontos de cruzamento. Existem alguns manuscritos que possuem pontos de cruzamentos em excesso, ou seja, eles possuem mais de um ponto de cruzamento, como mostrado na Figura 6.12.

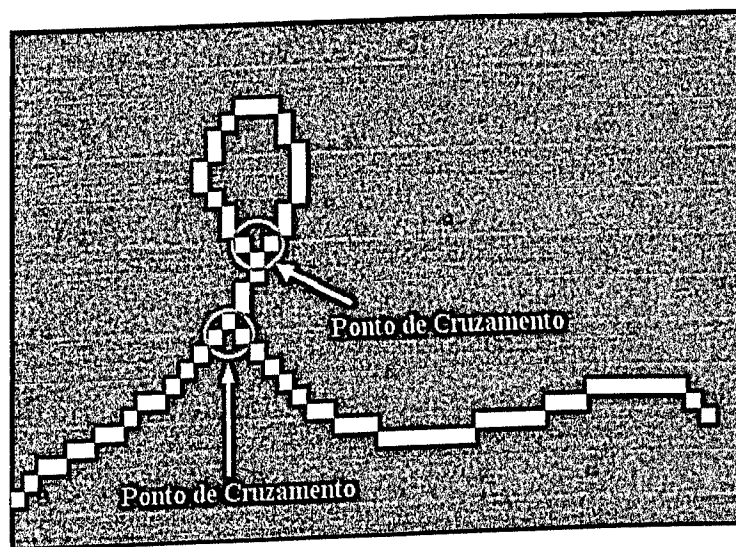


Figura 6.12 - Pontos de cruzamento em excesso na imagem.

Para eliminar os falsos pontos de cruzamentos, foi empregado neste trabalho uma janela que corresponde à espessura do traço calculado na imagem, tanto no eixo das linhas, quanto no eixo das colunas. A Figura 6.13 mostra esta janela que é aplicada na imagem afilada.

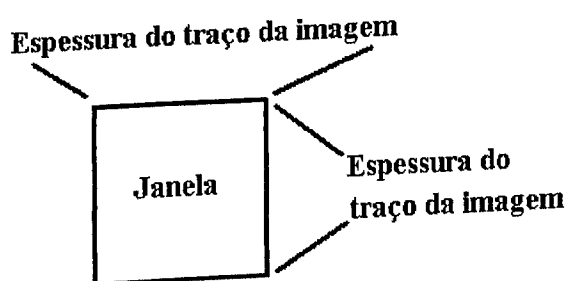


Figura 6.13 - Janela utilizada para verificar se existe ponto de cruzamento em excesso na imagem de uma palavra.

Para que os pontos de cruzamento sejam considerados redundantes, eles devem estar contidos na janela (somente as colunas). Se dois ou mais pontos estão presentes na janela, os

pontos de cruzamento com as linhas que estão mais distantes dos loops são eliminados, até que reste apenas um ponto de cruzamento.

6.2.6 - Verificação do conjunto de pontos finais na imagem de uma palavra

Os pontos finais na imagem de uma palavra são utilizados na verificação da existência de traços na zona superior, como o da letra **t**, e na zona mediana para evitar o corte da letra **u**, dependendo da abertura dessa letra, e também, para evitar que a letra **a** seja cortada quando ela não está completamente fechada.

Assim como no caso dos pontos de cruzamento, a localização dos pontos finais é feita através do afilamento da imagem, aplicando o cálculo das vizinhanças de acordo com o código da cadeia de oito direções [20]. Para que um pixel seja considerado pertencente ao conjunto dos pontos finais, basta que ele possua apenas um único pixel vizinho.

Da mesma forma que pode existir palavras com falsos pontos de cruzamento, também podem existir palavras com falsos pontos finais, por isso, é necessário fazer a verificação para eliminá-los. Além desses pontos estarem contidos na janela com dimensões igual a espessura do traço, eles devem pertencer à mesma zona. Após esta verificação, os pontos finais que obedecem a estas condições são descartados.

6.2.7 - Localização dos pontos de segmentação na imagem de uma palavra

Para localizar as coordenadas de segmentação na imagem de uma palavra, em primeiro lugar, é necessário obter os resultados dos seguintes processos: delimitação das

linhas da palavra (linha superior, linha de base superior, linha de base inferior, linha inferior), delimitação das zonas (zona superior, zona mediana, zona inferior), conjuntos de Mínimos, conjuntos de Máximos, pontos de segmentação dos loops, pontos de cruzamento, pontos finais, espessura do traço, coordenadas de segmentação natural.

Antes de iniciar o processo de segmentação da imagem de uma palavra, é fundamental distinguir se uma palavra pertence ao grupo das letras manuscritas de fôrma, ou das letras manuscritas cursivas. Para fazer esta distinção, o sistema faz uso das coordenadas de segmentação natural existentes na imagem da palavra. Se uma palavra possui mais de quatro segmentações naturais, ela é considerada pertencente ao grupo das letras de fôrma, caso contrário ela pertence ao grupo das letras cursivas. O algoritmo desenvolvido para esta função é mostrado a seguir:

Se (Quantidade Máxima de Segmentação Natural > 4)

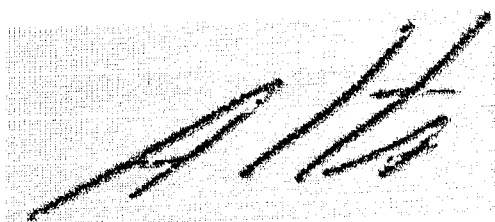
Gera Pontos De Corte Para Letras De Fôrma

Caso contrário

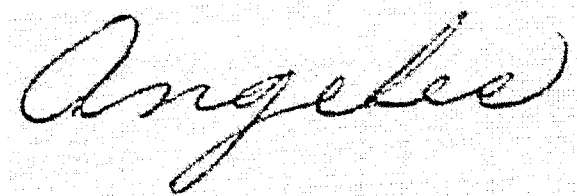
Gera Pontos De Corte Para Letras Manuscritas

Fim

A Figura 6.14 mostra a imagem de palavras escritas com letra de fôrma e letra cursiva.



(a)



(b)

Figura 6.14 – (a) Imagem da palavra escrita com letra de fôrma, (b) Imagem da palavra escrita com letra cursiva.

Como pode ser observado, o sistema desenvolvido neste trabalho deve decidir qual método que vai ser aplicado na imagem. A segmentação é composta de duas ramificações:

- geração de pontos de corte para letras de fôrma; e
- geração de pontos de corte para letras manuscritas.

6.2.7.1 - geração de pontos de corte para letras de fôrma na imagem de uma palavra

A geração dos pontos de corte na imagem de uma palavra escrita com letra de fôrma é baseada no tamanho médio de cada letra. Esse tamanho é calculado usando a operação da morfologia matemática denominada de espessamento [19, 20] ("thicken"), que consiste em dilatar os pixels pertencentes ao objeto até atingir uma espessura de um pixel, ficando exatamente as linhas correspondentes a operação de espessamento. Basicamente o cálculo da quantidade de letras na imagem da palavra está baseado nos traços presentes encontrados pela operação de espessamento na parte superior e na parte inferior da imagem da palavra, sendo feita a média aritmética entre os traços superiores e os traços inferiores, resultando no total médio de letras na imagem da palavra.

A Figura 6.15 mostra a imagem da palavra que é submetida a operação morfológica de espessamento.

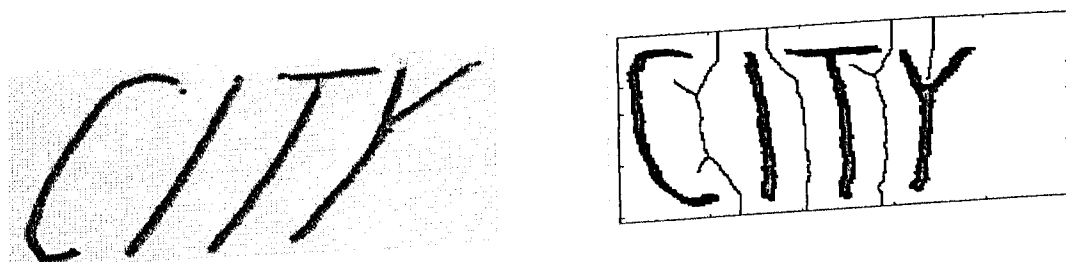


Figura 6.15 – (a) Imagem da palavra antes de ser submetida a operação morfológica de espessamento, (b) Imagem da palavra após a operação morfológica de espessamento.

Após o cálculo, da quantidade de letras na imagem da palavra é verificado se existe alguma coordenada de segmentação natural entre as coordenadas dos loops, e a coordenadas de tolerância, que corresponde à espessura do traço somada com as coordenadas dos loops. Se não existir nenhum ponto de corte entre essas coordenadas, então, é criado um ponto de corte em cima da coordenada do loop. A Figura 6.16 mostra a localização das coordenadas de segmentação do loop.

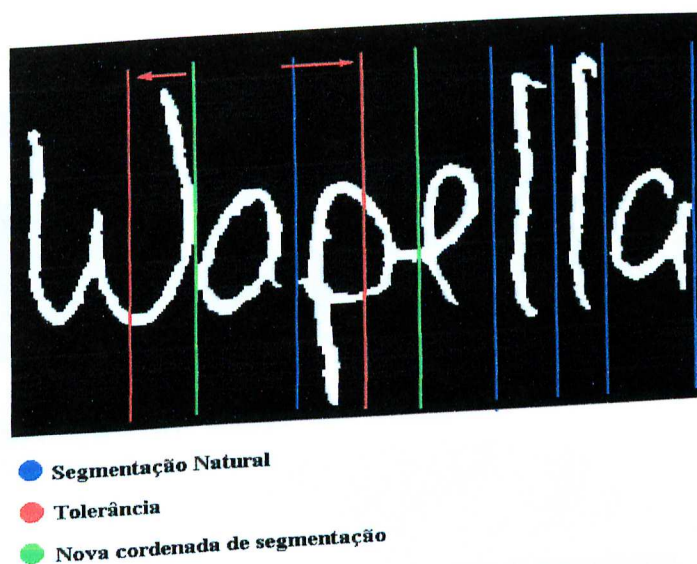


Figura 6.16 - Localização das coordenadas de segmentação do loop.

