

Universidade Estadual de Campinas  
Faculdade de Engenharia Elétrica e de Computação

# Melhorias para Sistemas de Reconhecimento da Placa de Licenciamento Veicular

**Autor: Eng. Fábio Gaiotto Dias**

**Orientador: Prof. Dr. Roberto de Alencar Lotufo**

Dissertação de Mestrado apresentada à Faculdade de Engenharia Elétrica e de Computação como parte dos requisitos para obtenção do título de Mestre em Engenharia Elétrica. Área de concentração: Engenharia de Computação.

## Banca Examinadora

Prof. Dr. Altamiro Amadeu Susin ..... DELET/UFRGS  
Prof. Dr. Clésio Luis Tozzi ..... DCA/FEEC/UNICAMP  
Prof. Dr. Fernando José Von Zuben ..... DCA/FEEC/UNICAMP

Campinas, SP

Outubro/2005

|   |                    |
|---|--------------------|
| Este exemplar corresponde a redação final de tese |                    |
| defendida por                                     | Fábio Gaiotto Dias |
| * aprovada pela Comissão                          |                    |
| Julgada em  | 25 / 10 / 2005     |
| Orientador  |                    |

200604640

|            |          |
|------------|----------|
| UNIDADE    | BC       |
| Nº CHAMADA |          |
| TITULO     |          |
| V.         | EX       |
| TOMBO BCI  | 67107    |
| PROC.      | 123106   |
| PREÇO      | 11,00    |
| DATA       | 27/10/06 |
| Nº CPD     |          |

161810-375882

FICHA CATALOGRÁFICA ELABORADA PELA  
BIBLIOTECA DA ÁREA DE ENGENHARIA - BAE - UNICAMP

|            |   |
|------------|---|
| D<br>M543m | <p>Dias, Fábio Gaiotto<br/>Melhorias para sistemas de reconhecimento da placa de licenciamento veicular / Fábio Gaiotto Dias. --Campinas, SP: [s.n.], 2005.</p> <p>Orientador: Roberto de Alencar Lotufo<br/>Dissertação (Mestrado) - Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação.</p> <p>1. Sistemas de reconhecimento de padrões. 2. Visão por computador. 3. Percepção visual. 4. Processamento de imagens. 5. Inteligência artificial. I. Lotufo, Roberto de Alencar. II. Universidade Estadual de Campinas. Faculdade de Engenharia Elétrica e de Computação. III. Título.</p> |
|------------|---|

Título em Inglês: Improvements for vehicle license plate recognition systems

Palavras-chave em Inglês: Vehicle automatic recognition, Computer vision, Image processing, Pattern recognition, Artificial intelligence

Área de concentração: Engenharia de Computação

Titulação: Mestre em Engenharia Elétrica

Banca examinadora: Altamiro Amadeu Susin, Clésio Luis Tozzi e Fernando José Von Zuben

Data da defesa: 25/10/2005

# Resumo

Sistemas de reconhecimento da placa de licenciamento veicular (SRPLV) identificam o código da placa de um veículo com base em sua imagem. Suas principais aplicações são: detecção de veículos roubados; controle de acesso em áreas restritas; monitoramento de tráfego; fiscalização de trânsito; controle automático de estacionamentos e de pedágios. Neste trabalho, descrevemos melhorias para SRPLV. As principais contribuições estão no método de localização da placa, efetuando a binarização com base nos mínimos regionais da imagem filtrada; no enquadramento com correção da distorção de perspectiva da placa, através da homografia; na separação dos caracteres com auxílio da projeção vertical; e no reconhecimento dos caracteres, utilizando um classificador hierárquico de distância mínima. Nossa proposta obteve resultado 1,17 vezes superior na localização da placa e 1,58 vezes superior no correto reconhecimento da placa em relação ao melhor resultado entre duas demonstrações de sistemas comerciais disponíveis na internet (SeeCar e SIAV 3.0), usando um banco de dados com 1200 imagens.

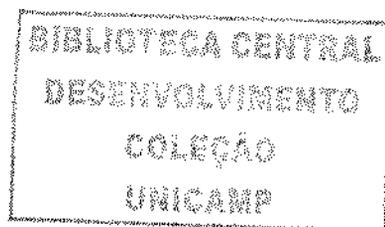
**Palavras-chave:** Reconhecimento Automático de Veículos, Visão Computacional, Processamento de Imagens, Reconhecimento de Padrões, Inteligência Artificial.

# Abstract

Vehicle license plate recognition systems (VLPRS) identify the plate number of a vehicle based on its image. Their main applications are: stolen vehicle detection; access control of restricted areas; traffic monitoring; traffic law enforcement; automatic control of parking and toll collection. In this work, we describe improvements for VLPRS. The main contributions are in the plate location method performing the binarization based on regional minima of the filtered image; in the plate framing with perspective distortion correction using the homography; in the character separation with the aid of the vertical projection; and in the character recognition, using a minimal distance hierarchical classifier. Our proposal presented a performance 1.17 times superior in the plate location and 1.58 times superior in the correct plate recognition, when compared to the best result of two commercial system demonstrations available in the internet (SeeCar and SIAV 3.0), using a database of 1200 images.

**Keywords:** Vehicle Automatic Recognition, Computer Vision, Image Processing, Pattern Recognition, Artificial Intelligence.

*Dedico esta dissertação aos meus pais, Irineu e Maria Inês, que sempre me apoiaram no que foi preciso e nunca mediram esforços para que eu pudesse atingir minhas metas, às minhas irmãs, Fabiana e Patrícia, parentes e amigos, que me incentivaram para alcançar este objetivo.*



# Agradecimentos

Ao meu orientador, Prof. Dr. Roberto de Alencar Lotufo, sou grato pela sugestão do tema, confiança, orientação e apoio no desenvolvimento deste trabalho, sem o qual isto não seria possível. Também agradeço pelos conhecimentos transmitidos, os quais foram fundamentais para minha formação.

Ao Dr. Roberto Fernandes Tavares Filho, por ter indicado esta linha de pesquisa, pelo auxílio para meu ingresso neste curso e pela recomendação do Prof. Dr. Roberto de Alencar Lotufo como meu orientador.

À Maria Aparecida Gaiotto, pela revisão ortográfica e gramatical desta dissertação.

À empresa SDC Engenharia e Sistemas, por ter fornecido as 1200 imagens utilizadas na análise dos resultados.

Aos professores Dr. Fernando José von Zuben e Dr. Clésio Luis Tozzi pelas sugestões de melhorias comentadas durante o exame de qualificação desta dissertação e pela aprendizagem adquirida em suas disciplinas.

Ao pessoal do Departamento de Engenharia de Computação e Automação Industrial (DCA) e da Faculdade de Engenharia Elétrica e de Computação (FEEC), pelo auxílio durante meu curso de mestrado.

Aos colegas André, Gustavo, Romaric e Rubens da pós-graduação, pelas críticas e sugestões.

Ao pessoal da Agência de Inovação (INOVA), pelo suporte técnico durante o depósito da patente e do registro do software.

À minha família, parentes e amigos, pelo incentivo, apoio e compreensão durante esta jornada.

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), pela bolsa de mestrado concedida durante este curso.

# Sumário

|   |             |
|---|-------------|
| <b>Lista de Figuras</b>   | <b>xiii</b> |
| <b>Lista de Tabelas</b>   | <b>xix</b>  |
| <b>1 Introdução</b>   | <b>1</b>    |
| 1.1 Sistemas de Identificação Automática de Veículo . . . . .               | 2           |
| 1.2 Sistemas de Reconhecimento da Placa de Licenciamento Veicular . . . . . | 4           |
| 1.2.1 Imagem do Veículo . . . . .   | 6           |
| 1.2.2 Pré-Processamento . . . . .   | 7           |
| 1.2.3 Localização da Placa . . . . .  | 8           |
| 1.2.4 Enquadramento da Placa . . . . .                                      | 9           |
| 1.2.5 Separação dos Caracteres . . . . .                                    | 10          |
| 1.2.6 Reconhecimento dos Caracteres . . . . .                               | 10          |
| 1.3 Organização da Dissertação . . . . .                                    | 11          |
| <b>2 Conceitos Fundamentais</b>   | <b>13</b>   |
| 2.1 Aquisição da Imagem . . . . .   | 13          |
| 2.1.1 Imagens Digitais . . . . .  | 14          |
| 2.1.2 Relacionamentos Básicos entre Pixels . . . . .                        | 15          |
| 2.2 Pré-Processamento . . . . .   | 16          |
| 2.3 Segmentação . . . . .   | 16          |
| 2.3.1 Binarização . . . . .   | 16          |
| 2.3.2 Projecção Vertical e Horizontal de uma Imagem . . . . .               | 17          |
| 2.3.3 Propriedades Geométricas dos Componentes Conexos . . . . .            | 18          |
| 2.3.4 Detecção de Contornos . . . . .                                       | 18          |
| 2.3.5 Detecção de Retas . . . . .   | 19          |

|          |  |           |
|----------|--|-----------|
| 2.3.6    | Homografia . . . . .                               | 19        |
| 2.3.7    | Morfologia Matemática . . . . .                    | 20        |
| 2.4      | Representação e Descrição . . . . .                | 22        |
| 2.5      | Reconhecimento e Interpretação . . . . .           | 22        |
| 2.5.1    | Classificador de Distância Mínima . . . . .        | 24        |
| <b>3</b> | <b>Análise Bibliográfica</b> . . . . .             | <b>27</b> |
| 3.1      | Métodos de Localização da Placa . . . . .          | 28        |
| 3.1.1    | Método da Morfologia . . . . .                     | 29        |
| 3.1.2    | Método da Variação Tonal . . . . .                 | 30        |
| 3.1.3    | Método da Correlação . . . . .                     | 31        |
| 3.1.4    | Método da Detecção das Arestas . . . . .           | 31        |
| 3.1.5    | Método Conexionista . . . . .                      | 32        |
| 3.1.6    | Método Híbrido . . . . .                           | 33        |
| 3.2      | Métodos de Enquadramento da Placa . . . . .        | 33        |
| 3.2.1    | Método da Morfologia . . . . .                     | 34        |
| 3.2.2    | Método das Projeções . . . . .                     | 35        |
| 3.2.3    | Método Iterativo . . . . .                         | 35        |
| 3.2.4    | Método da Detecção das Arestas . . . . .           | 36        |
| 3.3      | Métodos de Separação dos Caracteres . . . . .      | 36        |
| 3.3.1    | Método das Projeções . . . . .                     | 37        |
| 3.3.2    | Método da Morfologia . . . . .                     | 37        |
| 3.3.3    | Método Iterativo . . . . .                         | 38        |
| 3.4      | Métodos de Reconhecimento dos Caracteres . . . . . | 38        |
| 3.4.1    | Método Estatístico . . . . .                       | 39        |
| 3.4.2    | Método Conexionista . . . . .                      | 40        |
| 3.4.3    | Método Sintático . . . . .                         | 40        |
| 3.5      | Visão Geral das Propostas Analisadas . . . . .     | 41        |
| <b>4</b> | <b>Metodologia Adotada</b> . . . . .               | <b>43</b> |
| 4.1      | Imagem do Veículo . . . . .                        | 45        |
| 4.2      | Pré-Processamento . . . . .                        | 46        |
| 4.3      | Localização da Placa . . . . .                     | 46        |
| 4.3.1    | Binarização da Imagem . . . . .                    | 48        |

---

|          |  |            |
|----------|--|------------|
| 4.3.2    | Localização de Regiões Promissoras . . . . .                   | 56         |
| 4.3.3    | Filtragem dos Objetos . . . . .                                | 57         |
| 4.3.4    | Agrupamento dos Objetos Similares . . . . .                    | 58         |
| 4.3.5    | Filtragem dos Grupos de Objetos . . . . .                      | 60         |
| 4.3.6    | Resultado do Módulo de Localização da Placa . . . . .          | 61         |
| 4.4      | Enquadramento da Placa . . . . .                               | 61         |
| 4.4.1    | Região Promissora . . . . .                                    | 63         |
| 4.4.2    | Detecção de Contornos . . . . .                                | 64         |
| 4.4.3    | Detecção de Retas . . . . .                                    | 66         |
| 4.4.4    | Detecção de Segmentos . . . . .                                | 67         |
| 4.4.5    | Localização das Arestas da Placa . . . . .                     | 69         |
| 4.4.6    | Resultado do Módulo de Enquadramento da Placa . . . . .        | 92         |
| 4.5      | Separação dos Caracteres . . . . .                             | 92         |
| 4.5.1    | Enquadramento da Cadeia de Caracteres . . . . .                | 94         |
| 4.5.2    | Pré-Separação dos Caracteres . . . . .                         | 97         |
| 4.5.3    | Separação dos Caracteres Unidos . . . . .                      | 99         |
| 4.5.4    | Resultado do Módulo de Separação dos Caracteres . . . . .      | 104        |
| 4.6      | Reconhecimento dos Caracteres . . . . .                        | 104        |
| 4.6.1    | Extração de Características . . . . .                          | 105        |
| 4.6.2    | Classificação dos Caracteres . . . . .                         | 107        |
| 4.6.3    | Resultado do Módulo de Reconhecimento dos Caracteres . . . . . | 116        |
| <b>5</b> | <b>Análise dos Resultados</b>                                  | <b>117</b> |
| 5.1      | Análise da Localização da Placa . . . . .                      | 125        |
| 5.2      | Análise do Reconhecimento dos Caracteres . . . . .             | 128        |
| 5.3      | Análise do Custo Computacional . . . . .                       | 133        |
| 5.4      | Análise Comparativa . . . . .                                  | 133        |
| <b>6</b> | <b>Conclusões</b>  | <b>135</b> |
|          | <b>Referências bibliográficas</b>                              | <b>137</b> |

# Lista de Figuras

|      |   |    |
|------|---|----|
| 1.1  | Modelo da placa de licenciamento veicular utilizado no Brasil. . . . .            | 2  |
| 1.2  | Diagrama genérico de um <i>software</i> para SRPLV. . . . .                       | 6  |
| 1.3  | Imagem do veículo obtida por um SRPLV. . . . .                                    | 7  |
| 1.4  | Região promissora delimitada por um retângulo. . . . .                            | 8  |
| 1.5  | Enquadramento da placa representado por um polígono. . . . .                      | 9  |
| 1.6  | Imagem da placa com correção da distorção de perspectiva. . . . .                 | 10 |
| 1.7  | Separação dos caracteres representada por retângulos. . . . .                     | 10 |
| 1.8  | Exemplo de caracteres '2', '4', '6' e '9' com tipografia diferente. . . . .       | 11 |
| 1.9  | Exemplo da semelhança de tipografia dos caracteres '0', 'D', 'O', e 'Q'. . . . .  | 11 |
| 2.1  | Convenção dos eixos adotada nesta dissertação. . . . .                            | 14 |
| 2.2  | Representação do atributo volume de cada componente de nível. . . . .             | 22 |
| 2.3  | Interpretação de um dendrograma. . . . .  | 23 |
| 2.4  | Representação de um classificador de distância mínima. . . . .                    | 24 |
| 4.1  | Diagrama geral da nossa proposta. . . . .   | 44 |
| 4.2  | Imagem do veículo. . . . .  | 45 |
| 4.3  | Imagem resultante do pré-processamento da figura 4.2. . . . .                     | 46 |
| 4.4  | Diagrama em bloco da localização da placa. . . . .                                | 47 |
| 4.5  | Representação da linha 149 da figura 4.3. . . . .                                 | 48 |
| 4.6  | Histograma da figura 4.3 com limiar determinado pelo método de Otsu. . . . .      | 49 |
| 4.7  | Gráfico da binarização da linha 149 da figura 4.3 pelo método de Otsu. . . . .    | 50 |
| 4.8  | Binarização da figura 4.3 pelo método de Otsu. . . . .                            | 50 |
| 4.9  | Gráfico da binarização da linha 149 da figura 4.3 pelo método de Niblack. . . . . | 51 |
| 4.10 | Binarização da figura 4.3 pelo método de Niblack. . . . .                         | 51 |
| 4.11 | Gráfico da binarização da linha 149 da figura 4.3 pelo método do top-hat. . . . . | 52 |

|      |  |    |
|------|--|----|
| 4.12 | Binarização da figura 4.3 pelo método do top-hat. . . . .  | 52 |
| 4.13 | Mínimos regionais da figura 4.3. . . . .   | 53 |
| 4.14 | Imagem resultante da aplicação do filtro v-minima na figura 4.3. . . . .   | 54 |
| 4.15 | Imagem de profundidade da região da placa: (a) antes; (b) depois de aplicar o filtro v-minima. . . . .                   | 54 |
| 4.16 | Perfil de uma linha na região da placa: (a) sem; (b) com vales preenchidos pelo filtro v-minima. . . . .                 | 54 |
| 4.17 | Gráfico da binarização da linha 149 da figura 4.3 pelo método dos mínimos regionais da imagem filtrada. . . . .          | 55 |
| 4.18 | Binarização da figura 4.3 pelo método dos mínimos regionais da imagem filtrada. . .                                      | 56 |
| 4.19 | Rotulação da imagem binarizada (figura 4.18). . . . .  | 57 |
| 4.20 | Imagem resultante após a filtragem de objetos da figura 4.19. . . . .  | 58 |
| 4.21 | Objetos da figura 4.20 rotulados de acordo com o agrupamento a que pertencem. . .  | 59 |
| 4.22 | Grupos da figura 4.21 filtrados de acordo com a quantidade de objetos. . . . .   | 60 |
| 4.23 | Imagem com as regiões promissoras delimitadas por um retângulo. . . . .  | 61 |
| 4.24 | Diagrama em bloco do enquadramento da placa. . . . .   | 62 |
| 4.25 | Imagem de uma região promissora. . . . .   | 63 |
| 4.26 | Região promissora após aplicar os filtros morfológicos de reconstrução na figura 4.25. .                                 | 65 |
| 4.27 | Região promissora após aplicar o detector de contornos de Canny na figura 4.26. . .                                      | 65 |
| 4.28 | Região promissora após aplicar o operador morfológico de abertura por área na figura 4.27. . . . .                       | 65 |
| 4.29 | Transformada de Hough da figura 4.28: (a) antes da limiarização; (b) depois da limiarização. . . . .                     | 66 |
| 4.30 | Retas detectadas na figura 4.28. . . . .   | 67 |
| 4.31 | Contorno da região promissora, na imagem da figura 4.28, rotulado e dilatado. . . . .                                    | 67 |
| 4.32 | Segmentos rotulados de acordo com o objeto a que pertence. . . . .   | 68 |
| 4.33 | Exemplo de segmentos rotulados. . . . .  | 68 |
| 4.34 | Segmentos horizontais (a) e verticais (b) da figura 4.32, rotulados de acordo com o objeto a que pertence. . . . .       | 73 |
| 4.35 | Gráfico do valor da função-objetivo horizontal para cada quadrilátero. . . . .   | 75 |
| 4.36 | Imagem com as arestas horizontais encontradas. . . . .   | 75 |
| 4.37 | Segmentos horizontais e o par de segmentos verticais “virtuais” rotulados de acordo com o objeto a que pertence. . . . . | 77 |

|      |   |     |
|------|---|-----|
| 4.38 | Quadriláteros resultantes do conjunto Q. . . . .  | 78  |
| 4.39 | Correção da distorção de perspectiva: (a) arestas do quadrilátero 1; (b) arestas do quadrilátero 16; (c) arestas do quadrilátero 22; (d) imagem corrigida do quadrilátero 1; (e) imagem corrigida do quadrilátero 16 e (f) imagem corrigida do quadrilátero 22. . . . .   | 79  |
| 4.40 | Gráfico da projeção horizontal do gradiente da imagem corrigida. . . . .  | 80  |
| 4.41 | Gráfico do valor da função-objetivo vertical para cada quadrilátero. . . . .  | 83  |
| 4.42 | Imagem com as 4 arestas da placa encontradas. . . . .   | 83  |
| 4.43 | Segmentos horizontais e o par de segmentos verticais “virtuais” rotulados de acordo com o objeto a que pertence. . . . .  | 86  |
| 4.44 | Imagem com as arestas verticais “virtuais” esquerda e direita. . . . .  | 87  |
| 4.45 | Correção da distorção de perspectiva: (a) arestas do quadrilátero 1; (b) arestas do quadrilátero 4; (c) arestas do quadrilátero 5; (d) imagem corrigida do quadrilátero 1; (e) imagem corrigida do quadrilátero 4 e (f) imagem corrigida do quadrilátero 5. . . . .   | 89  |
| 4.46 | Enquadramento horizontal da imagem corrigida. . . . .   | 90  |
| 4.47 | Binarização da imagem enquadrada (figura 4.46). . . . .   | 90  |
| 4.48 | Abertura por área aplicada na imagem binarizada (figura 4.47). . . . .  | 90  |
| 4.49 | Imagem da placa de licenciamento veicular corrigida. . . . .  | 92  |
| 4.50 | Diagrama em bloco da separação dos caracteres. . . . .  | 93  |
| 4.51 | Enquadramento horizontal da figura 4.49. . . . .  | 94  |
| 4.52 | Binarização da imagem da cadeia de caracteres (figura 4.51). . . . .  | 95  |
| 4.53 | Abertura por área aplicada na imagem B (figura 4.52). . . . .   | 95  |
| 4.54 | Enquadramento vertical da imagem B (figura 4.53). . . . .   | 96  |
| 4.55 | Rotulação da imagem B (figura 4.54) com elemento estruturante em forma de linha horizontal com tamanho de 7 pixels. . . . .   | 96  |
| 4.56 | Imagem B resultante do enquadramento da cadeia de caracteres da placa. . . . .  | 97  |
| 4.57 | Projeção vertical da imagem B (figura 4.56). . . . .  | 98  |
| 4.58 | Projeção vertical da figura 4.56 com as coordenadas horizontais da separação dos caracteres. . . . .  | 99  |
| 4.59 | Imagem da cadeia de caracteres binarizada com caracteres unidos devido à: (a) distorção de perspectiva; (b) presença de ruído; e (c) baixa resolução espacial. Suas respectivas projeções verticais: (d), (e), e (f). Respectivas imagens binarizada com as coordenadas horizontais de separação: (g), (h) e (i). . . . . | 100 |

|      |  |     |
|------|--|-----|
| 4.60 | Imagem da região candidata com caracteres unidos das Figuras: (a) 4.59a; (b) 4.59b; e (c) 4.59c. . . . .     | 101 |
| 4.61 | Gráfico indicando os pontos de separação para a região da figura 4.60a. . . . .                              | 103 |
| 4.62 | Resultado da separação dos caracteres de uma placa que não precisou separar caracteres unidos. . . . .       | 104 |
| 4.63 | Resultado da separação dos caracteres de placas que precisaram separar caracteres unidos. . . . .            | 104 |
| 4.64 | Diagrama em bloco do reconhecimento dos caracteres. . . . .  | 105 |
| 4.65 | Imagem dos caracteres da figura 4.56 separados e enquadrados. . . . .  | 105 |
| 4.66 | Normalização do caractere 'B' da figura 4.65. . . . .  | 106 |
| 4.67 | Resultado da normalização dos caracteres da figura 4.65. . . . .   | 106 |
| 4.68 | Dendrograma das amostras rotuladas com ampliação parcial. . . . .  | 108 |
| 4.69 | Dendrograma dos protótipos resultantes da análise de agrupamento. . . . .                                    | 110 |
| 4.70 | Representação do fator de confiabilidade no reconhecimento. . . . .  | 112 |
| 4.71 | Representação dos fatores de confiabilidade no reconhecimento entre os protótipos. . . . .                   | 112 |
| 4.72 | Classificador hierárquico reconhecendo o caractere '8'. . . . .  | 115 |
| 4.73 | Classificador hierárquico reconhecendo o caractere 'D'. . . . .  | 115 |
| 5.1  | Algumas imagens do banco de dados com 1200 imagens utilizadas na análise. . . . .                            | 118 |
| 5.2  | Amostras de 128 regiões promissoras obtidas aleatoriamente das 1200 imagens (1 a 32). . . . .                | 119 |
| 5.3  | Amostras de 128 regiões promissoras obtidas aleatoriamente das 1200 imagens (33 a 64). . . . .               | 120 |
| 5.4  | Amostras de 128 regiões promissoras obtidas aleatoriamente das 1200 imagens (65 a 96). . . . .               | 121 |
| 5.5  | Amostras de 128 regiões promissoras obtidas aleatoriamente das 1200 imagens (97 a 128). . . . .              | 122 |
| 5.6  | Amostras de 128 placas enquadradas com correção da distorção de perspectiva (1 a 64). . . . .                | 123 |
| 5.7  | Amostras de 128 placas enquadradas com correção da distorção de perspectiva (65 a 128). . . . .              | 124 |
| 5.8  | Porcentagem de sucesso na localização da placa com diferentes métodos de binarização. . . . .                | 126 |
| 5.9  | Porcentagem de sucesso na localização da placa com diferentes combinações de métodos de binarização. . . . . | 127 |

---

|      |  |     |
|------|--|-----|
| 5.10 | Imagens cujas placas não foram localizadas. . . . .  | 128 |
| 5.11 | Distribuição de caracteres errados por placa. . . . .  | 129 |
| 5.12 | Porcentagem de reconhecimento por caracteres. . . . .  | 130 |
| 5.13 | Reconhecimento por caracteres. . . . .   | 131 |
| 5.14 | Teste comparativo com duas demonstrações de sistemas comerciais disponíveis na internet. . . . . | 134 |

# Lista de Tabelas

|     |  |     |
|-----|--|-----|
| 3.1 | Classificação das propostas analisadas. . . . .                                      | 42  |
| 4.1 | Classes obtidas após a análise de agrupamentos. . . . .                              | 109 |
| 4.2 | Distâncias entre os protótipos. . . . .  | 111 |
| 4.3 | Valores dos fatores de confiabilidade no reconhecimento entre os protótipos. . . . . | 113 |
| 4.4 | Resultado do módulo de reconhecimento dos caracteres. . . . .                        | 116 |
| 5.1 | Matriz de confusão. . . . .  | 132 |

# Capítulo 1

## Introdução

Um veículo precisa estar licenciado no órgão de trânsito competente para trafegar no país. No Brasil, o licenciamento atribui dois códigos diferentes ao veículo para que possa ser identificado, sendo único e exclusivo a cada veículo. O primeiro código é denominado **RENAVAN**, o qual é marcado somente na documentação do veículo. Portanto, um veículo só pode ser identificado por ele se tivermos posse de sua documentação. O segundo código é denominado **placa de licenciamento veicular**, a qual é fixada na frente e na traseira do veículo, em local visível. Deste modo, o segundo código é de domínio público, ou seja, qualquer pessoa pode identificar um veículo através dele.

Cada país possui uma legislação para padronizar a placa de licenciamento veicular. No Brasil, isto é feito conforme a **resolução nº 45 do CONTRAN**. Resumidamente a placa brasileira tem as seguintes características:

- Seu código é composto por 3 letras maiúsculas, um hífen e 4 números naturais de 0 a 9;
- Os caracteres da placa são estampados em alto relevo;
- O formato da placa é retangular com fundo uniforme;
- Possui uma tarjeta, fixada acima do código, informando a unidade federativa e a cidade na qual o veículo foi licenciado;
- As cores dos caracteres e da placa são contrastantes, porém suas cores dependem da categoria na qual o veículo está licenciado;
- São fixadas na frente e traseira do veículo por 4 parafusos, sendo que a placa traseira deve ser iluminada, para permitir visualização noturna, e lacrada, para evitar fraudes.

As características da placa são fundamentais para o desenvolvimento e/ou configuração de um sistema automático de reconhecimento da placa de licenciamento veicular. A figura 1.1 mostra o modelo da placa utilizado no Brasil.



Fig. 1.1: Modelo da placa de licenciamento veicular utilizado no Brasil.

## 1.1 Sistemas de Identificação Automática de Veículo

Embora o ser humano consiga identificar um veículo eficientemente, ele não consegue trabalhar com grande quantidade de informação em curto espaço de tempo. Outro problema é que o ser humano pode errar por distração ou cansaço.

A maioria das tarefas que necessitam da identificação do veículo consiste em verificar se o código dele pertence a uma lista.

Quando há poucos veículos em circulação, o processo de verificação de seu código em uma lista pode ser feito manualmente. Porém, isto se torna inviável com o aumento da quantidade de veículos. Considerando que um ser humano demora cerca de 1 segundo para comparar o código do veículo com cada registro de uma lista, a tarefa de verificar se o veículo identificado pertence a uma lista contendo 3000 veículos levaria até 50 minutos.

Com o advento da computação, estas listas passaram a ser armazenadas em bancos de dados. Deste modo, o usuário somente tem que digitar o código do veículo para verificar sua situação em um banco de dados. O tempo de procura é menor que 1 segundo, sendo que um usuário treinado demora cerca de 5 segundos para identificar o código do veículo e digitá-lo no sistema.

Devido ao grande fluxo de veículos, principalmente em estacionamentos e vias de trânsito, é necessário que o tempo de resposta do sistema seja menor que 1 segundo, o que não é possível por intermédio de um ser humano. Isto vem motivando o desenvolvimento de sistemas para a identificação automática de veículo. Por esta razão, surgiram várias propostas de sistemas, os quais podemos agrupar de acordo com a informação utilizada para a identificação, tais como:

- Sistemas que utilizam a imagem da placa de licenciamento veicular;
- Sistemas que utilizam um código via satélite;
- Sistemas que utilizam um código via rádio-frequência;
- Sistemas que utilizam um código de barras;
- Sistemas que utilizam um código via cartão.

Os sistemas que utilizam a imagem da placa de licenciamento veicular não têm 100% de confiabilidade, pois a imagem da placa é obtida em ambiente não controlado, sendo prejudicada por condições climáticas e de iluminação, além do mau estado de conservação de algumas placas. A vantagem é que não necessitam da instalação de equipamento no veículo. Com isso, todos os veículos licenciados estão habilitados para serem reconhecidos por estes sistemas.

Os demais sistemas, acima citados, têm 100% de confiabilidade na identificação, porém necessitam de um equipamento permanente (transmissor, código de barras) ou temporário (cartão) instalado no veículo para efetuar sua identificação. Deste modo, estão restritos a identificar apenas os veículos dotados destes equipamentos, os quais podem ser trocados ou recodificados sem o conhecimento do sistema, implicando em erro de identificação por fraude.

Devido à possibilidade de fraude, é necessário que todos os sistemas armazenem uma prova física, sendo esta a imagem do veículo. Os sistemas que utilizam a imagem da placa de licenciamento veicular já possuem a prova a ser armazenada. Portanto, não necessitam de equipamento adicional. Os demais sistemas necessitam da instalação de outro equipamento para obter a prova física da identificação. Com isso, têm maiores custos, inclusive com o equipamento a ser instalado nos veículos.

Cada sistema tem suas características, tais como:

- Identificação somente de veículos parados;
- Identificação de veículos parados ou em movimento;
- Porcentagem de confiabilidade;
- Necessidade de instalar equipamento no veículo;
- Necessidade de intervenção humana;
- Custos fixos e operacionais;
- Perímetro de identificação;
- Etc.

Portanto, a escolha do sistema depende do ambiente e de sua aplicação. As principais aplicações destes sistemas são:

- **Identificação de Veículo Roubado:** quando é comunicado ao órgão de trânsito competente o roubo ou furto de um veículo, a cadeia de caracteres da placa de licenciamento do mesmo é adicionada a um banco de dados. Em caso de suspeita de um veículo e em fiscalizações de trânsito é verificada no banco de dados a procedência do mesmo.
- **Controle de Acesso:** alguns estacionamentos precisam identificar cada veículo na entrada e saída para computar o tempo da estadia do mesmo. As praças de pedágio podem utilizar a

identificação do veículo para posterior cobrança da tarifa; esta informação também é utilizada para autorizar ou registrar a entrada de veículos em locais de acesso restrito;

- **Monitoramento do Tráfego:** a engenharia de tráfego precisa conhecer a velocidade média dos veículos em um determinado percurso, sendo que para calcular o tempo e velocidade do veículo no percurso é preciso identificá-lo no início e término do percurso, assim como seus respectivos horários; a velocidade média também pode ser utilizada para fiscalizar infrações de trânsito por excesso de velocidade. As administradoras de rodovias podem utilizar a identificação do veículo para acompanhar todo seu trajeto, com isso, conseguem detectar quando um veículo pára em um trecho da rodovia, acionando automaticamente o serviço de socorro, não necessitando que o motorista caminhe pelo acostamento até um telefone de emergência, tarefa esta perigosa que coloca em risco a vida do usuário;
- **Fiscalização:** quando um motorista comete uma infração de trânsito, a placa de licenciamento veicular do infrator é registrada para posterior indenização à parte prejudicada pelo responsável legal do veículo.

A primeira aplicabilidade vista pela sociedade é na fiscalização de trânsito, principalmente no processamento das multas obtidas por radares eletrônicos. Porém, a principal contribuição para a sociedade é na identificação de veículos roubados. Isto pode ser feito instalando o sistema em pontos estratégicos de vigilância e interligado ao banco de dados do DETRAN, no qual fica armazenada a situação do veículo (tal como: queixa de furto, roubo, não licenciado, etc). Também é possível equipar um veículo com o sistema e interligá-lo ao banco de dados de modo a identificar veículos roubados que passem pelo mesmo. Deste modo, é possível intensificar a recuperação dos mesmos.

## 1.2 Sistemas de Reconhecimento da Placa de Licenciamento Veicular

Os sistemas que utilizam a imagem da placa de licenciamento veicular para identificar um veículo têm maior facilidade na adaptação aos sistemas atuais, nos quais a identificação do veículo é feita pelo ser humano, do que os demais sistemas. Embora tenham confiabilidade menor que 100%, algumas vezes necessitando da intervenção humana, somente é preciso instalar equipamentos nas estações de monitoramento, sendo estes também utilizados para obter uma prova física (imagem do veículo) a ser armazenada.

Os sistemas de identificação automática de veículo que utilizam a imagem da placa de licenci-

amento são denominados “Sistemas de Reconhecimento da Placa de Licenciamento Veicular”. Doravante vamos denominar estes sistemas pela abreviação SRPLV. Eles são compostos por:

- Sensor de presença (capacitivo, indutivo, etc);
- Câmera de vídeo (analógica ou digital);
- Cabo para transmissão de sinal de vídeo da câmera ao computador;
- Computador (com porta USB ou IEEE 1394 para câmeras digitais; ou com placa digitalizadora de vídeo para câmeras analógicas);
- Software para reconhecimento da placa de licenciamento veicular.

O principal componente destes sistemas é o *software*, pois é ele que identifica o veículo através da imagem obtida pela câmera de vídeo.

Este trabalho visa pesquisar e desenvolver um *software* para SRPLV, sendo este um programa de computador que consiste em reconhecer a cadeia de caracteres da placa de licenciamento veicular de uma imagem digital da frente ou traseira do veículo, retornando o código ASCII (*American Standard Code for Information Interchange*) de cada caractere da placa.

Quando um veículo passa pelo sensor de presença, a câmera de vídeo captura a imagem e envia ao computador. Em seguida, o *software* processa a imagem do veículo em busca da placa de licenciamento veicular, enquadra a placa corrigindo eventuais distorções da imagem, separa os caracteres da placa e reconhece cada um deles retornando seu código ASCII, o qual é adotado por ser um padrão digital para a representação de caracteres, permitindo a integração com qualquer outro sistema.

Os SRPLV foram inicialmente propostos na década de 1990. Porém, até este momento nenhum deles conseguiu igualar à confiabilidade de identificação de veículo feita por um ser humano, o que motiva a elaboração de novas propostas. Neste trabalho, estamos colaborando com uma nova metodologia na intenção de atingir tal objetivo.

Visando facilitar o entendimento, dividimos o *software* de acordo com seus módulos fundamentais: *Pré-Processamento*; *Localização da Placa*; *Enquadramento da Placa*; *Separação dos Caracteres*; e *Reconhecimento dos Caracteres*. A figura 1.2 apresenta o diagrama genérico deste *software*.

Cada módulo pode ser desenvolvido por diversos métodos. Uma classificação destes métodos, acompanhada de uma breve descrição, será apresentada no capítulo 3.