Após localizar as coordenadas de segmentação do loop, é feita uma verificação no sentido de conferir o tamanho entre as segmentações existentes. Se for maior que o permitido, esse espaço é considerado como sendo mais de uma letra e, devido a isso, ele deve ser particionado onde a coluna candidata a ponto de corte deve oferecer um custo [26, 27] mínimo para a segmentação da imagem da palavra. O custo está definido na Equação 6.1 abaixo:

$$\text{Custo} = (\text{SomaPixelCortadosPelaReta} / (\text{EspessuraTraço} * 2)) \quad 6.1$$

Para cada ângulo que se encontra entre 120° a 60° , variando de $5,5^\circ$, dando um total de 21 ângulos, é projetada uma reta de acordo com o ângulo e calculado o custo [26, 27], entre as retas escolhe a que cortou a menor quantidade de pixel, ou seja, a reta que possuir o menor custo. A Figura 6.17 mostra a aplicação do custo na coluna de corte da imagem de uma palavra.



● Verificação do custo da coluna

Figura 6.17 - Aplicação do custo na coluna de corte na imagem de uma palavra.

Se ainda não foi encontrado nenhum ponto de corte, o algoritmo verifica se entre os pontos de cruzamento existe alguma coordenada de segmentação e se isso não ocorrer, é alocado um ponto de corte nesse intervalo.

6.2.7.2 - geração de pontos de corte na imagem da palavra manuscrita

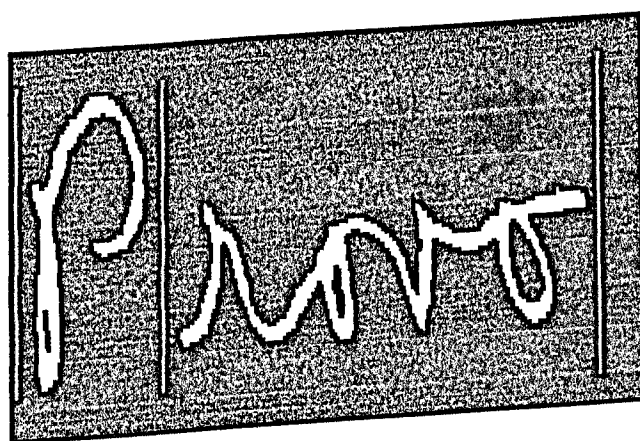
A geração dos pontos de corte na imagem das palavras manuscritas requer algumas regras a mais comparando com as imagens das palavras escritas com letras de fôrma. Estas

regras são: coordenadas dos pontos de mínimos, pontos de cruzamentos, pontos finais, pontos de segmentação natural, e os pontos do loop.

6.2.7.2.1 - segmentação natural da imagem da palavra manuscrita

O método inicia atribuindo a segmentação natural [8] ao conjunto de pontos de corte d imagem d palavra manuscrita. Estes pontos são as possíveis coordenadas dos pontos de segmentação.

A imagem da Figura 6.18 mostra as coordenadas de segmentação natural, que são adicionadas ao conjunto dos pontos de corte de imagem da palavra manuscrita.



○ Segmentação Natural

Figura 6.18 - Imagem da palavra manuscrita segmentada naturalmente.

6.2.7.2.2 - verificação dos pontos de mínimos da imagem da palavra manuscrita

Para localizar os pontos de mínimos, da imagem da palavra manuscrita são analisados os pontos de mínimos na zona mediana e na zona inferior da imagem, com o objetivo de eliminar os mínimos indesejáveis.

Primeiro, são analisados os mínimos que estão localizados na zona mediana, conforme descrito a seguir:

- **MÍNIMO PERTENCE AO LOOP:** eliminam-se os mínimos que estão dentro de um loop, na zona mediana, tais como os mínimos das letras **a**, **b**, **d** e **o**. A Figura 6.19, mostra a eliminação de pontos de mínimos nos loops da imagem de uma palavra manuscrita.

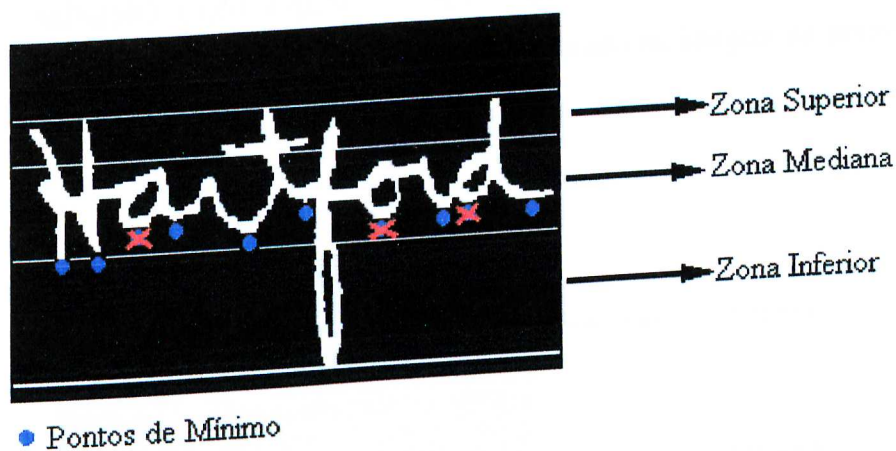


Figura 6.19 - Eliminação dos pontos de mínimos nos loops da imagem de uma palavra.

- **MÍNIMO COM CUSTO:** são eliminados os mínimos que tiverem custo ≥ 2 na zona mediana. Esta análise visa eliminar os mínimos indesejáveis, como está mostrado na letra **h** da Figura 6.20.

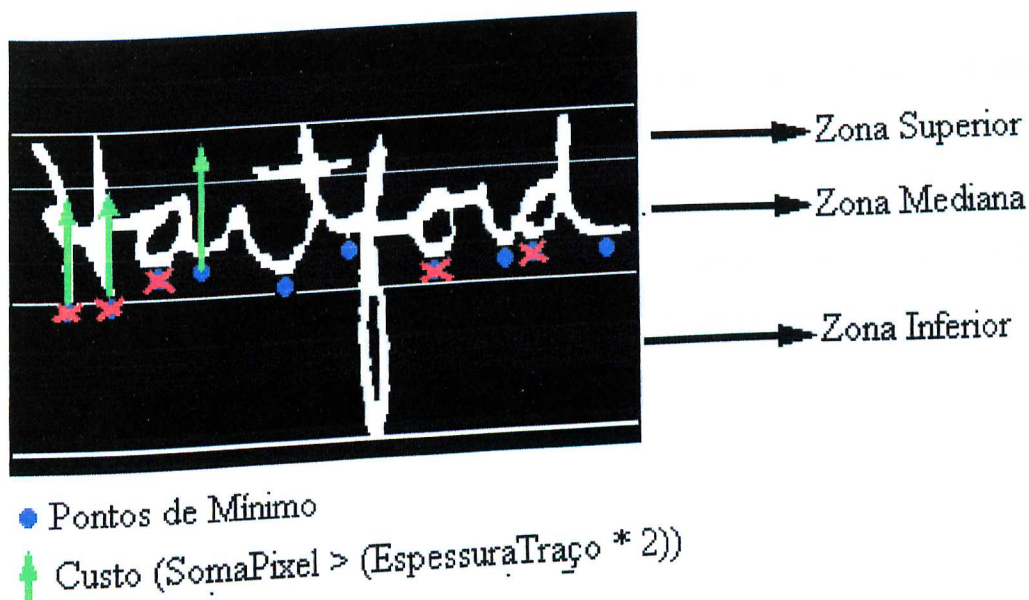


Figura 6.20 - Verificação do custo de corte da imagem de uma palavra manuscrita.

- **MÍNIMO COM FALSOS LOOPS:** elimina os mínimos das letras **a**, **o**, **s** que não formam um loop completo, como esta mostrado na imagem da primeira letra **a** na Figura 6.21.

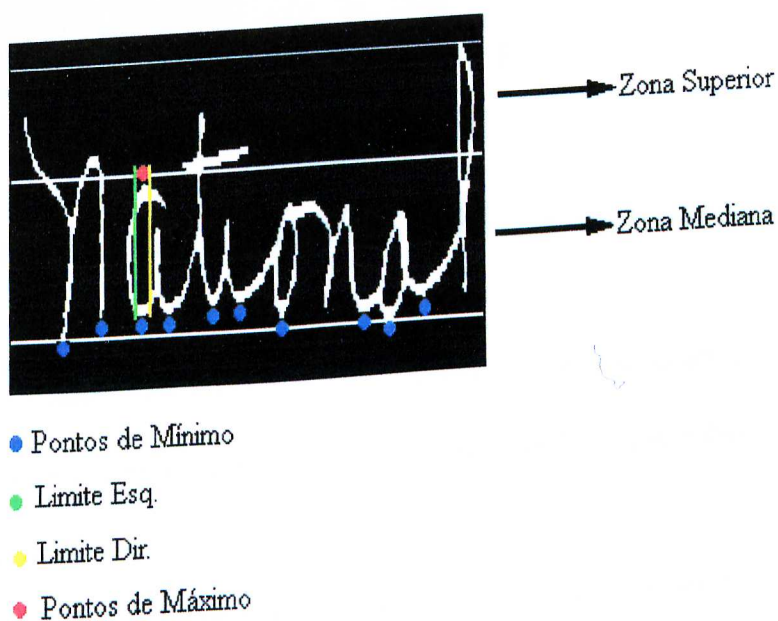


Figura 6.21 - Eliminação dos mínimos nos falsos loops de imagem de uma palavra manuscrita.

Com relação aos falsos loops, se for verificado que o mínimo pertence a eles, esse ponto deve ser apagado, gerando uma nova coordenada.

Uma vez analisados os mínimos na zona mediana, são analisados os mínimos na zona inferior, conforme descrito a seguir:

- É verificado se os mínimos estão dentro de um loop como ocorre nas letras **f**, **g**, **z**, **j** e **g**. Se estiver, o mínimo é apagado, sendo verificada a possibilidade de uma nova coordenada de corte, como mostrado na Figura 6.22.