O processamento dos módulos fundamentais é seqüencial, ou seja, cada módulo depende do resultado obtido por seu antecessor. Todos os módulos são importantes, pois a falha de um deles impossibilita o funcionamento do sistema.

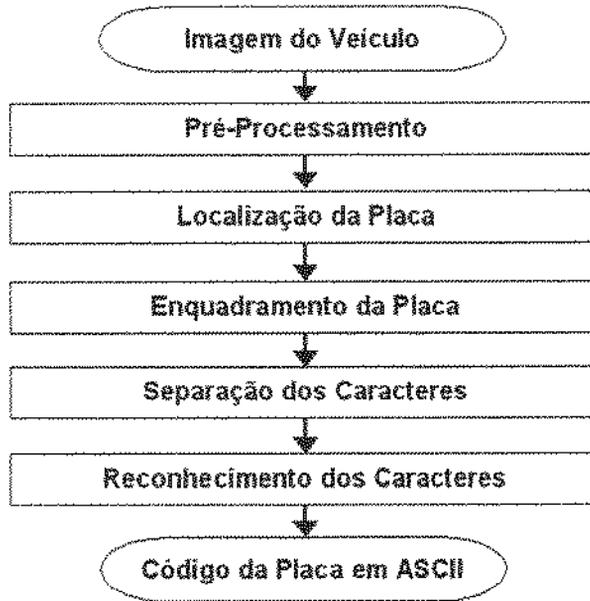


Fig. 1.2: Diagrama genérico de um software para SRPLV.

As subseções seguintes explicam sucintamente cada módulo com uma breve análise das dificuldades apresentadas.

### 1.2.1 Imagem do Veículo

Por ser obtida em ambiente não controlado, a imagem do veículo (figura 1.3) não contém apenas a placa de licenciamento, mas também o veículo e demais objetos que estejam no campo de visão da câmera. Portanto, a tarefa de segmentar a imagem não é trivial. Outro problema é a fonte de iluminação; onde a quantidade, posição e intensidade variam no tempo. De fato, um veículo pode estar com o farol alto, baixo ou lanterna; acesos ou apagados. A maioria das instalações de SRPLV estão em ambiente aberto, ou seja, utilizam iluminação natural (luz solar), a qual varia sua posição (nascente no leste e poente no oeste) e intensidade (nuvens bloqueiam os raios luminosos) no tempo. Considerando que estes sistemas tenham que operar em período noturno, também deve ser utilizada uma iluminação artificial (lâmpadas). Também está exposto a diversas condições climáticas, tais como: chuva, neblina, etc. Deste modo, para desenvolver um *software* para SRPLV é importante fazer uma diversificada coleta de dados, visando garantir sua funcionalidade nas diversas condições a que será exposto.

A qualidade da imagem de entrada é de fundamental importância para todos os módulos, pois é dela que são extraídas todas as informações. A imagem de entrada é digital e pode ser obtida de duas



Fig. 1.3: Imagem do veículo obtida por um SRPLV.

maneiras:

1. Câmera analógica, os dados são digitalizados por uma placa de captura (*frame grabber*);
2. Câmera digital, os dados digitais são transmitidos via USB ou IEEE 1394 (*Firewire*).

Por motivos econômicos, a maioria dos SRPLV utilizam câmeras analógicas e placas de captura de vídeo de baixo custo. A transmissão analógica da câmera à placa de captura recebe ruídos por ondas eletromagnéticas, causando a degradação da imagem. Devido aos baixos valores de *amostragem* e *quantização* proporcionados pelas placas de captura no processo de digitalização, algumas informações da imagem são perdidas. Com isso, a maioria das imagens de entrada têm baixa qualidade comparada com imagens obtidas em câmeras digitais. Outro problema é que as placas de licenciamento veicular estão expostas ao ambiente, com isso ocorre degradação das mesmas com o tempo, o que também prejudica a funcionalidade deste sistema. Também devemos levar em consideração que as placas de licenciamento veicular não possuem superfície lambertiana, com isso podem refletir luz, ocasionando uma região branca na imagem.

### 1.2.2 Pré-Processamento

O módulo de *Pré-Processamento* não é obrigatório, visto que em alguns SRPLV a imagem de entrada já está no formato necessário para ser utilizada pelo módulo de *Localização da Placa*.

As principais tarefas que são efetuadas neste módulo são:

- Descompactar a imagem;
- Descriptografar a imagem;
- Desentrelaçar a imagem;
- Converter uma imagem colorida para níveis de cinza;
- Realçar o contraste da imagem;
- Remover ruídos da imagem;

### 1.2.3 Localização da Placa

O módulo de *Localização da Placa* tem a tarefa de analisar a imagem de entrada para encontrar as regiões que envolvam um conjunto de objetos com características semelhantes às de uma placa de licenciamento veicular, as quais denominamos de **regiões promissoras**. As principais características destas placas são:

- Cor dos caracteres diferente da cor do fundo da placa;
- Placa em formato retangular com fundo uniforme;
- Os caracteres da placa possuem altura e espaçamento regular, sendo alinhados horizontalmente;
- Quantidade determinada de caracteres.



Fig. 1.4: Região promissora delimitada por um retângulo.

A imagem de entrada pode conter várias regiões promissoras, tais como: prefixo do veículo, adesivos, modelo do veículo, marca do fabricante, etc. Por esta razão, muitos autores optam por

identificar todas as regiões promissoras, aplicando os demais módulos a cada região encontrada. Com isso, tentam minimizar o problema de não encontrar a região correta. A figura 1.4 ilustra uma região promissora delimitada por um retângulo.

#### 1.2.4 Enquadramento da Placa

A imagem da região promissora, obtida no módulo de *Localização da Placa*, não contém apenas a placa de licenciamento veicular, mas também partes do veículo que estão ao seu redor.

Outro problema é que, na maioria das vezes, a câmera não fica posicionada de modo ortonormal à placa do veículo, pois impediria a passagem do mesmo, ocasionando um efeito denominado **distorção de perspectiva**. Geralmente, a câmera fica posicionada ao lado do veículo, sendo que a distância da placa ao solo é diferente da distância da câmera ao solo. Portanto, é necessário rotacioná-la em torno de dois eixos: panorâmico e inclinação, conhecidos tecnicamente por *pan* e *tilt*. Deste modo, temos 2 graus de liberdade, podendo ocasionar: rotação da placa, inclinação dos caracteres da placa e o efeito trapézio. Esses efeitos prejudicam as etapas de separação e reconhecimento dos caracteres. Portanto, este módulo pode ter duas tarefas:

1. Enquadrar corretamente a placa de licenciamento veicular, de modo a eliminar os objetos que não pertencem a ela;
2. Corrigir a distorção de perspectiva.

A figura 1.5 ilustra o enquadramento da placa por um polígono. A correção da distorção de perspectiva é ilustrada na figura 1.6.



Fig. 1.5: Enquadramento da placa representado por um polígono.

A correção da distorção de perspectiva e o correto enquadramento da placa implicam diretamente em melhorias significativas na eficiência dos demais módulos.



Fig. 1.6: Imagem da placa com correção da distorção de perspectiva.

### 1.2.5 Separação dos Caracteres

A tarefa do módulo de *Separação dos Caracteres* é enquadrar cada caractere da placa para que eles sejam processados individualmente pelo módulo de *Reconhecimento dos Caracteres*.

A figura 1.7 ilustra o enquadramento de cada caractere por um retângulo.



Fig. 1.7: Separação dos caracteres representada por retângulos.

Embora os caracteres sejam marcados na placa de licenciamento veicular em fontes tipográficas, dois ou mais caracteres podem estar unidos. Isto ocorre devido à distorção de perspectiva, presença de ruídos ou baixa resolução espacial da imagem, o que implica em maior dificuldade desta tarefa.

### 1.2.6 Reconhecimento dos Caracteres

O módulo de *Reconhecimento dos Caracteres* retorna o código ASCII correspondente à imagem de cada caractere.

Temos o conhecimento a priori que a cadeia de caracteres da placa de licenciamento veicular é composta por letras maiúsculas do alfabeto latino e números naturais de 0 a 9. Deste modo, determinamos 36 símbolos (caracteres), sendo 26 letras e 10 números.

No Brasil, existem diferentes fabricantes de placas e não há uma resolução do CONTRAN que normalize a tipografia. Deste modo, existem placas com diferentes tipografias, o que pode ocasionar em imagens com formas geométricas diferentes que representam o mesmo caractere (figura 1.8). Também existem caracteres diferentes com tipografia parecida, é o caso dos caracteres: 'O', 'D', 'Q' e '0' (figura 1.9). Estes dois casos dificultam o correto reconhecimento dos caracteres.



Fig. 1.8: Exemplo de caracteres '2', '4', '6' e '9' com tipografia diferente.



Fig. 1.9: Exemplo da semelhança de tipografia dos caracteres '0', 'D', 'O', e 'Q'.

Mais importante que a escolha do método a ser utilizado para reconhecer os caracteres é a escolha do conjunto de características utilizadas para descrevê-los. Também é interessante considerar que algumas delas são mais descritivas. Deste modo, elas podem ser ponderadas para dar maior credibilidade às características de maior importância.

### 1.3 Organização da Dissertação

A presente dissertação é composta por 6 capítulos, além da referência bibliográfica.

Neste capítulo, foi apresentada a proposta de trabalho e a motivação para seu desenvolvimento. Também foi explicado sucintamente o funcionamento de um SRPLV, assim como as dificuldades no seu desenvolvimento. A proposta de trabalho será explicada detalhadamente nos capítulos seguintes.

No capítulo 2, são apresentados conceitos fundamentais de processamento de imagens e reconhecimento de padrões, sendo conhecimentos fundamentais para o entendimento dos demais capítulos.

O capítulo 3 apresenta uma análise bibliográfica dos *softwares* para SRPLV, no qual contribuimos com uma classificação das propostas com base nos módulos fundamentais: *Localização da Placa*, *Enquadramento da Placa*, *Separação dos Caracteres* e *Reconhecimento dos Caracteres*. Esta classificação é de grande ajuda para o desenvolvimento de novas propostas.

No capítulo 4, é apresentada a metodologia adotada para o desenvolvimento do *software* de SRPLV, no qual descrevemos detalhadamente o método utilizado em cada módulo.

No capítulo 5, os resultados obtidos pela metodologia adotada são analisados e comparados com algumas propostas atuais.

Finalmente, o capítulo 6 apresenta as conclusões e perspectivas de melhorias deste trabalho.

O artigo (Dias and Lotufo, 2005), apresenta um resumo deste trabalho, o qual foi apresentado no IV Workshop de Teses e Dissertações em Computação Gráfica e Processamento de Imagens do SIBGRAPI 2005, sendo premiado na categoria de mestrado.

## Capítulo 2

# Conceitos Fundamentais

Neste capítulo, apresentamos alguns conceitos fundamentais necessários para o bom entendimento desta dissertação.

Os fundamentos necessários para processar uma imagem digital são (Gonzalez and Woods, 2000):

1. Aquisição da Imagem;
2. Pré-Processamento;
3. Segmentação;
4. Representação e Descrição;
5. Reconhecimento e Interpretação.

Este capítulo foi baseado nos seguintes livros: Gonzalez and Woods (2000); Trucco and Verri (1998); Dougherty and Lotufo (2003); Kohn (1998); Duda and Hart (1973)

As seções seguintes descrevem os fundamentos de cada uma das etapas acima citadas.

### 2.1 Aquisição da Imagem

Para obter uma imagem, é necessário um sensor de imageamento (câmera fotográfica, câmera de vídeo, scanner, etc). Caso o sinal de saída deste sensor seja analógico, um equipamento de digitalização (*frame grabber*, etc) deve ser utilizado para converter o sinal em digital, pois trabalhamos com o processamento de *imagens digitais*.

### 2.1.1 Imagens Digitais

Uma imagem digital pode ser colorida ou monocromática, visando facilitar o entendimento, vamos explicar como é representada uma *imagem monocromática*, sendo a imagem colorida composta pela sobreposição de 3 camadas monocromáticas, denominadas de *RGB* (*Red*, *Green* e *Blue*), ou seja, uma camada para armazenar a intensidade de cada uma das cores: vermelha, verde e azul.

**Imagem monocromática:** é uma função bidimensional de intensidade de luz  $f(x, y)$ , onde  $x$  e  $y$  denotam as coordenadas espaciais e o valor de  $f$  em qualquer ponto  $(x, y)$  é proporcional ao brilho (ou *nível de cinza*) da imagem naquele ponto. A figura 2.1 ilustra a convenção dos eixos utilizada ao longo desta dissertação.

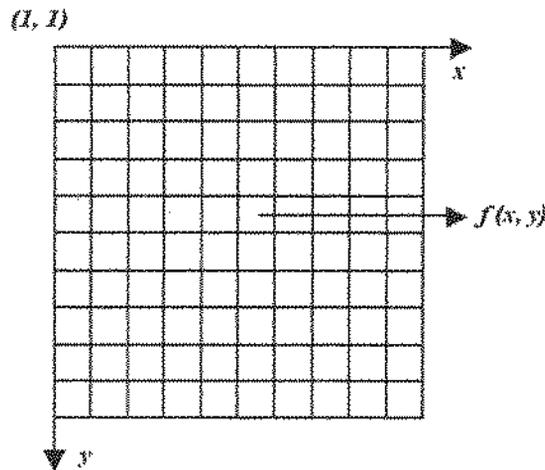


Fig. 2.1: Convenção dos eixos adotada nesta dissertação.

**Imagem digital:** é uma imagem  $f(x, y)$  discretizada tanto em coordenadas espaciais (*resolução espacial*) quanto em brilho (*resolução nos níveis de cinza*). Ela pode ser considerada como uma matriz cujos índices de linhas e colunas identificam um ponto (*pixel*), sendo o valor da intensidade luminosa deste ponto armazenada na respectiva coordenada da matriz.

**Pixel:** abreviação de *picture element* (elemento de figura), é a denominação dada a cada ponto da imagem.

**Resolução espacial:** também conhecida como **amostragem**, indica a quantidade de linhas e colunas igualmente espaçadas e arranjadas na forma de uma matriz  $N \times M$  na qual a imagem é discretizada durante sua digitalização.

**Resolução nos níveis de cinza:** também conhecida como **quantização**, indica a quantidade discreta da luminância (brilho) que pode ser armazenada em cada *pixel* da imagem durante sua digitalização.

### 2.1.2 Relacionamentos Básicos entre Pixels

Muitas tarefas de processamento de imagens digitais necessitam analisar a *vizinhança* de um *pixel* para tomar uma decisão sobre seu valor. As vizinhanças mais utilizadas são *N4* e *N8*.

**Vizinhança-N4:** um *pixel*  $p$  nas coordenadas  $(x, y)$  possui 4 vizinhos com uma unidade de distância, sendo 2 horizontais e 2 verticais, cujas coordenadas são dadas pela equação 2.1. Estes *pixels* são denominados **vizinhos-4**.

$$(x + 1, y), (x - 1, y), (x, y + 1), (x, y - 1) \quad (2.1)$$

**Vizinhança-N8:** um *pixel*  $p$  nas coordenadas  $(x, y)$  possui 8 vizinhos com uma unidade de distância, sendo 2 horizontais, 2 verticais e 4 diagonais, cujas coordenadas são dadas pela equação 2.2. Estes *pixels* são denominados **vizinhos-8**.

$$(x+1, y), (x-1, y), (x, y+1), (x, y-1), (x+1, y+1), (x+1, y-1), (x-1, y+1), (x-1, y-1) \quad (2.2)$$

**Caminho:** um caminho de um *pixel* com coordenadas  $(x, y)$  a um *pixel* com coordenadas  $(s, t)$  é uma seqüência de *pixels* distintos com coordenadas  $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$ , em que  $(x_0, y_0) = (x, y)$  e  $(x_n, y_n) = (s, t)$ ,  $(x_i, y_i)$  é vizinho a  $(x_{i-1}, y_{i-1})$ ,  $1 \leq i \leq n$ , onde  $n$  é o comprimento do caminho. Dizemos então que este é um **caminho-4** (ou **caminho-8**) dependendo se a vizinhança for 4 ou 8 ao longo do caminho.

**Componente conexo:** dois *pixels*  $p$  e  $q$  de um subconjunto de *pixels*  $S$  da imagem são ditos conexos em  $S$  se existir um **caminho** de  $p$  a  $q$  inteiramente contido em  $S$ . Para qualquer *pixel*  $p$  em  $S$ , o conjunto de *pixels* em  $S$  que são conexos a  $p$  é denominado **componente conexo** (também conhecido como **blob**).

**Componentes de nível:** é o conjunto dos **componentes conexos** com mesmo valor de nível de cinza.

**Rotulação de componentes conexos:** atribui um valor diferente, denominado rótulo, para cada componente conexo da imagem.

**Zona plana:** é o máximo **componente conexo** com mesmo valor de nível de cinza.

## 2.2 Pré-Processamento

Uma das tarefas do pré-processamento é a conversão de imagem colorida em níveis de cinza. O modelo mais utilizado é o sistema YIQ, sendo este utilizado na transmissão comercial de TV colorida.

**Modelo de cores YIQ:** este modelo foi projetado para tirar vantagem do sistema de maior sensibilidade da visão a mudanças de luminância do que nas mudanças de matiz ou saturação. A principal vantagem em processamento de imagens é que a luminância fica armazenada na matriz  $Y$ , a qual é dada pela equação 2.3:

$$Y(x, y) = 0,299R(x, y) + 0,587G(x, y) + 0,114B(x, y) \quad (2.3)$$

onde:

$Y$ : é a matriz da imagem no formato em 256 níveis de cinza;

$R$ ,  $G$  e  $B$ : são as matrizes da imagem colorida (Vermelha, Verde e Azul);

$x$  e  $y$ : são respectivamente a coordenada da coluna e linha na matriz da imagem.

## 2.3 Segmentação

A segmentação subdivide uma imagem em suas partes ou objetos constituintes (Gonzalez and Woods, 2000). Sendo a tarefa de maior importância em um processo de análise de imagem.

### 2.3.1 Binarização

A binarização é uma segmentação que separa os objetos da imagem em dois grupos, denominados tecnicamente de *fundo* e *objeto*.

**Limiarização:** é uma binarização que separa os objetos do fundo por um limiar de luminância (*threshold*)  $t$ , conforme a equação 2.4:

$$g(x, y) = \begin{cases} 0 & \text{se } f(x, y) > t \\ 1 & \text{se } f(x, y) \leq t \end{cases} \quad (2.4)$$

onde:

$g$ : é a imagem binarizada;

$f$ : é a imagem em níveis de cinza;

$t$ : é o limiar de separação;

$x$  e  $y$ : são respectivamente a coordenada da coluna e linha na imagem.

**Binarização global manual:** esta técnica define o valor do limiar de separação empiricamente, com base na amostragem das imagens a serem utilizadas.

**Binarização global automática:** esta técnica estima o limiar (threshold) com base na função densidade de probabilidade da luminância, sendo que a função de densidade global é a soma ou mistura de duas densidades unimodais, uma para a região clara e outra para a região escura. O método mais utilizado é o de Otsu (1979).

**Binarização local adaptativa:** esta técnica binariza cada *pixel* com base em sua vizinhança. O método mais utilizado é o de Niblack (1986), o qual utiliza a média e o desvio padrão da vizinhança do *pixel* para determinar seu limiar.

**Binarização iterativa:** consiste em mudar o valor do limiar iterativamente até binarizar corretamente os componentes conexos dos caracteres da placa.

### 2.3.2 Projeção Vertical e Horizontal de uma Imagem

O principal uso da projeção horizontal em SRPLV é no enquadramento da cadeia de caracteres da placa. A projeção vertical é normalmente utilizada na separação dos caracteres. Ambas projeções também podem ser utilizadas na representação de uma imagem.

**Projeção horizontal:** consiste da soma dos valores de cada linha da imagem, ou seja, soma da imagem na direção horizontal (eixo  $x$ ), conforme a equação 2.5:

$$P_y = \sum_{x=1}^{x \leq m} f(x, y), \text{ para } y \in [1, n] \quad (2.5)$$

onde:

$n$ : é a altura da imagem  $f$ ;

$m$ : é a largura da imagem  $f$ ;

$x$ : é a abscissa da imagem  $f$ ;

$y$ : é a ordenada da imagem  $f$ .

**Projeção vertical:** consiste da soma dos valores de cada coluna da imagem, ou seja, soma da

imagem na direção vertical (eixo  $y$ ), conforme a equação 2.6:

$$P_x = \sum_{y=1}^{y \leq n} f(x, y), \text{ para } x \in [1, m] \quad (2.6)$$

onde:

- $n$ : é a altura da imagem  $f$ ;
- $m$ : é a largura da imagem  $f$ ;
- $x$ : é a abscissa da imagem  $f$ ;
- $y$ : é a ordenada da imagem  $f$ .

### 2.3.3 Propriedades Geométricas dos Componentes Conexos

As propriedades geométricas dos *componentes conexos* podem ser utilizadas na segmentação e representação de objetos.

**Retângulo envolvente:** é definido pela coordenada superior esquerda e inferior direita do menor retângulo que contém o componente conexo.

**Altura:** é a distância na ordenada (vertical) da coordenada superior esquerda à inferior direita.

**Largura:** é a distância na abscissa (horizontal) da coordenada superior esquerda à inferior direita.

**Área:** é a quantidade de *pixels* de um componente conexo.

**Volume:** é o somatório dos valores de todos os *pixels* de um componente conexo.

### 2.3.4 Detecção de Contornos

Um detector de contornos possui três passos fundamentais: *suavização de ruídos*, *realce dos contornos* e *localização dos contornos*. Existem vários métodos de detecção de contornos: *Sobel*, *Roberts*, *Prewitt*, *Laplaciano*, *Cruzamento por Zero*, *Canny*, entre outros. Cada qual com características específicas.

**Detector de contorno de Canny:** este método encontra os contornos nos máximos locais do gradiente da imagem, sendo o gradiente calculado pela derivada do filtro *Gaussiano*, o qual tem seu desvio padrão caracterizado pelo parâmetro  $\sigma$ . Utiliza dois limiares (*thresholds*), um para detectar os contornos “fortes” e outro para os “fracos”, sendo que os contornos “fracos” somente serão utilizados se forem conexos aos “fortes”. Com isso, também é possível detectar partes dos contornos com baixo gradiente.

### 2.3.5 Detecção de Retas

A detecção de retas consiste em encontrar, numa imagem binária, subconjuntos de pontos que formem uma reta. O método mais utilizado na detecção de retas é a *Transformada de Hough*, também conhecida como *HT*.

**Transformada de Hough:** a idéia chave deste método é o mapeamento do *problema de localizar um conjunto de pontos colineares* para o simples *problema de encontrar picos no espaço de busca*. Sua transformação converte uma imagem binária contendo apenas os contornos da imagem para um espaço de parâmetros formado pelos ângulos ( $\theta$ ) das retas na abscissa e pelas posições ( $\rho$ ) das retas na ordenada. Para manter o espaço de parâmetros finito é adotado o sistema de coordenadas polares, deste modo, a equação da reta é representada por:  $\rho = x \cos \theta + y \sin \theta$ .

### 2.3.6 Homografia

A **homografia** é uma transformação projetiva linear simples entre planos, mapeando pontos de um plano para outro, ou seja, de 2D para 2D. Para aplicar esta transformação geométrica, é necessário determinar a *matriz de homografia*, a qual possui 9 coeficientes, correspondendo a 3 graus de liberdade: *translação*, *panorâmico (pan)* e *inclinação (tilt)*. A operação matricial da transformação geométrica é feita conforme equação 2.7:

$$\begin{aligned} x' &= \frac{u}{w} \\ y' &= \frac{v}{w} \end{aligned}$$

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (2.7)$$

onde:

$x$  e  $y$ : são as coordenadas da coluna e linha da imagem original;

$x'$  e  $y'$ : são as coordenadas da imagem transformada;

$a$ : são os coeficientes da matriz de homografia a serem determinados.

A metodologia adotada para determinar os coeficientes da *matriz de homografia* é a mesma utilizada para determinar a *matriz de projeção*, porém a *matriz de homografia* contém apenas 9 coeficientes (transformação 2D para 2D), ao contrário da *matriz de projeção* que contém 12 coeficientes

(transformação 3D para 2D). Como um coeficiente é determinado, pois  $a_{33} = 1$ , é preciso ter 8 equações para encontrar os demais coeficientes. Com 4 pontos nas coordenadas cartesianas, conhecidos na imagem original  $(x, y)$  e na imagem corrigida  $(u, v)$ , são obtidas as 8 equações necessárias, conforme sistema de equações homogêneo 2.8:

$$\underbrace{\begin{bmatrix} U_1 & V_1 & 1 & 0 & 0 & 0 & -X_1U_1 & -X_1V_1 \\ 0 & 0 & 0 & U_1 & V_1 & 1 & -Y_1U_1 & -Y_1V_1 \\ U_2 & V_2 & 2 & 0 & 0 & 0 & -X_2U_2 & -X_2V_2 \\ 0 & 0 & 0 & U_2 & V_2 & 1 & -Y_2U_2 & -Y_2V_2 \\ U_3 & V_3 & 1 & 0 & 0 & 0 & -X_3U_3 & -X_3V_3 \\ 0 & 0 & 0 & U_3 & V_3 & 1 & -Y_3U_3 & -Y_3V_3 \\ U_4 & V_4 & 1 & 0 & 0 & 0 & -X_4U_4 & -X_4V_4 \\ 0 & 0 & 0 & U_4 & V_4 & 1 & -Y_4U_4 & -Y_4V_4 \end{bmatrix}}_C \underbrace{\begin{bmatrix} a_{11} \\ a_{12} \\ a_{13} \\ a_{21} \\ a_{22} \\ a_{23} \\ a_{31} \\ a_{32} \end{bmatrix}}_A = \underbrace{\begin{bmatrix} X_1 \\ Y_1 \\ X_2 \\ Y_2 \\ X_3 \\ Y_3 \\ X_4 \\ Y_4 \end{bmatrix}}_R \quad (2.8)$$

onde:

$X_i$ : é a coordenada da coluna do  $i$ -ésimo ponto da imagem original;

$Y_i$ : é a coordenada da linha do  $i$ -ésimo ponto da imagem original;

$U_i$ : é a coordenada da coluna do  $i$ -ésimo ponto da imagem corrigida;

$V_i$ : é a coordenada da linha do  $i$ -ésimo ponto da imagem corrigida;

$a$ : são os coeficientes da matriz de homografia a serem determinados.

Por se tratar de um sistema determinado, pode ser resolvido a partir da técnica da matriz inversa:  
 $A = RC^{-1}$ .

### 2.3.7 Morfologia Matemática

A morfologia matemática é uma ramificação do processamento e análise não-linear de imagens, inicialmente desenvolvida por George Matheron e Jean Serra, com base na estrutura geométrica (morfologia) de imagens (Dougherty and Lotufo, 2003).

**Elemento estruturante:** é uma imagem de amostra com características geométricas e/ou topológicas relacionadas com a informação que pretendemos extrair da imagem de interesse.

**Erosão:** é a primeira operação básica da morfologia matemática binária. Consiste em marcar o centro do *elemento estruturante* em todas as regiões nas quais este elemento se encaixe por completo dentro dos objetos da imagem.

**Dilatação:** é a segunda operação básica da morfologia matemática binária. Consiste em marcar a forma do *elemento estruturante* em todas as regiões nas quais o centro deste elemento esteja posicionado sobre um *pixel* pertencente aos objetos da imagem.

**Abertura:** consiste de uma *erosão* seguida por uma *dilatação* com o mesmo elemento estruturante. Esta operação pode apagar pequenos objetos claros em um fundo escuro. Um objeto será apagado se o elemento estruturante não encaixar dentro dele.

**Fechamento:** consiste de uma *dilatação* seguida por uma *erosão* com o mesmo elemento estruturante. Esta operação apaga pequenos objetos escuros em fundo claro. Um objeto será apagado se o elemento estruturante não encaixar dentro dele.

**Abertura por área:** remove todos os componentes conexos com área menor que  $\alpha$ .

**Mínimo regional:** é uma *zona plana* não adjacente a uma zona plana de nível de cinza mais baixo.

**Máximo regional:** é uma *zona plana* não adjacente a uma zona plana de nível de cinza mais alto.

**Dinâmica dos mínimos regionais:** é a menor altura que temos que subir de um *mínimo regional* para encontrar outro mínimo de menor altitude. Esta ferramenta permite selecionar os mínimos regionais de acordo com um critério de contraste.

**Filtro conexo:** é um operador que somente une *zonas planas*, sendo a imagem filtrada composta por zonas planas da imagem de entrada. Uma vantagem deste filtro é que se um contorno existe na imagem de saída, ele também existe na imagem de entrada.

**V-minima:** é um *filtro conexo* que utiliza critério de volume, o qual combina área e contraste para preencher os vales da imagem com valor volumétrico especificado. Para facilitar o entendimento explicaremos o funcionamento de seu complemento, ou seja, do filtro *v-maxima*. Para compreender este filtro é importante visualizar a imagem de forma topográfica, similar a maquete de relevo feita com pedaços de cartolina. Primeiro a cartolina é cortada no formato de cada linha de nível (linha de mesma altura), então todas as peças são empilhadas para formar a topografia. Cada peça de cartolina é um componente conexo do conjunto de um nível da imagem. Iremos nos referir a estes como *componentes de nível*. O volume de um componente de nível é a soma de todas as áreas dos componentes de nível acima dele, incluindo ele, representado na figura 2.2. O filtro *v-maxima* remove todos os componentes de nível com volume menor que  $v$ . Enquanto que o filtro *v-minima* preenche todos os componentes de nível abaixo dele.

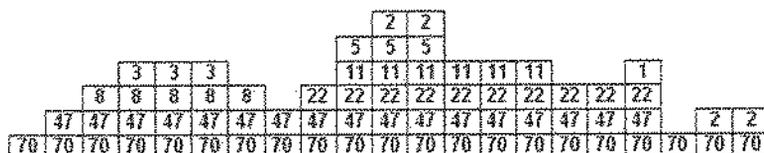


Fig. 2.2: Representação do atributo volume de cada componente de nível.

**Abertura reconstrutiva:** esta operação utiliza dois elementos estruturantes: um para especificar a forma do critério de encaixe e outro para especificar a conectividade. Consiste em passar para a imagem resultante todos os componentes conexos que foram marcados pelo operador de *abertura*.

**Fechamento reconstrutivo:** esta operação utiliza dois elementos estruturantes: um para especificar a forma do critério de encaixe e outro para especificar a conectividade. Consiste em passar para a imagem resultante todos os componentes conexos que foram marcados pelo operador de *fechamento*.

**Filtro alternado seqüencial reconstrutivo (ASF):** é um *filtro conexo* definido pela alternância seqüencial dos operadores de *abertura* e *fechamento reconstrutivos*.

**Top-hat:** é obtido subtraindo da imagem original o resultado de seu *fechamento* morfológico por um elemento estruturante apropriado. Esta operação é utilizada para detectar os objetos eliminados pelo fechamento.

## 2.4 Representação e Descrição

Para reconhecer um objeto visualmente, é necessário representar as características extraídas de sua imagem de forma apropriada.

**Normalização espacial e de níveis de cinza do caractere:** é uma das formas de representar um objeto por meio de informações visuais. Consiste em padronizar a imagem obtida de um objeto numa determinada resolução *espacial* e *nos níveis de cinza*, o que permite comparar esta imagem com algumas amostras na mesma padronização.

## 2.5 Reconhecimento e Interpretação

A tarefa de reconhecimento de imagens, também conhecido como *análise automática de imagens*, é associar um rótulo ao objeto com base em sua representação e no conhecimento adquirido por um conjunto de amostras. Para isto, são utilizadas técnicas de *reconhecimento de padrões*.

**Padrão:** é uma descrição quantitativa ou estrutural de um objeto ou alguma outra entidade de interesse em uma imagem. Geralmente, um padrão é representado pelo *vetor de características*.

**Vetor de características:** é um vetor composto pelas características extraídas da imagem.

**Classe de padrões:** é uma família de padrões que compartilham algumas propriedades comuns, ou seja, é um grupo de padrões com características similares.

**Reconhecimento de padrões:** também denominado de classificação de padrões, consiste na atribuição dos padrões a suas respectivas classes automaticamente e com a mínima intervenção humana possível.

**Protótipo:** é o representante de uma classe, sendo definido pela média dos padrões de uma classe, conforme a equação 2.9:

$$m_j = \frac{1}{N_j} \sum_{x \in \omega_j} x, \text{ para } j \in [1, M] \quad (2.9)$$

onde:

$W_j$ : é a  $j$ -ésima classe de padrões;  $N_j$ : é o número de padrões da classe  $\omega_j$ ;

$M$ : é a quantidade de classes de padrões.

**Vetor de pesos:** este vetor é utilizado para ponderar o vetor de características, de modo a valorizar as que melhor descrevam as classes.

**Dendrograma:** é uma estrutura do tipo árvore de agrupamento binário, a qual utiliza a forma de 'U' invertido para representar a interligação hierárquica dos objetos. A distância entre dois vetores de características interligados é equivalente à altura desta ligação. A figura 2.3 ilustra 5 vetores de características representados no plano cartesiano e seu respectivo dendrograma elaborado pela técnica de agrupamento hierárquico *single-link*, tendo como medida de similaridade a *distância euclidiana*.

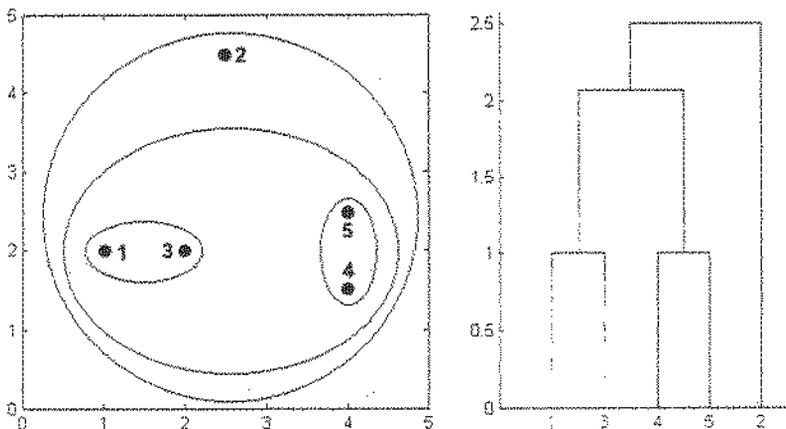


Fig. 2.3: Interpretação de um dendrograma.

**Single-link:** é uma técnica de agrupamento hierárquico, a qual utiliza as distâncias entre os padrões para definir as classes de acordo com uma distância mínima estipulada (*threshold*).

### 2.5.1 Classificador de Distância Mínima

O classificador de distância mínima, também é conhecido como **casamento de modelos**, consiste em comparar o padrão a ser reconhecido com os *protótipo (modelo)* representantes de cada *classe de padrões*. Para isto utiliza uma *medida de similaridade*. Como resultado, temos o padrão associado à classe mais próxima, ou seja, à classe de menor distância do padrão ao seu protótipo. A figura 2.4 ilustra este classificador, na qual temos as classes ( $W$ ), os protótipos de cada classe ( $m$ ) e um padrão a ser classificado ( $x$ ). Nesta imagem podemos verificar que se for utilizada a distância euclidiana como medida de similaridade, o padrão será reconhecido como pertencente à classe  $W_2$ , pois ela é representada pelo protótipo mais próximo do padrão. Também podemos visualizar as *fronteiras de decisão*.

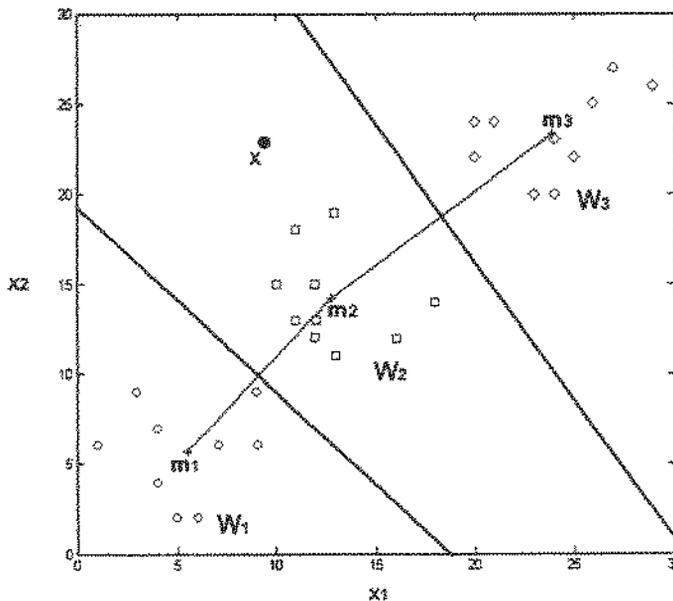


Fig. 2.4: Representação de um classificador de distância mínima.

**Medidas de similaridade:** esta medida retorna a distância do padrão ao protótipo.

**Fronteiras de decisão:** são definidas como retas perpendiculares ao segmento que liga dois protótipos, sendo que a fronteira intercepta a mediatriz desse segmento.

**Distância euclidiana:** é uma medida de distância, de forma radial, definida pela equação 2.10:

$$D(x, m_j) = \sqrt{\sum_{i=1}^n (x_i - m_{ji})^2} \quad (2.10)$$

onde:

$x$ : é o vetor de características de um padrão a ser reconhecido;

$m_j$ : é o vetor de características do protótipo da classe  $W_j$ ;

$n$ : é a quantidade de características extraídas da imagem.

**Distância euclidiana ponderada:** é uma medida de distância, normalmente de forma elíptica. Sua principal característica é ponderar o vetor de características de todas as classes por um único vetor de pesos definidos empiricamente ou através de medidas estatísticas. Esta medida de similaridade é definida pela equação 2.11:

$$D(x, m_j) = \sqrt{\sum_{i=1}^n p_i (x_i - m_{ji})^2} \quad (2.11)$$

onde:

$x$ : é o vetor de características de um padrão a ser reconhecido;

$m_j$ : é o vetor de características do protótipo da classe  $W_j$ ;

$p$ : é o vetor de pesos;

$n$ : é a quantidade de características extraídas da imagem.

## Capítulo 3

# Análise Bibliográfica

Devido à grande motivação no desenvolvimento de Sistemas de Reconhecimento da Placa de Licenciamento Veicular (SRPLV), pois ainda nenhum deles conseguiu obter a mesma confiabilidade que na identificação de veículo feita por um ser humano, muitas propostas foram publicadas.

Existem centenas de propostas diferentes de *softwares* para SRPLV, sendo que neste momento vários desenvolvedores estão trabalhando em novas propostas. Durante a elaboração desta dissertação, analisamos 42 propostas publicadas em artigos, teses ou patentes.

Considerando que um *software* para SRPLV é composto por módulos fundamentais e que cada módulo pode ser desenvolvido por métodos diferentes, conclui-se que ele pode ser composto por inúmeras combinações, sendo inviável classificar todas as propostas de modo global. Portanto, classificamos as propostas de acordo com os seguintes módulos fundamentais:

- Localização da Placa;
- Enquadramento da Placa;
- Separação dos Caracteres;
- Reconhecimento dos Caracteres.

O módulo de *Pré-Processamento* não foi utilizado na classificação, pois seu desenvolvimento depende de vários fatores, tais como: qualidade da imagem de entrada, métodos utilizados nos demais módulos, etc.

Como não é possível definir todas as possibilidades de métodos de desenvolvimento para cada módulo, podem existir ou surgir propostas que não se enquadram em nenhuma das classificações que sugerimos, tornando necessário formar uma nova classe de métodos.

As seções seguintes definem os principais métodos utilizados no desenvolvimento de cada módulo fundamental, os quais são explicados sucintamente, sendo também classificadas as propostas analisadas em seus respectivos métodos.

### 3.1 Métodos de Localização da Placa

No Brasil, as cores dos caracteres e do fundo da placa de licenciamento veicular dependem da categoria do veículo, conforme a **resolução nº 45 do CONTRAN**. A maioria dos veículos em circulação pertence à categoria particular, os quais tem os caracteres escuros e o fundo da placa claro. Portanto, é comum desenvolver os métodos de localização da placa para trabalharem com as características desta categoria. Deste modo, para localizar a placa de um veículo cujos caracteres sejam claros e o fundo da placa seja escuro, é necessário gerar o negativo da imagem, o qual complementa o valor dos pixels, ou seja, os pixels de claros tornam-se escuros e vice-versa.

Analisando as características da placa de licenciamento veicular utilizadas em cada método constatamos que podemos dividi-los em 3 grupos:

1. Métodos baseados na morfologia dos caracteres;
2. Métodos baseados na morfologia do fundo da placa;
3. Métodos híbridos.

Os métodos baseados na morfologia dos caracteres utilizam as seguintes características da placa de licenciamento veicular: cor dos caracteres diferente da cor da placa; os caracteres da placa possuem altura e espaçamento regular, sendo alinhados horizontalmente; e quantidade determinada de caracteres.

Os principais métodos do primeiro grupo são:

- Método da Morfologia;
- Método da Variação Tonal;
- Método da Correlação.

Os métodos baseados na morfologia do fundo da placa utilizam a seguinte característica: placa em formato retangular com fundo uniforme.

Os principais métodos do segundo grupo são:

- Método da Detecção das Arestas;
- Método Conexionista.

Os métodos híbridos visam melhorar a eficiência na localização da placa, sendo compostos pela combinação de dois ou mais métodos diferentes. Estes métodos são classificados no terceiro grupo.

### 3.1.1 Método da Morfologia

Este método utiliza as informações geométricas dos caracteres para encontrar as regiões promissoras.

Primeiramente é feita a binarização da imagem. As técnicas de binarização mais utilizadas são:

- *Binarização global manual;*
- *Binarização global automática;*
- *Binarização local adaptativa;*
- *Binarização iterativa.*

As técnicas de binarização acima citadas têm melhor desempenho com uma iluminação homogênea, ou seja, não pode existir foco de luz nem refletância excessiva.

Algumas propostas recentes desenvolveram técnicas específicas para binarizar a imagem de veículo visando separar os caracteres da placa dos demais objetos. Dentre elas podemos citar a proposta de Martín et al. (2000). Também desenvolvemos uma técnica inovadora, a qual denominamos de **binarização dos mínimos regionais da imagem filtrada**, a qual é descrita no capítulo 4.

Devido à complexidade da imagem, o resultado da binarização contém outros elementos conexos além dos correspondentes aos caracteres da placa. Portanto, é necessário filtrá-los de modo que resultem apenas os *componentes conexos* dos caracteres da placa. Para isto, são extraídas informações morfológicas (estruturais) dos componentes conexos da imagem. Os quais são filtrados de acordo com as seguintes características: os caracteres da placa possuem altura e espaçamento regular, sendo alinhados horizontalmente; e quantidade determinada de caracteres.

A principal dificuldade deste método é a existência de outras cadeias de caracteres com características semelhantes à placa de licenciamento, tais como: marca do veículo, modelo do veículo, prefixo do veículo, etc.

A proposta mais conhecida que utiliza o método da morfologia é a de Coetzee et al. (1998), a qual utiliza o método de Niblack (1986) para binarizar a imagem.

As seguintes propostas também utilizam o método da morfologia para localizar a placa: Lotufo and Dias (2004); Koval et al. (2003); Huang and Guo (2003); Hontani and Koga (2001); Martín et al. (2000); Naito et al. (2000, 1999); Salagado et al. (1999).

Nossa proposta também utiliza este método. Primeiro é aplicada uma binarização dos mínimos regionais da imagem filtrada. Para isto utiliza os mínimos regionais da imagem após ter as regiões homogêneas expandidas pelo filtro de volume mínimo (*v-minima*). Em seguida são aplicados filtros morfológicos com base nas características dos caracteres da placa citadas anteriormente. Uma explicação detalhada de nossa proposta é apresentada no capítulo 4.

### 3.1.2 Método da Variação Tonal

Este método consiste na procura, em cada linha horizontal da imagem, de um padrão de frequência e amplitude, ou seja, uma “*assinatura*”. O padrão procurado é uma região com grande quantidade de transição da cor, criado pelo contraste entre a cor dos caracteres e do fundo da placa.

A proposta que utiliza o método da variação tonal mais conhecida é a de Soh et al. (1994). Consiste basicamente de quatro etapas. Na primeira etapa é feita uma amostragem de linhas horizontais em intervalos regulares  $\tau$ , visando reduzir o tempo de busca. Onde  $\tau$  deve ser menor ou igual à altura dos caracteres da placa. A segunda etapa consiste em codificar cada linha amostrada, utilizando o gradiente unidimensional de Sobel. Os picos positivos da onda resultante representam as transições de baixa para alta luminância e os picos negativos vice-versa. Os valores absolutos dos picos menores que  $\theta$  são descartados (filtro pela amplitude). Em seguida, um tipo de pico é adotado (positivo ou negativo) de acordo com a cor dos caracteres e do fundo da placa, sendo os picos complementares descartados, ou seja, se forem adotados os picos positivos então os negativos serão descartados e vice-versa. Na terceira etapa, a distância entre os picos (intervalos) remanescente da etapa anterior que estiverem dentro de uma faixa pré-determinada são agrupados (filtro pela frequência), resultando em vários grupos com quantidades diferentes de intervalos válidos. Na quarta etapa, os grupos formados pela etapa anterior são filtrados por uma quantidade mínima e máxima de intervalos. As regiões resultantes deste filtro são as regiões promissoras a placa de licenciamento veicular.

O principal problema deste método é confundir a placa de licenciamento veicular com alguns padrões de textura, tais como: paralelepípedos, grade do radiador do veículo, galhos de árvore, etc.

Outras propostas que utilizam o método da variação tonal para localizar a placa são: Hsieh et al. (2005); Huang et al. (2004); Taleb-Ahmed et al. (2003); Wang and Lee (2003); Borba et al. (2002); Yan et al. (2001); Barroso et al. (1997); Miyamoto et al. (1991).

### 3.1.3 Método da Correlação

A correlação permite encontrar uma sub-imagem dentro de uma imagem, sendo que as dimensões da sub-imagem devem ser menores ou iguais que as dimensões da imagem.

Para encontrar um objeto na imagem  $f$ , deve-se fazer a correlação com a imagem do objeto procurado  $w$ , sendo que a imagem do objeto em  $w$  deve ter o mesmo tamanho e orientação que na imagem  $f$ . A imagem resultante contém valores altos nas regiões semelhantes ao objeto procurado.

Utilizar este método para localizar uma placa de licenciamento implica em restrições do sistema, visto que a orientação da placa depende da posição do veículo, da fixação da placa e da posição da câmera; assim como o tamanho da placa também depende da distância do veículo em relação à câmera e da distância focal. Deste modo é necessário configurar o sistema de acordo com: a posição e orientação da câmera; a distância do sensor que detecta a presença do veículo em relação à câmera; e da distância focal. Outro problema é que a imagem da placa depende dos caracteres que a compõe, deste modo, não é possível ter uma imagem real da placa a ser procurada, mas apenas uma imagem aproximada.

Portanto, existem poucas propostas que utilizam este método. A proposta de Sáenz et al. (2001) faz a correlação normalizada utilizando, como imagem padrão para busca, o somatório de várias imagens de placas amostradas.

Outra proposta que utiliza o método da correlação para localizar a placa é: Shapiro et al. (2003).

### 3.1.4 Método da Detecção das Arestas

Uma das características da placa de licenciamento veicular é a de ter formato retangular com fundo uniforme, a qual é utilizada por este método para localizar a placa.

A primeira etapa deste método consiste em aplicar um *detector de contornos*, também conhecido como detector de bordas, tais como: *Sobel*, *Roberts*, *Prewitt*, *Laplaciano*, *Cruzamento por Zero*, *Canny*, entre outros, sendo necessário realçar todo o contorno da placa.

Em seguida é utilizado um método de *detecção de retas* para encontrar as arestas da placa. O método de detecção de retas mais utilizado é a *transformada de Hough* (HT).

A etapa final consiste em procurar, no conjunto das retas detectadas, quatro segmentos de retas concorrentes 2 a 2. Sendo que dois segmentos são orientados horizontalmente e os outros dois verticalmente, podendo ser oblíquos. Embora a distorção de perspectiva aplique um pequeno efeito trapézio, pode-se considerar que os segmentos horizontais tendem a ser paralelos entre si, assim como os segmentos verticais. Também podemos dizer que 2 segmentos concorrentes tendem a ser ortogo-

nais. Outra característica importante é do conhecimento a priori da razão de aspecto da placa (razão do comprimento do segmento horizontal pelo segmento vertical).

Isto faz com que este método tenha algumas vantagens sobre os demais: é invariante a distorção de perspectiva, aliás pode ser utilizado para corrigir esta distorção; consegue detectar as placas em que seus caracteres estejam borrados ou apagados; é independente da distância do veículo em relação à câmara e também da distância focal da câmara; etc.

Porém, como todo método, também tem algumas desvantagens: necessita que a borda da placa de licenciamento veicular tenha bom contraste, ou seja, a cor do fundo da placa deve ser diferente da cor do local em que a placa está fixada; geralmente a imagem de um veículo contém outros quadriláteros irregulares (faróis do veículo, grade do radiador, etc), o que dificulta a busca; exige maior custo computacional que os demais métodos; etc.

O item mais importante deste método é conseguir detectar o contorno da placa de licenciamento veicular corretamente. Isto depende do contraste entre a placa e o local em que está fixada.

Algumas propostas, como a de Yu and Kim (2000), são mais simplificadas. Primeiro é aplicado um detector de contorno vertical. Em seguida, os contornos são filtrados com critérios de forma e posição (altura e distância entre dois segmentos) para localizar a placa. Embora tenha um menor custo computacional está mais sujeito a erro.

Também encontramos as seguintes propostas que utilizam método da detecção das arestas para localizar a placa: Duan et al. (2005); Sarfraz et al. (2003); Ahmed et al. (2003); Lee et al. (2002); Hegt et al. (1998); Kamat and Ganesan (1995).

### 3.1.5 Método Conexionista

Este método é baseado na procura por regiões da imagem que tenham cor e textura semelhantes às de uma placa de licenciamento veicular.

Os principais métodos conexionistas utilizados são:

- Redes neurais;
- SVM (*Support Vector Machine*);
- *Art Map Fuzzy*;
- SOM (*Self Organization Map*).

Primeiramente, a imagem de entrada deve ser dividida em regiões menores que a placa de licenciamento veicular. Em seguida, são aplicados métodos conexionistas para processar cada região da

imagem, visando classificar esta região como pertencente ou não à placa de licenciamento veicular. Como resultado final temos demarcadas as regiões promissoras a conterem a placa de licenciamento veicular.

Este método é sensível ao treinamento, ou seja, para que funcione corretamente com as variações da intensidade da fonte de iluminação, comum em ambiente não controlado, é necessário cuidado no processo de amostragem das imagens para o treinamento. Também é sensível à refletância da placa.

Algumas propostas utilizam as informações de cores, necessitando assim de imagem colorida. Dentre elas podemos citar: Nijhuis et al. (1995).

Outras propostas que utilizam o método conexionista para localizar a placa são: Nathan et al. (2004); Chang et al. (2004); Jianfeng et al. (2003); Guingo et al. (2002); Kim et al. (2000); Brugge et al. (1998); Kim et al. (1996).

### 3.1.6 Método Híbrido

Muitos autores concluem que não é possível desenvolver um método que trate de todos os problemas da localização da placa. Devido a isto, recomendam o uso de mais de um método, sendo estes complementares.

Algumas propostas que utilizam o método híbrido são: Wang et al. (2004); Souza and Susin (2000).

## 3.2 Métodos de Enquadramento da Placa

Geralmente a região promissora, obtida no módulo de *Localização da Placa*, contém outros objetos ao redor da placa de licenciamento (para-choque, grade de ar, etc). Também pode ocorrer a distorção de perspectiva devido ao posicionamento da câmera de modo não ortogonal à placa de licenciamento veicular. Esses problemas prejudicam a eficiência dos demais módulos.

O enquadramento da placa visa aplicar um ajuste fino na sua localização, podendo também corrigir a distorção de perspectiva.

As propostas que utilizam o *método de detecção das arestas* para localização da placa, já possuem a região promissora com correto enquadramento da placa, portanto somente precisam corrigir a distorção de perspectiva.

Os métodos de enquadramento da placa podem ser divididos em 2 grupos:

1. Métodos que não corrigem a distorção de perspectiva da placa;
2. Métodos que corrigem a distorção de perspectiva da placa.

Os principais métodos do primeiro grupo são:

- Método da Morfologia;
- Método das Projeções.

Os principais métodos do segundo grupo são:

- Método Iterativo;
- Método da Detecção das Arestas.

Algumas propostas não implementam este módulo, porém necessitam de um bom posicionamento da câmera. Outras desenvolvem somente o enquadramento, não dando importância à distorção de perspectiva. Porém, as propostas que efetuam as duas correções resultam em ganhos significativos na eficiência da separação e reconhecimento dos caracteres.

### 3.2.1 Método da Morfologia

O método da morfologia utiliza as características morfológicas dos caracteres para remover os demais objetos presentes na região promissora. Este método não efetua a correção da distorção de perspectiva.

Primeiro são desenvolvidos filtros baseados na morfologia dos caracteres de modo a eliminar os demais objetos. Em seguida é aplicado o enquadramento de modo a conter somente os componentes conexos resultantes após a aplicação dos filtros.

Este método é de simples implementação, sendo utilizado pela maioria dos autores.

Algumas propostas que utilizam o método de morfologia para enquadrar a placa são: Ahmed et al. (2003); Koval et al. (2003); Wang and Lee (2003); Martín et al. (2000); Yu and Kim (2000); Souza and Susin (2000); Naito et al. (2000).

### 3.2.2 Método das Projeções

Este método é geralmente utilizado em propostas que localizam a placa pelo *método de variação tonal*. O método das projeções não efetua a correção da distorção de perspectiva.

O enquadramento é efetuado com base nas projeções vertical e horizontal. Geralmente utiliza a detecção de contornos ou a binarização da região promissora para obter as projeções.

A proposta de Barroso et al. (1997) utiliza este método para efetuar o enquadramento da placa.

Outras propostas que utilizam o método das projeções para enquadrar a placa são: Shapiro et al. (2003); Taleb-Ahmed et al. (2003); Rahman et al. (2003); Guingo et al. (2002); Yan et al. (2001).

### 3.2.3 Método Iterativo

Este método primeiro efetua a correção da distorção de perspectiva para em seguida efetuar o enquadramento da placa.

A distorção de perspectiva é ocasionada por dois graus de liberdade da câmara: *pan* e *tilt*. Portanto precisa de duas correções: a rotação da placa e a inclinação dos caracteres da placa.

Ambas as correções são iterativas, as quais utilizam uma função-objetivo. Para corrigir a rotação da placa são aplicadas rotações na região promissora de um grau mínimo a um grau máximo, estipulados empiricamente. Para cada transformada de rotação é calculada a função-objetivo com base na projeção horizontal da detecção de contornos da região promissora. O grau que corrige a rotação da placa é o grau com maior valor retornado pela função-objetivo, o qual deve ser aplicado para corrigir a rotação. A correção da inclinação dos caracteres é efetuada da mesma maneira, porém em vez da transformada de rotação é aplicada a transformada de inclinação (*skew*), sendo que a função-objetivo é baseada na projeção vertical da detecção de contornos da região promissora. O grau de correção da inclinação também é obtido da mesma forma que o da rotação.

Depois de corrigidas a rotação e a inclinação é efetuado o enquadramento da placa utilizando as projeções horizontal e vertical da detecção de contornos da imagem corrigida.

Uma patente da Siemens (Tyan and Neubauer, 2002), utiliza este método de enquadramento da placa, obtendo bons resultados.

Outras propostas que utilizam o método iterativo para enquadrar a placa são: Huang et al. (2004); Salgado et al. (1999).

### 3.2.4 Método da Detecção das Arestas

Este método opera identicamente ao método de mesmo nome para localização da placa, descrito na seção 3.1.4. Porém utiliza uma área restrita da imagem (região promissora), com isto tem menor custo computacional, viabilizando sua adoção.

Portanto, caso não seja utilizado o *método da detecção das arestas* para localizar a placa, descrito na seção 3.1.4, é necessário aplicá-lo na região promissora para obter os 4 vértices da placa de licenciamento, os quais formam o quadrilátero irregular utilizado para o correto enquadramento da placa.

Para corrigir a distorção de perspectiva é aplicada uma transformação projetiva linear simples, também conhecida como *homografia*, a qual utiliza as coordenadas dos 4 vértices detectados e seus respectivos pares a serem representados na imagem corrigida, os quais deverão compor um retângulo. Deste modo, transforma o quadrilátero irregular em um retângulo com uma única transformação geométrica.

Esse método apresenta melhores resultados tanto no enquadramento quanto na correção da distorção de perspectiva.

Nossa proposta utiliza este método para efetuar o enquadramento da placa, obtendo ótimos resultados.

Dentre as propostas que utilizam o método de detecção das arestas para enquadrar a placa podemos citar: Duan et al. (2005); Zhang and Zhang (2003); Hegt et al. (1998).

## 3.3 Métodos de Separação dos Caracteres

O módulo de *Reconhecimento dos Caracteres* necessita da imagem do caractere devidamente enquadrada para extrair suas características corretamente. Esta tarefa pode ser simples, caso a imagem da placa de licenciamento não contenha ruídos e nenhum caractere toque em outro, ou complexa nos demais casos.

Esses métodos operam na imagem binarizada resultante do módulo de *Enquadramento da Placa*. Portanto, a técnica utilizada para a binarização é de fundamental importância. Os principais problemas da binarização das imagens da placa são: presença de ruído, iluminação heterogênea e reflectância da mesma. Devido à diversidade de técnicas utilizadas para binarização não foi viável classificá-las. Porém, constatamos que as mais utilizadas são: *binarização global automática* (Otsu, 1979) e *binarização local adaptativa* (Niblack, 1986). Sendo que existem técnicas mais avançadas

que podem contribuir para a melhoria deste módulo.

Os principais métodos para a separação dos caracteres são:

- Método das Projeções;
- Método da Morfologia;
- Método Iterativo.

Devido às condições do ambiente em que o sistema opera e das placas de licenciamento veicular, esta tarefa geralmente é complexa, exigindo métodos mais robustos. Portanto, nestas condições operacionais, os métodos simples, tais como: *Método das Projeções* e *Método da Morfologia* têm menor eficiência que os demais.

### 3.3.1 Método das Projeções

O método das projeções é de fácil implementação, mas exige que os caracteres não se toquem e que os retângulos envolventes dos caracteres não se sobreponham.

Este método foi inicialmente proposto por Lu (1995), chamado "*peak-to-valley*". O método consiste em procurar por vales da projeção vertical da imagem binarizada.

Primeiro a cadeia de caracteres da placa é enquadrada por meio das projeções horizontal e vertical da imagem binarizada. Em seguida os caracteres são separados com base nos mínimos regionais (vales) da projeção vertical da imagem resultante.

As seguintes propostas utilizam o método das projeções para separar os caracteres: Duan et al. (2005); Wang et al. (2004); Sarfraz et al. (2003); Rahman et al. (2003); Ahmed et al. (2003); Zhang and Zhang (2003); Guingo et al. (2002); Lee et al. (2002); Yan et al. (2001); Martín et al. (2000); Yu and Kim (2000); Kim et al. (2000); Salagado et al. (1999); Hegt et al. (1998); Barroso et al. (1997).

### 3.3.2 Método da Morfologia

Este método separa os caracteres com base nos componentes conexos. Além de exigir que os caracteres não se toquem e que os retângulos envolventes dos caracteres não se sobreponham, também exige que os caracteres não estejam rompidos (caractere composto por mais de um componente conexo).

Primeiro os componentes conexos da imagem binarizada são filtrados de acordo com conhecimento a priori de sua geometria. Em seguida os caracteres são enquadrados de acordo com o retângulo envolvente de seu respectivo componente conexo.

As seguintes propostas que utilizam métodos da morfologia para separar os caracteres são: Taleb-Ahmed et al. (2003); Koval et al. (2003); Wang and Lee (2003); Hontani and Koga (2001); Souza and Susin (2000); Nijhuis et al. (1995); Miyamoto et al. (1991).

### 3.3.3 Método Iterativo

O método iterativo possui a habilidade de separar caracteres unidos de modo intrínseco. Porém tem seu custo computacional maior que as demais classes.

Consiste basicamente em montar uma estrutura do tipo *árvore* com todas as separações possíveis de caracteres, onde cada *nó* da *árvore* representa as coordenadas horizontais do retângulo envolvente do candidato a caractere, sendo que as coordenadas verticais são as mesmas para todos os candidatos. Em seguida é aplicada uma função-objetivo, normalmente composta pelo valor da confiabilidade de reconhecimento, em cada candidato a caractere. Finalmente a *árvore* é percorrida de modo a encontrar o caminho que contém os *nós* com maior valor da função-objetivo.

Uma patente da Siemens (Tyan and Neubauer, 2002) utiliza este método de separação dos caracteres, obtendo bons resultados.

Outras propostas que utilizam o método iterativo para separar os caracteres são: Huang et al. (2004); Naito et al. (2000, 1999).

## 3.4 Métodos de Reconhecimento dos Caracteres

Para o reconhecimento dos caracteres é necessário efetuar uma amostragem rotulada de todos os possíveis caracteres a comporem a placa, obtendo mais de uma amostra para cada caractere. As placas de licenciamento veicular brasileiras são compostas por 3 letras maiúsculas do alfabeto latino, um hífen e 4 números naturais de 0 a 9, totalizando 36 caracteres diferentes.

O reconhecimento dos caracteres consiste em extrair características da imagem do candidato a caractere e compará-lo com as características das amostras. Tendo como resultado o rótulo da amostra de maior semelhança, o qual normalmente corresponde ao código ASCII do mesmo.

As principais dificuldades são:

- Diferença na tipografia do mesmo caractere (figura 1.8);
- Semelhança na tipografia de caracteres diferentes (figura 1.9);

Os principais métodos para o reconhecimento dos caracteres são:

- Método Estatístico;
- Método Conexionista;
- Método Sintático;

### 3.4.1 Método Estatístico

O método estatístico utiliza um classificador estatístico, também conhecido como “casamento de modelos” (*template matching*), para reconhecer os caracteres. Dentre eles podemos citar:

- Classificador de distância mínima;
- Classificador bayesiano;
- Classificador por correlação.

Este método consiste em encontrar o *padrão* de maior similaridade em um conjunto de amostras rotuladas. Para isto utiliza uma medida de distância, também conhecida como *medida de similaridade*. A escolha desta medida implica na eficiência do método. As principais medidas de similaridade são:

- Distância Euclidiana;
- Distância Euclidiana Ponderada;
- Distância Euclidiana Normalizada;
- Distância de Mahalanobis;
- Distância de Manhattan;
- Distância de Minkowsky;

Devido à base estatística deste método é possível definir uma medida real de confiabilidade, o que não é possível nos demais métodos. A medida de confiabilidade é utilizada pelo *método iterativo* de separação dos caracteres, sendo também utilizada para detectar falhas no reconhecimento. Portanto, se for necessário utilizar uma medida de confiabilidade este método é o mais recomendado.

A principal desvantagem deste método é que as fronteiras de decisões são *lineares* e dependente do conjunto de amostras. Portanto é necessário selecionar cuidadosamente o conjunto de amostras.

Algumas propostas que utilizam o método estatístico para reconhecer os caracteres são: Wang et al. (2004); Huang et al. (2004); Sarfraz et al. (2003); Rahman et al. (2003); Ahmed et al. (2003); Wang and Lee (2003); Tyan and Neubauer (2002); Lee et al. (2002); Yan et al. (2001); Martín et al.

(2000); Yu and Kim (2000); Naito et al. (2000, 1999); Salagado et al. (1999); Hegt et al. (1998); Barroso et al. (1997); Miyamoto et al. (1991).

Nossa proposta utiliza um classificador hierárquico de distância mínima, tendo como medida de similaridade a distância euclidiana.

### 3.4.2 Método Conexionista

Este método utiliza técnicas conexionistas para reconhecer os caracteres. Dentre as principais técnicas conexionistas podemos citar:

- Redes neurais;
- SVM (*Support Vector Machine*);
- *Art Map Fuzzy*;
- SOM (*Self Organization Map*).

O método conexionista utiliza um conjunto de amostras rotuladas para treinar uma rede de elementos. O treinamento consiste em ponderar as conexões entre os elementos de forma que o sinal de saída corresponda ao rótulo correto do padrão de entrada da rede. Com isto a estrutura tem a habilidade de “*aprender*” durante o processo (aprendizagem supervisionada).

As fronteiras de decisão são *não-lineares*, definidas iterativamente pelo método de treinamento e pelo conjunto de amostras. Portanto deve-se tomar cuidado tanto na definição do método de treinamento quanto na seleção do conjunto de amostras.

As seguintes propostas utilizam o método conexionista para reconhecer os caracteres: Duan et al. (2005); Chang et al. (2004); Koval et al. (2003); Taleb-Ahmed et al. (2003); Guingo et al. (2002); Souza and Susin (2000); Kim et al. (2000); Nijhuis et al. (1995).

### 3.4.3 Método Sintático

Este método utiliza um conjunto de primitivas de padrões; um conjunto de regras de produção, na forma de uma gramática; e um reconhecedor (*autômato*), tendo sua estrutura determinada pelo conjunto de regras.

Consiste basicamente em extrair as primitivas do candidato a caractere, onde cada primitiva é representada por uma letra minúscula do alfabeto; compor uma sentença com as primitivas utilizando uma gramática, sendo que a gramática é um conjunto de regras de sintaxe; e utilizar um *autômato finito* para reconhecer a sentença, o qual resulta no reconhecimento do caractere.

Neste método um padrão é representado por uma sentença e uma classe é definida pelo conjunto de sentenças geradas por uma mesma gramática  $G$ , o qual também é denominado de sua linguagem  $L(G)$ .

Sua principal vantagem é necessitar de apenas uma amostra de cada caractere para extrair as primitivas e definir a gramática.

Uma das principais propostas que utilizam este método foi apresentada por Cowell (1995).

Outra proposta que utiliza o método sintático para reconhecer os caracteres é: Polidório and Borges (1996).

### 3.5 Visão Geral das Propostas Analisadas

Visando facilitar a visualização geral da classificação das propostas analisadas, elaboramos a tabela 3.1. Nela podemos verificar o método utilizado em cada módulo das 42 propostas que analisamos durante o período de elaboração desta dissertação.

| Propostas                   |      | Métodos        |               |              |                |
|-----------------------------|------|----------------|---------------|--------------|----------------|
| Referência                  | Pub. | Localização    | Enquadramento | Separação    | Reconhecimento |
| Hsieh et al. (2005)         | cnf  | Variação Tonal | Projeções     | -            | -              |
| Duan et al. (2005)          | cnf  | Det. Arestas   | Det. Arestas  | Projeções    | Conexionista   |
| Lotufo and Dias (2004)      | ptn  | Morfologia     | -             | -            | -              |
| Nathan et al. (2004)        | cnf  | Conexionista   | -             | -            | -              |
| Wang et al. (2004)          | cnf  | Híbrido        | -             | Projeções    | Estatístico    |
| Chang et al. (2004)         | cnf  | Conexionista   | -             | -            | Conexionista   |
| Huang et al. (2004)         | cnf  | Variação Tonal | Iterativo     | Iterativo    | Estatístico    |
| Jianfeng et al. (2003)      | cnf  | Conexionista   | -             | -            | -              |
| Shapiro et al. (2003)       | cnf  | Correlação     | Projeções     | -            | -              |
| Taleb-Ahmed et al. (2003)   | cnf  | Variação Tonal | Projeções     | Morfologia   | Conexionista   |
| Sarfraz et al. (2003)       | cnf  | Det. Arestas   | -             | Projeções    | Estatístico    |
| Rahman et al. (2003)        | cnf  | -              | Projeções     | Projeções    | Estatístico    |
| Ahmed et al. (2003)         | cnf  | Det. Arestas   | Morfologia    | Projeções    | Estatístico    |
| Koval et al. (2003)         | cnf  | Morfologia     | Morfologia    | Morfologia   | Conexionista   |
| Wang and Lee (2003)         | cnf  | Variação Tonal | Morfologia    | Morfologia   | Estatístico    |
| Huang and Guo (2003)        | cnf  | Morfologia     | -             | -            | -              |
| Zhang and Zhang (2003)      | cnf  | -              | Det. Arestas  | Projeções    | -              |
| Tyan and Neubauer (2002)    | ptn  | -              | Iterativo     | Iterativo    | Estatístico    |
| Guingo et al. (2002)        | cnf  | Conexionista   | Projeções     | Projeções    | Conexionista   |
| Borba et al. (2002)         | cnf  | Variação Tonal | -             | -            | -              |
| Lee et al. (2002)           | cnf  | Det. Arestas   | -             | Projeções    | Estatístico    |
| Sáenz et al. (2001)         | cnf  | Correlação     | -             | -            | -              |
| Yan et al. (2001)           | cnf  | Variação Tonal | Projeções     | Projeções    | Estatístico    |
| Hontani and Koga (2001)     | cnf  | Morfologia     | -             | Morfologia   | -              |
| Martin et al. (2000)        | cnf  | Morfologia     | Morfologia    | Projeções    | Estatístico    |
| Yu and Kim (2000)           | cnf  | Det. Arestas   | Morfologia    | Projeções    | Estatístico    |
| Souza and Susin (2000)      | cnf  | Híbrido        | Morfologia    | Morfologia   | Conexionista   |
| Kim et al. (2000)           | cnf  | Conexionista   | -             | Projeções    | Conexionista   |
| Naito et al. (2000)         | cnf  | Morfologia     | Morfologia    | Iterativo    | Estatístico    |
| Naito et al. (1999)         | cnf  | Morfologia     | -             | Iterativo    | Estatístico    |
| Salgado et al. (1999)       | cnf  | Morfologia     | Iterativo     | Projeções    | Estatístico    |
| Coetzee et al. (1998)       | cnf  | Morfologia     | -             | Morfologia   | Conexionista   |
| Hegt et al. (1998)          | cnf  | Det. Arestas   | Det. Arestas  | Projeções    | Estatístico    |
| Brugge et al. (1998)        | cnf  | Conexionista   | -             | Conexionista | -              |
| Barroso et al. (1997)       | cnf  | Variação Tonal | Projeções     | Projeções    | Estatístico    |
| Polidório and Borges (1996) | cnf  | -              | -             | -            | Sintático      |
| Kim et al. (1996)           | cnf  | Conexionista   | -             | -            | -              |
| Nijhuis et al. (1995)       | cnf  | Conexionista   | -             | Morfologia   | Conexionista   |
| Cowell (1995)               | prd  | Morfologia     | -             | Morfologia   | Sintático      |
| Kamat and Ganesan (1995)    | cnf  | Det. Arestas   | -             | -            | -              |
| Soh et al. (1994)           | cnf  | Variação Tonal | -             | -            | -              |
| Miyamoto et al. (1991)      | cnf  | Variação Tonal | -             | Morfologia   | Estatístico    |

cnf: conferência prd: periódico ptn: patente

Tab. 3.1: Classificação das propostas analisadas.

## Capítulo 4

### Metodologia Adotada

A análise bibliográfica, explanada no capítulo 3, apresentou os diversos métodos utilizados para efetuar o reconhecimento visual da placa de licenciamento veicular por intermédio da computação, assim como suas vantagens e desvantagens.

Com base nesta análise, conseguimos identificar uma combinação de métodos que tendem a maximizar a eficiência no reconhecimento da placa de licenciamento veicular.

Resumidamente, nossa metodologia é composta pelos seguintes métodos:

- Localização da Placa: *Método da Morfologia*;
- Enquadramento da Placa: *Método da Detecção das Arestas*;
- Separação dos Caracteres: *Método das Projeções*;
- Reconhecimento dos Caracteres: *Método Estatístico*.