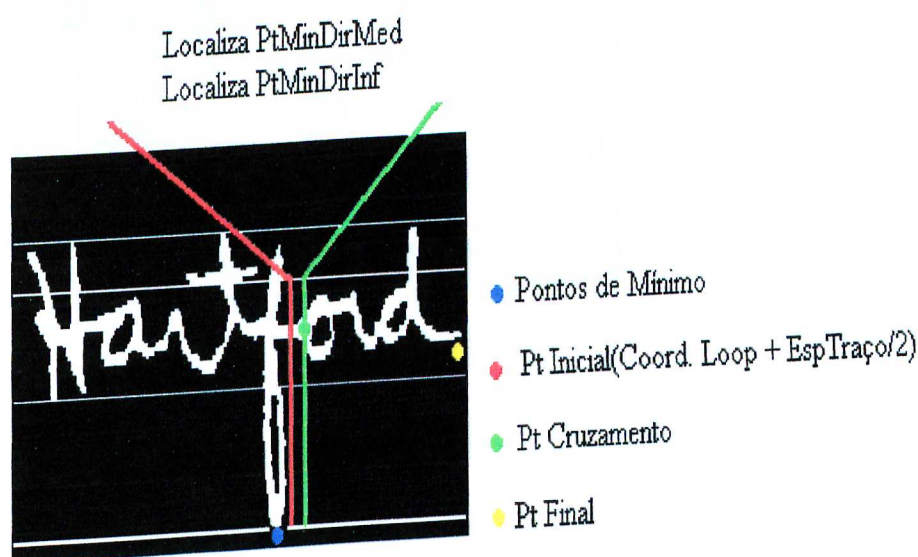


Figura 6.22 - Eliminação dos mínimos na zona inferior da imagem de uma palavra manuscrita.

Como o mínimo da linha foi eliminado, deve-se localizar uma nova coordenada entre o ponto de cruzamento e o ponto final, escolhendo o que estiver mais próximo do ponto de mínimo analisado.

Os mínimos que não foram apagados, são adicionados ao conjunto de pontos de corte, observando a distância entre os mínimos e as coordenadas de segmentação natural, pois, se eles estiverem muito próximos, um deles deve ser eliminado.

6.2.7.2.3 - verificação dos loops na imagem da palavra manuscrita

A verificação dos loops na imagem da palavra é importante, pois é verificado onde existe a possibilidade de segmentação entre as letras que possuem um loop, tais como as letras: **a, o, l, q, b, d, e p**, como mostrado na Figura 6.23.

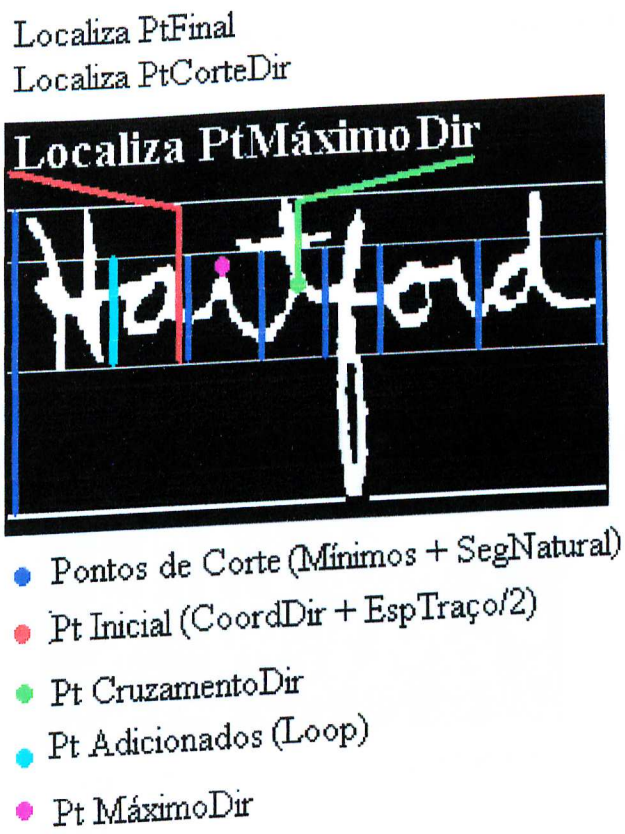


Figura 6.23 - Verificação das coordenadas dos loops na imagem de uma palavra manuscrita.

Na verificação das coordenadas dos loops da imagem de uma palavra manuscrita, cada coluna do LOOP é analisada observando se ela está contida em um espaço onde existem dois pontos de cruzamentos (Pc), ou seja, Pc1 e Pc2. Se isto acontecer, e não existir nenhum ponto de mínimo entre Pc1 e Pc2, a coordenada da coluna do loop pertence ao conjunto de mínimos.

6.2.7.2.4 - verificação dos pontos cruzamento na imagem de uma palavra manuscrita

É necessário a verificação dos pontos de cruzamentos na zona superior na imagem de uma palavra manuscrita, devido a existência de escritores que possuem o costume de emendar o traço da letra **t** com outro traço de outra letra **t**, como mostrado na Figura 6.24.

Na zona mediana, é feita uma verificação se entre as letras másculas não existe mais de um ponto de corte.

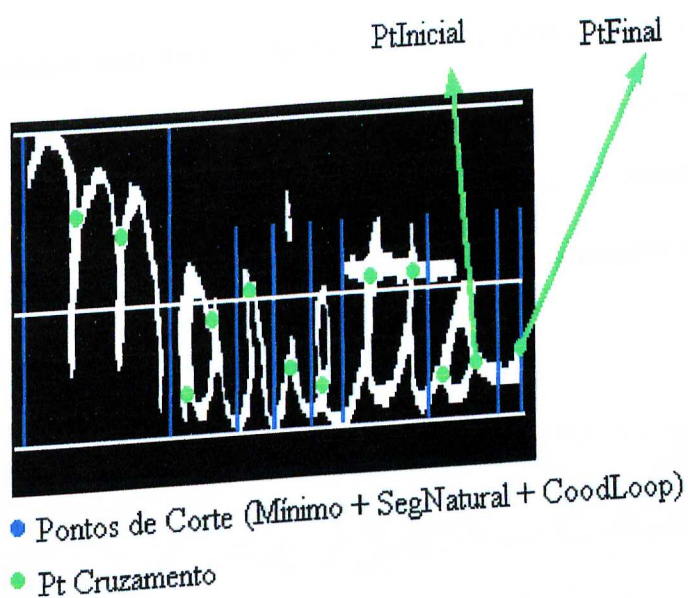


Figura 6.24 - Verificação dos pontos de cruzamento na zona mediana da imagem de uma palavra manuscrita.

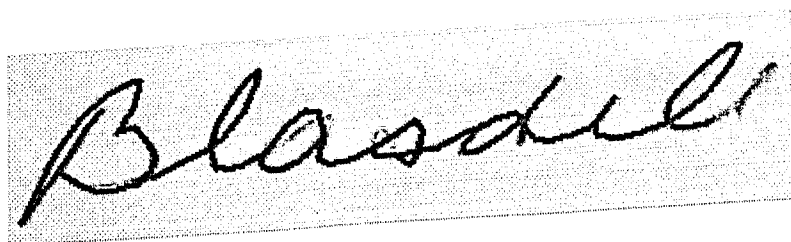
6.2.8 - Corte da imagem da palavra manuscrita nos pontos de segmentação

Após localizar os pontos de corte, é necessário realizar a segmentação das prováveis letras, baseando-se nas coordenadas desses pontos. Sendo assim, os pixels da linha de corte são marcados com o valor dois.

6.3 - Resultados Obtidos

Os resultados obtidos nos testes realizados com as imagens das palavras manuscritas foram obtidos considerando as características existentes na própria imagem da palavra, tais como: loops, pontos finais, pontos de cruzamento, pontos de mínimos e pontos de máximo.

As Figuras 6.25, 6.26, 6.27 e 6.28 mostram as imagens das palavras manuscrita com tons de níveis de cinza antes de ser segmentada, e a imagem desta palavra após ser binarizada e segmentada.

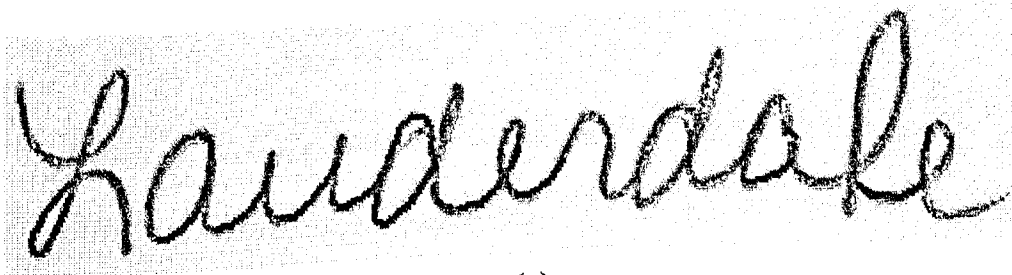


(a)

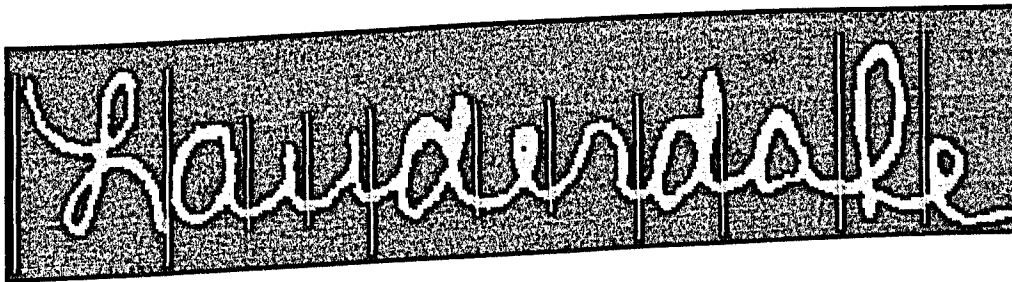


(b)

Figura 6.25 – (a) Imagem da palavra manuscrita com tons de níveis de cinza antes de ser segmentada, (b) Imagem da palavra manuscrita após ser binarizada e segmentada.

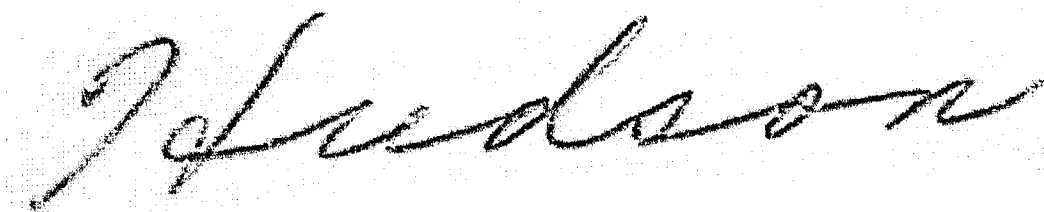


(a)

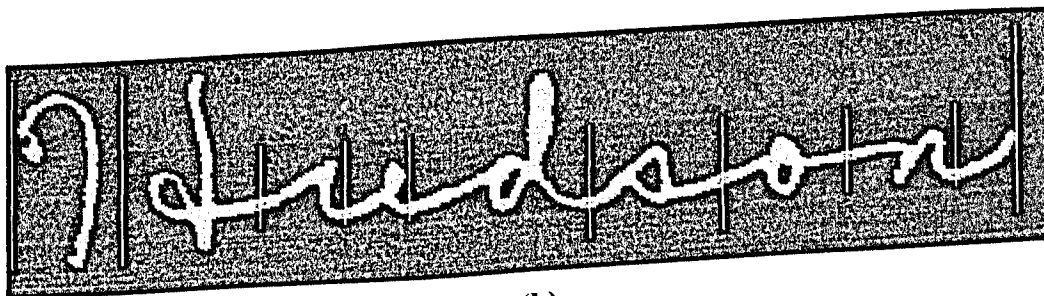


(b)

Figura 6.26 - (a) Imagem da palavra manuscrita com tons de níveis de cinza antes de ser segmentada, (b) Imagem da palavra manuscrita após ser binarizada e segmentada.

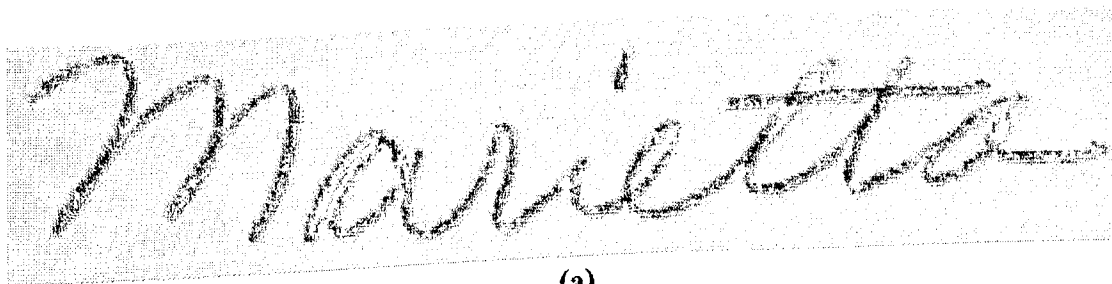


(a)

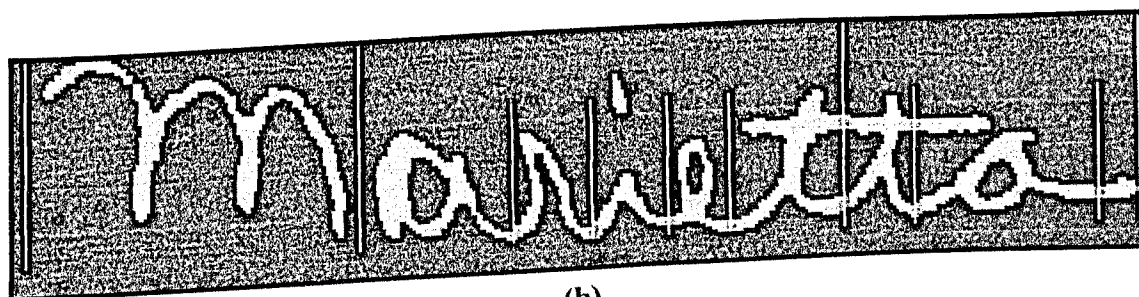


(b)

Figura 6.27 - (a) Imagem da palavra manuscrita com tons de níveis de cinza antes de ser segmentada, (b) Imagem da palavra manuscrita após ser binarizada e segmentada.



(a)



(b)

Figura 6.28 - (a) Imagem da palavra manuscrita com tons de níveis de cinza antes de ser segmentada, (b) Imagem da palavra manuscrita após ser binarizada e segmentada.

6.4 - Conclusão Deste Capítulo

A segmentação de palavras em seus caracteres constituintes é uma tarefa não muito trivial. As palavras cursivas apresentam como um desafio na determinação dos pontos de corte, pois, as letras estão interligadas dificultando a localização do início e do fim de cada uma delas.

Os resultados obtidos com o sistema de segmentação desenvolvido neste trabalho mostram a sua robustez no tratamento de palavras manuscritas. O sistema é independente de quem realizou a escrita. O segmentador é composto de um conjunto de algoritmos que localizam os pontos de segmentação adequados, utilizando apenas informações estáticas da

imagem de entrada, uma vez que a escrita já foi realizada em um momento anterior à sua análise.

Capítulo 7

RESULTADOS OBTIDOS

7.1 - Introdução

As etapas de pré-processamento juntamente com a etapa de segmentação, desenvolvidas neste trabalho, são fundamentais para o bom desempenho do sistema reconhecedor, atuando como um padronizador das imagens das palavras e minimizando as distorções entre elas.

A linguagem utilizada no desenvolvimento deste trabalho foi o MatLab.

7.2 - Resultados Do Processo De Binarização

Como foi descrito no Capítulo 4, deste trabalho, a idéia básica do processo de binarização das imagens, consiste na aplicação do método de Otsu [14]. As imagens originais, utilizadas nos testes foram gravadas com tons de níveis de cinza de alta resolução, na escala de 300 ppi, com 8 bits.

A Figura 7.1 mostra a imagem de uma palavra manuscrita gravada com tons de níveis de cinza de alta resolução, na escala de 300 ppi, com 8 bits.



Figura 7.1 - Imagem de uma palavra manuscrita gravada com tons de níveis de cinza de alta resolução na escala de 300 ppi, com 8 bits.

Nos testes realizados inicialmente, aplicando somente o método de Otsu, mas, como as imagens da base de dados CEDAR [26, 27], é uma representação fiel das condições reais das imagens originais, possuem os ruídos existentes nas correspondências originais. Os resultados da aplicação do método de Otsu está mostrado na Figura 7.2.



Figura 7.2 - Imagem binarizada utilizando somente o método de Otsu.

Para melhorar o resultado final da binarização utilizando o método de Otsu foi necessário, a aplicação prévia do filtro mediano, com a finalidade de reduzir os ruídos, existentes na imagem da palavra, suavizando-a. Os filtros que removem os vales e os picos na

imagem, são aplicados logo após a binarização, com o intuito de remover o pixel que esta de forma irregular na imagem, formando um pico, de espessura de um pixel, ou completando um vale com espessura de um pixel. A Figura 7.3 mostra uma imagem após ser aplicada nela o filtro mediano, o método de Otsu e os filtros de remoção de picos e vales.

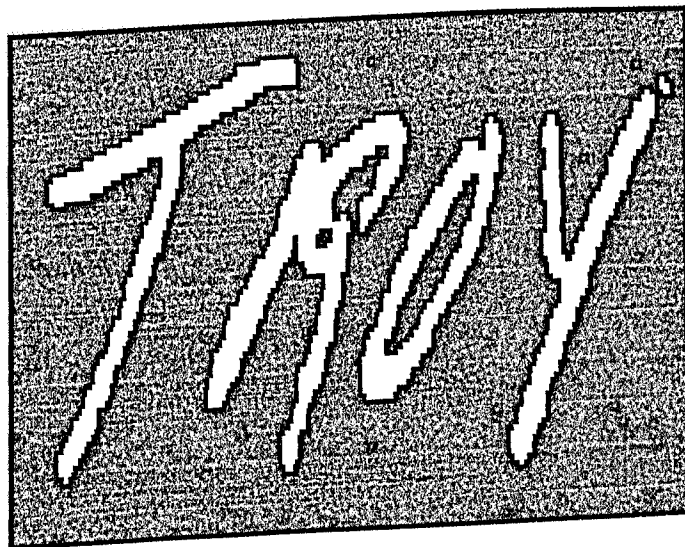


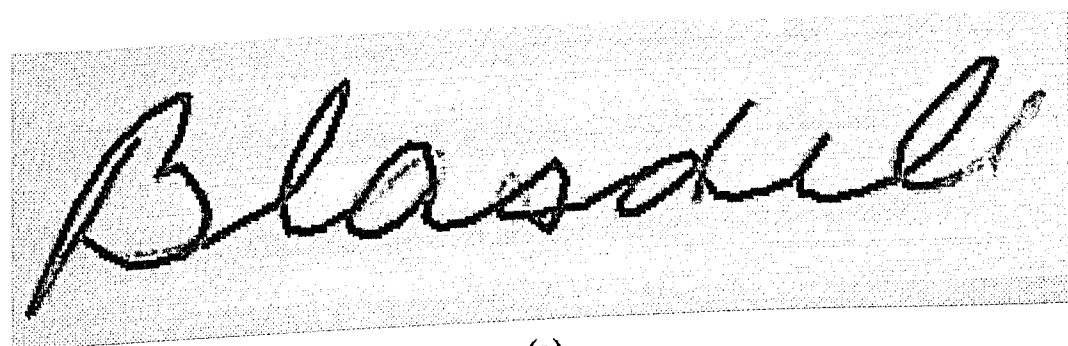
Figura 7.3 - Imagem após ser aplicado nela o filtro mediano, o método de Otsu e os filtros de remoção de picos e vales.

7.3 - Resultados Da Correção Horizontal

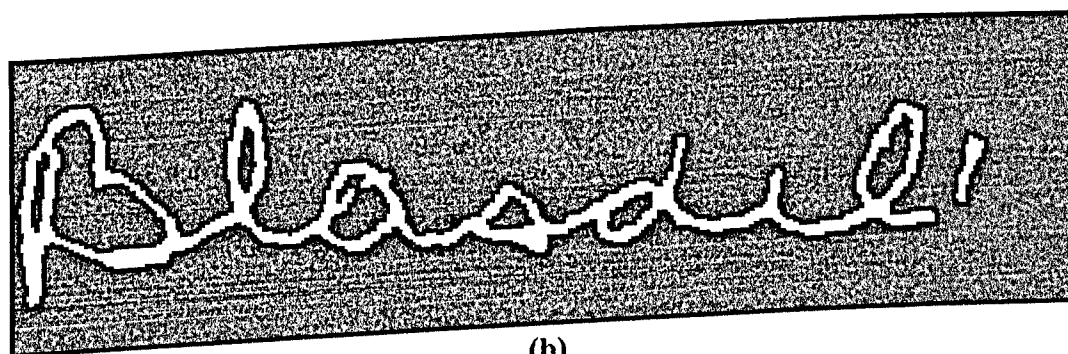
O método de correção horizontal foi baseado no algoritmo de Morita [16], tendo em vista seu melhoramento computacional, e a eliminação dos limiares empíricos, que são empregados na determinação do número de iterações nas etapas da morfologia matemática.