As principais contribuições estão no método de localização da placa, efetuando a binarização com base nos mínimos regionais da imagem filtrada; no enquadramento com correção da distorção de perspectiva da placa, através da homografia; na separação dos caracteres com auxílio da projeção vertical; e no reconhecimento dos caracteres, utilizando um classificador hierárquico de distância mínima.

O método que adotamos para a localização da placa utiliza as propriedades de luminância, contraste e parte da morfologia dos caracteres; sendo menos sensível a padrões de textura (tais como: paralelepípedos, grade do radiador do veículo, galhos de árvore, etc) do que os outros métodos. Para minimizar o problema de não encontrar a placa, todas as regiões promissoras são processadas pelos demais módulos.

Efetuamos o enquadramento da placa através de suas arestas, sendo este método eficiente na eliminação de objetos que não estejam contidos na placa. Também corrigimos a distorção de perspectiva

da placa, contribuindo com melhorias significativas na separação e reconhecimento dos caracteres.

A separação dos caracteres é baseada no modelo geométrico da placa e na disposição dos caracteres, permitindo que todos eles tenham o mesmo enquadramento horizontal e facilitando o enquadramento vertical de cada um deles.

Para o reconhecimento dos caracteres utilizamos um classificador hierárquico de distância mínima, o qual divide o problema da classificação em duas etapas. Deste modo, o tratamento de conflitos entre caracteres semelhantes é simplificado. Também analisamos os agrupamentos de amostras rotuladas com auxílio de um dendrograma elaborado pela técnica de agrupamento hierárquico denominada *single-link*.

A figura 4.1 apresenta o diagrama geral da metodologia adotada, o qual é explicado detalhadamente nas seções seguintes.

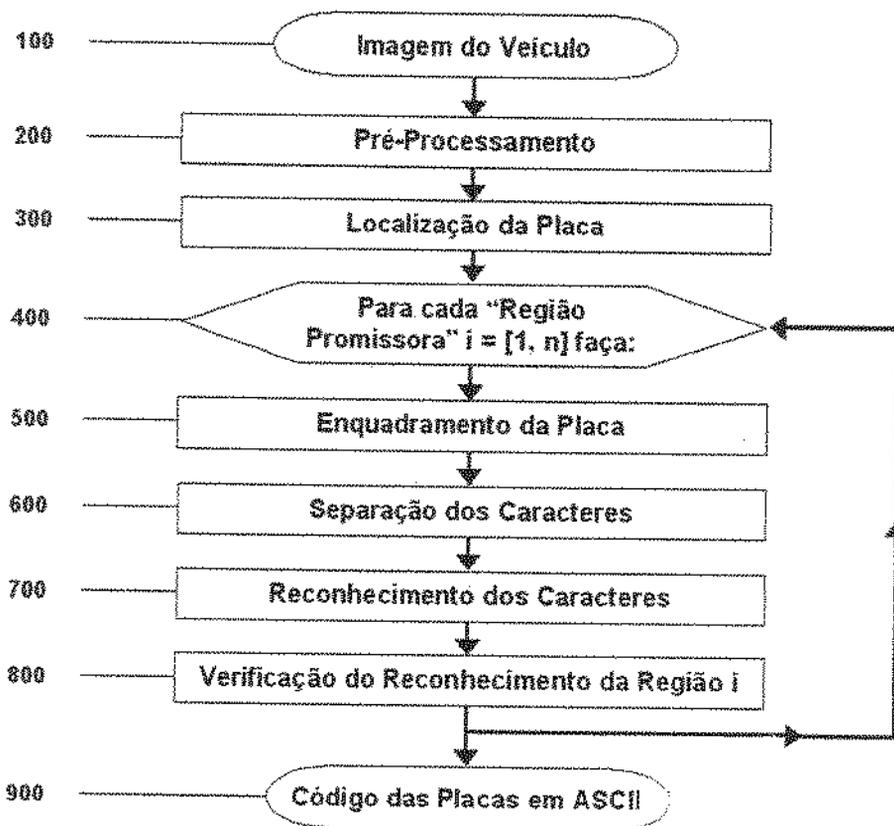


Fig. 4.1: Diagrama geral da nossa proposta.

## 4.1 Imagem do Veículo

Como imagem de entrada, bloco 100 do diagrama da figura 4.1, temos uma foto da frente ou traseira do veículo, a qual deve conter a placa de licenciamento veicular, figura 4.2.



Fig. 4.2: Imagem do veículo.

Nossa metodologia somente necessita da luminância da imagem para efetuar o reconhecimento da placa de licenciamento veicular. Portanto, não é necessário que a imagem de entrada seja colorida.

Duas características importantes de uma imagem digital são: *resolução nos níveis de cinza* e *resolução espacial*, as quais são utilizadas para discretizar uma imagem real. Um SRPLV necessita de uma resolução nos níveis de cinza mínima que separe os caracteres do fundo da placa. A *resolução nos níveis de cinza* recomendável é de 256 níveis. Porém, não adianta ter uma boa resolução nos níveis de cinza se a imagem não tiver um bom contraste e não for obtida em boas condições de iluminação. Também necessita de uma resolução espacial mínima que permita extrair informações morfológicas de cada caractere para o reconhecimento. Observe que a resolução espacial está ligada diretamente à confiabilidade no reconhecimento, mas inversamente ao tempo de processamento. Nós constatamos que a *resolução espacial* mínima da placa de licenciamento veicular é: 91x30 *pixels*, onde cada caractere tem resolução média de 10x15 *pixels*, sendo a resolução recomendável da placa de: 122x40 *pixels*, onde cada caractere tem resolução média de 14x20 *pixels*.

## 4.2 Pré-Processamento

O bloco 200 do diagrama da figura 4.1, consiste em preparar a imagem do veículo, figura 4.2, para ser processada pelo módulo de *Localização da Placa*.

Nesta etapa, somente convertemos a imagem do veículo para o formato de 256 níveis de cinza. Caso a imagem já esteja neste formato é enviada para o próximo bloco sem nenhum processamento.

Para converter uma imagem colorida (formato RGB) em 256 níveis de cinza utilizamos o *modelo de cores YIQ*.

A figura 4.3 apresenta a imagem resultante deste processamento aplicado na figura 4.2.



Fig. 4.3: Imagem resultante do pré-processamento da figura 4.2.

## 4.3 Localização da Placa

O bloco 300 do diagrama da figura 4.1, tem a finalidade de encontrar toda região que contenha uma placa de licenciamento veicular. Este bloco está representado na figura 4.4, sendo descrito detalhadamente nesta seção.

Dentre os métodos de localização da placa classificados na análise bibliográfica, o *método da morfologia* mostrou ser o mais eficiente. Portanto, adotamos este método para localizar a placa.

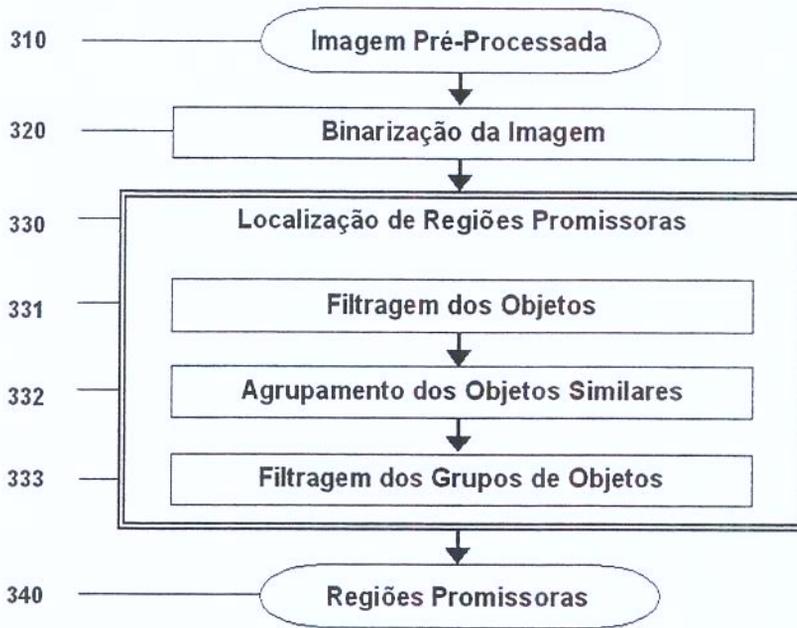


Fig. 4.4: Diagrama em bloco da localização da placa.

Desenvolvemos as duas etapas fundamentais do *método de morfologia* da seguinte maneira:

1. *Binarização da imagem* (bloco 320 do diagrama da figura 4.4): utiliza uma técnica de binarização para separar os caracteres dos demais objetos da imagem;
2. *Localização de regiões promissoras* (bloco 330 do diagrama da figura 4.4): utiliza informações morfológicas dos caracteres da placa para filtrar grupos de objetos visando encontrar as regiões promissoras.

O método que desenvolvemos para localizar a placa tem a vantagem de ser pouco sensível a padrões de textura e não precisar da localização correta da câmera ou padronização do tamanho da placa. Para isto, utiliza conjuntamente as propriedades de luminância, contraste e a morfologia dos caracteres (estimada dinamicamente). Nele constatamos nossa maior contribuição, o que levou ao depósito de uma patente pela UNICAMP (Lotufo and Dias, 2004).

Devido à possibilidade de existirem outros objetos na imagem com o mesmo conjunto de características da placa (tais como: prefixo do veículo, adesivos, modelo do veículo, marca do fabricante, etc.) resolvemos aplicar os demais módulos em cada região promissora encontrada, visando minimizar o problema de não encontrar a região correta. Isto está representado no bloco 400 do diagrama da figura 4.1.

As subseções seguintes descrevem detalhadamente cada etapa do módulo de *Localização da Placa*.

### 4.3.1 Binarização da Imagem

Na binarização da imagem, bloco 320 do diagrama da figura 4.4, pretendemos separar os objetos, de modo a ressaltar os caracteres da placa.

Em uma imagem composta de vários objetos, como a de um veículo, a *segmentação* da imagem, visando a localização da placa, é um processo de extrema complexidade.

Existem várias técnicas de segmentação, cada qual com suas características. De acordo com o nosso objetivo, podemos pensar na segmentação de maneira simplificada, ou seja, em separar os caracteres da placa dos demais objetos, esta técnica é denominada *binarização*. As principais técnicas de binarização utilizam a *limiarização*.

Devido à complexidade e importância na correta binarização da imagem do veículo, escolhemos 3 métodos para esta tarefa: *binarização global automática*, *binarização local adaptativa* e *binarização global no top-hat da imagem*. Também elaboramos um método que utilize informações complementares aos métodos anteriormente citados, o qual denominamos de *binarização dos mínimos regionais da imagem filtrada*. Em seguida descrevemos estes 4 métodos, tendo seus desempenhos analisados no capítulo 5.

Para descrever estes métodos, utilizaremos como referência a imagem obtida pelo módulo de *Pré-Processamento*, representada na figura 4.3, doravante denominada de imagem  $f$ . Visando facilitar o entendimento, apresentaremos um gráfico da binarização da linha 149 da imagem  $f$ , a qual é representada na figura 4.5.



Fig. 4.5: Representação da linha 149 da figura 4.3.

Uma imagem que contenha apenas objeto e fundo (ex.: imagem da placa de licenciamento veicular) pode ser facilmente binarizada por um método de *binarização global automática*, sendo que o método mais conhecido foi proposto por Otsu (1979). No qual a escolha do valor do limiar  $t$  pode ser facilmente determinado por métodos estatísticos, pois seu limiar depende somente dos *pixels* da imagem. Porém, imagens que contêm vários objetos (ex.: imagem de veículo) possuem vários limiares de separação de difícil determinação, dificultando o processo de binarização.

Podemos constatar esta dificuldade com base no histograma da imagem  $f$ , figura 4.6, a qual também representa o limiar de separação obtido pelo método de Otsu. Neste caso, o limiar conseguiu separar os caracteres do fundo da placa devido a grande quantidade de *pixels* escuros na imagem, principalmente pela cor do veículo, o que não acontece sempre.

O gráfico da binarização da linha 149 da imagem  $f$  pelo método de Otsu, figura 4.7, mostra a dificuldade em separar os caracteres da placa dos demais objetos com base em um único limiar do nível de cinza. Na figura 4.8, temos o resultado desta binarização.

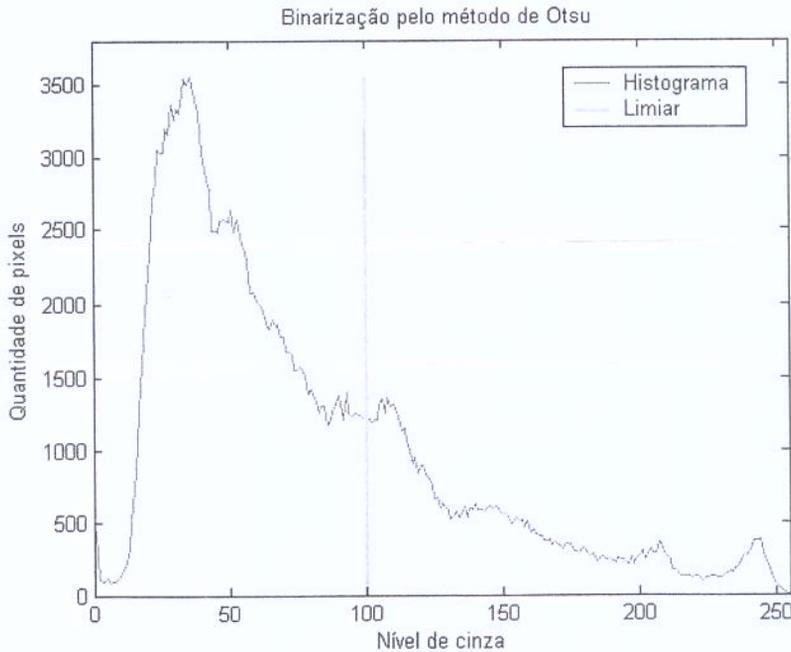


Fig. 4.6: Histograma da figura 4.3 com limiar determinado pelo método de Otsu.

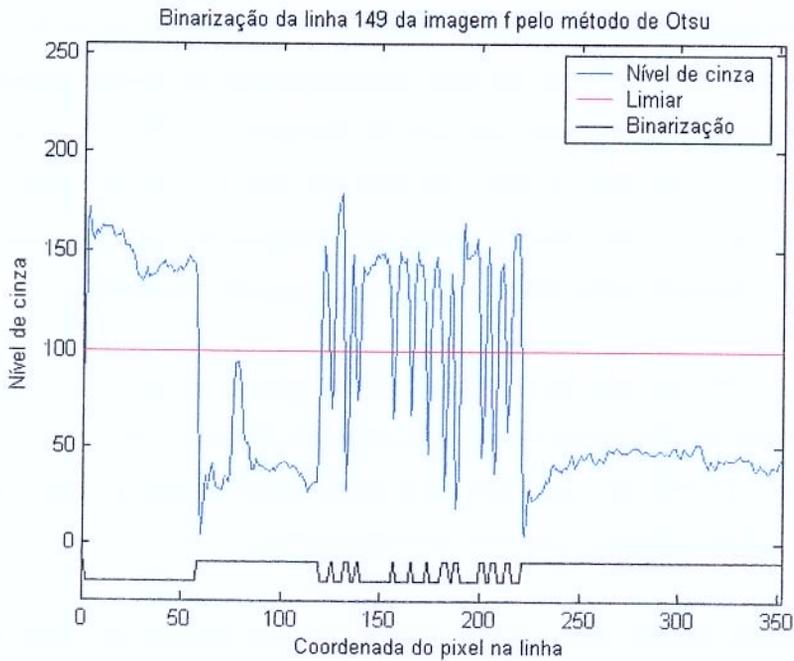


Fig. 4.7: Gráfico da binarização da linha 149 da figura 4.3 pelo método de Otsu.



Fig. 4.8: Binarização da figura 4.3 pelo método de Otsu.

Outro método utilizado é a **binarização local adaptativa**. A proposta mais conhecida é a de Niblack (1986). Este método consiste em definir, para cada *pixels* da imagem, um limiar baseado em sua vizinhança. Deste modo, permite determinar um *limiar dinâmico* baseado no nível de cinza.

A principal desvantagem desta técnica é de ser muito sensível à ruídos, o que é comum nas imagens utilizadas em SRPLV. A maioria das propostas utilizam uma vizinhança quadrada de 15x15

*pixels*, com coeficiente de proporcionalidade  $\alpha = -0,2$  (ponderação do desvio padrão).

O gráfico da binarização da linha 149 da imagem  $f$  pelo método de Niblack, figura 4.9, mostra como o limiar dinâmico binariza os caracteres da placa. A figura 4.10 mostra o resultado deste método aplicado na imagem  $f$ .

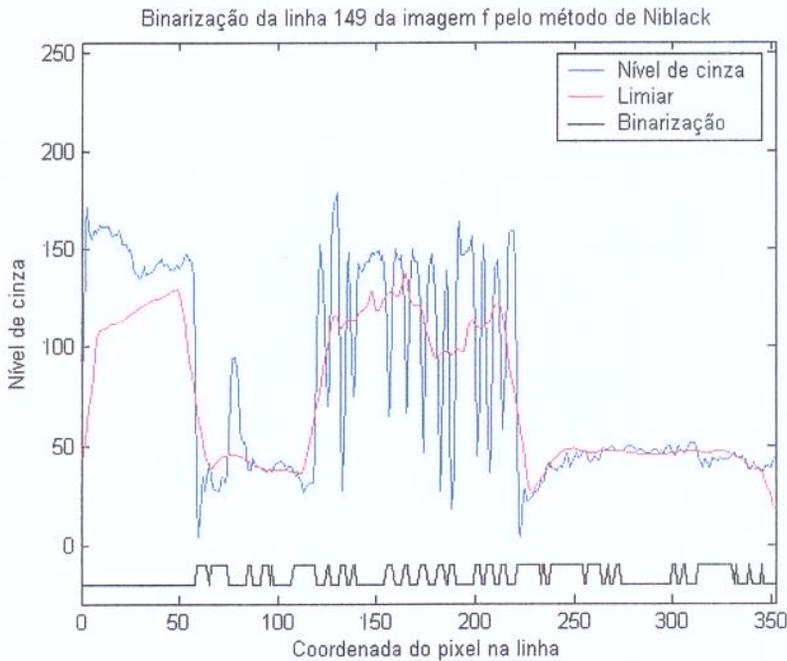


Fig. 4.9: Gráfico da binarização da linha 149 da figura 4.3 pelo método de Niblack.

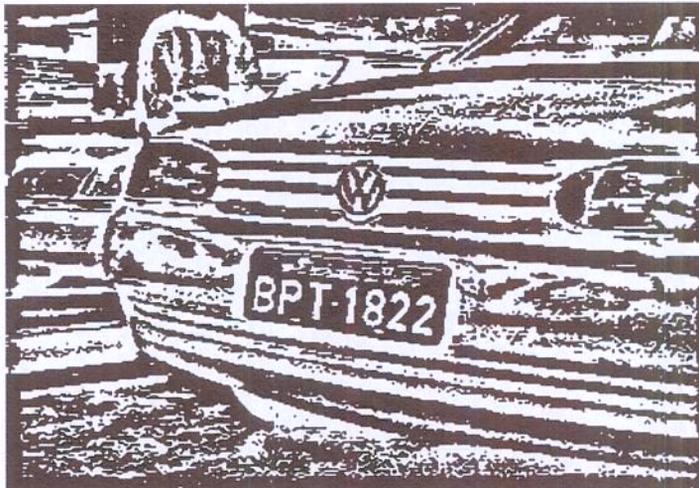
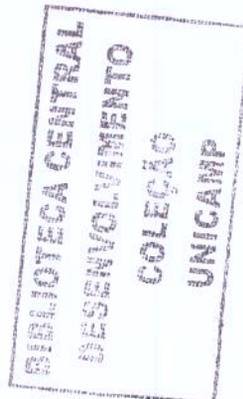


Fig. 4.10: Binarização da figura 4.3 pelo método de Niblack.



Recentemente, foi proposto por F. Martín et al. (2000) um método específico para binarizar as imagens de veículos visando separar os caracteres da placa dos demais objetos da imagem. Denominamos este método de *binarização global no top-hat da imagem*. Consiste em aplicar o operador de morfologia matemática denominado de *top-hat*, com um elemento estruturante um pouco maior que a espessura do caractere, na imagem do veículo; tendo seu resultado binarizado pelo método de Otsu.

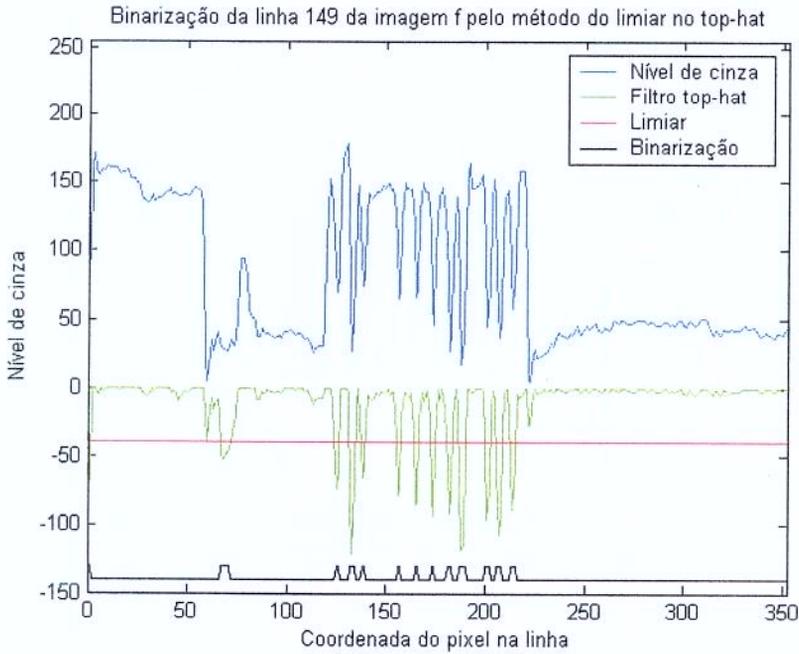


Fig. 4.11: Gráfico da binarização da linha 149 da figura 4.3 pelo método do top-hat.



Fig. 4.12: Binarização da figura 4.3 pelo método do top-hat.

O gráfico da binarização da linha 149 da imagem  $f$  pelo método do *top-hat*, figura 4.11, mostra como a imagem é binarizada com o limiar único, sendo este determinado pelo método de Otsu no resultado do operador *top-hat* aplicado na imagem  $f$ . A imagem resultante deste método é mostrada na figura 4.12.

Para elaborar o método de *binarização dos mínimos regionais da imagem filtrada*, nós consideramos que os caracteres são mais escuros do que a placa. Portanto é possível encontrá-los através dos valores *mínimos regionais* da imagem, podendo ser visto como uma estratégia de binarização independente de um valor de nível de cinza para a limiarização. Esta é a grande vantagem da técnica, pois para aplicar uma binarização por limiarização é necessário conhecer o valor do nível de cinza que separa o fundo do objeto. Com os mínimos regionais não temos este problema. No entanto, a maioria das imagens possuem um grande número de mínimos regionais, conforme ilustra a figura 4.13, isto se deve à presença de ruído durante o processo de aquisição. Deste modo é necessário aplicar um filtro de suavização de ruído que não modifique as bordas das regiões. Uma maneira de fazer isto é utilizando o filtro morfológico conexo denominado *v-minima*, o qual estende as regiões homogêneas da imagem.



Fig. 4.13: *Mínimos regionais da figura 4.3.*

Nosso método de binarização é efetuado da seguinte maneira:

Primeiramente é aplicado o filtro morfológico *v-minima* com 1500 de volume e conexidade com *elemento estruturante* em forma de cruz unitária (*vizinhança-N4*). O resultado da filtragem é apresentado na figura 4.14.



Fig. 4.14: Imagem resultante da aplicação do filtro v-minima na figura 4.3.



Fig. 4.15: Imagem de profundidade da região da placa: (a) antes; (b) depois de aplicar o filtro v-minima.

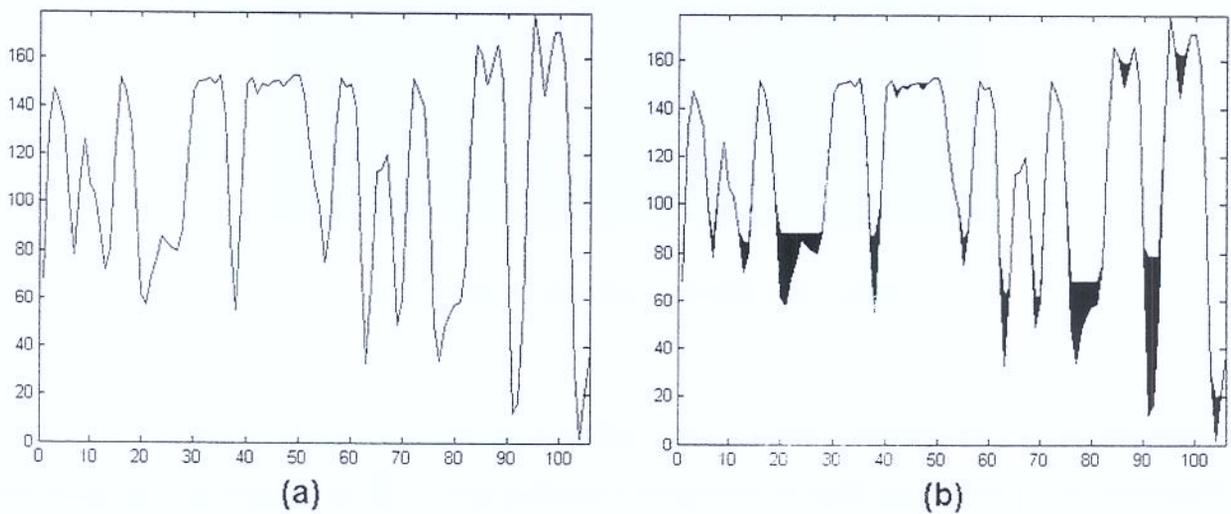


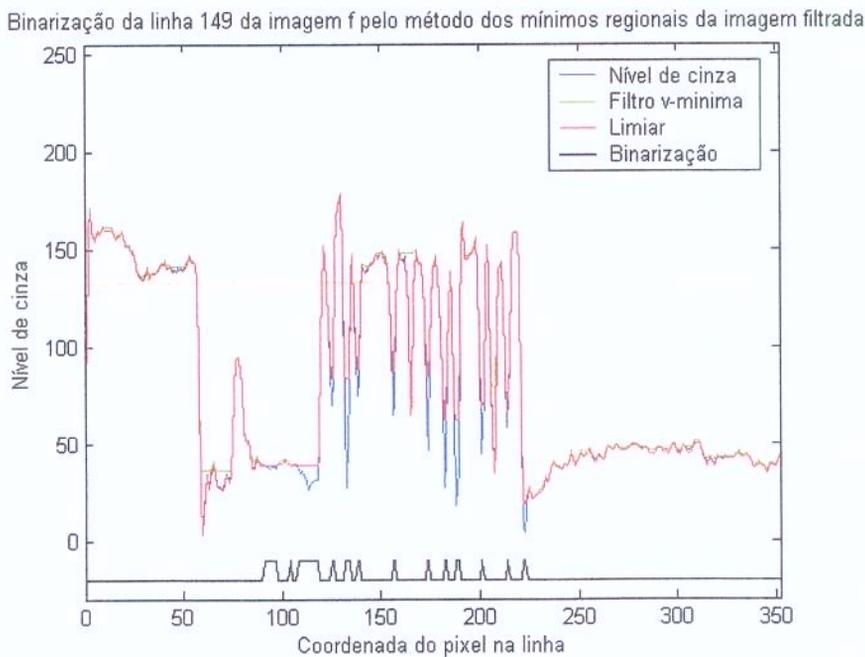
Fig. 4.16: Perfil de uma linha na região da placa: (a) sem; (b) com vales preenchidos pelo filtro v-minima.

Também é mostrado o preenchimento dos vales da imagem na região onde se encontra a placa (figura 4.15a e 4.15b), nela podemos constatar que o hífen e os furos de fixação da placa foram completamente preenchidos, sendo que a tarjeta de identificação da localidade do veículo foi parcialmente preenchida. O gráfico do perfil de uma linha desta região é ilustrado na figura 4.16a e o preenchimento dos vales por esse filtro na figura 4.16b.

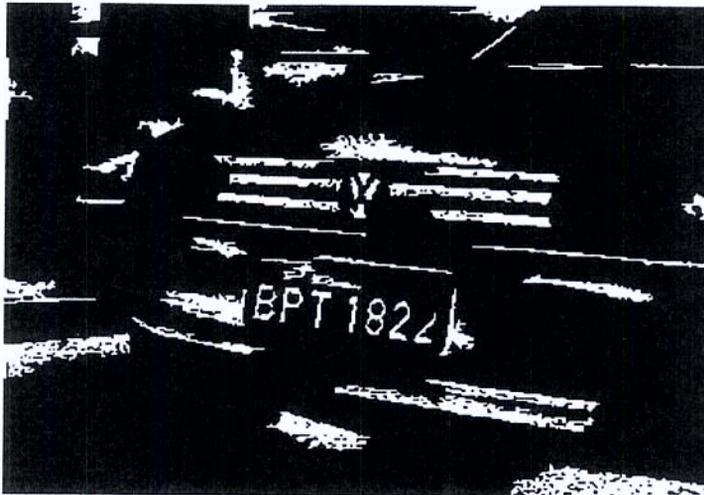
Em seguida, é aplicado o filtro de mínimo regional com elemento estruturante em forma de caixa unitária (*vizinhança-N8*). É importante não confundir o conceito de *mínimo regional* com *mínimo local*, sendo que o primeiro é uma propriedade regional e o segundo é uma propriedade pontual.

O gráfico da binarização da linha 149 da imagem  $f$  por nosso método, figura 4.17, mostra como a binarização é feita utilizando o limiar dinâmico definido pelos mínimos regionais da imagem resultante após a aplicação do filtro *v-minima*.

Como resultado da binarização, figura 4.18, obtemos os objetos compostos por mínimos regionais, tendo volume maior ou igual a 1500. Este volume foi calculado empiricamente, utilizando imagens com resolução de 320x240 e 640x480 que contenham todo o veículo. Portanto, para outras resoluções ou imagens que contenham apenas parte do veículo, é necessário estimar um novo valor de volume.



**Fig. 4.17:** Gráfico da binarização da linha 149 da figura 4.3 pelo método dos mínimos regionais da imagem filtrada.



**Fig. 4.18:** Binarização da figura 4.3 pelo método dos mínimos regionais da imagem filtrada.

Os 4 métodos de binarização apresentados utilizam informações complementares. Por esta razão, eles podem ser utilizados em conjunto, visando a correta binarização nas diversas condições de aquisição da imagem. No capítulo 5 faremos uma análise da eficiência com as possíveis combinações destes métodos.

### 4.3.2 Localização de Regiões Promissoras

A localização de regiões promissoras, bloco 330 do diagrama da figura 4.4, pretende encontrar as regiões com probabilidade de conter uma placa de licenciamento veicular.

A imagem binarizada ainda contém objetos que não são os caracteres da placa de licenciamento veicular. Portanto, é necessário aplicar filtros para eliminar os objetos indesejados. Existem várias técnicas para filtrar uma imagem binarizada de veículo visando obter como resultado apenas os caracteres da placa. A maioria dos autores utilizam conhecimento a priori da morfologia (geometria e posição) dos caracteres da placa para filtrá-los.

Algumas propostas estimam empiricamente a altura do caractere. Porém, a resolução espacial da imagem da placa de licenciamento depende de vários fatores, dentre eles podemos citar: distância do carro à câmera, distância focal, etc. Portanto, estimar empiricamente a altura do caractere implica em parametrizar tais fatores, exigindo uma configuração do SRPLV de acordo com o ambiente no qual irá operar.

Devido à heterogeneidade de ambientes utilizados em aplicações deste sistema, elaboramos um método auto-ajustável com base na propriedade de que todos os caracteres da placa possuem altura e

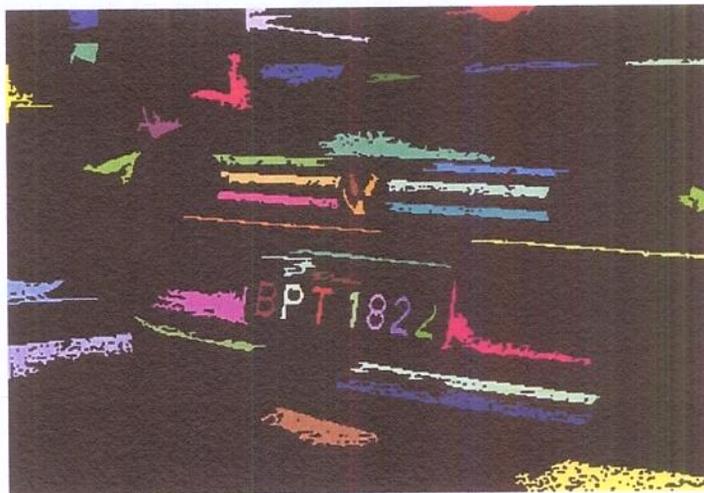
espaçamento regular, sendo alinhados horizontalmente.

Conforme o bloco 330 do diagrama da figura 4.4, este processo é dividido em 3 etapas:

1. *Filtragem dos Objetos* (bloco 331 do diagrama da figura 4.4), a qual utiliza a altura mínima e a razão de aspecto para filtrar os objetos;
2. *Agrupamento dos Objetos Similares* (bloco 332 do diagrama da figura 4.4), nesta etapa são formados agrupamentos com base na altura, espaçamento horizontal e vertical dos objetos, visando agrupar objetos com altura e espaçamento regular alinhados horizontalmente;
3. *Filtragem dos Grupos de Objetos* (bloco 333 do diagrama da figura 4.4), consiste em filtrar os grupos de acordo com a quantidade de objetos.

A primeira etapa consiste na *rotulação* da imagem binarizada, a qual atribui uma cor diferente para cada componente conexo, sendo extraídas as seguintes características de cada um deles: altura, largura, coordenada vertical e horizontal do centróide. Doravante denominaremos de objetos os componentes conexos da imagem binarizada.

A figura 4.19 ilustra o resultado da rotulação da imagem binarizada (figura 4.18).



**Fig. 4.19:** Rotulação da imagem binarizada (figura 4.18).

As subseções seguintes explicam em detalhe cada uma das 3 etapas de filtragem.

### 4.3.3 Filtragem dos Objetos

A filtragem dos objetos, bloco 331 do diagrama da figura 4.4, filtra os objetos baseado em sua altura mínima e na razão de aspecto.

Este filtro remove da imagem rotulada (figura 4.19) todos os objetos com largura maior que o dobro da altura ou com altura menor do que 8, pois um caractere com altura inferior a 8 não é reconhecido corretamente. A figura 4.20 mostra o resultado deste processo. O conjunto de objetos resultante ( $F$ ) é representado pela equação 4.1:

$$F = \{O_i | (W_i \leq 2H_i) \wedge (H_i \geq 8), \text{ para } i \in [1, n]\} \quad (4.1)$$

onde:

$O_i$ : é o  $i$ -ésimo objeto da imagem rotulada;

$W_i$ : é a largura do  $i$ -ésimo objeto da imagem rotulada;

$H_i$ : é a altura do  $i$ -ésimo objeto da imagem rotulada;

$n$ : é a quantidade de objetos da imagem rotulada.