O método de correção horizontal desenvolvido, neste trabalho não emprega a operação morfologia envelope pseudo convexo, reduzindo consideravelmente o tempo de processamento, evitando também a determinação dos fatores empíricos relacionados a essa operação. A operação empregada, para fazer a ligação entre as letras, realiza a localização da

linha média horizontal da palavra. O resultado obtido ao utilizar esse método é mostrado na Figura 7.4.



(a)



(b)

Figura 7.4 – (a) Imagem com tons de níveis de cinza antes de aplicar o método de correção horizontal desenvolvido neste trabalho, (b) Imagem após aplicar o processo de binarização e o método de correção horizontal desenvolvido neste trabalho.

7.4 - Resultados Obtidos Da Segmentação De Palavras

Manuscritas

As imagens das palavras manuscritas empregadas na análise de desempenho do sistema desenvolvido neste trabalho foram extraídas, aleatoriamente, da base de dados CEDAR [28]. As imagens das palavras foram divididas em três grandes grupos, baseando-se

no tipo de letras: de fôrma ou cursiva, e no grau de dificuldade que apresentam para o processo de segmentação. Estes grupos estão mostrados a seguir.

- Grupo 1: pertencem a este grupo, as imagens das palavras escritas com letras de fôrma.
- Grupo 2: contém as imagens das palavras cursivas bem comportadas.
- Grupo 3: pertencem a este grupo as imagens das palavras totalmente cursivas, que possuem maior dificuldade de segmentação, mesmo para um leitor humano, devido ao tipo de caligrafia do escritor, e a presença de alguns tipos de letras e suas combinações, tais como: **m**, **n**, **nr**, **tt**.

7.5 - Critérios De Avaliação

Para poder comparar o desempenho do sistema de segmentação desenvolvido neste trabalho com o desempenho de outros trabalhos [30], foram adotados alguns critérios de classificação de segmentação. Considerando-se estes critérios, a segmentação realizada pelo sistema desenvolvido neste trabalho pode ser classificada como:

- segmentação correta: quando foi encontrado o limite exato entre as letras na palavra.
- segmentação ausente: quando o limite entre as duas letras não foi encontrado.
- segmentação ruim: quando uma letra foi segmentada em até três segmentos.
- segmentação exagerada: quando uma letra foi segmentada em mais de três segmentos.

7.6 - Resultados Experimentais Utilizando-se O Processo De Segmentação Desenvolvido Neste Trabalho

A seguir, são apresentados alguns exemplos de segmentação, por grupo, realizado pelo sistema desenvolvido neste trabalho, baseados nos critérios de avaliação explicados no item 7.5 desta dissertação.

7.6.1 - Grupo 1

A Figura 7.5, mostra a imagem da palavra "Troy" em tons de níveis de cinza escrita em letra de fôrma, inclinada para a direita em relação a vertical. Pode-se observar que, desta maneira, o histograma vertical dessa palavra não conseguiria encontrar os pontos naturais de segmentação.



Figura 7.5 – Imagem da palavra "Troy" inclinada para a direita gravada em tons de níveis de cinza.

Após a utilização dos algoritmos de pré-processamento, os pontos de segmentação natural são determinados pelas posições onde o histograma possui valor igual a zero. A Figura 7.6 mostra a segmentação utilizando o histograma vertical. Como todas as letras da imagem da palavra foram segmentadas corretamente, a taxa de segmentação correta foi de 100%.

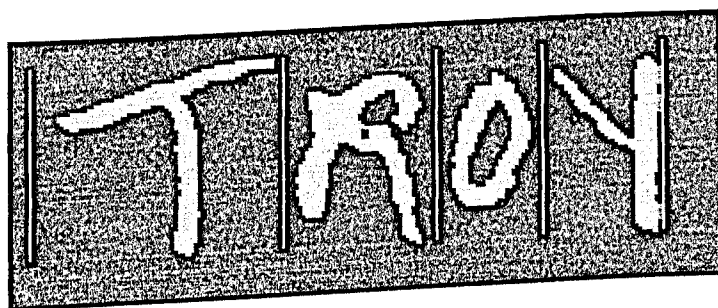


Figura 7.6 – Segmentação utilizando o histograma vertical.

Na imagem da palavra da Figura 7.7, embora existam pontos de segmentação natural, o seu histograma vertical não consegue detectar o ponto de segmentação entre as letras **i** e **n**, antes do pré-processamento da imagem. Um outro problema está na conexão entre a letra **t** e a letra **o**.

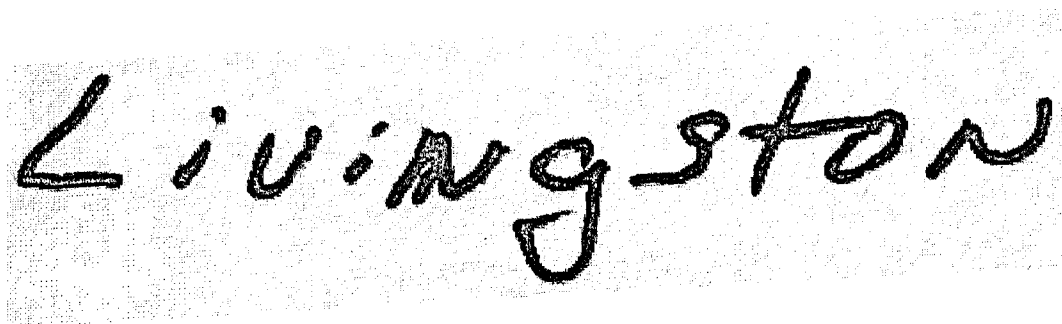


Figura 7.7 – Imagem da palavra “Livingston” gravada em tons de níveis de cinza.

Pode-se observar na Figura 7.8 que, após a etapa de pré-processamento, todos os pontos de segmentação natural foram localizados e, o sistema de segmentação fez corretamente a separação das letras conectadas. Assim, a taxa de segmentação correta desta palavra foi de 100%.

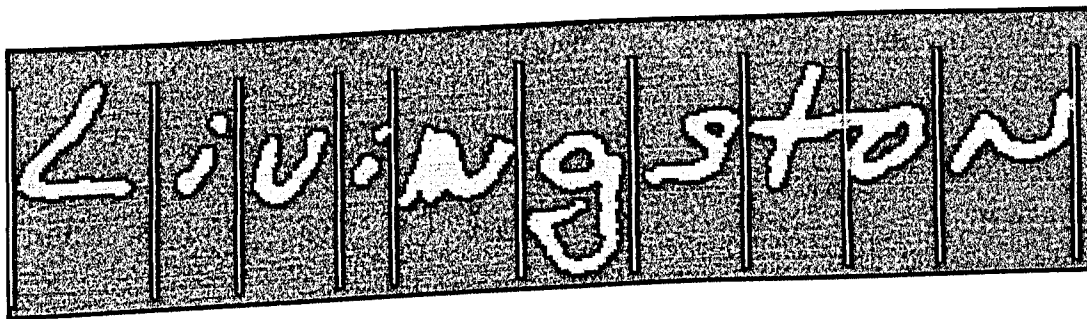


Figura 7.8 – Resultado da segmentação natural da palavra “Livingston”.

7.6.2 - Grupo 2

A seguir são apresentados exemplos de segmentação de palavras puramente cursivas.

A Figura 7.9 mostra a imagem da palavra “Wheat” gravada com tons de níveis de cinza.

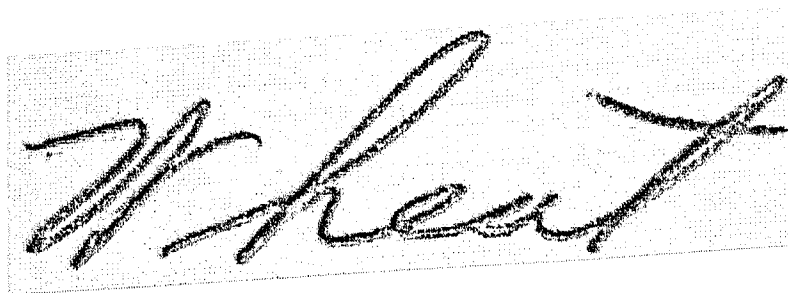


Figura 7.9 – Imagem da palavra “Wheat” gravada com tons de cinza.

A Figura 7.10 mostra a imagem da palavra “Wheat” segmentada naturalmente.

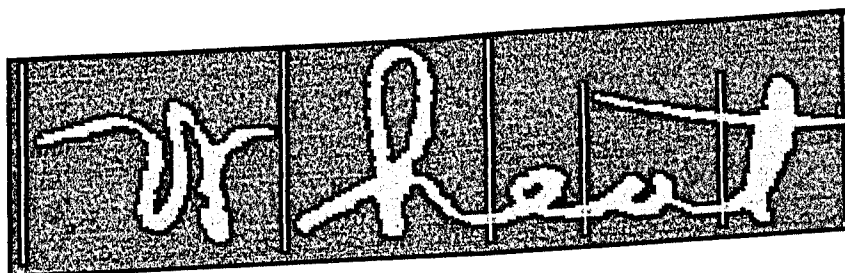


Figura 7.10 – Imagem da palavra “Wheat” segmentada naturalmente.

Pode-se observar na Figura 7.10 que a porcentagem de segmentação correta obtida, com o algoritmo utilizado neste trabalho foi de 100%.

Mesmo tendo na imagem dessa palavra a letra w e a letra a que não estava fechada completamente.

A Figura 7.11 mostra a imagem da palavra "Stuttgart" gravada com tons de níveis de cinza.