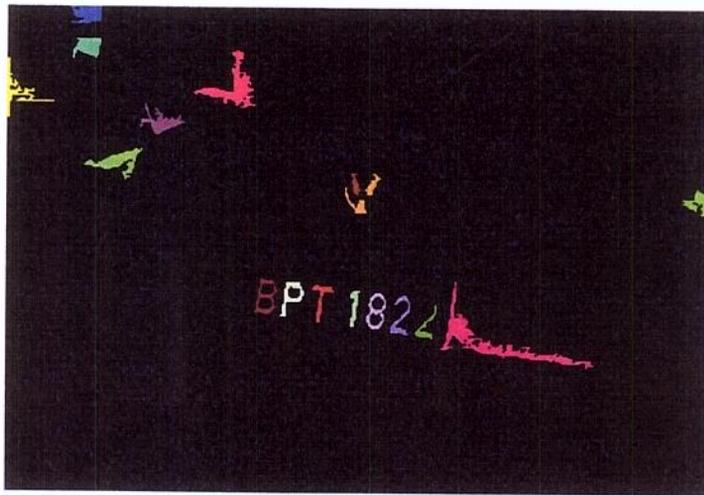


Fig. 4.20: Imagem resultante após a filtragem de objetos da figura 4.19.

#### 4.3.4 Agrupamento dos Objetos Similares

Nesta etapa, bloco 332 do diagrama da figura 4.4, são agrupados os objetos com altura e espaçamento regular alinhados horizontalmente.

Os agrupamentos são definidos de acordo com as seguintes propriedades morfológicas:

- altura com desvio máximo de  $\pm 15\%$ ;
- posição vertical do centróide com desvio máximo de  $\pm 1/3$  da altura;
- posição horizontal do centróide com desvio máximo de  $\pm$  o dobro da altura.

Para compor os grupos, cada um dos objetos da imagem filtrada (figura 4.20) é utilizado como referência para detectar quais objetos são similares a ele. Se um ou mais objetos detectados pertencerem à outro grupo, então estes grupos serão unidos resultando em um único grupo. Esta tarefa é realizada conforme a equação 4.2:

$$G_j = \{F_i | ((H_i \leq 1,15H_j) \wedge (H_i \geq 0,85H_j)) \wedge (|Y_i - Y_j| \leq 1/3H_j) \wedge (|X_i - X_j| \leq 2H_j), \text{ para } i \in [1, n]\} \quad (4.2)$$

onde:

$G_j$ : é o  $j$ -ésimo grupo formado;

$H_j$ : é a altura do  $j$ -ésimo objeto da imagem filtrada ( $F$ );

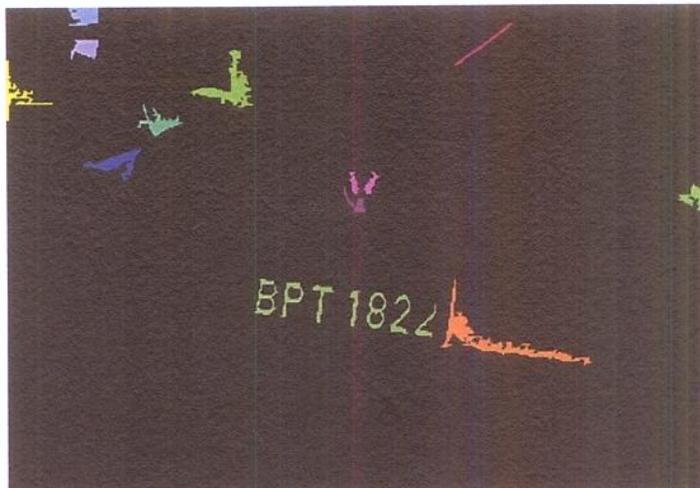
$H_i$ : é a altura do  $i$ -ésimo objeto da imagem filtrada ( $F$ );

$Y_i$ : é a posição vertical do centróide do  $i$ -ésimo objeto da imagem filtrada ( $F$ );

$X_i$ : é a posição horizontal do centróide do  $i$ -ésimo objeto da imagem filtrada ( $F$ );

$n$ : é a quantidade de objetos da imagem filtrada ( $F$ ).

Cada agrupamento recebe um rótulo diferente, de modo que os objetos tenham o mesmo rótulo do agrupamento a que pertencem. A figura 4.21 mostra os objetos rotulados de acordo com seu agrupamento.



**Fig. 4.21:** Objetos da figura 4.20 rotulados de acordo com o agrupamento a que pertencem.

### 4.3.5 Filtragem dos Grupos de Objetos

Esta etapa, bloco 333 do diagrama da figura 4.4, utiliza a quantidade determinada de objetos, para localizar os grupos de objetos com as características morfológicas da cadeia de caracteres da placa.

A imagem de um veículo normalmente possui vários grupos de objetos (Ex: cadeia de caracteres da placa do veículo, tarjeta de identificação, marca do veículo, modelo do veículo, adesivos, etc). Portanto, é importante localizar os grupos de objetos mais semelhantes à cadeia de caracteres da placa. Uma característica importante da placa de licenciamento é a quantidade de objetos. Esta informação permite desenvolver um filtro que retorne os grupos de objetos acima de uma quantidade mínima e abaixo de uma máxima. Lembramos que não utilizamos a quantidade exata devido à possibilidade de alguns objetos estarem unidos a outro ou que alguns objetos tenham sido eliminados na binarização (devido a ruídos aditivos ou subtrativos), o que causaria uma falha no critério de quantidade exata.

Para elaborar este filtro, consideramos que a cadeia de caracteres da placa tenha no mínimo 3 e no máximo 10 objetos, pois alguns caracteres podem estar unidos e outros objetos podem ser confundidos com caracteres. A equação 4.3 representa este filtro:

$$G' = \{G_i | (3 \leq \text{card}(G_i) \leq 10), \text{ para } i \in [1, n]\} \quad (4.3)$$

onde:

$G_i$ : é o  $i$ -ésimo grupo formado;

$\text{card}$ : é a cardinalidade de conjunto;

$n$ : é a quantidade de grupos no conjunto  $G$ .



Fig. 4.22: Grupos da figura 4.21 filtrados de acordo com a quantidade de objetos.

A figura 4.22 mostra os grupos resultantes, os quais são utilizados para definir as regiões promissoras.

### 4.3.6 Resultado do Módulo de Localização da Placa

Como resultado da etapa anterior, temos uma imagem com os grupos de objetos, cada qual definido por componentes conexos de mesmo rótulo, doravante iremos denominar as regiões destes grupos de *regiões promissoras*.

O módulo de *Enquadramento da Placa* necessita que o retângulo delimitador da região promissora envolva as arestas da placa de licenciamento veicular, de modo que as arestas horizontais sejam maiores que  $1/3$  da largura desta imagem e as verticais maiores que  $1/5$  da altura. Portanto torna-se necessário manipular cada região promissora de modo a satisfazer tal exigência.

Na figura 4.23, o resultado é representado através de um retângulo envolvendo a região promissora encontrada.



Fig. 4.23: Imagem com as regiões promissoras delimitadas por um retângulo.

## 4.4 Enquadramento da Placa

O bloco 500 do diagrama da figura 4.1, tem a finalidade de enquadrar a placa de licenciamento veicular e corrigir a distorção de perspectiva da placa. Este bloco está detalhado na figura 4.24, sendo descrito nesta seção.

A imagem da região promissora, obtida no módulo de *Localização da Placa*, não contém apenas a placa, mas também partes do veículo, pois na maioria das vezes a placa está levemente inclinada. Isso

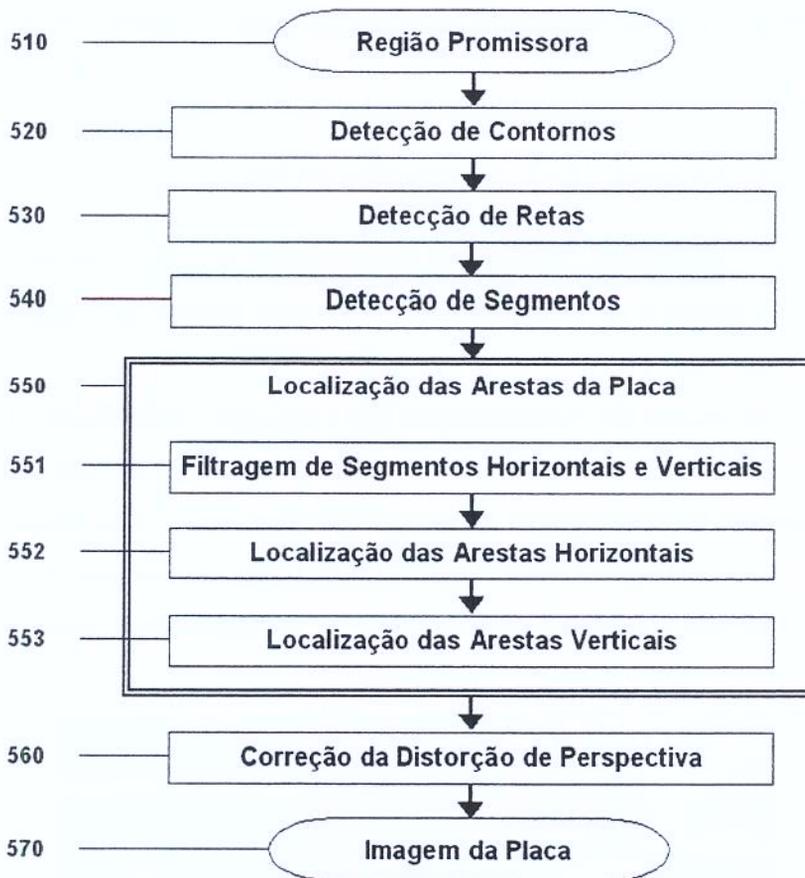


Fig. 4.24: Diagrama em bloco do enquadramento da placa.

ocorre devido a câmera não ficar posicionada de modo ortonormal à placa do veículo, ocasionando um efeito denominado *distorção de perspectiva*. A maioria das câmeras ficam posicionadas no lado esquerdo ou direito da passagem do veículo. Deste modo, para obter a imagem da placa é necessário rotacionar a câmera em torno de dois eixos: panorâmico e inclinação, conhecidos tecnicamente por *pan* e *tilt*. Deste modo temos 2 graus de liberdade, podendo ocasionar: rotação da placa, inclinação dos caracteres e o efeito trapézio. Estes efeitos prejudicam as etapas de binarização da placa, separação e reconhecimento dos caracteres.

Portanto, a correção da distorção de perspectiva e o correto enquadramento da placa implicam diretamente em melhorias fundamentais nos demais módulos.

Nossa proposta visa corrigir a distorção de perspectiva com o uso da *homografia*, sendo o enquadramento feito eficientemente na imagem corrigida.

Para aplicar a homografia é necessário conhecer 4 pontos da imagem distorcida e seus respectivos pontos na imagem corrigida. Como a placa de licenciamento é formada por um plano retangular, é

possível obter os 4 cantos do retângulo, os quais são coplanares, reduzindo nossa tarefa a duas dimensões. O principal problema é encontrar os 4 cantos da placa. Para isso, utilizamos as arestas da placa, pois os 4 cantos são formados pelas interseções das arestas 2 a 2. Para encontrar as arestas, utilizamos o método de *detecção de contorno de Canny* e o método de *detecção de retas* pela *transformada de Hough*.

#### 4.4.1 Região Promissora

A região promissora, bloco **510** do diagrama da figura 4.24, é obtida do módulo de *Localização da Placa*, figura 4.25. Para encontrar as arestas da placa é necessário que as mesmas apareçam por completo na imagem. Devido a critérios, definidos empiricamente, utilizados durante o processo de enquadramento é necessário que as arestas horizontais sejam maiores que 1/3 da largura desta imagem e as verticais maiores que 1/5 da altura.



Fig. 4.25: Imagem de uma região promissora.

Vários filtros e operadores utilizam como parâmetro valores espaciais (ex.: largura, altura, área). Devido à aquisição de imagens da placa com diferentes *resoluções espaciais*, é necessário normalizar sua resolução para um determinado padrão. Como a imagem da região promissora pode ter diferentes razões de aspecto, escolhemos a altura da mesma como parâmetro de normalização geométrica. Empiricamente, nós definimos o valor padrão da altura em 150 *pixels*. Desenvolvemos este processo da seguinte maneira:

Primeiramente é calculado o valor do **fator de normalização geométrica**, definido na equação 4.4:

$$Fator = \frac{150}{h} \quad (4.4)$$

onde:  $h$  é a altura da imagem da região promissora.

Em seguida é aplicada a transformação geométrica, apresentada na equação 4.5:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} Fator & 0 & 0 \\ 0 & Fator & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (4.5)$$

onde:

$x$  e  $y$ : são, respectivamente, as coordenadas da coluna e linha da imagem original;

$x'$  e  $y'$ : são, respectivamente, as coordenadas da coluna e linha da imagem transformada.

#### 4.4.2 Detecção de Contornos

Este módulo, bloco **520** do diagrama da figura 4.24, tem a tarefa de fornecer uma imagem binária que contenha apenas os contornos dos objetos ao operador de detecção de retas, bloco **530** do mesmo diagrama.

Para encontrar as arestas da placa é necessário obter o contorno da placa, o qual depende do fundo do objeto que está envolta da mesma, podendo ser de cor diferente da placa, facilitando a detecção do contorno ou de cor igual a da placa, dificultando ou até mesmo impossibilitando a detecção. Outra dificuldade é a presença de sombras e objetos sobrepostos à placa.

Devido às dificuldades existentes, adotamos o método de *detecção de contorno de Canny*.

Os contornos dos caracteres da placa prejudicam a identificação das arestas, deste modo é importante minimizar este problema antes de aplicar o detector de contornos. Para isso são utilizados dois operadores de morfologia matemática: *fechamento reconstrutivo* e *filtro alternado seqüencial reconstrutivo (ASF)*. O primeiro operador preenche regiões homogêneas menores que o elemento estruturante utilizado com os valores dos *pixels* da região na qual está contido. O segundo operador utiliza o conceito de *zonas planas* para simplificar a imagem, expandindo regiões homogêneas.

Portanto, primeiramente é aplicado o operador de fechamento reconstrutivo, com elemento estruturante em forma de disco com raio 7, na região promissora. Em seguida, aplicado o filtro alternado seqüencial reconstrutivo na imagem resultante, tendo como seqüência de operadores: *abertura* e *fechamento* morfológicos, ambos com elemento estruturante em forma de disco com raio unitário, aplicado durante 3 iterações.

O resultado desta etapa pode ser visualizado na figura 4.26, na qual constatamos a remoção dos caracteres e demais objetos da placa.

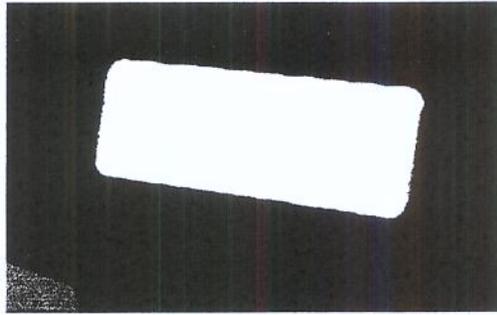


Fig. 4.26: Região promissora após aplicar os filtros morfológicos de reconstrução na figura 4.25.

Em seguida é aplicado o detector de contornos de Canny com  $\sigma = 1$  e *threshold* automático. A figura 4.27 mostra o resultado deste operador.

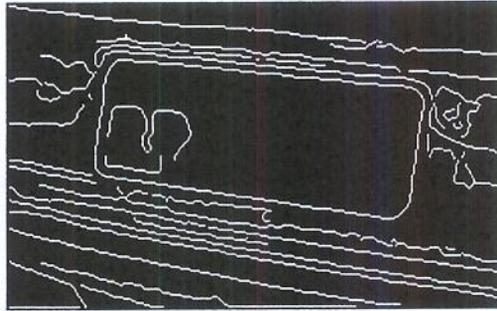


Fig. 4.27: Região promissora após aplicar o detector de contornos de Canny na figura 4.26.

Para remover os ruídos que atrapalham a detecção das retas, é aplicado o operador morfológico de *abertura por área* com elemento estruturante em forma de caixa unitária (vizinhança-N8) eliminando componentes conexos com área menor que 50. O resultado deste operador pode ser visualizado na figura 4.28.

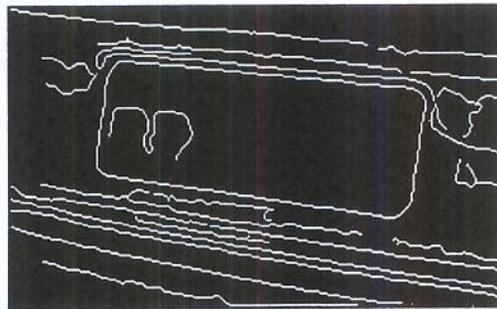


Fig. 4.28: Região promissora após aplicar o operador morfológico de abertura por área na figura 4.27.

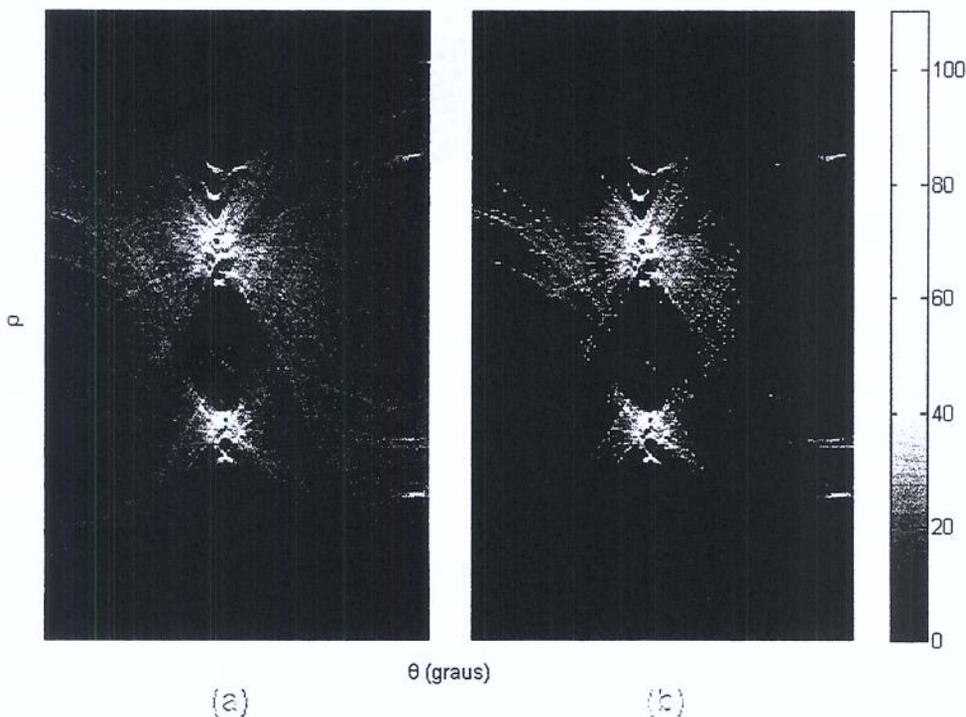
### 4.4.3 Detecção de Retas

Desenvolvemos o operador de detecção de retas, bloco 530 do diagrama da figura 4.24, da seguinte maneira:

Primeiro é aplicada a *transformada de Hough*, sendo o espaço de parâmetros delimitado entre ângulos de 1 a 180 graus e posição da reta entre a altura da região promissora, ambos com passo unitário.

Em seguida é necessário encontrar os picos, representantes das retas na imagem. Para isso os valores obtidos pela transformada de Hough são limiarizados por um valor definido empiricamente, onde somente são considerados os valores menores que 20. Em seguida é aplicado o filtro morfológico de *máximo regional* com elemento estruturante em forma de retângulo com 21 de base e 7 de altura.

O resultado da transformada de Hough pode ser visualizado na figura 4.29a e o resultado do filtro na figura 4.29b.



**Fig. 4.29:** Transformada de Hough da figura 4.28: (a) antes da limiarização; (b) depois da limiarização.

Cada máximo regional de coordenadas  $(\theta, \rho)$  representa uma reta de equação:  $\rho = x \cos \theta + y \sin \theta$ . A figura 4.30 mostra todas as retas encontradas neste processo.

Uma reta é definida por sua posição e orientação no espaço, sendo infinita em ambas as dire-

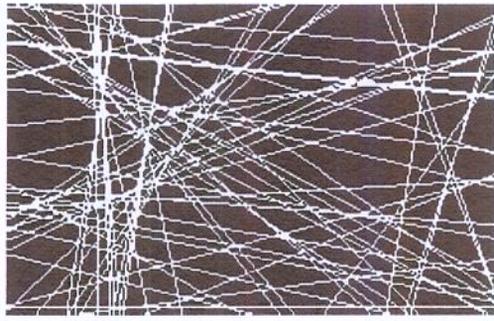


Fig. 4.30: Retas detectadas na figura 4.28.

ções. Porém, como estamos trabalhando em um espaço finito (região promissora), as retas iniciam e terminam nas bordas da imagem.

#### 4.4.4 Detecção de Segmentos

Encontrar as 4 retas que representam as arestas da placa a partir do conjunto de retas da região promissora, figura 4.30, é uma tarefa difícil. Como os objetos não são formados por retas, mas por segmentos de retas, é necessário convertê-las para os respectivos segmentos de retas representantes dos objetos da região promissora. Esta operação está representada no bloco 540 do diagrama da figura 4.24. Desenvolvemos isto da seguinte maneira:

Primeiramente, a imagem obtida pelo detector de contornos de Canny e com ruídos filtrados pelo operador morfológico de *abertura por área*, figura 4.28, é rotulada utilizando um elemento estruturante em forma de caixa unitária (vizinhança- $N_8$ ). Em seguida, é aplicado o operador morfológico de *dilatação* com elemento estruturante em forma de disco com raio 2. O resultado desta operação pode ser visualizado na figura 4.31.

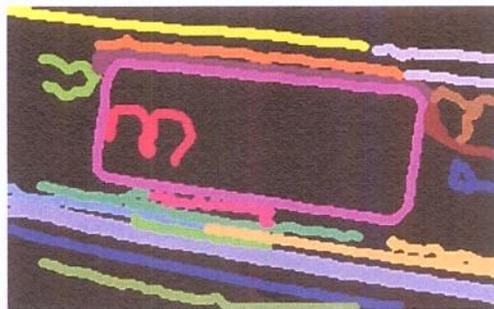


Fig. 4.31: Contorno da região promissora, na imagem da figura 4.28, rotulado e dilatado.

Para obtermos os segmentos, a imagem rotulada e dilatada (figura 4.31) é projetada nos *pixels* pertencentes às retas da figura, resultando na imagem ilustrada pela figura 4.32

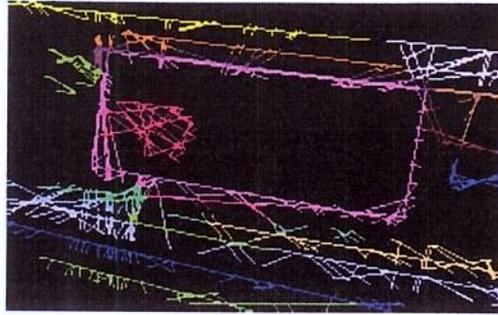


Fig. 4.32: Segmentos rotulados de acordo com o objeto a que pertence.

Os segmentos são extraídos de cada reta, sendo compostos pelos *pixels* de mesmo rótulo, podendo ter no máximo 30 *pixels* de rótulos diferentes entre eles. O conjunto  $S$  representa os segmentos detectados, conforme a equação 4.6:

$$S_i = \{(x_{1_i}, y_{1_i}), (x_{2_i}, y_{2_i}), o_i\} \quad (4.6)$$

onde:

- $x_{1_i}$ : é a abscissa do ponto inicial do *i-ésimo* segmento;
- $y_{1_i}$ : é a ordenada do ponto inicial do *i-ésimo* segmento;
- $x_{2_i}$ : é a abscissa do ponto final do *i-ésimo* segmento;
- $y_{2_i}$ : é a ordenada do ponto final do *i-ésimo* segmento;
- $o_i$ : é o rótulo do objeto a que pertence o *i-ésimo* segmento.

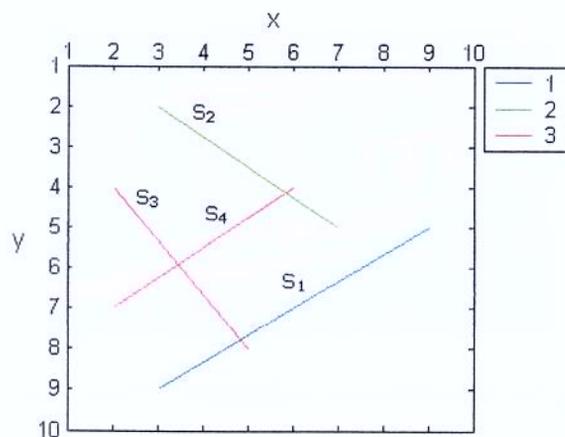


Fig. 4.33: Exemplo de segmentos rotulados.

A figura 4.33 exemplifica os segmentos rotulados, os quais são armazenados na lista de segmentos  $S$  da seguinte maneira:

$$S_1 = \{(3, 9), (9, 5), 1\}$$

$$S_2 = \{(3, 2), (7, 5), 2\}$$

$$S_3 = \{(2, 4), (5, 8), 3\}$$

$$S_4 = \{(2, 7), (6, 4), 3\}$$

#### 4.4.5 Localização das Arestas da Placa

Este conjunto de operações, representado pelo bloco 550 do diagrama da figura 4.24 tem a tarefa de encontrar as 4 arestas da placa de licenciamento veicular, sendo a tarefa de maior importância do módulo de *Enquadramento da Placa*.

Muitos dos segmentos encontrados não pertencem às arestas da placa. Assim, é necessário filtrá-los de modo a minimizar a quantidade de segmentos candidatos a arestas. Para isto desenvolvemos um filtro que separa os segmentos de acordo com sua orientação (horizontal ou vertical), além de filtrá-los por tamanho mínimo estipulado empiricamente. Como resultado temos o conjunto  $H$  contendo os segmentos horizontais e o conjunto  $V$  contendo os segmentos verticais.

Devido à grande quantidade de combinações de segmentos para compor as 4 arestas da placa, dividimos esta tarefa em duas. A primeira consiste em encontrar as arestas horizontais. Caso não sejam encontradas, o módulo de *Enquadramento da Placa* falha por completo, necessitando de um método complementar. A segunda etapa consiste em encontrar as arestas verticais. Por terem comprimento reduzido, as arestas verticais estão mais sujeitas a não serem detectadas. Visando minimizar este problema, utilizamos as arestas horizontais da placa, encontradas na etapa anterior, para estipular pares de segmentos verticais “virtuais”, os quais são utilizados na procura das arestas verticais, no caso de eventual falta de uma ou das duas arestas.

Normalmente, a imagem da região promissora é composta por vários segmentos, dificultando a tarefa de encontrar as arestas da placa. Deste modo, é necessário obter informações da imagem corrigida de cada quadrilátero do conjunto  $Q$ , “penalizando” proporcionalmente quanto as diferenças em relação às características conhecidas da placa de licenciamento veicular. Para isto definimos duas funções-objetivo, uma para encontrar as arestas horizontais e outra para encontrar as arestas verticais.

Resumidamente, tanto as arestas horizontais quanto as arestas verticais da placa são encontradas da seguinte maneira:

1. São formados quadriláteros com combinações de 2 segmentos horizontais e 2 verticais;
2. Para cada quadrilátero é aplicada uma função-objetivo da seguinte maneira:
  - (a) A distorção de perspectiva da imagem envolvida pelo quadrilátero é corrigida, resultando na imagem corrigida;
  - (b) Da imagem corrigida são extraídas características;
  - (c) As características da imagem corrigida são utilizadas para “*penalizar*” o quadrilátero de acordo com a diferença das características conhecidas de uma placa de licenciamento veicular, sendo esta “*penalidade*” o valor retornado pela função-objetivo;
3. O par de arestas da placa procurado é obtido do quadrilátero com menor valor retornado pela função-objetivo.

A maneira de formar os quadriláteros, a função-objetivo e as características da imagem por ela utilizada são específicas para cada orientação (horizontal ou vertical) do par de arestas a ser encontrado. Portanto são descritos em suas respectivas subseções: **Localização das Arestas Horizontais** e **Localização das Arestas Verticais**.

A correção da distorção de perspectiva é utilizada para encontrar o par de arestas horizontais, o par de arestas verticais e para corrigir a imagem envolvida pelas quatro arestas encontradas. Portanto explicaremos esta correção na subseção seguinte.

### Correção da Distorção de Perspectiva

Os quadriláteros, conjunto  $Q$ , são definidos por 4 segmentos, os quais representam suas arestas, na seguinte ordem: horizontal superior, vertical esquerdo, horizontal inferior e vertical direito, conforme a equação 4.7:

$$Q_i = \{h_{sup}, v_{esq}, h_{inf}, v_{dir}\} \quad (4.7)$$

onde:

- $i$ : é o  $i$ -ésimo quadrilátero do conjunto  $Q$ ;
- $h_{sup}$ : é um segmento horizontal, diferente de  $h_{inf}$ ;
- $v_{esq}$ : é um segmento vertical, diferente de  $v_{dir}$ ;
- $h_{inf}$ : é um segmento horizontal, diferente de  $h_{sup}$ ;
- $v_{dir}$ : é um segmento vertical, diferente de  $v_{esq}$ .

Para corrigir a distorção de perspectiva é utilizada a **homografia**, a qual foi descrita na subseção 2.3.6 do capítulo 2.

Os pontos da imagem de entrada correspondem aos vértices do quadrilátero atual, os quais podem ser encontrados pelo método de *Cramer*.

Dados os coeficientes da equação da reta ( $a_1, b_1, a_2$  e  $b_2$ ) no formato inclinação e posição ( $y = ax + b$ ), as coordenadas dos vértices são encontradas conforme as equações 4.8 e 4.9:

$$x_v = \frac{\det \begin{bmatrix} b_1 & 1 \\ b_2 & 1 \end{bmatrix}}{\det \begin{bmatrix} -a_1 & 1 \\ -a_2 & 1 \end{bmatrix}} \quad (4.8)$$

$$y_v = a_1 x_v + b_1 \quad (4.9)$$

onde:

$x_v$ : é a abscissa de cada vértice do quadrilátero;

$y_v$ : é a ordenada de cada vértice do quadrilátero;

$a_1$  e  $a_2$ : são os coeficientes de inclinação de duas retas concorrentes;

$b_1$  e  $b_2$ : são os coeficientes de posição de duas retas concorrentes.

Para corrigir a distorção de perspectiva é necessário vincular os pontos da imagem original com os pontos da imagem corrigida. Portanto é necessário que seus vértices estejam na seguinte ordem:

1. Intersecção da reta horizontal superior com a reta vertical esquerda;
2. Intersecção da reta horizontal inferior com a reta vertical esquerda;
3. Intersecção da reta horizontal inferior com a reta vertical direita;
4. Intersecção da reta horizontal superior com a reta vertical direita;

Os conjuntos  $X$  e  $Y$ , equações 4.10 e 4.11, representam as coordenadas dos vértices de cada quadrilátero na imagem original.

$$X = \{x_v(h_{sup} \cap v_{esq}), x_v(h_{inf} \cap v_{esq}), x_v(h_{inf} \cap v_{dir}), x_v(h_{sup} \cap v_{dir})\} \quad (4.10)$$

$$Y = \{y_v(h_{sup} \cap v_{esq}), y_v(h_{inf} \cap v_{esq}), y_v(h_{inf} \cap v_{dir}), y_v(h_{sup} \cap v_{dir})\} \quad (4.11)$$

Para determinar os pontos da imagem corrigida, utilizamos o conhecimento a priori de que a placa de licenciamento é retangular, sendo que existem em diferentes razões de aspecto. Deste modo, é necessário estimar a altura ( $h_{calc}$ ) e largura ( $w_{calc}$ ) da placa corrigida para não ocorrer distorção, conforme as equações 4.13 e 4.12:

$$w_{calc} = \max (\| (X_4, Y_4) - (X_1, Y_1) \|, \| (X_3, Y_3) - (X_2, Y_2) \|) \quad (4.12)$$

$$h_{calc} = \max (\| (X_2, Y_2) - (X_1, Y_1) \|, \| (X_3, Y_3) - (X_4, Y_4) \|) \quad (4.13)$$

onde:

$w_{calc}$ : é a largura estimada;

$h_{calc}$ : é a altura estimada;

$X_i$ : é a abscissa do  $i$ -ésimo ponto na imagem original;

$Y_i$ : é a ordenada do  $i$ -ésimo ponto na imagem original;

Os pontos são calculados de modo a transformar um quadrilátero irregular em regular, ou seja, em um retângulo. Portanto, os pontos da imagem corrigida são representados pelos conjuntos  $U$  e  $V$ , conforme as equações 4.14 e 4.15:

$$U = \{1, 1, w_{est}, 1 + w_{est}\} \quad (4.14)$$

$$V = \{1, 1 + h_{est}, 1 + h_{est}, 1\} \quad (4.15)$$

### Filtragem de Segmentos Horizontais e Verticais

A filtragem dos segmentos horizontais e verticais, bloco 551 do diagrama da figura 4.24, consiste em separar os segmentos verticais dos horizontais de acordo com sua orientação, sendo em seguida filtrados por tamanho mínimo definido empiricamente. Desenvolvemos este filtro da seguinte maneira:

Primeiramente, os segmentos são associados ao conjunto  $H$  ou  $V$  de acordo com sua orientação, onde definimos que os segmentos horizontais possuem inclinação menor que  $45^\circ$  e os verticais têm inclinação maior ou igual a  $45^\circ$ .

Em seguida, os segmentos horizontais com largura menor  $1/5$  da altura da região promissora e os segmentos verticais com altura menor que  $1/3$  da largura da região promissora são removidos de seus respectivos conjuntos. Este procedimento está representado nas equações 4.18 e 4.19:

$$dx_i = \{(|x_{2_i} - x_{1_i}| + 1) \mid x_{1_i} \text{ e } x_{2_i} \in S_i\} \quad (4.16)$$

$$dy_i = \{(|y_{2_i} - y_{1_i}| + 1) \mid y_{1_i} \text{ e } y_{2_i} \in S_i\} \quad (4.17)$$

$$H = \{S_i \mid ((dy_i < dx_i) \wedge (dx_i \geq 1/3w)), \text{ para } i \in [1, n]\} \quad (4.18)$$

$$V = \{S_i \mid ((dy_i \geq dx_i) \wedge (dy_i \geq 1/5h)), \text{ para } i \in [1, n]\} \quad (4.19)$$

onde:

$dx$ : é a distância do ponto inicial ao final na abscissa;

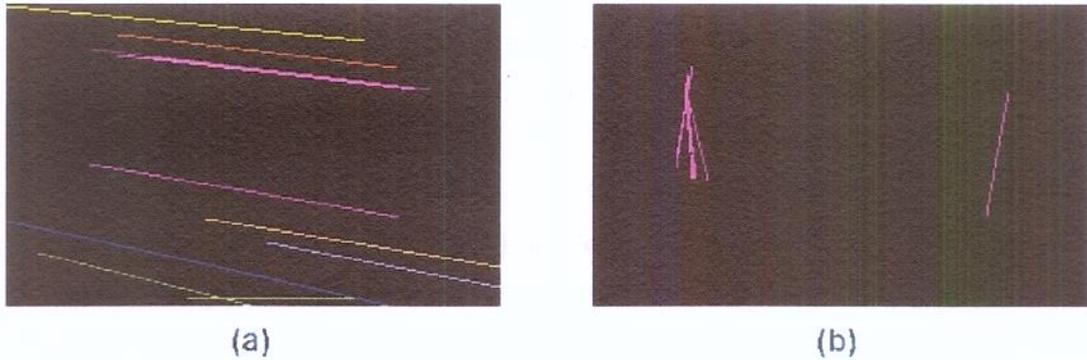
$dy$ : é a distância do ponto inicial ao final na ordenada;

$h$ : é a altura da região promissora;

$w$ : é a largura da região promissora;

$n$ : é a quantidade de segmentos encontrados e armazenados no conjunto  $S$ .

Os segmentos horizontais encontrados podem ser visualizados na figura 4.34 (a) e os verticais na figura 4.34 (b), as quais utilizam cores diferentes para representar os objetos a que pertencem cada segmento.



**Fig. 4.34:** Segmentos horizontais (a) e verticais (b) da figura 4.32, rotulados de acordo com o objeto a que pertence.

### Localização das Arestas Horizontais

Dado o conjunto de segmentos horizontais ( $H$ ), é necessário encontrar neste conjunto o par de segmentos que representam as arestas horizontais: **superior** e **inferior**. Esta operação está representada no bloco **552** do diagrama da figura 4.24.

Para encontrar as arestas horizontais é aplicada a **função-objetivo horizontal** ( $f_{obj_H}$ ) no conjunto de quadriláteros  $Q$  formados com combinações de 2 segmentos horizontais e 2 segmentos verticais. Esta função-objetivo utiliza os seguintes parâmetros:

- Paralelismo entre as arestas horizontais da placa ( $h_{sup}$  e  $h_{inf}$ );
- Intensidade da projeção horizontal na região da cadeia de caracteres.
- Espaço vertical entre a aresta superior e a coordenada superior da cadeia de caracteres da placa;
- Espaço vertical entre a aresta inferior e a coordenada inferior da cadeia de caracteres da placa;
- Maior probabilidade de ambas as arestas horizontais pertencerem ao mesmo objeto.

A função-objetivo também utiliza dois parâmetros calculados empiricamente:  $Dist\_Sup = 0,35$  e  $Dist\_Inf = 0,05$ . Sendo que o primeiro parâmetro representa o espaço vertical entre a aresta superior e a coordenada superior da cadeia de caracteres da placa; e o segundo representa o espaço vertical entre a aresta inferior e a coordenada inferior da cadeia de caracteres da placa, ambos normalizados pela altura da placa. A equação 4.20 representa a função-objetivo horizontal ( $f_{obj_H}$ ):

$$f_{obj_H}(Q_i) = \lambda + 100p + \gamma(e_{sup} + e_{inf}) \quad (4.20)$$

onde:

$Q_i$ : é o  $i$ -ésimo quadrilátero do conjunto  $Q$ ;

$\lambda$ : é a diferença entre os ângulos dos segmentos horizontais (equação 4.33);

$p$ : é a intensidade de projeção horizontal na região da cadeia de caracteres (equação 4.36);

$e_{sup}$ : é a diferença entre o parâmetro  $Dist\_Sup$  e a distância vertical normalizada da aresta superior da placa à coordenada superior da cadeia de caracteres (equação 4.38);

$e_{inf}$ : é a diferença entre o parâmetro  $Dist\_Inf$  e a distância vertical normalizada da coordenada inferior da cadeia de caracteres à aresta inferior da placa (equação 4.39);

$\gamma$ : é a ponderação aplicada à  $e_{sup}$  e  $e_{inf}$ , sendo  $1/2$  quando ambas as arestas horizontais pertencem ao mesmo objeto e  $1$  quando pertencem a objetos diferentes (equação 4.37).

As arestas horizontais ( $a_{sup}$  e  $a_{inf}$ ) da placa de licenciamento veicular estão contidas no quadrilátero com menor valor de “penalidade”, conforme as equações 4.21 e 4.22:

$$a_{sup} = \{h_{sup} \in Q_i \mid f_{obj_H}(Q_i) = \min(\{f_{obj_H}(Q_k), \text{para } k \in [1, n]\}), \text{para } i \in [1, n]\} \quad (4.21)$$

$$a_{inf} = \{h_{inf} \in Q_i \mid f_{obj_H}(Q_i) = \min(\{f_{obj_H}(Q_k), \text{para } k \in [1, n]\}), \text{para } i \in [1, n]\} \quad (4.22)$$

onde:

$n$ : é a quantidade de quadriláteros em  $Q$ ;

$h_{sup}$ : é um segmento horizontal, diferente de  $h_{inf}$ ;

$h_{inf}$ : é um segmento horizontal, diferente de  $h_{sup}$ .

O valor que a função-objetivo retorna para cada quadrilátero do conjunto  $Q$  pode ser visualizada na figura 4.35, onde constatamos que o quadrilátero 22 possui o menor valor de “penalidade”. Portanto este quadrilátero deve conter as arestas horizontais da placa. A figura 4.36 mostra as arestas horizontais encontradas.

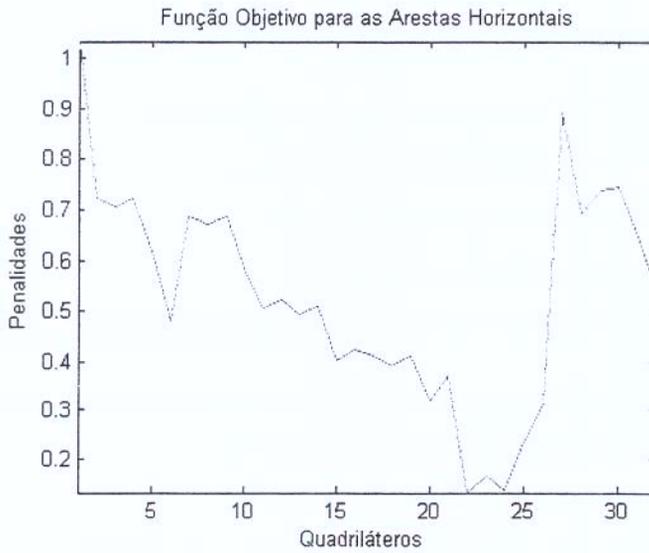


Fig. 4.35: Gráfico do valor da função-objetivo horizontal para cada quadrilátero.



Fig. 4.36: Imagem com as arestas horizontais encontradas.

Os quadriláteros que compõe o conjunto  $Q$  são formados com pares de segmentos horizontais do conjunto  $H$  e dois segmentos verticais “virtuais”, os quais representam suas quatro arestas.

As arestas horizontais a comporem os quadriláteros são definidas pelo par de segmentos obtidos com a combinação 2 a 2 dos segmentos do conjunto  $H$ . Os segmentos são separados em dois conjuntos: o conjunto  $H_{sup}$  contendo os segmentos superiores e o conjunto  $H_{inf}$  contendo os segmentos inferiores, conforme a equação 4.24:

$$m = \binom{n}{2} \quad (4.23)$$

$$(H_{sup}, H_{inf}) = \{H_{n,2} \mid \{\min(y_{1_i}, y_{2_i}), y_{1_i} e y_{2_i} \in H_{sup_i}\} > \{\min(y_{1_i}, y_{2_i}), y_{1_i} e y_{2_i} \in H_{inf_i}\}, \text{ para } i \in [1, m]\} \quad (4.24)$$

onde:

$H_{n,2}$ : é a combinação 2 a 2 dos segmentos do conjunto  $H$ ;

$y_{1_i}$ : é a ordenada do ponto inicial do segmento  $H_i$ ;

$y_{2_i}$ : é a ordenada do ponto final do segmento  $H_i$ ;

$n$ : é a quantidade de segmentos no conjunto  $H$ ;

$m$ : é a quantidade de combinações 2 a 2 dos segmentos no conjunto  $H$ .

As arestas verticais são definidas por 2 segmentos “virtuais”, estimados com base no conjunto  $H$ , denominados  $v_{esq}$  e  $v_{dir}$ , conforme as equações 4.28 e 4.29:

$$\bar{e} = \frac{1}{n} \sum_{i=1}^n \{\min(x_1, x_2), x_1 e x_2 \in H_i\} \quad (4.25)$$

$$\bar{d} = \frac{1}{n} \sum_{i=1}^n \{\max(x_1, x_2), x_1 e x_2 \in H_i\} \quad (4.26)$$

$$o' = \{(\max(o) + 1), o \in H\} \quad (4.27)$$

$$v_{esq} = \{(\bar{e}, h), (\bar{e}, 1), o\} \quad (4.28)$$

$$v_{dir} = \{(\bar{d}, h), (\bar{d}, 1), o\} \quad (4.29)$$

onde:

$\bar{e}$ : é média das ordenadas de menor valor dos segmentos;

$\bar{d}$ : é média das ordenadas de maior valor dos segmentos;

$x_1$ : é a abscissa do ponto inicial do segmento atual;

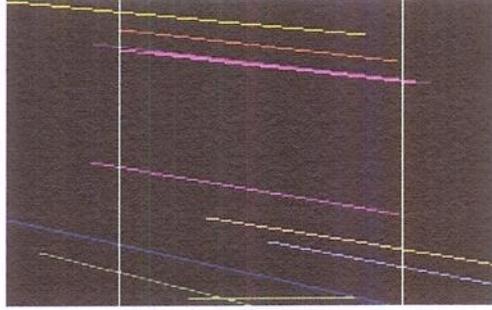
$x_2$ : é a abscissa do ponto final do segmento atual;

$o'$ : é o rótulo atribuído ao segmento “*virtual*”;

$h$ : é a altura da região promissora;

$n$ : é a quantidade de segmentos no conjunto  $H$ .

A figura 4.37 mostra os segmentos horizontais e o par de segmentos verticais “*virtuais*” rotulados de acordo com o objeto a que pertence.



**Fig. 4.37:** Segmentos horizontais e o par de segmentos verticais “*virtuais*” rotulados de acordo com o objeto a que pertence.

Deste modo, temos o conjunto de quadriláteros  $Q$  formado conforme a equação 4.30:

$$Q = \left\{ \left\{ H_{sup_i}, v_{esq}, H_{inf_j}, v_{dir} \right\}, \text{ para } i \in [1, n] \right\} \quad (4.30)$$

onde:  $n$  é a quantidade de segmentos no conjunto  $H_{sup}$  ou  $H_{inf}$ .

No conjunto  $Q$  existem quadriláteros que não são representantes das arestas da placa. Visando minimizar a quantidade de falsos candidatos é aplicado um filtro para remover quadriláteros com distância mínima entre as arestas horizontais menor que  $1/4$  da altura da região promissora, conforme equação 4.32:

$$dy_i = \{ \min(|y'_1 - y''_1| + 1), (|y'_2 - y''_2| + 1) \},$$

$$y'_1 \text{ e } y'_2 \in H_{sup_i}, y''_1 \text{ e } y''_2 \in H_{inf_i} \quad (4.31)$$

$$Q = \{ Q_i | dy_i > 1/4h, \text{ para } i \in [1, n] \} \quad (4.32)$$

onde:

$dy_i$ : é a distância mínima na ordenada entre as arestas horizontais do  $i$ -ésimo quadrilátero em  $Q$ ;

$y'_1$ : é a ordenada do ponto inicial da aresta  $H_{sup_i}$ ;

$y''_1$ : é a ordenada do ponto inicial da aresta  $H_{inf_i}$ ;

$y'_2$ : é a ordenada do ponto final da aresta  $H_{sup_i}$ ;

$y''_2$ : é a ordenada do ponto final da aresta  $H_{inf_i}$ ;

$h$ : é a altura da região promissora;

$n$ : é a quantidade de quadriláteros no conjunto  $Q$ .

A figura 4.38 mostra os quadriláteros resultantes da filtragem no conjunto  $Q$ . Este exemplo de região promissora continha 55 quadriláteros no conjunto  $Q$  antes de aplicar o filtro, sendo reduzido para 32 quadriláteros com a utilização do filtro. Para este caso, o filtro reduziu o espaço de busca em 41,82%, implicando em uma melhora significativa no custo computacional da busca pelas arestas horizontais da placa.



Fig. 4.38: Quadriláteros resultantes do conjunto  $Q$ .

Para o cálculo do paralelismo entre as arestas horizontais do quadrilátero ( $h_{sup}$  e  $h_{inf}$ ) não é necessário aplicar a transformação de homografia na região promissora com base no quadrilátero  $Q_i$ . Portanto, podemos efetuar este cálculo neste momento.

Devido ao efeito trapézio provocado pela distorção de perspectiva, não teremos o paralelismo, mas uma tendência ao mesmo. Deste modo, utilizamos a diferença entre os ângulos das arestas horizontais do quadrilátero  $Q_i$ , conforme a equação 4.33:

$$\lambda = |\theta_{sup} - \theta_{inf}| \quad (4.33)$$

onde:

$\theta_{sup}$ : é o ângulo de inclinação da aresta  $h_{sup}$  em radianos;

$\theta_{inf}$ : é o ângulo de inclinação da aresta  $h_{inf}$  em radianos.

Para os demais cálculos da função-objetivo horizontal ( $f_{obj_H}$ ) é necessário corrigir a distorção de perspectiva da região promissora, conforme descrito na subseção 4.4.5. Os pontos que determinam a

transformação são obtidos dos vértices do quadrilátero  $Q_i$ . As Figuras 4.39a, 4.39b e 4.39c mostram respectivamente os quadriláteros 1, 16 e 22 do conjunto  $Q$ , tendo suas imagens corrigidas nas Figuras 4.39d, 4.39e e 4.39f.



**Fig. 4.39:** Correção da distorção de perspectiva: (a) arestas do quadrilátero 1; (b) arestas do quadrilátero 16; (c) arestas do quadrilátero 22; (d) imagem corrigida do quadrilátero 1; (e) imagem corrigida do quadrilátero 16 e (f) imagem corrigida do quadrilátero 22.

Em seguida, é calculada a projeção horizontal  $P$  do gradiente, na direção horizontal da imagem corrigida  $G$ , conforme a equação 4.34:

$$P_y = \sum_{x=1}^{x < m} |G_{(x+1,y)} - G_{(x,y)}| \quad (4.34)$$

onde:

$G$ : imagem corrigida;

$n$ : é a altura da imagem corrigida;

$m$ : é a largura da imagem corrigida;

$y$ : é a ordenada da imagem corrigida;

$x$ : é a abscissa da imagem corrigida.

Para remover ruídos, é aplicado o operador morfológico de *abertura*, com elemento estruturante em forma de caixa com 21 de lado, na projeção horizontal  $P$ .

As posições superior ( $p_{sup}$ ) e inferior ( $p_{inf}$ ) da cadeia de caracteres da placa são, respectivamente, o ponto inicial da curva ascendente de  $P$  com maior altura e o ponto final da curva descendente de  $P$  com maior altura, podendo conter no máximo uma linha vizinha acima e outra abaixo de mesmo valor.

A figura 4.40 mostra o gráfico da projeção horizontal do gradiente antes e depois de aplicado o operador morfológico de abertura, também mostrando a região da cadeia de caracteres delimitada pelas posições superior ( $p_{sup}$ ) e inferior ( $p_{inf}$ ).



Fig. 4.40: Gráfico da projeção horizontal do gradiente da imagem corrigida.

Caso a mínima distância vertical entre  $p_{sup}$  e  $p_{inf}$  seja inferior à 8 *pixels*, então a função-objetivo horizontal ( $f_{obj_H}$ ) deve retornar valor máximo de “*penalidade*” (infinito), pois a imagem não tem altura suficiente para conter os caracteres da placa.

A intensidade de projeção  $p$  é calculada conforme equação 4.36:

$$h' = (p_{inf} - p_{sup}) + 1 \quad (4.35)$$

$$p = \begin{cases} \infty & (h' \leq 8) \\ \frac{h'}{\sum_{y=p_{sup}}^{p_{inf}} P_y} & (h' > 8) \end{cases} \quad (4.36)$$

onde:

$P_y$ : é a projeção horizontal do gradiente da linha  $y$  (equação 4.34);

$p_{sup}$ : é a posição superior obtida na projeção do gradiente da imagem corrigida;

$p_{inf}$ : é a posição inferior obtida na projeção do gradiente da imagem corrigida.

O cálculo de  $\gamma$ , valor utilizado para ponderar a diferença dos espaços verticais ( $e_{sup}$  e  $e_{inf}$ ), é efetuado conforme a equação 4.37:

$$\gamma = \begin{cases} 1/2 & (o' = o'' \mid o' \in h_{sup} \ o'' \in h_{inf}) \\ 1 & (caso \ contr\acute{a}rio) \end{cases} \quad (4.37)$$

onde:

$h_{sup}$ : é a aresta superior do quadrilátero;

$h_{inf}$ : é a aresta inferior do quadrilátero;

$o'$ : é o rótulo do segmento representante da aresta  $h_{sup}$ ;

$o''$ : é o rótulo do segmento representante da aresta  $h_{inf}$ .

A diferença do espaço vertical superior e inferior entre as arestas horizontais e a região da cadeia de caracteres são calculadas, respectivamente, pelas equações 4.38 e 4.39:

$$e_{sup} = \left| Dist\_Sup - \left( \frac{p_{sup}}{h} \right) \right| \quad (4.38)$$

$$e_{inf} = \left| Dist\_Inf - \left( \frac{h - p_{inf}}{h} \right) \right| \quad (4.39)$$

onde:

$Dist\_Sup$ : é o espaço vertical entre a aresta superior e a coordenada superior da cadeia de caracteres da placa normalizado pela altura da placa (0, 35);

$Dist\_Inf$ : é o espaço vertical entre a aresta inferior e a coordenada inferior da cadeia de caracteres da placa normalizado pela altura da placa (0, 05);

$p_{sup}$ : é a posição superior obtida na projeção do gradiente da imagem corrigida;

$p_{inf}$ : é a posição inferior obtida na projeção do gradiente da imagem corrigida;

$h$ : é a altura da imagem corrigida.

## Localização das Arestas Verticais

Dado o conjunto de segmentos verticais ( $V$ ), é necessário encontrar neste conjunto o par de segmentos que representam as arestas verticais: **direita** e **esquerda**. Esta operação está representada no bloco 553 do diagrama da figura 4.24.

Para encontrar as arestas verticais é aplicada a **função-objetivo vertical** ( $f_{obj_V}$ ) no conjunto de quadriláteros  $Q$  formados com combinações de 2 segmentos horizontais e 2 segmentos verticais. Esta função-objetivo utiliza os seguintes parâmetros:

- Paralelismo entre as arestas verticais da placa ( $v_{esq}$  e  $v_{dir}$ );
- Semelhança entre a quantidade de representantes dos caracteres e da sintaxe da cadeia de caracteres da placa;
- Mediana da razão de aspecto de cada representante dos caracteres (largura dividida pela altura);
- Maior probabilidade de ambas as arestas verticais pertencerem ao mesmo objeto.

A função-objetivo também utiliza como parâmetro, conhecido a priori, a sintaxe da placa. Para o caso das placas brasileiras temos:  $Sintaxe = AAA - 9999$ , onde a letra 'A' representa uma letra maiúscula do alfabeto latino e o número '9' representa um número natural de 0 a 9.

A equação 4.40 representa a função-objetivo vertical ( $f_{obj_V}$ ):

$$f_{obj_V}(Q_i) = \lambda + e_{sintaxe} + \phi - \gamma \quad (4.40)$$

onde:

$Q_i$ : é o  $i$ -ésimo quadrilátero do conjunto  $Q$ ;

$\lambda$ : é a diferença entre os ângulos dos segmentos verticais (equação 4.67);

$e_{sintaxe}$ : é a diferença entre a quantidade de representantes dos caracteres e da sintaxe da cadeia de caracteres;

$\phi$ : é a mediana da razão de aspecto de cada representante dos caracteres (equação 4.70);

$\gamma$ : é a ponderação aplicada à  $v_{esq}$  e  $v_{dir}$ , sendo 0,15 quando ambas as arestas verticais pertencem ao mesmo objeto e 0 quando pertencem a objetos diferentes (equação 4.71).

As arestas verticais da placa de licenciamento veicular estão contidas no quadrilátero com menor valor de "penalidade", conforme as equações 4.41 e 4.42:

$$a_{esq} = \{v_{esq} \in Q_i \mid f_{obj_V}(Q_i) = \min(\{f_{obj_V}(Q_k), \text{para } k \in [1, n]\}),$$

$$\text{para } i \in [1, n] \quad (4.41)$$

$$a_{dir} = \{v_{dir} \in Q_i \mid f_{obj_V}(Q_i) = \min(\{f_{obj_V}(Q_k), \text{para } k \in [1, n]\}),$$

$$\text{para } i \in [1, n] \quad (4.42)$$

onde:  $n$  é a quantidade de quadriláteros em  $Q$ .

O valor que a função-objetivo retorna para cada quadrilátero do conjunto  $Q$  pode ser visualizado na figura 4.41, onde constatamos que o quadrilátero 5 possui o menor valor de “penalidade”. Portanto este quadrilátero deve conter as arestas verticais da placa.

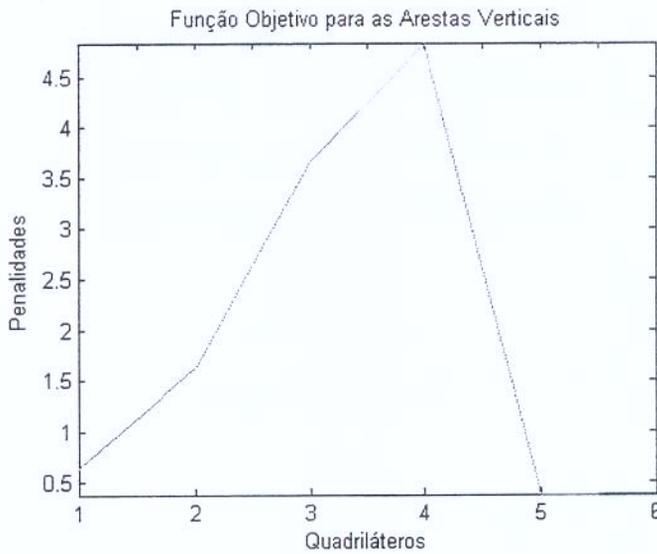
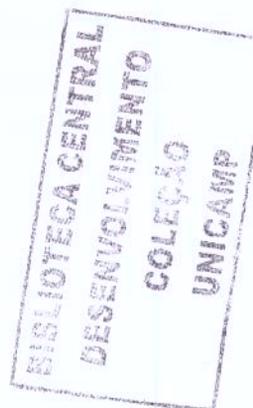


Fig. 4.41: Gráfico do valor da função-objetivo vertical para cada quadrilátero.

A figura 4.42 mostra as arestas horizontais e verticais encontradas.



Fig. 4.42: Imagem com as 4 arestas da placa encontradas.



Os quadriláteros que compõem o conjunto  $Q$  são formados com as duas arestas horizontais ( $a_{sup}$  e  $a_{inf}$ ) encontradas na etapa anterior e segmentos verticais do conjunto  $V$ , os quais representam suas 4 arestas.

Para definir os quadriláteros, os segmentos verticais do conjunto  $V$  são filtrados e separados de acordo com sua posição, ou seja, entre direito e esquerdo, visando minimizar a quantidade de candidatos a arestas verticais da placa. Este filtro utiliza as arestas horizontais ( $a_{sup}$  e  $a_{inf}$ ), encontradas pelo bloco 552 do diagrama da figura 4.24.

Primeiramente, a reta horizontal média  $\bar{r}_{horiz}$  (com equação no formato inclinação, posição) é calculada conforme a equação 4.49:

$$\bar{x}_1 = \left\{ \frac{x'_1 + x''_1}{2} \mid x'_1 \in a_{sup}, x''_1 \in a_{inf} \right\} \quad (4.43)$$

$$\bar{y}_1 = \left\{ \frac{y'_1 + y''_1}{2} \mid y'_1 \in a_{sup}, y''_1 \in a_{inf} \right\} \quad (4.44)$$

$$\bar{x}_2 = \left\{ \frac{x'_2 + x''_2}{2} \mid x'_2 \in a_{sup}, x''_2 \in a_{inf} \right\} \quad (4.45)$$

$$\bar{y}_2 = \left\{ \frac{y'_2 + y''_2}{2} \mid y'_2 \in a_{sup}, y''_2 \in a_{inf} \right\} \quad (4.46)$$

$$a = \frac{|\bar{y}_2 - \bar{y}_1|}{|\bar{x}_2 - \bar{x}_1|} \quad (4.47)$$

$$b = \bar{y}_1 - \bar{x}_1 a \quad (4.48)$$

$$\bar{r}_{horiz}(x) = ax + b \quad (4.49)$$

onde:

$x_1$ : é a abscissa na imagem do ponto inicial de um segmento;

$y_1$ : é a ordenada na imagem do ponto inicial de um segmento;

$x_2$ : é a abscissa na imagem do ponto final de um segmento;

$y_2$ : é a ordenada na imagem do ponto final de um segmento;

$a$ : é o coeficiente angular de uma reta horizontal média ( $\bar{r}_{horiz}$ );

$b$ : é o ponto de posicionamento de uma reta horizontal média ( $\bar{r}_{horiz}$ ).

Em seguida, o conjunto segmentos verticais ( $V$ ) é filtrado de modo a conter somente os segmentos que tenham pelo menos um ponto superior à linha horizontal média  $\bar{r}_{horiz}$  e pelo menos um inferior a ela, conforme a equação 4.50:

$$V = \{V_i \mid ((y_1 > \bar{r}_{horiz}(x_1)) \wedge (y_2 < \bar{r}_{horiz}(x_2))) \vee (y_1 < \bar{r}_{horiz}(x_1)) \wedge (y_2 > \bar{r}_{horiz}(x_2))\}, \text{ para } i \in [1, n]; x_1, y_1, x_2 \text{ e } y_2 \in V_i \} \quad (4.50)$$

onde:

- $V_i$ : é o  $i$ -ésimo segmento do conjunto  $V$ ;
- $x_1$ : é a abscissa na imagem do ponto inicial de um segmento;
- $y_1$ : é a ordenada na imagem do ponto inicial de um segmento;
- $x_2$ : é a abscissa na imagem do ponto final de um segmento;
- $y_2$ : é a ordenada na imagem do ponto final de um segmento;
- $\bar{w}_{horiz}$ : é a reta horizontal média determinada na etapa anterior.

Nesta etapa, o conjunto de segmentos verticais ( $V$ ) é filtrado para que contenha somente segmentos verticais pertencentes ao mesmo objeto que uma das arestas horizontais da placa ( $a_{sup}$  e  $a_{inf}$ ), sendo também separado de acordo com sua posição em relação ao centro da imagem, ou seja, em segmento vertical esquerdo ( $V_{esq}$ ) e direito ( $V_{dir}$ ), conforme equações 4.52 e 4.53:

$$\bar{w} = \frac{w}{2} \quad (4.51)$$

$$V_{esq} = \{V_i | ((o' = o'') \vee (o' = o''')) \wedge ((x_1 < \bar{w}) \vee (x_2 < \bar{w})), \\ \text{para } i \in [1, n], o', x_1 \text{ e } x_2 \in V_i; o'' \in a_{sup}, o''' \in a_{inf}\} \quad (4.52)$$

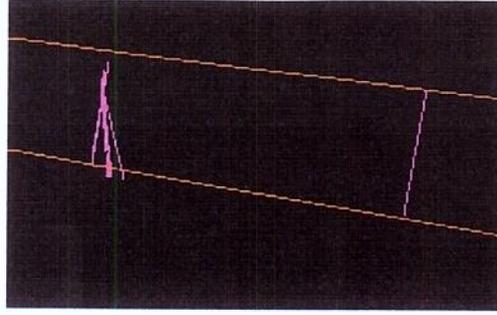
$$V_{dir} = \{V_i | ((o' = o'') \vee (o' = o''')) \wedge ((x_1 > \bar{w}) \vee (x_2 > \bar{w})), \\ \text{para } i \in [1, n], o', x_1 \text{ e } x_2 \in V_i; o'' \in a_{sup}, o''' \in a_{inf}\} \quad (4.53)$$

onde:

- $o'$ : é o rótulo do  $i$ -ésimo segmento do conjunto  $V$ ;
- $o''$ : é o rótulo do segmento representante da aresta  $a_{sup}$ ;
- $o'''$ : é o rótulo do segmento representante da aresta  $a_{inf}$ ;
- $x_1$ : é a abscissa na imagem do ponto inicial do  $i$ -ésimo segmento do conjunto  $V$ ;
- $x_2$ : é a abscissa na imagem do ponto final do  $i$ -ésimo segmento do conjunto  $V$ ;
- $w$ : é a largura da região promissora;
- $\bar{w}$ : é abscissa do centro da região promissora.

A figura 4.43 mostra os segmentos verticais e o par de arestas horizontais ( $a_{sup}$  e  $a_{inf}$ ) rotulados de acordo com o objeto a que pertence.

Caso não exista nenhum **segmento esquerdo** pertencente ao mesmo objeto que uma das arestas horizontais, o conjunto  $V_{esq}$  é redefinido de modo a conter todos os segmentos esquerdos, mesmo que não pertençam ao objeto, também sendo criados segmentos “*virtuais*” com diferentes inclinações e



**Fig. 4.43:** Segmentos horizontais e o par de segmentos verticais “virtuais” rotulados de acordo com o objeto a que pertence.

posicionados na extremidade esquerda das arestas horizontais ( $a_{sup}$  e  $a_{inf}$ ), conforme a equação 4.59:

$$V'_{esq} = \{V_i | ((x_1 < \bar{w}) \vee (x_2 < \bar{w})), \\ x_1 \text{ e } x_2 \in V_i, \text{ para } i \in [1, n]\} \quad (4.54)$$

$$h' = \{|y'_1 - y''_1| + 1, y'_1 \in a_{sup}, y''_1 \in a_{inf}\} \quad (4.55)$$

$$o' = \begin{cases} \{(\max(o) + 1), o \in V'_{esq}\} & (V'_{esq} \neq \emptyset) \\ 1 & (\text{caso contrário}) \end{cases} \quad (4.56)$$

$$V''_{esq} = \{\{(x'_1 + kh', y'_1), (x''_1, y''_1), o'\}, \text{ para } k \in \{0; 0, 1; 0, 2; 0, 3; 0, 4\}, \\ x'_1 \text{ e } y'_1 \in a_{sup}, x''_1 \text{ e } y''_1 \in a_{inf}\} \quad (4.57)$$

$$V'''_{esq} = \{\{(x'_1, y'_1), (x''_1 + kh', y''_1), o'\}, \text{ para } k \in \{0, 1; 0, 2; 0, 3; 0, 4\}, \\ x'_1 \text{ e } y'_1 \in a_{sup}, x''_1 \text{ e } y''_1 \in a_{inf}\} \quad (4.58)$$

$$V_{esq} = \begin{cases} V'_{esq} \cup V''_{esq} \cup V'''_{esq} & (V_{esq} \neq \emptyset) \\ V_{esq} & (\text{caso contrário}) \end{cases} \quad (4.59)$$

onde:

$x_1, x_2, x'_1$  e  $x''_1$ : são as abscissas do ponto inicial de um segmento;

$y_1, y'_1$  e  $y''_1$ : são as ordenadas do ponto inicial de um segmento;

$o'$ : é o rótulo atribuído ao objeto “virtual”;

$h'$ : é a distância entre os pontos iniciais das arestas horizontais ( $a_{sup}$  e  $a_{inf}$ );

$\bar{w}$ : é a abscissa do ponto médio do segmento;

$n$ : é a quantidade de segmentos do conjunto  $V$ .

Caso não exista nenhum **segmento direito** pertencente ao mesmo objeto que uma das arestas horizontais, o conjunto  $V_{dir}$  é redefinido do mesmo modo que o conjunto  $V_{esq}$ , conforme a equação

4.65:

$$V'_{dir} = \{V_i | ((x_1 > \bar{w}) \vee (x_2 > \bar{w})), \\ x_1 \text{ e } x_2 \in V_i, \text{ para } i \in [1, n]\} \quad (4.60)$$

$$h' = \{|y'_2 - y''_2| + 1, y'_2 \in a_{sup}, y''_2 \in a_{inf}\} \quad (4.61)$$

$$o' = \begin{cases} \{(\max(o) + 1), o \in V'_{dir}\} & (V'_{dir} \neq \emptyset) \\ 1 & (\text{caso contrário}) \end{cases} \quad (4.62)$$

$$V''_{dir} = \{\{(x'_2 + kh', y'_2), (x''_2, y''_2), o'\}, \text{ para } k \in \{0; 0, 1; 0, 2; 0, 3; 0, 4\}, \\ x'_2 \text{ e } y'_2 \in a_{sup}, x''_2 \text{ e } y''_2 \in a_{inf}\} \quad (4.63)$$

$$V'''_{dir} = \{\{(x'_2, y'_2), (x''_2 + kh', y''_2), o'\}, \text{ para } k \in \{0, 1; 0, 2; 0, 3; 0, 4\}, \\ x'_2 \text{ e } y'_2 \in a_{sup}, x''_2 \text{ e } y''_2 \in a_{inf}\} \quad (4.64)$$

$$V_{dir} = \begin{cases} V'_{dir} \cup V''_{dir} \cup V'''_{dir} & (V_{dir} = \emptyset) \\ V_{dir} & (\text{caso contrário}) \end{cases} \quad (4.65)$$

onde:

$x_1, x_2, x'_2$  e  $x''_2$ : são as abscissas do ponto inicial de um segmento;

$y_1, y'_2$  e  $y''_2$ : são as ordenadas do ponto inicial de um segmento;

$o'$ : é o rótulo atribuído ao objeto “virtual”;

$h'$ : é a distância entre os pontos iniciais das arestas horizontais ( $a_{sup}$  e  $a_{inf}$ );

$\bar{w}$ : é a abscissa do ponto médio do segmento;

$n$ : é a quantidade de segmentos do conjunto  $V$ .

Na figura 4.44, são mostradas as arestas verticais “virtuais” definidas com base nas arestas horizontais da placa ( $a_{sup}$  e  $a_{inf}$ ).

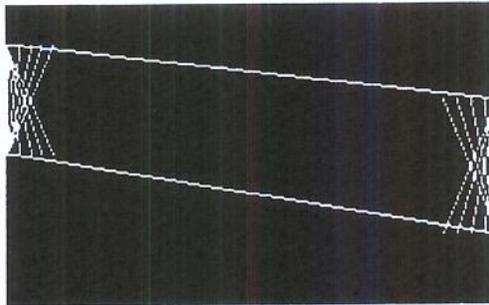


Fig. 4.44: Imagem com as arestas verticais “virtuais” esquerda e direita.

Em seguida, são formados quadriláteros com a combinação de pares de segmentos verticais (esquerdo e direito) e com o par de arestas horizontais ( $a_{sup}$  e  $a_{inf}$ ). Deste modo, temos o conjunto de quadriláteros  $Q$  formado conforme a equação 4.66:

$$Q = \left\{ \left\{ \left\{ a_{sup}, V_{esq_i}, a_{inf}, V_{dir_j} \neq V_{dir_j} \right\}, \text{ para } i \in [1, n] \right\}, \text{ para } j \in [1, m] \right\} \quad (4.66)$$

onde:

$n$ : é a quantidade de segmentos no conjunto  $V_{esq}$ ;

$m$ : é a quantidade de segmentos no conjunto  $V_{dir}$ .

Para o cálculo do paralelismo entre as arestas verticais do quadrilátero ( $v_{esq}$  e  $v_{dir}$ ) não é necessário aplicar a transformação de homografia na região promissora com base no quadrilátero  $Q_i$ . Portanto, podemos efetuar este cálculo neste momento.

Devido ao efeito trapézio provocado pela distorção de perspectiva, não teremos o paralelismo, mas uma tendência ao mesmo. Deste modo, a diferença entre os ângulos das arestas verticais do quadrilátero  $Q_i$ , conforme a equação 4.67:

$$\lambda = |\theta_{esq} - \theta_{dir}| \quad (4.67)$$

onde:

$\theta_{esq}$ : é o ângulo de inclinação da aresta  $v_{esq}$  em radianos;

$\theta_{dir}$ : é o ângulo de inclinação da aresta  $v_{dir}$  em radianos.

Para os demais cálculos da função-objetivo vertical ( $f_{obj_v}$ ) é necessário corrigir a distorção de perspectiva da região promissora, conforme descrito anteriormente na subseção 2.3.6. Os pontos que determinam a transformação são obtidos dos vértices do quadrilátero  $Q_i$ . As Figuras 4.45a, 4.45b e 4.45c mostram respectivamente os quadriláteros 1, 4 e 5 do conjunto  $Q$ , tendo suas imagens corrigidas nas Figuras 4.45d, 4.45e e 4.45f.