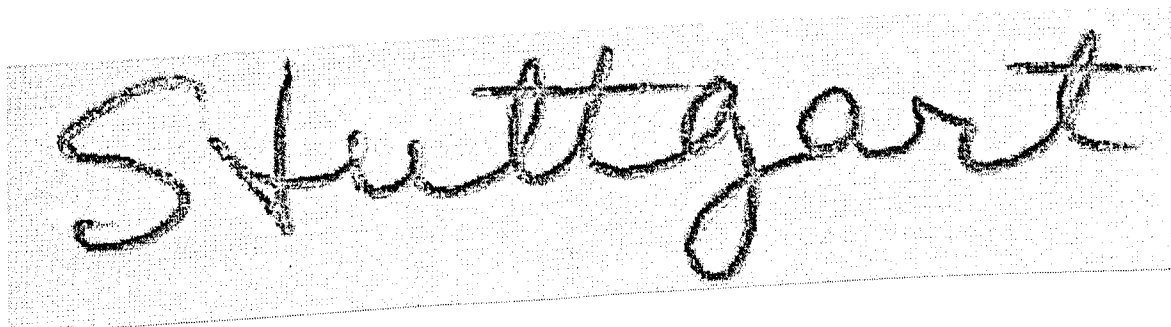


Figura 7.11- Imagem da palavra "Stuttgart" gravada com tons de níveis de cinza.

A Figura 7.12 mostra a imagem da palavra "Stuttgart" segmentada pelo algoritmo utilizado neste trabalho.

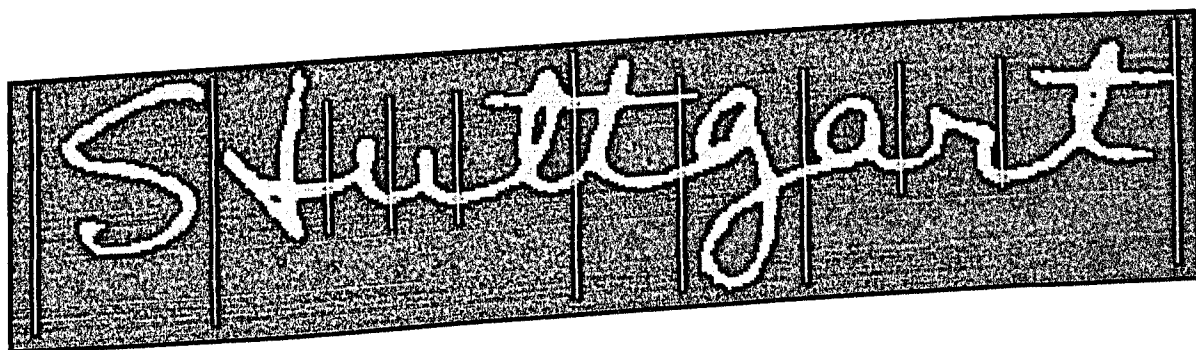


Figura 7.12 – Imagem da palavra "Stuttgart" segmentada pelo algoritmo utilizado neste trabalho.

Pode-se observar na Figura 7.12 que a letra u foi segmentada em duas partes. Isto se deve ao fato do sistema interpretá-la como sendo constituída de duas letras i ou duas letras e. Já, as duas letras t, que o escritor emendou com um traço superior, o sistema conseguiu

realizar a segmentação corretamente. A porcentagem de segmentação correta do sistema para esta palavra foi de 88,88%.

A Figura 7.13 mostra a imagem da palavra "Fairfield" gravada com tons de níveis de cinza.

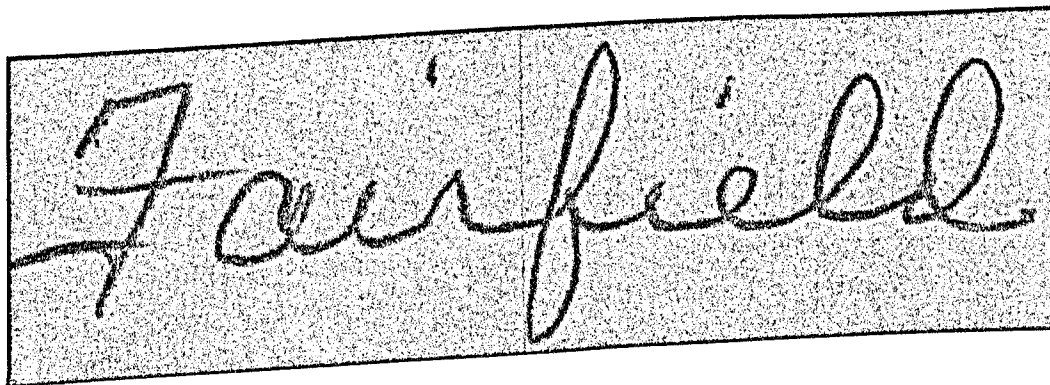


Figura 7.13 – Imagem da palavra "Fairfield" gravada com tons de níveis de cinza.

A Figura 7.14 mostra a imagem da palavra "Fairfield" segmentada pelo algoritmo utilizado neste trabalho.

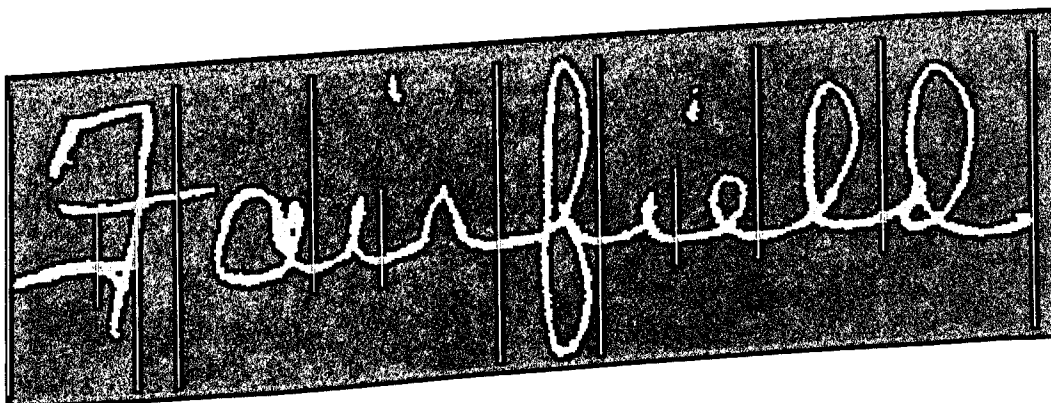


Figura 7.14 - Imagem da palavra "Fairfield" segmentada pelo algoritmo utilizado neste trabalho

Pode-se observar na Figura 7.14, que a letra "F" da palavra, foi dividida em 3 segmentos devido a presença de um pequeno loop na sua parte inferior. A porcentagem de segmentação correta do sistema para esta palavra foi de 88,88%.

7.6.3 - Grupo 3

A seguir, são mostrados exemplos de palavras que são mais difíceis de serem segmentadas. A Figura 7.15 mostra a imagem da palavra "Tanauaanda" gravada com tons de níveis de cinza.

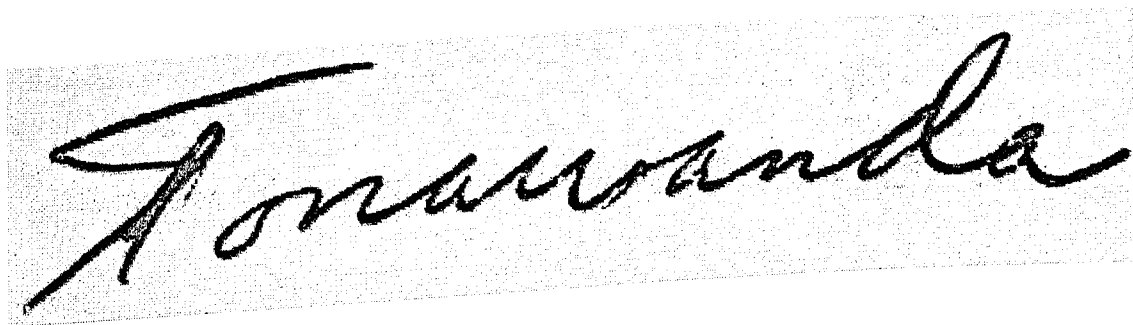


Figura 7.15 - Imagem da palavra "Tanauaanda" gravada com tons de níveis de cinza.

A Figura 7.16 mostra a imagem da palavra "Tanauaanda" segmentada pelo algoritmo utilizado neste trabalho.

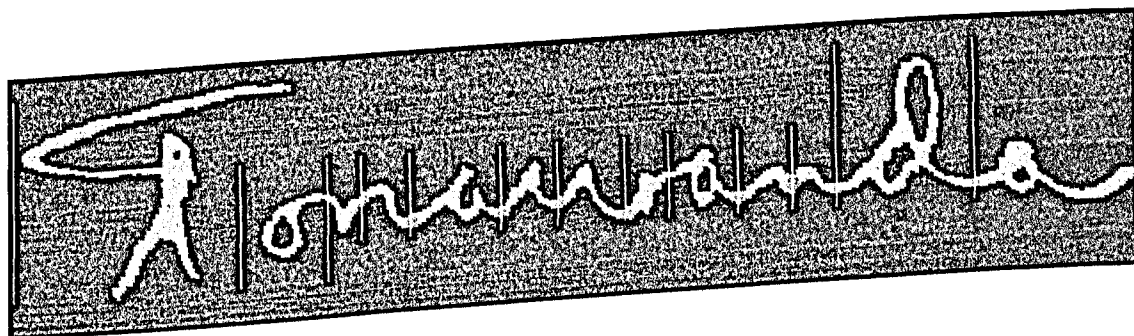


Figura 7.16 - Imagem da palavra "Tanauaanda" segmentada pelo algoritmo utilizado neste trabalho.

Na Figura 7.16 pode-se observar a complexidade de algumas escritas manuais. Caso o leitor humano desconheça a palavra, dificilmente ela poderá ser lida corretamente. Um sistema que trabalha apenas com as informações estáticas da imagem, também, terá dificuldade na segmentação. A porcentagem de segmentação correta da imagem da palavra dessa figura foi de 70%.

A Figura 7.17 mostra a imagem da palavra "Ardmore" gravada com tons de níveis de cinza.