A função-objetivo vertical ( $f_{obj_v}$ ) utiliza informações morfológicas dos componentes conexos da imagem corrigida binarizada ( $B$ ) para calcular os valores de  $e_{sintaxe}$  e  $\phi$ , sendo calculados da seguinte maneira:

Primeiramente, a imagem corrigida ( $G$ ) é enquadrada horizontalmente com base nas posições ( $p_{sup}$  e  $p_{inf}$ ), as quais são, respectivamente, o ponto inicial da curva ascendente de  $P$  com maior



**Fig. 4.45:** Correção da distorção de perspectiva: (a) arestas do quadrilátero 1; (b) arestas do quadrilátero 4; (c) arestas do quadrilátero 5; (d) imagem corrigida do quadrilátero 1; (e) imagem corrigida do quadrilátero 4 e (f) imagem corrigida do quadrilátero 5.

altura e o ponto final da curva descendente de  $P$  com maior altura, podendo conter no máximo uma linha vizinha acima e outra abaixo de mesmo valor, onde  $P$  é a projeção horizontal, após ser aplicado o operador de *abertura*, com elemento estruturante em forma de caixa com 21 de lado. Este processo foi descrito anteriormente na subseção *Localização das Arestas Horizontais*, sendo a projeção horizontal ( $P$ ) representada pela figura 4.40 e descrita pela equação 4.34:

$$R = \{G(x, y), \text{ para } x \in [1, m], y \in [p_{sup}, p_{inf}]\} \quad (4.68)$$

onde:

$x$ : é a abscissa do ponto na imagem;

$y$ : é a ordenada do ponto na imagem;

$m$ : é a largura da imagem corrigida ( $G$ );

$p_{sup}$ : é a posição superior obtida na projeção do gradiente da imagem corrigida;

$p_{inf}$ : é a posição inferior obtida na projeção do gradiente da imagem corrigida.

A figura 4.46 mostra o resultado do enquadramento horizontal da imagem corrigida.



**Fig. 4.46:** *Enquadramento horizontal da imagem corrigida.*

A imagem resultante da etapa anterior ( $R$ ) pode conter partes do veículo próximas à placa de licenciamento, o que prejudica sua binarização pelo método de Otsu (1979), pois ele utiliza informações estatísticas da região a ser binarizada. Portanto, modificamos este método de modo que a amostra para a binarização seja obtida em 1/3 da largura da imagem centrada na mesma, mantendo sua altura. A figura 4.47 ilustra a binarização da imagem  $R$  (figura 4.46).



**Fig. 4.47:** *Binarização da imagem enquadrada (figura 4.46).*

Para remover possíveis ruídos da imagem binarizada ( $B$ ), figura 4.47, é aplicado o operador morfológico de abertura por área, o qual elimina os componentes conexos com área menor que 10, com elemento estruturante em forma de caixa unitária (vizinhança-N8). O resultado desta operação é ilustrado na figura 4.48.



**Fig. 4.48:** *Abertura por área aplicada na imagem binarizada (figura 4.47).*

Para o cálculo de  $e_{sintaxe}$ , é necessário detectar o hífen e desconsiderar os ruídos que não foram filtrados pela etapa anterior. Para isto, a imagem  $B$  (4.48) é rotulada utilizando uma vizinhança-N8.

Em seguida, os componentes conexos resultantes são rotulados com a seguinte codificação: '?' é um candidato a caractere; '-' é um hífen; e '\*' é um ruído. Os candidatos a caracteres têm altura maior que 30% da altura da imagem. Tanto o hífen quanto os ruídos possuem altura menor ou igual a 30% da altura da imagem, sendo que o hífen fica localizado em uma região de 40% centralizada verticalmente. Os componentes conexos são processados da esquerda para a direita, tendo seus rótulos armazenados no conjunto  $S$ , conforme a equação 4.69:

$$S_i = \begin{cases} '?' & (h_i > 0, 3h') \\ '-' & ((h_i \leq 0, 3h') \wedge (0, 3h' < y_i < 0, 7h')) \\ '*' & ((h_i \leq 0, 3h') \wedge (0, 3h' \geq y_i \geq 0, 7h')) \end{cases} \quad (4.69)$$

onde:

$h'$ : é a altura da imagem binarizada ( $B$ );

$h_i$ : é a altura do  $i$ -ésimo componente conexo;

$y_i$ : é a ordenada do centróide do  $i$ -ésimo componente conexo.

No cálculo do erro na sintaxe ( $e_{sintaxe}$ ) somente são considerados o posicionamento do hífen e a quantidade de caracteres à direita e à esquerda do hífen. Caso a sintaxe não contenha hífen, então somente será considerada a diferença na quantidade de caracteres. Os componentes conexos rotulados como ruído ('\*') não são considerados neste cálculo. Exemplos:

|                         |                           |                    |
|-------------------------|---------------------------|--------------------|
| 1: Sintaxe = "AAA-9999" | Conjunto $S$ = "???"      | $e_{sintaxe} = 5;$ |
| 2: Sintaxe = "AAA-9999" | Conjunto $S$ = "???????"  | $e_{sintaxe} = 2;$ |
| 3: Sintaxe = "AAA-9999" | Conjunto $S$ = "???-???"  | $e_{sintaxe} = 1;$ |
| 4: Sintaxe = "AAA-9999" | Conjunto $S$ = "??-????"  | $e_{sintaxe} = 1;$ |
| 5: Sintaxe = "AAA9999"  | Conjunto $S$ = "???????"  | $e_{sintaxe} = 0;$ |
| 6: Sintaxe = "AAA-9999" | Conjunto $S$ = "???-????" | $e_{sintaxe} = 0.$ |

O valor de  $\phi$  é composto pela mediana da razão de aspecto dos componentes conexos rotulados como candidato a caractere ('?'), conforme a equação 4.70:

$$\phi = \text{mediana} \left( \left\{ \frac{w_i}{h} \mid S_i q = '?', \text{ para } i \in [1, n] \right\} \right) \quad (4.70)$$

onde:

$h'$ : é a altura da imagem binarizada ( $B$ );

$w_i$ : é a largura do  $i$ -ésimo componente conexo;

$S_i$ : é o rótulo do  $i$ -ésimo componente conexo.

O cálculo de  $\gamma$  é efetuado conforme a equação 4.71:

$$\gamma = \begin{cases} 0.15 & (o' = o'', o' \in v_{esq} \text{ e } o'' \in v_{dir}) \\ 0 & (\text{caso contrário}) \end{cases} \quad (4.71)$$

onde:

$o'$ : é o rótulo do segmento representante da aresta  $v_{esq}$ ;

$o''$ : é o rótulo do segmento representante da aresta  $v_{dir}$ .

#### 4.4.6 Resultado do Módulo de Enquadramento da Placa

O resultado do módulo de *Enquadramento da Placa* é obtido corrigindo a distorção de perspectiva da imagem da placa, bloco 560 do diagrama da figura 4.24, tendo como pontos da imagem os 4 vértices formados pelas arestas da placa encontradas na etapa anterior.

Com isto, temos a imagem da placa de licenciamento veicular corrigida com seu correto enquadramento, bloco 570 do diagrama da figura 4.24, conforme figura 4.49.



Fig. 4.49: Imagem da placa de licenciamento veicular corrigida.

### 4.5 Separação dos Caracteres

O módulo de *Separação dos Caracteres*, bloco 600 do diagrama da figura 4.1, tem a finalidade de enquadrar devidamente a imagem de cada caractere, visando a correta extração de suas características pelo módulo de *Reconhecimento dos Caracteres*. Este bloco está detalhado na figura 4.50, sendo descrito nesta seção.

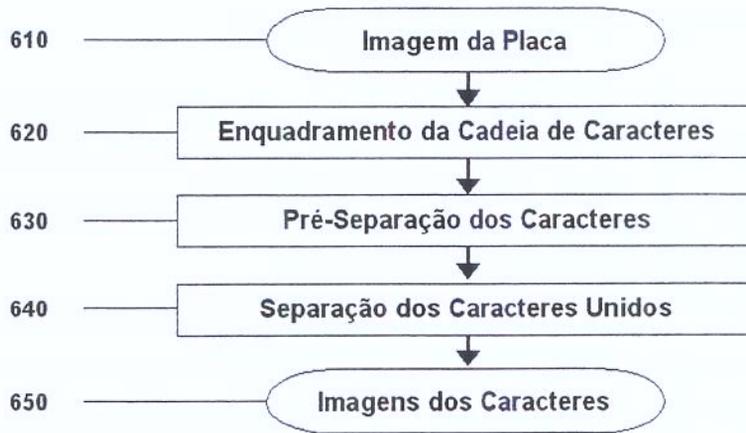


Fig. 4.50: Diagrama em bloco da separação dos caracteres.

A tarefa deste módulo pode ser simples, nos casos em que a imagem da placa de licenciamento não contém ruídos e nenhum caractere toca em outro, ou complexa nos outros casos. Portanto, dividimos esta tarefa em três etapas fundamentais:

1. *Enquadramento da cadeia de caracteres*: enquadrar a cadeia de caracteres da placa com base nas projeções horizontal e vertical da imagem corrigida, deste modo temos o mesmo enquadramento horizontal para todos os caracteres;
2. *Pré-separação dos caracteres*: separa os caracteres com base em sua projeção vertical do enquadramento da cadeia de caracteres, tendo êxito com imagens sem ruídos e nas quais nenhum caractere toca em outro;
3. *Separação dos caracteres unidos*: separa dois ou mais caracteres unidos utilizando uma técnica iterativa, deste modo consegue solucionar o problema da separação dos caracteres nos casos em que algum caractere toca em outro.

Nós optamos por não tratar a união de caracteres *cortados* devido à baixa porcentagem das imagens apresentarem este problema e por utilizarmos o método das projeções para separá-los, o qual somente apresenta problemas quando o corte tiver inclinação de  $90^\circ$ . Decidimos utilizar o tempo restrito de desenvolvimento em etapas mais críticas, visando um maior ganho global da metodologia. Porém, pretendemos resolver este problema em trabalhos futuros.

O processamento deste módulo utiliza apenas a imagem corrigida obtida no módulo de *Enquadramento da Placa* (figura 4.49), sendo representada pelo bloco 610 do diagrama da figura 4.50. As subseções seguintes descrevem detalhadamente cada uma das três etapas acima citadas.

### 4.5.1 Enquadramento da Cadeia de Caracteres

O enquadramento da cadeia de caracteres determina um único enquadramento horizontal para todos os caracteres, também facilitando o enquadramento vertical de cada caractere. Esta etapa, bloco 620 do diagrama da figura 4.50, utiliza as projeções horizontal e vertical para enquadrar a cadeia de caracteres da placa de licenciamento veicular.

Para isto, primeiramente são obtidas as posições superior ( $p_{sup}$ ) e inferior ( $p_{inf}$ ) da imagem corrigida da placa, as quais são, respectivamente, o ponto inicial da curva ascendente de  $P$  com maior altura e o ponto final da curva descendente de  $P$  com maior altura, podendo conter no máximo uma linha vizinha acima e outra abaixo de mesmo valor, onde  $P$  é a projeção horizontal, após ser aplicado o operador de *abertura*, com elemento estruturante em forma de caixa com 21 de lado. Este processo foi descrito anteriormente na subseção *Localização das Arestas Horizontais*, sendo a projeção horizontal ( $P$ ) descrita pela equação 4.34 e representada pela figura 4.40.

Como resultado, temos o enquadramento horizontal da imagem corrigida  $G$  entre as posições  $p_{sup}$  e  $p_{inf}$ , resultando na imagem  $R$ , conforme equação 4.72:

$$R = \{G(x, y), \text{ para } x \in [1, m] \text{ e } y \in [p_{sup}, p_{inf}]\} \quad (4.72)$$

onde:

$x$ : é a abscissa do ponto na imagem;

$y$ : é a ordenada do ponto na imagem;

$m$ : é a largura da imagem  $G$ .

A figura 4.51 mostra o resultado do enquadramento horizontal da imagem corrigida.



Fig. 4.51: Enquadramento horizontal da figura 4.49.

A imagem resultante da etapa anterior ( $R$ ) pode conter partes do veículo próximas à placa de licenciamento, o que prejudica sua binarização pelo método de Otsu (1979), pois ele utiliza informações estatísticas da região a ser binarizada. Portanto, modificamos este método de modo que a amostra para a binarização seja obtida em 1/3 da largura da imagem centrada na mesma, mantendo sua altura. A figura 4.52 ilustra a binarização da imagem  $R$  (figura 4.51), a qual doravante denominaremos de  $B$ .



**Fig. 4.52:** Binarização da imagem da cadeia de caracteres (figura 4.51).

Para remover possíveis ruídos é aplicado o operador morfológico de abertura por área, o qual elimina os componentes conexos com área menor que 10, com elemento estruturante em forma de caixa unitária (vizinhança-N8). A imagem resultante é ilustrada na figura 4.53.



**Fig. 4.53:** Abertura por área aplicada na imagem  $B$  (figura 4.52).

A imagem  $B$  é rotulada utilizando uma vizinhança retangular com largura de 3 *pixels* e altura de  $h$  *pixels*, onde  $h$  é a altura da imagem  $B$ , sendo extraídas as coordenadas do ponto superior esquerdo  $(x_1, y_1)$  e inferior direito  $(x_2, y_2)$  do retângulo envolvente de cada componente conexo. As posições esquerda ( $p_{esc}$ ) e direita ( $p_{dir}$ ) do enquadramento vertical são encontradas conforme a equações 4.73 e 4.74:

$$p_{esc} = \min(\{x_{1_i} \mid (dx_i > 5) \wedge (dx_i < \frac{h}{2}), \text{ para } i \in [1, n]\}) \quad (4.73)$$

$$p_{dir} = \max(\{x_{2_i} \mid (dx_i > 5) \wedge (dx_i < \frac{h}{2}), \text{ para } i \in [1, n]\}) \quad (4.74)$$

onde:

$x_{1_i}$ : é a abscissa do ponto superior esquerdo do  $i$ -ésimo componente conexo;

$x_{2_i}$ : é a abscissa do ponto inferior direito do  $i$ -ésimo componente conexo;

$dx_i$ : é a largura do  $i$ -ésimo componente conexo;

$n$ : é a quantidade de componentes conexos na imagem  $B$ ;

$h$ : é a altura da imagem  $B$ .

A imagem  $B$ , agora é enquadrada na horizontal e na vertical entre as posições  $p_{sup}$ ,  $p_{inf}$ ,  $p_{esq}$  e  $p_{dir}$ , conforme equação 4.75:

$$B = \{B(x, y), \text{ para } x \in [p_{esq}, p_{dir}] \text{ e } y \in [p_{sup}, p_{inf}]\} \quad (4.75)$$

onde:

$x$ : é a abscissa do ponto na imagem  $B$ ;

$y$ : é a ordenada do ponto na imagem  $B$ .

A figura 4.54 ilustra o enquadramento vertical da figura 4.53.



**Fig. 4.54:** Enquadramento vertical da imagem  $B$  (figura 4.53).

Para concluir o enquadramento da cadeia de caracteres é necessário aplicar um ajuste fino no enquadramento horizontal. Para isto, primeiro a imagem  $B$  é rotulada utilizando um elemento estruturante em forma de linha horizontal com tamanho de 7 pixels, visando segmentar cada caractere da placa em linhas horizontais. A figura 4.55 ilustra a rotulação desta imagem.



**Fig. 4.55:** Rotulação da imagem  $B$  (figura 4.54) com elemento estruturante em forma de linha horizontal com tamanho de 7 pixels.

Em seguida, é computada a quantidade de componentes conexos em cada linha da imagem. O novo valor da posição superior ( $p_{sup}$ ) é a primeira linha que contenha mais de 6 componentes conexos (equação 4.76) e o valor da posição inferior é a última linha que contenha mais de 6 componentes (equação 4.77):

$$p_{sup} = \min(\{y \mid (c_y > 6), \text{ para } i \in [1, h]\}) \quad (4.76)$$

$$p_{inf} = \max(\{y \mid (c_y > 6), \text{ para } i \in [1, h]\}) \quad (4.77)$$

onde:

$y$ : é uma linha da imagem  $B$ ;

$C_y$ : é a quantidade de componentes conexos da linha  $y$  na imagem  $B$ ;

$h$ : é a altura da imagem  $B$ .

O enquadramento final do conjunto de caracteres da placa é obtido aplicando-se os novos valores do limite na imagem  $B$ , conforme equação 4.78:

$$B = \{B(x, y), \text{ para } x \in [p_{esq}, p_{dir}] \text{ e } y \in [p_{sup}, p_{inf}]\} \quad (4.78)$$

onde:

$x$ : é a abscissa do ponto na imagem  $B$ ;

$y$ : é a ordenada do ponto na imagem  $B$ .

Como resultados temos a nova imagem  $B$ , ilustrada na figura 4.56.



Fig. 4.56: Imagem  $B$  resultante do enquadramento da cadeia de caracteres da placa.

### 4.5.2 Pré-Separação dos Caracteres

A pré-separação dos caracteres, utiliza a projeção vertical  $P_x$  da imagem obtida na etapa anterior ( $B$ ), conforme a equação 4.79:

$$P_x = \sum_{y=1}^{y \leq n} B''(x, y), \text{ para } x \in [1, m] \quad (4.79)$$

onde:

$n$ : é a altura da imagem  $f$ ;

$m$ : é a largura da imagem  $f$ ;

$x$ : é a abscissa da imagem  $f$ ;

$y$ : é a ordenada da imagem  $f$ .

A figura 4.57 ilustra a projeção vertical da imagem  $B$ . Em seguida, a projeção vertical é binarizada conforme a equação 4.80:

$$P'_x = \begin{cases} 0 & (P_x = 0) \\ 1 & (\text{caso contrário}) \end{cases} \quad (4.80)$$

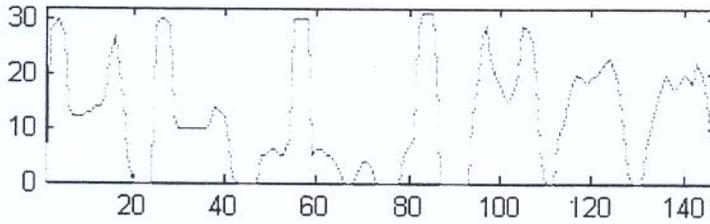


Fig. 4.57: Projeção vertical da imagem B (figura 4.56).

Em seguida, são extraídas as posições inicial e final  $(x_{1_i}, x_{2_i})$  de cada componente conexo da projeção vertical binarizada  $(P'_x)$ , utilizando um critério de vizinhança-N4, sendo armazenados no conjunto  $C$ , conforme a equação 4.81:

$$C_i = \{(x_{1_i}, x_{2_i})\} \quad (4.81)$$

onde:

$C_i$ : é o  $i$ -ésimo componente conexo de  $P'$ ;

$x_{1_i}$ : é a abscissa do ponto inicial do  $i$ -ésimo componente conexo de  $P'$ ;

$x_{2_i}$ : é a abscissa do ponto final do  $i$ -ésimo componente conexo de  $P'$ .

Doravante, denominaremos de **região candidata a caractere** cada região da projeção vertical  $P_x$  delimitada pelas coordenadas horizontais de um elemento do conjunto  $C$ .

Para comparar as regiões candidatas a caracteres com a sintaxe da placa, é necessário identificar o hífen e a presença de ruídos. Portanto, nós atribuímos rótulos a cada região candidata de acordo com a seguinte codificação: '?' é um candidato a caractere; '-' é um hífen; e '\*' é um ruído. Os candidatos a caracteres possuem valor máximo de projeção vertical  $(P_x)$  maior que 30% da altura da região candidata. Tanto o hífen quanto os ruídos possuem um valor máximo de projeção vertical menor ou igual a 30% da altura da região candidata. Porém, o hífen possui *pixels* com valor '1' somente em uma região de 40% centralizada verticalmente. Os rótulos atribuídos a cada região candidata a caractere são armazenados no conjunto  $S$ , sendo este processo representado pela equação 4.82:

$$S_i = \begin{cases} '?' & (\max(\{P_x \mid \text{para } x \in [x_{1_i}, x_{2_i}], x_{1_i} \text{ e } x_{2_i} \in C_i\}) > 0,3h) \\ '-' & ((\sum_{j=1}^{j < 0,3n} \sum_{i=x_{1_i}}^{i < x_{2_i}} |B_{(i+1,j)} - B''_{(i,j)}| = 0) \wedge \\ & (\sum_{j=0,7n}^{j < n} \sum_{i=x_{1_i}}^{i < x_{2_i}} |B_{(i+1,j)} - B_{(i,j)}| = 0)) \\ '*' & (\text{caso contrário}) \end{cases} \quad (4.82)$$

onde:

- $i$ : é a  $i$ -ésima região do conjunto  $C$ ;
- $B$ : é a imagem binarizada da cadeia de caracteres;
- $x_{1_i}$ : é a abscissa do ponto inicial da  $i$ -ésima região do conjunto  $C$ ;
- $x_{2_i}$ : é a abscissa do ponto final da  $i$ -ésima região do conjunto  $C$ ;
- $n$ : é a altura da imagem  $B$ .

Concluindo a etapa de pré-separação dos caracteres, as regiões candidatas a caracteres que foram rotuladas como ruído ('\*') são removidas dos conjuntos  $C$  e  $S$ . Como resultado desta etapa aplicada na imagem  $B$  (figura 4.56) temos:

$$C = \{(2, 19); (25, 42); (48, 65); (69, 72); (79, 86); (94, 109); (113, 128); (132, 147)\}$$

$$S = \{{"???-????"}\}$$

A figura 4.58 ilustra as coordenadas horizontais da pré-separação dos caracteres (conjunto  $C$ ).

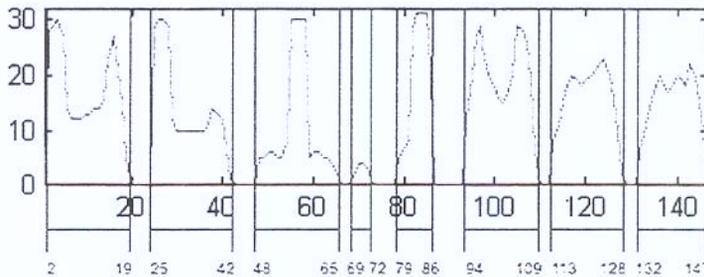


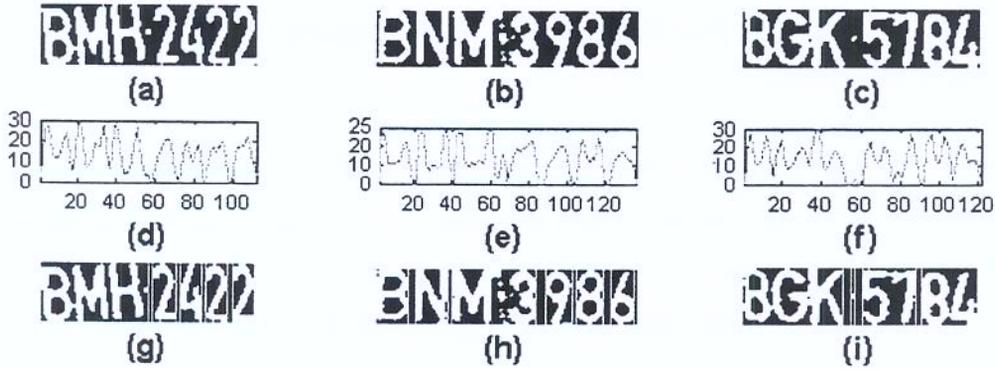
Fig. 4.58: Projeção vertical da figura 4.56 com as coordenadas horizontais da separação dos caracteres.

### 4.5.3 Separação dos Caracteres Unidos

Devido ao efeito da distorção de perspectiva, à presença de ruídos e à baixa resolução espacial da imagem da placa, a imagem binarizada pode conter caracteres unidos, os quais não foram separados pela etapa de *Pré-separação dos Caracteres*. Esta etapa, bloco 640 do diagrama da figura 4.50, tem a tarefa de detectar e separar caracteres unidos.

A imagem da cadeia de caracteres binarizada, ilustrada na figura 4.56, foi separada corretamente na etapa de *Pré-separação dos Caracteres*, não apresentando caracteres unidos. As Figuras 4.59a, 4.59b e 4.59c ilustram imagens de cadeias de caracteres que contêm caracteres unidos devido à distorção de perspectiva, presença de ruído e baixa resolução espacial. A figura 4.59a é utilizada para exemplificar esta etapa.

A projeção vertical das Figuras 4.59a, 4.59b e 4.59c são ilustradas nas respectivas Figuras 4.59d, 4.59e e 4.59f; nas quais constatamos que as duas primeiras figuras possuem apenas uma região com caracteres unidos e a última figura possui três regiões com caracteres unidos. As Figuras 4.59g, 4.59h e 4.59i ilustram as coordenadas horizontais, obtidas na etapa de pré-separação dos caracteres.



**Fig. 4.59:** Imagem da cadeia de caracteres binarizada com caracteres unidos devido à: (a) distorção de perspectiva; (b) presença de ruído; e (c) baixa resolução espacial. Suas respectivas projeções verticais: (d), (e), e (f). Respective imagens binarizada com as coordenadas horizontais de separação: (g), (h) e (i).

Os conjuntos de enquadramentos horizontais ( $C$ ) e de rótulos ( $S$ ) das regiões candidatas a caracteres obtidos para a figura 4.59a foram os seguintes:

$$C = \{(1, 57); (59, 71); (73, 84); (86, 97); (100, 111)\}$$

$$S = \{“?????”\}$$

Para verificar se algum caractere está unido a outro, utilizamos o critério de que a quantidade de regiões candidatas deve ser igual a quantidade de caracteres da sintaxe, incluindo o hífen. Deste modo, se a quantidade das regiões for menor que a quantidade da sintaxe, consideramos que existem caracteres unidos na imagem. No exemplo ilustrado pela figura 4.59a temos o seguinte resultado:

“?????”                      Quantidade de regiões candidatas: 5

“AAA-9999”                Quantidade de caracteres da sintaxe: 8

Portanto, a imagem deste exemplo contém caracteres unidos. Uma vez detectada esta existência, é necessário identificar qual é a região que contém a união e quantos caracteres estão unidos à ela, além de verificar se um dos caracteres unidos é o hífen. Para isto utilizamos a largura estimada das regiões ( $w_{est}$ ), na qual consideramos que a maioria dos caracteres (exceto: '1', 'I', 'M' e 'W') tem aproximadamente a mesma largura, sendo a abscissa de seu centro geométrico regularmente espaçada; e que o hífen tem largura de aproximadamente 1/4 da largura de um caractere. Os caracteres '1' e 'I' são mais finos que os demais e possuem a abscissa de seu centro geométrico igualmente

espaçada a outros caracteres, com isto o espaço entre estes é maior que nos demais casos, o que implica em menor probabilidade de se unirem à outros caracteres. Já os caracteres 'M' e 'W' possuem largura pouco maior que os demais caracteres, o que aumenta a probabilidade de se unirem à outros caracteres, sendo que sua largura estimada  $w_{est}$  é sempre maior que a largura média dos caracteres. Portanto, não temos problemas com a largura destes caracteres ser pouco maior que a dos demais. A largura estimada do caractere, baseada no formato da cadeia de caracteres da placa de licenciamento brasileira ("AAA-9999"), é calculada conforme a equação 4.83:

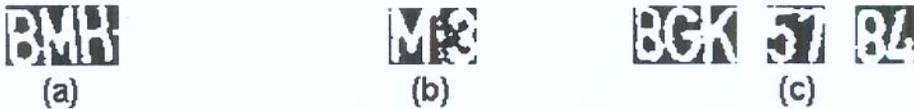
$$w_{est} = \left\lfloor \left( \frac{nw}{(n(n-1)) + 1} \right) + 0,5 \right\rfloor \quad (4.83)$$

onde:

$w$ : é a largura da imagem da cadeia de caracteres;

$n$ : é a quantidade de caracteres da sintaxe, incluindo o hífen ( $n = 8$ ).

A largura estimada da cadeia de caracteres ilustrada na figura 4.59a foi a seguinte:  $w_{est} = 15$ . Definimos empiricamente que uma região candidata possui caracteres unidos quando sua largura for maior que  $1,5w_{est}$ . Portanto, com base no conjunto do enquadramento horizontal ( $C$ ) apresentado anteriormente, constatamos que somente a primeira região candidata da figura 4.59a é detectada com caracteres unidos. As Figuras 4.60a, 4.60b e 4.60c ilustram as regiões detectadas de suas respectivas imagens da cadeia de caracteres: 4.59a, 4.59b e 4.59c.



**Fig. 4.60:** Imagem da região candidata com caracteres unidos das Figuras: (a) 4.59a; (b) 4.59b; e (c) 4.59c.

Uma vez detectadas as regiões com caracteres unidos, é necessário identificar quantos caracteres estão unidos em cada região com base na sintaxe da cadeia de caracteres da placa. Também é necessário verificar se o hífen está unido a algum caractere desta região. Para exemplificar esta etapa utilizamos a imagem da cadeia de caracteres da figura 4.59a.

Primeiramente marcamos a região candidata detectada no conjunto de rótulos ( $S$ ) com o símbolo '#', conforme ilustrado abaixo:

$$S = \{ \text{"#????"} \}$$

Em seguida removemos as regiões candidatas à direita e à esquerda que não contenham caracteres unidos, removendo também os respectivos caracteres da sintaxe.

$$S = \{\text{"#"}\}$$

$$\text{Sintaxe} = \{\text{"AAA-"}\}$$

Neste caso, temos apenas uma região candidata com caracteres unidos, portanto a sintaxe acima ilustra a quantidade de caracteres unidas nesta região, indicando também a presença de um hífen.

Nos casos em que existam mais de uma região, é necessário estimar quantos caracteres estão unidos em cada região, eliminando as regiões candidatas que não possuam caracteres unidos. Para isto, o conjunto de rótulos ( $S$ ) é percorrido da esquerda para a direita, onde em cada região candidata com caracteres unidos ('#') são atribuídos,  $k$  caracteres da *Sintaxe*, onde  $k = \lfloor (q_{reg} + 0, 5) \rfloor$ , sendo este obtido pela equação 4.84:

$$q_{reg} = \left\{ \frac{|x_{2_i} - x_{1_i}| + 1}{w_{est}} \mid x_{2_i} \text{ e } x_{1_i} \in C_i \right\} \quad (4.84)$$

onde:

$C_i$ : é a  $i$ -ésima região candidata;

$x_{1_i}$ : é a abscissa do ponto inicial da  $i$ -ésima região candidata ( $C_i$ );

$x_{2_i}$ : é a abscissa do ponto final da  $i$ -ésima região candidata ( $C_i$ );

$w_{est}$ : é a largura estimada do caractere (equação 4.83).

Caso  $q_{reg} > k$  e o hífen for um dos  $k$  caracteres da sintaxe ou for o  $k + 1$  caractere da *Sintaxe* e não tiver sido detectado pela etapa de *Pré-Separação dos Caracteres*, então também é atribuído à esta região o caractere  $k + 1$  da sintaxe. Para cada região candidata sem caracteres unidos ('?') é removida esta região do conjunto codificado ( $S$ ) e um caractere da sintaxe.

Finalmente, para a última região candidata com caracteres unidos são atribuídos os caracteres restantes da *Sintaxe*. Deste modo, para a imagem da cadeia de caracteres da figura 4.59c, temos as seguintes sintaxes para cada região:

$$S = \{\text{"#-##"}\}$$

Região 1: *Sintaxe* = "AAA"

Região 2: *Sintaxe* = "99"

Região 3: *Sintaxe* = "99"

A separação dos caracteres unidos descrita a seguir é aplicada em cada região detectada com caracteres unidos. Primeiro é obtida a **projeção vertical** da imagem binarizada da cadeia de caracteres.

Em seguida é calculada a *dinâmica dos mínimos regionais* desta projeção, sendo colocados *marcadores de separação* no centro de cada posição cujo valor da dinâmica do mínimo regional seja maior que 10% do valor máximo da projeção. Este processo é ilustrado na figura 4.61.

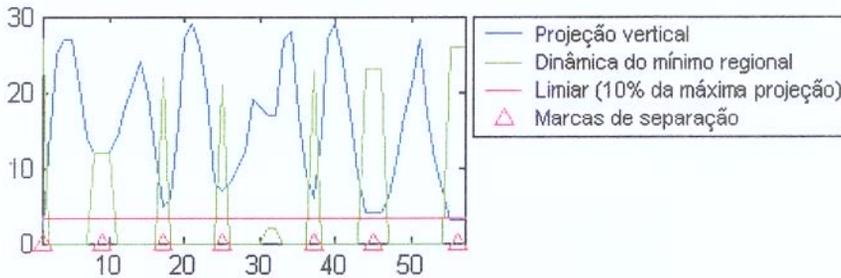


Fig. 4.61: Gráfico indicando os pontos de separação para a região da figura 4.60a.

Para determinar quais são os marcadores que melhor separam os caracteres unidos, definimos a **função-objetivo separação** ( $f_{obj_S}$ ), a qual recebe como entrada um conjunto  $M$  contendo uma combinação de marcadores, retornando a soma das diferenças das distâncias entre dois marcadores que delimitam uma região candidata com o valor estimado da largura desta região ( $w_{est}$ ) no caso de uma letra ou  $(1/4w_{est})$  no caso de um hífen. A equação 4.85 representa a função-objetivo separação ( $f_{obj_S}$ ):

$$f_{obj_S}(M) = \sum_{i=1}^{n-1} \begin{cases} |(M_{i+1} - M_i) + 1) - w_{est}| & (Sintaxe_i = '?') \\ |(M_{i+1} - M_i) + 1) - 1/4w_{est}| & (Sintaxe_i = '-')$$

onde:

$M_i$ : é a abscissa do  $i$ -ésimo marcador do conjunto  $M$ ;

$Sintaxe_i$ : é o rótulo associado à  $i$ -ésima região candidata;  $w_{est}$ : é a largura estimada do caractere (equação 4.83).

A combinação de marcadores com menor valor retornado pela  $f_{obj_S}$  é a que separa da melhor forma possível os caracteres unidos. Portanto, devem ser formados todas as possíveis combinações (representadas por  $M$ ) contendo  $n + 1$  marcadores, onde  $n$  é a quantidade de caracteres na região unida, aplicando a função-objetivo separação em cada uma delas.

Concluindo esta etapa, a posição de cada região candidata com caracteres unidos são substituídas pelas posições encontradas para os caracteres desta região.

#### 4.5.4 Resultado do Módulo de Separação dos Caracteres

Como resultado do módulo de *Separação dos Caracteres*, bloco 650 do diagrama da figura 4.50, temos os caracteres da placa de licenciamento devidamente enquadrados.

As Figuras 4.62, 4.63a, 4.63b e 4.63c ilustram com um retângulo o enquadramento de cada caractere. O resultado apresentado na figura 4.62 não precisou utilizar a etapa de *Separação dos Caracteres Unidos*, já o resultado das Figuras 4.63a, 4.63b e 4.63c utilizaram esta etapa, obtendo êxito no resultado.

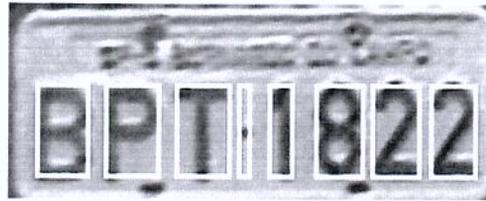
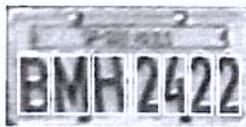


Fig. 4.62: Resultado da separação dos caracteres de uma placa que não precisou separar caracteres unidos.



(a)



(b)



(c)

Fig. 4.63: Resultado da separação dos caracteres de placas que precisaram separar caracteres unidos.

Podemos verificar, nas Figuras 4.62, 4.63a, 4.63b e 4.63c, a precisão no enquadramento de cada caractere. Isto somente foi possível devido à correção da distorção de perspectiva da placa de licenciamento. Com esta qualidade de enquadramento o conjunto de características extraídas da imagem de cada caractere são mais representativas, melhorando a eficiência no reconhecimento dos mesmos.

## 4.6 Reconhecimento dos Caracteres

O bloco 700 do diagrama da figura 4.1, tem a finalidade de reconhecer a imagem de cada caractere com base em suas características, retornando seu código *ASCII* (*American Standard Code for Information Interchange*). Adotamos este código por ser um padrão digital para a representação de caracteres, permitindo a integração com qualquer outro sistema. Este bloco está detalhado na figura 4.64, sendo descrito nesta seção.