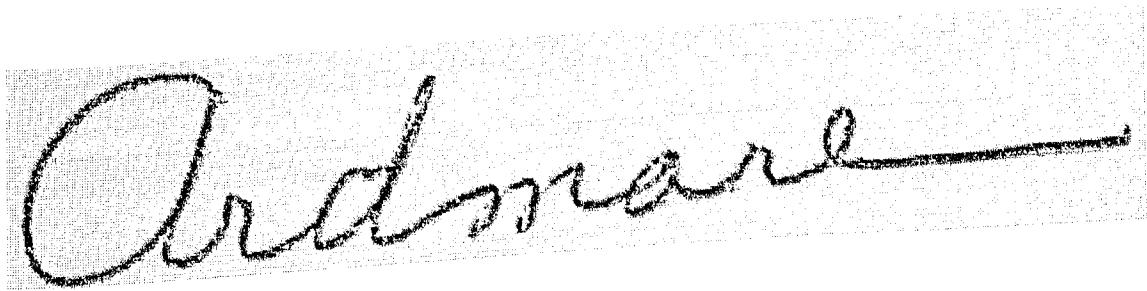


Figura 7.17 - Imagem da palavra "Ardmore" gravada com tons de níveis de cinza.

A Figura 7.18 mostra a imagem da palavra "Ardmore" segmentada pelo algoritmo utilizado neste trabalho.

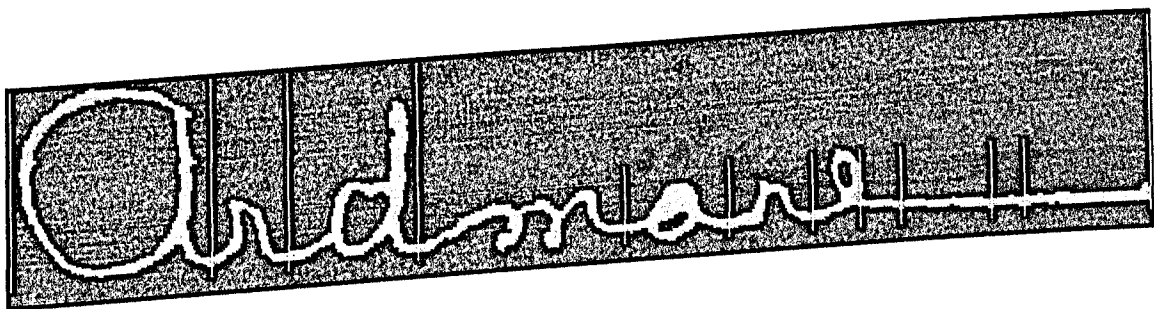


Figura 7.18 - Imagem da palavra "Ardmore" segmentada pelo algoritmo utilizado neste trabalho.

Pode-se observar na Figura 7.18 que o sistema de segmentação considera a letra e da palavra como sendo mais de uma letra, pois ela possui um traço fora do padrão. A porcentagem de segmentação correta do sistema para esta palavra foi de 85,71%.

7.7 - Resultados Obtidos

As técnicas de pré-processamento e segmentação desenvolvidas neste trabalho foram aplicadas a 300 palavras, que possuem um total de 2045 letras. Estas imagens de palavras

foram extraídas aleatoriamente da base de dados do CEDAR [29]. São imagens manuscritas reais de documentos que passaram por postos de correios americanos

A tabela 7.1 mostra os resultados obtidos utilizando-se as imagens das palavras escritas com letras de fôrma (Grupo 1), com letras cursivas bem comportadas (Grupo 2) e com letras cursivas com maior dificuldade de segmentação (Grupo 3) utilizando-se as técnicas de pré-processamento e segmentação desenvolvidas neste trabalho.

Tabela 7.1 – Resultados obtidos nos grupos 1, 2, 3 utilizando as técnicas de pré-processamento e segmentação desenvolvidas neste trabalho.

Tipos de grupos	Tipos de segmentação				Total de letras	Total de Palavra
	Exagerada	Ruim	Ausente	Correta		
Quantidade segmentada do Grupo 1	0	2	24	668	694	100
Porcentagem de acerto do Grupo 1	0	0,288	3,458	96,25		
Quantidade segmentada do Grupo 2	0	2	47	564	613	100
Porcentagem de acerto do Grupo 2	0	0,326	7,667	92,01		
Quantidade segmentada do Grupo 3	1	7	96	634	738	100
Porcentagem de acerto do Grupo 3	0,136	0,949	13,01	85,91		

Ao observar a Tabela 7.1 pode-se observar que a maior porcentagem de segmentação obtida foi de 96,25% no Grupo 1. Neste grupo, estão as imagens das palavras escritas com letras de fôrma, o que facilita a localização dos pontos de segmentação, mesmo sendo estas palavras escritas com inclinação, vertical ou horizontal, o pré-processamento desenvolvido neste trabalho faz o tratamento destas inclinações e possibilita esta porcentagem de segmentação.

Ao analisar a tabela 7.1 pode-se verificar que a porcentagem de segmentação obtida no Grupo 2 foi de 92%. Ao comparar a porcentagem de segmentação deste Grupo, com a porcentagem de segmentação do Grupo 1, pode-se verificar que a porcentagem de segmentação do Grupo 2 é menor, devido as imagens das palavras neste grupo serem puramente cursivas, e apresentarem um grau de dificuldade maior na determinação dos limites de cada letra, pois cada letra está conectada às outras letras da palavra.

Ao observar a Tabela 7.1, pode-se verificar que a porcentagem de segmentação do Grupo 3 foi de 85,90%. As palavras deste grupo apresentam deformações e letras que dificultam a sua leitura até para um leitor humano, caso não haja um prévio conhecimento das mesmas. Algumas destas letras são: **ph, an, m, w, n, rm, vv e do.**

A tabela 7.2 mostra o resultado geral, sendo a média entre os resultados conseguidos com cada grupo de palavra.

Tabela 7.2 – Resultado final obtido com a média dos grupos 1, 2, 3.

Resultado Geral	Tipos de segmentação				Total de letras	Total de Palavra
	Exagerada	Ruim	Ausente	Correta		
Quantidade segmentada geral	1	11	167	1866	2045	300
Porcentagem de acerto geral	0,049	0,538	8,163	91,25		

Ao analisar a Tabela 7.2 pode-se verificar que a porcentagem de segmentação das imagens das 300 palavras constituídas de 2045 letras foi de 91,25%, com 1866 letras segmentadas corretamente. Xiao e Leedham [30] fizeram experimentos com 200 imagens de palavras, ou seja, menos 100 imagens de palavras comparado com o total das imagens de palavras utilizadas neste trabalho. As imagens de palavras utilizadas por Xiao e Leedham foram nomes de cidades obtidos usando a mesma base de dados do CEDAR. Nestas imagens

das palavras 66 foram consideradas de difícil segmentação, onde estas palavras são compostas por letras: **vv, w, ui, m e n**.

A Tabela 7.3 mostra os resultados obtidos por Xiao e Leedham [30].

Tabela 7.3 - Resultados obtidos por Xiao e Leedham.

Segmentação Exagerada(%)	Segmentação Inexistente(%)	Segmentação Ruim(%)	Segmentação Correta(%)
0	0.5	1.8	82.9

Ao comparar a porcentagem de segmentação da Tabela 7.2 com a porcentagem de segmentação da Tabela 7.3, pode-se verificar que a porcentagem de segmentação da Tabela 7.2 foi bem maior, ou seja os resultados obtidos utilizando-se as técnicas de pré-processamento e segmentação desenvolvidas neste trabalho são bem melhores do que os resultados obtidos por Xiao e Leedham.

7.8 - Conclusão

Neste capítulo foram mostrados os resultados obtidos com os testes realizados com as técnicas de pré-processamento e segmentação desenvolvidos neste trabalho.

Capítulo 8

CONCLUSÕES, CONTRIBUIÇÕES DO TRABALHO E SUGESTÕES PARA FUTUROS TRABALHOS.

8.1 - Conclusões

Neste trabalho, foram desenvolvidos os módulos de pré-processamento e segmentação de caracteres para o desenvolvimento futuro de um sistema automático de reconhecimento de palavras manuscritas. As imagens utilizadas nesses módulos foram obtidas de uma base de dados construída a partir de imagens de correspondências que passaram por um posto de correio, por um período de tempo. O sistema desenvolvido neste trabalho é independente do estilo do autor da escrita. Para tratar os problemas que se apresentam no processamento de imagens de palavras manuscritas, foram desenvolvidos algoritmos baseados em técnicas de processamento digital de imagens.

No módulo de pré-processamento, foram desenvolvidas as seguintes etapas: filtragem, binarização, correção horizontal, correção vertical e afilamento. Verificou-se que a aplicação imediata do método de Otsu na imagem original, produzia uma binarização com excesso de pixels espúrios. A aplicação prévia do filtro mediano resolveu este problema, permitindo a geração de uma imagem binarizada melhor e isenta de eventuais ruídos presentes nas imagens originais, e com isso preservando as informações essenciais para os módulos subsequentes do sistema reconhecedor. No cálculo da correção horizontal, a utilização das linhas superior, inferior e central permitiram a determinação correta dos pontos de mínimo do contorno inferior da palavra. Com a aplicação do método dos mínimos quadrados sobre estes

pontos, foi realizado o cálculo da reta que melhor reflete a inclinação da palavra. Este algoritmo mostrou-se eficiente no tratamento de inclinação em aclave e declive da escrita. O algoritmo desenvolvido para o tratamento da correção vertical, foi baseado na geração de histogramas verticais da palavra rotacionada em um intervalo de 60° . Foram obtidos 21 histogramas e, dentre eles, os três que apresentaram os maiores picos foram utilizados no cálculo da inclinação vertical. Este algoritmo se mostrou eficiente e preciso na correção vertical, tanto à esquerda quanto à direita.

A partir dos resultados obtidos no módulo de pré-processamento, foram aplicadas as técnicas de segmentação desenvolvidas neste trabalho. A segmentação de caracteres se baseia na determinação dos pontos de mínimos do contorno inferior da imagem da palavra afilada. As informações geradas na etapa de correção vertical, ou seja, localização dos pontos de mínimos e pontos de máximo são utilizadas neste módulo. Neste módulo foram implementadas as seguintes etapas: localização da segmentação natural, verificação dos pontos de mínimo, determinação do conjunto de pontos dos loops, verificação do conjunto de pontos de máximo, verificação do conjunto de pontos de cruzamento, verificação do conjunto de pontos finais, localização dos pontos de segmentação da palavra e corte da imagem da palavra nos pontos de segmentação.

O histograma vertical mostrou-se adequado para a determinação dos pontos de segmentação natural. Para que a imagem das letras que apresentam loops não sejam segmentadas incorretamente, foi desenvolvido um algoritmo que faz a sua localização na imagem da palavra. O algoritmo desenvolvido neste trabalho para a localização dos loops baseia-se no conceito de vizinhança longa, e os resultados obtidos foram bastante satisfatórios. Para a determinação dos pontos de cruzamento em excesso, na imagem da palavra foi definida uma janela cujo tamanho é baseado na espessura do traço da palavra corrigida verticalmente. Esta janela é aplicada na imagem da palavra afilada e, se existirem

dois ou mais pontos de cruzamento dentro dela, o que estiver mais próximo do loop é preservado, e os pontos de cruzamentos restantes considerados em excesso são eliminados. Os resultados obtidos com a aplicação dessa janela foram satisfatórios.

Nos teste realizados neste trabalho foram utilizados as imagens das palavras escritas com letra de fôrma (Grupo 1), com letras cursivas bem comportadas (Grupo 2) e com letras cursivas com maior dificuldade de segmentação (Grupo 3). A maior porcentagem de segmentação obtida nos teste foi de 96,25% no Grupo 1. As imagens das palavras deste grupo facilitaram a localização dos pontos de segmentação, mesmo sendo estas palavras escritas com inclinação vertical ou horizontal, o pré-processamento desenvolvido neste trabalho faz o tratamento destas inclinações e possibilita esta porcentagem de segmentação alta.

A porcentagem de segmentação obtida no Grupo 2 foi de 92%. Este valor obtido é menor que a porcentagem de segmentação do Grupo 1. Isto ocorreu devido as imagens das palavras do Grupo 2 serem puramente cursivas e apresentarem um grau de dificuldade maior na determinação dos limites de cada letra, pois cada letra está conectada as outras na palavra.

A porcentagem de segmentação do Grupo 3 foi de 85,9%. As imagens das palavras deste grupo apresentam deformações e letras que dificultam a sua leitura até para o ser humano, caso não haja um prévio conhecimento das mesmas. Algumas das letras são: **ph, na, m, w,n, rm, vv e do.**

A porcentagem de segmentação das imagens das 300 palavras constituídas de 2045 letras foi de 91,25%, com 1866 letras segmentadas corretamente.

A porcentagem de segmentação dos testes realizados neste trabalho foi bem maior que a porcentagem de segmentação dos testes realizados por Xiao e Leedham. Isto significa que os resultados obtidos utilizando-se as técnicas de pré-processamento e segmentação desenvolvidos neste trabalho foram bem melhores do que os resultados obtidos por Xiao e Leedham.

8.2 - Contribuições Do Trabalho

Os desafios que se apresentam no desenvolvimento de um sistema de reconhecimento automático de palavras manuscritas estão, principalmente, na variedade de estilos de cada escritor. As etapas de pré-processamento são de fundamental importância no tratamento desta variação da escrita, atuando como padronizador dos diversos estilos de escrever que cada pessoa possui, elevando o índice do reconhecimento resultante, uma vez que o desempenho da etapa de classificação de palavras é fortemente dependente da qualidade de estilos das imagens que recebe. A etapa de segmentação visa a divisão da palavra em letras. A dificuldade em segmentar uma imagem de uma palavra, está no posicionamento exato do corte, evitando que as letras da palavra sejam separadas incorretamente. A proposta deste trabalho foi a realização de uma contribuição no aprimoramento do processo de manipulação da imagem de entrada de tal forma que o sistema possa ser robusto o suficiente no tratamento de escritas que não são bem comportadas.

8.3 - Sugestões Para Trabalhos Futuros

Alguns trabalhos que poderão ser desenvolvidos futuramente, são eles:

- desenvolvimento de mecanismos de consistência dos resultados do processo de segmentação.
- implementação da etapa de reconhecimento de imagem de palavras, que poderá ser uma fonte de informação para a correção de possíveis erros no processo de segmentação.
- criação de uma base de dados com imagens de palavras da língua portuguesa.

REFERÊNCIAS

- [1] FLÔRES, E. L., Reconhecimento de Caracteres Numéricos Manuscritos que Considera as Características Intrínsecas e Comuns de Cada Dígito, Tese (Doutorado em Engenharia Elétrica), Universidade Estadual de Campinas, vol. 1, 1997.
- [2] SENIOR, A. W., Off-Line Cursive Handwriting Recognition Using Recurrent Neural Networks, Thesis is submitted for consideration for the degree of Doctor of Philosophy, at the University of Cambridge, September-1994.
- [3] NISHIDA H., AND MORI S., Algebraic Description of Curve Structure, IEEE, Transactions on pattern analysis and machine intelligence, IEEE, vol 14, No. 5, May 1992.
- [4] BOZINOVIC R. C., SHIHARI S. N., Off-line Cursive Script Word Recognition, IEEE Transactions on patten analysus and machine intelgence, Vol 11, No 1, January 1989.
- [5] BEIJI, H. S. M., NATHAN K., CLARY G. J., SUBRAHMONIA J., Size Normalization In On-Line Unconstrained Handwriting Recognition, T. J. Watson Research Center, IBM, Yorktown Heights, New York 10598.
- [6] YANIKOGLU B. A., SANDON P. A., Off-line Cursive handwriting Recognition Using Style Parameters, Department of Mathematics and Computer Science Dartmouth College Hanover, NH, 03755, June, 1993
- [7] CHING Y.S.,CHRISTINE N.,RAYMOND L.,TUAN A. M., LOUISA L., Computer Recognition of Unconstrained Handwritten Numerals, IEEE, Vol. 80, July-1992.

- [8] FUJISAWA H., NAKANO Y., KURINO K., Segmentation Methods for Character Recognition: From Segmentation to Document Structure Analysis, IEEE, Vol 80, July-1992.
- [9] BLUMENSTEIN M., VERMA B., An Artificial Neural Network Based Segmentation Algorithm for Off-line Handwriting Recognition, Faculty of Information and Communication Technology Griffith University, Gold Coast Campus, Australia.
- [10] VELOSO L. R., SOUSA R. P., CARVALHO J. M., Morphological Cursive Word Segmentation, Universidade Federal da Paraíba, IEEE, 2000.
- [11] GOMES N. R., Reconhecimento de Palavras Manuscritas Baseado em HMM e no Emprego de Características Topológicas e Geométricas, Tese, Universidade Estadual de Campinas, 2000.
- [12] HAN K., AND SETHI I. K., Off-line Cursive Handwriting Segmentation, Vision and Neural Networks Laboratory, Department of Computer Science Wayne State University, Detroit, MI 48202, IEEE, 1995.
- [13] MO, S., and MATHEWS, V. J., Adaptive, Quadratic Preprocessing of Document Images for Binarization. IEEE Trans. on Image Processing, vol. 7, pp. 992-999, Jul. 1998.
- [14] OTSU N., A threshold Selection Method From Gray Level Histograms, IEEE Trans. Sys. vol. Smc-9, no. 1, january-1979.
- [15] TSURUOKA S., WATANABE N., MINAMIDE N., KIMURA F., MIYAKE Y., SHRIDHAR M., Base Line Correction for Handwritten Word Recognition, IEEE, Vol 95, 1995.
- [16] MORITA M. E., Estudo Para a Melhoria da Correção da Inclinação da Linha de Base de Palavras Mnauscritas, Dissertação, CEFET-PR, 1998.

- [17] BUSE R., LUI Z., CAELLI T., A., Structural and Relational Approach to Handwritten Word Recognition, IEEE Transactions on Systems, Man. And Cybernetics, Part B, Cybernetics Vol. 27, no. 5, October 1997.
- [18] HAN N. H., LA C. W., RHEE P. K., An Efficient Fully Parallel Thinning Algorithm, IEEE, 1997.
- [19] PARKER, J. R., Algorithms for Image Processing and Computer Vision, John Wiley and Sons, 1996.
- [20] GONZALES R. C. AND WOODS R. E., Processamento de Imagens Digitais, Editora Edgard Blücher LTDA, 2000.
- [21] FACON, J., Morfologia Matemática: teoria e exemplos, Editora Universitária Champagnat PUC-PR, Curitiba, Paraná, 1996.
- [22] SERRA, J., Image Analysis and Mathematical Morphology, Academic Press, London, 1982.
- [23] MENDENHALL W., Probabilidde e Estatística, Vol. 2, Editora Campus Ltda, 1985.
- [24] BERGER M., Computer Graphics, The Benjamin/Cummings Publishing Company, Inc 1986.
- [25] BORGES, MARC., Computer Graphics with Pascal, The Benjamin/Cummings Publishing Company, Inc., 1985.
- [26] PAVEL S., STEPAN H., MICHAL M., Off_line Recognition of Cursive Handwritten Czech Text, Faculty of Informatics Masaryk University, February, 1998.
- [27] KIMURA F., TSURUOKA S, MIYAKE Y., SHIDHAR M., A lexicon Directed Algorithm for Recognition of Unconstrained Handwritten Words, IEICE, trans. Inf. & syst. Vol. E77-D, No. 7, July 1994.

- [28] <http://www.cedar.buffalo.edu/Databases/CDROM1/>, Usp's Office Of Advanced Technology of Handwritten Cities, States, Zip Codes, Digits, and Alphabetic Characteres, 2002.
- [29] HULL H. J., A Database For Handwritten Text Recognition Research, Center of Excellence for Document Analysis and Recognition(CEDAR), Department of Computer Science State University of New York at Buffalo, IEEE, 1993.
- [30] XIAO X., LEEDHAM, Knowledge-based English cursive script segmentation, Patter Recognition Letters 21, 945-954, 30 May 2000.

LISTA DE PUBLICAÇÕES

- [1] PARREIRA F. J., YAMANAKA K., Segmentation Of Handwriting Founding At Images Processing Techinique, ACIVS 2002, submetido.
- [2] PARREIRA F. J., YAMANAKA K., Pré-Processamento e Segmentação de Palavras Manuscritas baseadas em Técnicas de Processamento de Imagens, REVISTA CIÊNCIA & ENGENHARIA, submetido.