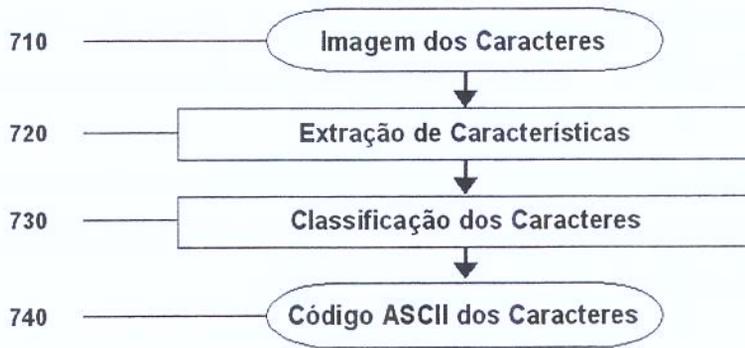


Fig. 4.64: Diagrama em bloco do reconhecimento dos caracteres.

Como entrada deste módulo, temos os caracteres da imagem binarizada (figura 4.56), separados e enquadrados, conforme ilustrado pela figura 4.65.



Fig. 4.65: Imagem dos caracteres da figura 4.56 separados e enquadrados.

Conforme ilustrado no diagrama da figura 4.64, esta tarefa é dividida em duas etapas, as quais são descritas nas subseções seguintes:

1. *Extração de características;*
2. *Classificação dos caracteres.*

### 4.6.1 Extração de Características

A extração de características, representada pelo bloco 720 do diagrama da figura 4.64, é a principal etapa para o reconhecimento dos caracteres, pois o conjunto de características, também conhecido como *vetor de características*, é utilizado para associar um caractere a ser reconhecido com sua respectiva *classe* independente do método adotado para o reconhecimento do mesmo (*Estatístico*, *Conexionista* ou *Sintático*).

A maior dificuldade desta etapa é definir quais são as características da imagem que melhor representam as classes de caracteres.

O vetor de características que adotamos é composto pelos valores dos *pixels* da normalização das dimensões da imagem binária do caractere para a resolução 5x8, sendo estes valores reais compreendidos entre 0 e 1.

Para efetuar a normalização, a imagem de cada caractere é dividida em 5 partes horizontais e 8 partes verticais, resultando em 40 retângulos de mesma dimensão, sendo calculada a porcentagem de preenchimento de cada retângulo, conforme a equação 4.86:

$$P_i = \frac{k_i}{a} \quad (4.86)$$

onde:

$P_i$ : é a porcentagem de preenchimento do  $i$ -ésimo retângulo;

$k_i$ : é a quantidade de *pixels* com valor máximo (brancos) do  $i$ -ésimo retângulo;

$a$ : é a área do retângulo.

As figuras 4.66a, 4.66b e 4.66c ilustram este processo, onde: a figura 4.66a exemplifica a divisão da imagem binária do caractere 'B' em 8 partes verticais e cinco partes horizontais; a figura 4.66b mostra os valores da porcentagem de preenchimento de cada retângulo; e a 4.66c ilustra o resultado em nível de cinza, possibilitando uma comparação visual do mesmo. A figura 4.67 ilustra o resultado da normalização dos caracteres da figura 4.65.

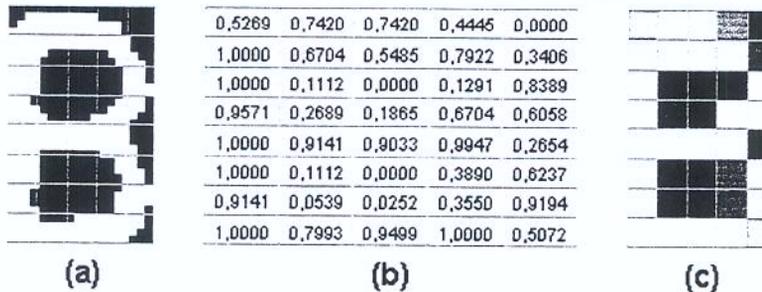


Fig. 4.66: Normalização do caractere 'B' da figura 4.65.

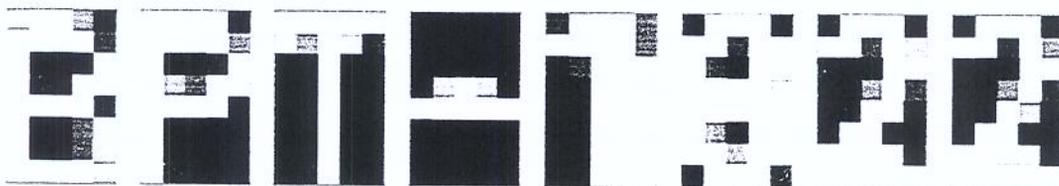


Fig. 4.67: Resultado da normalização dos caracteres da figura 4.65.

## 4.6.2 Classificação dos Caracteres

Para a classificação dos caracteres, representada pelo bloco 730 do diagrama da figura 4.64, é adotado um *classificador hierárquico de distância mínima*, também conhecido como *casamento de modelos*. Tendo como *medida de similaridade* a *distância euclidiana*.

Os conflitos entre caracteres alfabéticos e numéricos das placas brasileiras são resolvidos utilizando a sintaxe da placa de licenciamento veicular brasileira (“AAA-9999”) como auxílio ao contexto.

As principais dificuldades desta etapa são:

1. *Diferença na tipografia do mesmo caractere* (ex.: figura 1.8);
2. *Semelhança na tipografia de caracteres diferentes* (ex.: figura 1.9).

Visando resolver a primeira dificuldade, efetuamos a *análise de agrupamentos* e para resolver a segunda dificuldade, utilizamos um *classificador hierárquico de distância mínima*.

### Análise de Agrupamentos

Para definir as classes e seus representantes, primeiramente amostramos e rotulamos 10 vetores de características para cada um dos 36 possíveis caracteres da placa de licenciamento. Em seguida, analisamos os agrupamentos no dendrograma das amostras rotuladas elaborado com a técnica de agrupamento hierárquico denominada *single-link*, tendo como medida de similaridade a *distância euclidiana*.

Nós optamos por analisar os números junto com as letras, pois a sintaxe da placa de licenciamento veicular de alguns países não permite determinar se um caractere é numérico ou alfabético com base em sua posição, sendo necessário utilizar um único classificador para os números e as letras. Porém, com a sintaxe das placas brasileiras não temos este problema, podendo utilizar um classificador para os números e outro para as letras, facilitando a resolução de conflitos entre alguns caracteres.

Aplicando diferentes limiarizações (*thresholds*) no dendrograma, podemos verificar quais são os prováveis conflitos entre classes e detectar quando mais de dois agrupamentos são definidos por amostras de mesmo rótulo, neste caso é importante definir uma classe para cada agrupamento. Também podemos detectar amostras que não se agrupam em nenhuma classe devido à má qualidade, as quais prejudicam a performance do classificador, sendo importante removê-las do conjunto de amostras.

A figura 4.68 ilustra este dendrograma, tendo uma ampliação parcial da parte envolvida por um retângulo, na qual podemos verificar as amostras que não se agruparam e alguns grupos formados por uma determinada limiarização.

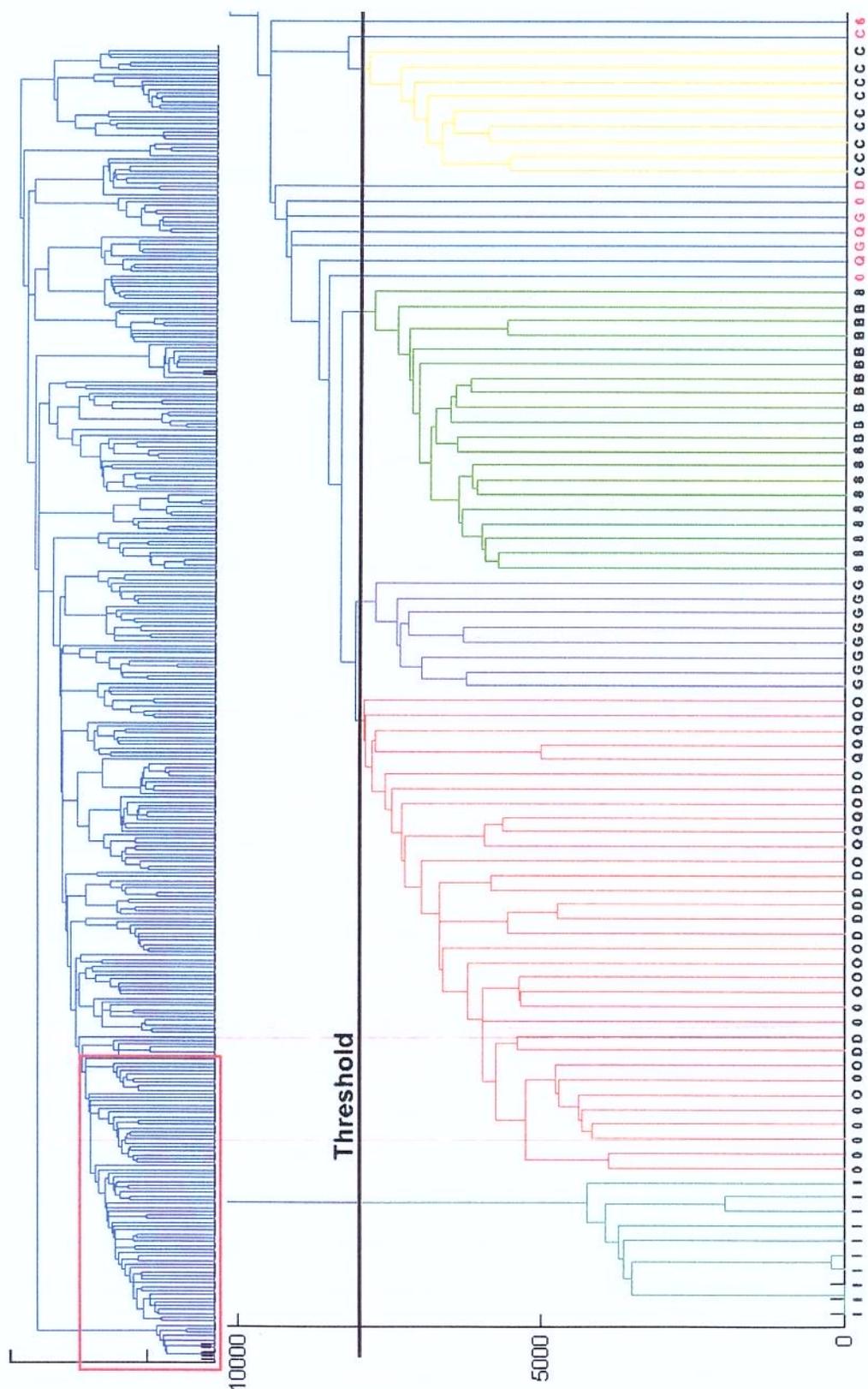


Fig. 4.68: Dendrograma das amostras rotuladas com ampliação parcial.

| Classe | Caractere Representante | Quantidade de Amostras Rotuladas |
|--------|-------------------------|----------------------------------|
| 1      | '0'                     | 10                               |
| 2      | '1'                     | 10                               |
| 3      | '2'                     | 7                                |
| 4      | '2'                     | 3                                |
| 5      | '3'                     | 10                               |
| 6      | '4'                     | 2                                |
| 7      | '4'                     | 6                                |
| 8      | '5'                     | 10                               |
| 9      | '6'                     | 6                                |
| 10     | '6'                     | 4                                |
| 11     | '7'                     | 10                               |
| 12     | '8'                     | 6                                |
| 13     | '9'                     | 4                                |
| 14     | '9'                     | 5                                |
| 15     | 'A'                     | 10                               |
| 16     | 'B'                     | 9                                |
| 17     | 'C'                     | 9                                |
| 18     | 'D'                     | 7                                |
| 19     | 'E'                     | 10                               |
| 20     | 'F'                     | 10                               |
| 21     | 'G'                     | 10                               |
| 22     | 'H'                     | 10                               |
| 23     | 'I'                     | 10                               |
| 24     | 'J'                     | 10                               |
| 25     | 'K'                     | 10                               |
| 26     | 'L'                     | 10                               |
| 27     | 'M'                     | 10                               |
| 28     | 'N'                     | 10                               |
| 29     | 'O'                     | 10                               |
| 30     | 'P'                     | 10                               |
| 31     | 'Q'                     | 9                                |
| 32     | 'R'                     | 9                                |
| 33     | 'S'                     | 10                               |
| 34     | 'T'                     | 9                                |
| 35     | 'U'                     | 10                               |
| 36     | 'V'                     | 9                                |
| 37     | 'W'                     | 10                               |
| 38     | 'X'                     | 10                               |
| 39     | 'Y'                     | 10                               |
| 40     | 'Z'                     | 10                               |

**Tab. 4.1:** Classes obtidas após a análise de agrupamentos.

As classes resultantes podem ser visualizadas na tabela 4.1. Nela constatamos que os seguintes caracteres apresentaram duas classes: '2', '4', '6' e '9'. A figura 4.69 ilustra o dendrograma dos *protótipos* resultantes.

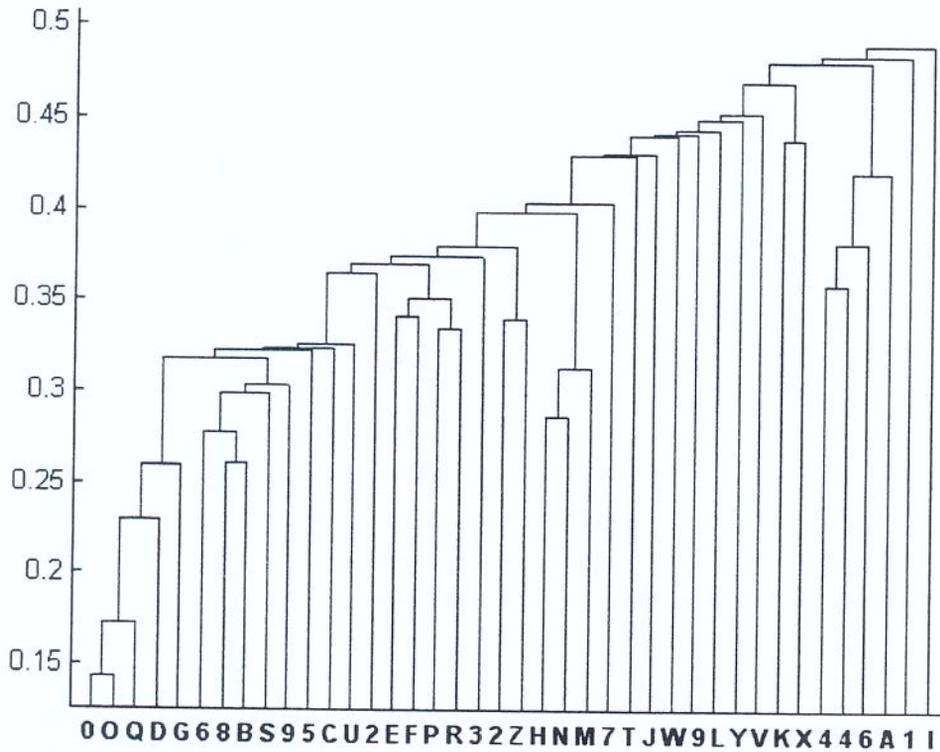


Fig. 4.69: Dendrograma dos protótipos resultantes da análise de agrupamento.

Para uma melhor análise das distâncias entre os protótipos elaboramos a tabela 4.2, na qual as distâncias estão normalizadas entre 0 e 1. Nela podemos constatar uma pequena distância entre caracteres semelhantes (Ex.: '0', 'D', 'O', 'Q').

Os dados apresentados anteriormente não permitem analisar a distância entre as fronteiras de decisão. Portanto, definimos um fator de confiabilidade do reconhecimento utilizando uma medida relativa normalizada entre 0% e 100%, conforme a equação 4.87:

$$C_{fator} = \left( \frac{d_n - d_s}{d_n + d_s} \right) 100 \quad (4.87)$$

onde:

$d_s$ : é a distância entre o padrão a ser reconhecido e a amostra classificada;

$d_n$ : é a distância entre o padrão a ser reconhecido e a amostra mais próxima dele que represente um caractere diferente da amostra classificada.

|   | 0    | 1    | 2    | 3    | 4    | 5    | 6    | 7    | 8    | 9    | A    | B    | C    | D    | E    | F    | G    | H    | I    | J    | K    | L    | M    | H    | O    | P    | Q    | R    | S    | T    | U    | V    | W    | X    | Y    | Z    |      |      |      |      |      |      |
|---|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| 0 | 0.00 | 0.67 | 0.38 | 0.48 | 0.43 | 0.59 | 0.66 | 0.45 | 0.60 | 0.37 | 0.67 | 0.32 | 0.63 | 0.34 | 0.70 | 0.32 | 0.39 | 0.23 | 0.33 | 0.61 | 0.30 | 0.48 | 0.58 | 0.55 | 0.69 | 0.61 | 0.39 | 0.52 | 0.14 | 0.53 | 0.20 | 0.47 | 0.44 | 0.81 | 0.33 | 0.63 | 0.38 | 0.76 | 0.78 | 0.64 |      |      |
| 1 | 0.67 | 0.00 | 0.68 | 0.68 | 0.48 | 0.58 | 0.72 | 0.65 | 0.75 | 0.69 | 0.62 | 0.64 | 0.74 | 0.59 | 0.74 | 0.71 | 0.81 | 0.74 | 0.65 | 0.39 | 0.69 | 0.74 | 0.65 | 0.64 | 0.82 | 1.00 | 0.71 | 0.67 | 0.69 | 0.85 | 0.64 | 0.71 | 0.63 | 0.67 | 0.70 | 0.72 | 0.66 | 0.70 | 0.65 | 0.70 |      |      |
| 2 | 0.38 | 0.68 | 0.00 | 0.38 | 0.42 | 0.75 | 0.64 | 0.69 | 0.40 | 0.54 | 0.49 | 0.58 | 0.40 | 0.54 | 0.66 | 0.58 | 0.54 | 0.68 | 0.68 | 0.71 | 0.61 | 0.72 | 0.68 | 0.69 | 0.62 | 0.50 | 0.63 | 0.62 | 0.62 | 0.60 | 0.56 | 0.63 | 0.62 | 0.62 | 0.63 | 0.62 | 0.59 | 0.62 | 0.60 | 0.58 | 0.62 | 0.60 |
| 3 | 0.48 | 0.68 | 0.38 | 0.00 | 0.46 | 0.62 | 0.69 | 0.63 | 0.57 | 0.44 | 0.57 | 0.36 | 0.52 | 0.47 | 0.66 | 0.49 | 0.59 | 0.52 | 0.59 | 0.59 | 0.67 | 0.68 | 0.68 | 0.67 | 0.67 | 0.50 | 0.48 | 0.52 | 0.49 | 0.52 | 0.72 | 0.60 | 0.68 | 0.68 | 0.58 | 0.68 | 0.68 | 0.66 | 0.71 | 0.45 | 0.52 |      |
| 4 | 0.43 | 0.48 | 0.46 | 0.00 | 0.64 | 0.69 | 0.47 | 0.59 | 0.48 | 0.47 | 0.59 | 0.37 | 0.69 | 0.44 | 0.59 | 0.46 | 0.63 | 0.72 | 0.45 | 0.60 | 0.67 | 0.49 | 0.73 | 0.68 | 0.62 | 0.45 | 0.63 | 0.63 | 0.39 | 0.68 | 0.51 | 0.63 | 0.68 | 0.67 | 0.63 | 0.67 | 0.71 | 0.55 | 0.61 | 0.74 | 0.69 |      |
| 5 | 0.59 | 0.58 | 0.75 | 0.62 | 0.64 | 0.00 | 0.36 | 0.65 | 0.49 | 0.58 | 0.74 | 0.54 | 0.74 | 0.63 | 0.42 | 0.63 | 0.69 | 0.71 | 0.74 | 0.78 | 0.63 | 0.63 | 0.58 | 0.69 | 0.63 | 0.85 | 0.66 | 0.62 | 0.60 | 0.74 | 0.58 | 0.63 | 0.65 | 0.78 | 0.67 | 0.71 | 0.55 | 0.61 | 0.74 | 0.69 |      |      |
| 6 | 0.67 | 0.72 | 0.69 | 0.69 | 0.63 | 0.36 | 0.00 | 0.69 | 0.38 | 0.61 | 0.74 | 0.63 | 0.77 | 0.74 | 0.42 | 0.69 | 0.72 | 0.73 | 0.75 | 0.61 | 0.68 | 0.75 | 0.66 | 0.59 | 0.62 | 0.68 | 0.64 | 0.72 | 0.67 | 0.68 | 0.62 | 0.68 | 0.64 | 0.68 | 0.64 | 0.70 | 0.65 | 0.69 | 0.59 | 0.35 | 0.74 | 0.63 |
| 7 | 0.45 | 0.65 | 0.64 | 0.63 | 0.47 | 0.65 | 0.69 | 0.00 | 0.59 | 0.39 | 0.66 | 0.47 | 0.67 | 0.38 | 0.63 | 0.51 | 0.48 | 0.49 | 0.56 | 0.45 | 0.56 | 0.63 | 0.64 | 0.66 | 0.64 | 0.56 | 0.63 | 0.56 | 0.49 | 0.64 | 0.51 | 0.32 | 0.70 | 0.49 | 0.68 | 0.58 | 0.59 | 0.70 | 0.59 | 0.70 | 0.59 |      |
| 8 | 0.60 | 0.75 | 0.68 | 0.67 | 0.65 | 0.69 | 0.00 | 0.48 | 0.71 | 0.50 | 0.74 | 0.63 | 0.51 | 0.37 | 0.58 | 0.65 | 0.61 | 0.67 | 0.52 | 0.65 | 0.73 | 0.60 | 0.60 | 0.68 | 0.73 | 0.68 | 0.72 | 0.62 | 0.67 | 0.50 | 0.67 | 0.61 | 0.68 | 0.56 | 0.59 | 0.70 | 0.59 | 0.70 | 0.59 | 0.70 | 0.59 |      |
| 9 | 0.37 | 0.69 | 0.38 | 0.44 | 0.48 | 0.58 | 0.61 | 0.39 | 0.48 | 0.00 | 0.69 | 0.28 | 0.63 | 0.33 | 0.65 | 0.31 | 0.47 | 0.46 | 0.38 | 0.53 | 0.46 | 0.51 | 0.66 | 0.59 | 0.60 | 0.60 | 0.52 | 0.43 | 0.57 | 0.44 | 0.46 | 0.30 | 0.76 | 0.47 | 0.67 | 0.52 | 0.66 | 0.66 | 0.75 | 0.57 | 0.42 |      |
| 0 | 0.67 | 0.62 | 0.40 | 0.57 | 0.48 | 0.74 | 0.66 | 0.71 | 0.69 | 0.00 | 0.66 | 0.46 | 0.66 | 0.67 | 0.65 | 0.67 | 0.63 | 0.68 | 0.47 | 0.49 | 0.63 | 0.68 | 0.57 | 0.54 | 0.37 | 0.51 | 0.40 | 0.43 | 0.37 | 0.78 | 0.48 | 0.63 | 0.49 | 0.68 | 0.71 | 0.58 | 0.71 | 0.58 | 0.71 | 0.58 |      |      |
| 1 | 0.32 | 0.64 | 0.54 | 0.36 | 0.41 | 0.54 | 0.63 | 0.47 | 0.50 | 0.28 | 0.66 | 0.00 | 0.56 | 0.30 | 0.64 | 0.26 | 0.49 | 0.45 | 0.50 | 0.60 | 0.39 | 0.47 | 0.49 | 0.63 | 0.68 | 0.57 | 0.54 | 0.37 | 0.51 | 0.40 | 0.43 | 0.37 | 0.78 | 0.48 | 0.63 | 0.49 | 0.68 | 0.71 | 0.58 | 0.71 | 0.58 |      |
| 2 | 0.63 | 0.74 | 0.49 | 0.52 | 0.59 | 0.74 | 0.67 | 0.74 | 0.63 | 0.46 | 0.56 | 0.00 | 0.55 | 0.67 | 0.59 | 0.65 | 0.67 | 0.62 | 0.55 | 0.64 | 0.66 | 0.73 | 0.73 | 0.69 | 0.77 | 0.60 | 0.68 | 0.63 | 0.44 | 0.87 | 0.51 | 0.55 | 0.64 | 0.70 | 0.58 | 0.64 | 0.70 | 0.58 | 0.64 | 0.70 | 0.58 |      |
| 3 | 0.43 | 0.59 | 0.55 | 0.47 | 0.37 | 0.63 | 0.74 | 0.38 | 0.63 | 0.33 | 0.66 | 0.30 | 0.55 | 0.00 | 0.72 | 0.39 | 0.57 | 0.43 | 0.55 | 0.64 | 0.43 | 0.46 | 0.52 | 0.59 | 0.68 | 0.72 | 0.58 | 0.50 | 0.42 | 0.56 | 0.42 | 0.45 | 0.32 | 0.79 | 0.46 | 0.65 | 0.53 | 0.71 | 0.72 | 0.64 |      |      |
| 4 | 0.70 | 0.74 | 0.70 | 0.66 | 0.69 | 0.42 | 0.42 | 0.69 | 0.51 | 0.65 | 0.70 | 0.64 | 0.67 | 0.72 | 0.60 | 0.68 | 0.72 | 0.75 | 0.72 | 0.71 | 0.68 | 0.69 | 0.72 | 0.74 | 0.61 | 0.76 | 0.68 | 0.68 | 0.70 | 0.71 | 0.68 | 0.64 | 0.66 | 0.73 | 0.75 | 0.70 | 0.59 | 0.51 | 0.73 | 0.66 | 0.64 |      |
| 5 | 0.32 | 0.71 | 0.58 | 0.44 | 0.63 | 0.69 | 0.42 | 0.57 | 0.31 | 0.65 | 0.26 | 0.59 | 0.33 | 0.68 | 0.00 | 0.50 | 0.34 | 0.44 | 0.54 | 0.37 | 0.44 | 0.52 | 0.67 | 0.58 | 0.61 | 0.53 | 0.52 | 0.38 | 0.44 | 0.41 | 0.37 | 0.38 | 0.79 | 0.46 | 0.67 | 0.53 | 0.71 | 0.74 | 0.58 | 0.64 |      |      |
| 6 | 0.39 | 0.81 | 0.61 | 0.57 | 0.58 | 0.69 | 0.64 | 0.57 | 0.58 | 0.47 | 0.67 | 0.49 | 0.65 | 0.57 | 0.72 | 0.50 | 0.00 | 0.43 | 0.48 | 0.56 | 0.32 | 0.66 | 0.77 | 0.67 | 0.65 | 0.48 | 0.71 | 0.65 | 0.35 | 0.59 | 0.39 | 0.57 | 0.47 | 0.71 | 0.47 | 0.66 | 0.68 | 0.73 | 0.74 | 0.61 |      |      |
| 7 | 0.23 | 0.74 | 0.59 | 0.53 | 0.46 | 0.71 | 0.72 | 0.48 | 0.65 | 0.46 | 0.67 | 0.45 | 0.67 | 0.43 | 0.75 | 0.34 | 0.43 | 0.00 | 0.53 | 0.60 | 0.33 | 0.60 | 0.33 | 0.67 | 0.57 | 0.70 | 0.57 | 0.63 | 0.56 | 0.23 | 0.51 | 0.29 | 0.49 | 0.49 | 0.79 | 0.36 | 0.69 | 0.64 | 0.79 | 0.80 | 0.64 |      |
| 8 | 0.53 | 0.85 | 0.54 | 0.47 | 0.63 | 0.74 | 0.73 | 0.49 | 0.61 | 0.38 | 0.65 | 0.50 | 0.62 | 0.55 | 0.72 | 0.44 | 0.48 | 0.53 | 0.00 | 0.34 | 0.49 | 0.58 | 0.67 | 0.30 | 0.54 | 0.44 | 0.66 | 0.63 | 0.56 | 0.48 | 0.58 | 0.47 | 0.47 | 0.69 | 0.58 | 0.64 | 0.79 | 0.80 | 0.64 |      |      |      |
| 9 | 0.61 | 0.93 | 0.65 | 0.59 | 0.72 | 0.78 | 0.75 | 0.58 | 0.67 | 0.33 | 0.61 | 0.60 | 0.55 | 0.64 | 0.71 | 0.54 | 0.56 | 0.60 | 0.34 | 0.00 | 0.58 | 0.59 | 0.78 | 0.85 | 0.53 | 0.56 | 0.62 | 0.61 | 0.35 | 0.69 | 0.47 | 0.57 | 0.69 | 0.64 | 0.68 | 0.61 | 0.65 | 0.71 | 0.63 | 0.63 |      |      |
| A | 0.30 | 0.69 | 0.60 | 0.52 | 0.45 | 0.63 | 0.61 | 0.45 | 0.52 | 0.35 | 0.66 | 0.39 | 0.64 | 0.43 | 0.68 | 0.37 | 0.32 | 0.33 | 0.49 | 0.58 | 0.00 | 0.57 | 0.68 | 0.59 | 0.64 | 0.56 | 0.66 | 0.57 | 0.26 | 0.57 | 0.28 | 0.51 | 0.34 | 0.72 | 0.42 | 0.67 | 0.62 | 0.72 | 0.56 | 0.64 |      |      |
| B | 0.48 | 0.74 | 0.68 | 0.59 | 0.60 | 0.63 | 0.68 | 0.56 | 0.65 | 0.46 | 0.73 | 0.47 | 0.66 | 0.46 | 0.69 | 0.44 | 0.66 | 0.53 | 0.58 | 0.57 | 0.00 | 0.61 | 0.64 | 0.51 | 0.66 | 0.31 | 0.29 | 0.33 | 0.50 | 0.56 | 0.41 | 0.57 | 0.88 | 0.40 | 0.64 | 0.44 | 0.66 | 0.75 | 0.70 | 0.64 |      |      |
| C | 0.58 | 0.65 | 0.71 | 0.59 | 0.67 | 0.58 | 0.75 | 0.63 | 0.73 | 0.51 | 0.80 | 0.49 | 0.73 | 0.52 | 0.72 | 0.32 | 0.77 | 0.67 | 0.67 | 0.78 | 0.68 | 0.64 | 0.00 | 0.86 | 0.70 | 0.63 | 0.60 | 0.66 | 0.73 | 0.61 | 0.60 | 0.66 | 0.87 | 0.68 | 0.76 | 0.52 | 0.71 | 0.81 | 0.70 | 0.64 |      |      |
| D | 0.55 | 0.64 | 0.61 | 0.67 | 0.43 | 0.69 | 0.66 | 0.64 | 0.60 | 0.66 | 0.64 | 0.63 | 0.75 | 0.59 | 0.74 | 0.67 | 0.67 | 0.57 | 0.80 | 0.85 | 0.59 | 0.64 | 0.86 | 0.00 | 0.80 | 0.80 | 0.70 | 0.67 | 0.59 | 0.79 | 0.61 | 0.80 | 0.47 | 0.64 | 0.70 | 0.73 | 0.61 | 0.80 | 0.47 | 0.64 | 0.66 |      |
| E | 0.69 | 0.92 | 0.72 | 0.68 | 0.75 | 0.65 | 0.59 | 0.66 | 0.60 | 0.59 | 0.69 | 0.63 | 0.69 | 0.68 | 0.61 | 0.38 | 0.65 | 0.70 | 0.54 | 0.53 | 0.64 | 0.51 | 0.70 | 0.80 | 0.00 | 0.60 | 0.53 | 0.47 | 0.69 | 0.56 | 0.69 | 0.48 | 0.62 | 0.73 | 0.63 | 0.64 | 0.49 | 0.44 | 0.64 | 0.62 |      |      |
| F | 0.61 | 0.00 | 0.69 | 0.68 | 0.68 | 0.76 | 0.85 | 0.74 | 0.64 | 0.68 | 0.80 | 0.80 | 0.88 | 0.77 | 0.72 | 0.76 | 0.81 | 0.48 | 0.57 | 0.44 | 0.56 | 0.66 | 0.68 | 0.87 | 0.80 | 0.60 | 0.00 | 0.72 | 0.67 | 0.59 | 0.63 | 0.61 | 0.64 | 0.62 | 0.82 | 0.54 | 0.67 | 0.72 | 0.70 | 0.77 | 0.68 |      |
| G | 0.59 | 0.71 | 0.69 | 0.67 | 0.68 | 0.66 | 0.72 | 0.65 | 0.73 | 0.60 | 0.68 | 0.57 | 0.60 | 0.66 | 0.50 | 0.71 | 0.63 | 0.68 | 0.62 | 0.66 | 0.31 | 0.65 | 0.70 | 0.53 | 0.72 | 0.00 | 0.35 | 0.62 | 0.52 | 0.64 | 0.49 | 0.65 | 0.82 | 0.51 | 0.52 | 0.63 | 0.63 | 0.73 | 0.68 | 0.73 |      |      |
| H | 0.52 | 0.67 | 0.72 | 0.67 | 0.62 | 0.67 | 0.66 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 | 0.68 |      |
| I | 0.14 | 0.69 | 0.59 | 0.50 | 0.45 | 0.60 | 0.63 | 0.49 | 0.58 | 0.37 | 0.63 | 0.42 | 0.70 | 0.38 | 0.35 | 0.23 | 0.36 | 0.61 | 0.26 | 0.59 | 0.66 | 0.39 | 0.69 | 0.59 | 0.62 | 0.55 | 0.00 | 0.53 | 0.17 | 0.50 | 0.46 | 0.71 | 0.34 | 0.63 | 0.60 | 0.76 | 0.66 | 0.65 | 0.63 |      |      |      |
| J | 0.53 | 0.85 | 0.58 | 0.48 | 0.63 | 0.74 | 0.78 | 0.64 | 0.72 | 0.57 | 0.57 | 0.51 | 0.44 | 0.58 | 0.51 | 0.48 | 0.35 | 0.37 | 0.50 | 0.73 | 0.79 | 0.56 | 0.63 | 0.52 | 0.57 | 0.53 | 0.00 | 0.61 | 0.33 | 0.60 | 0.74 | 0.60 | 0.65 | 0.56 | 0.70 | 0.70 | 0.66 | 0.65 | 0.56 | 0.70 | 0.66 |      |
| K | 0.20 | 0.64 | 0.60 | 0.52 | 0.45 | 0.58 | 0.62 | 0.51 | 0.62 | 0.40 | 0.70 | 0.40 | 0.67 | 0.42 | 0.69 | 0.41 | 0.39 | 0.29 | 0.38 | 0.69 | 0.28 | 0.56 | 0.61 | 0.55 | 0.69 | 0.61 | 0.64 | 0.54 | 0.17 | 0.61 | 0.00 | 0.51 | 0.48 | 0.80 | 0.38 | 0.65 | 0.61 | 0.76 | 0.79 | 0.66 |      |      |
| L | 0.47 | 0.71 | 0.58 | 0.49 | 0.53 | 0.63 | 0.68 | 0.55 | 0.67 | 0.46 | 0.59 | 0.43 | 0.51 | 0.45 | 0.64 | 0.37 | 0.57 | 0.49 | 0.47 | 0.47 | 0.51 | 0.41 | 0.60 | 0.73 | 0.48 | 0.64 | 0.49 | 0.43 | 0.50 | 0.33 | 0.51 | 0.00 |      |      |      |      |      |      |      |      |      |      |

A figura 4.70 ilustra o conceito do fator de confiabilidade.

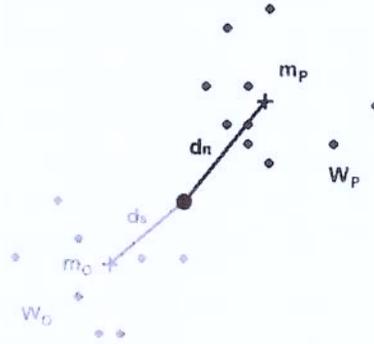


Fig. 4.70: Representação do fator de confiabilidade no reconhecimento.

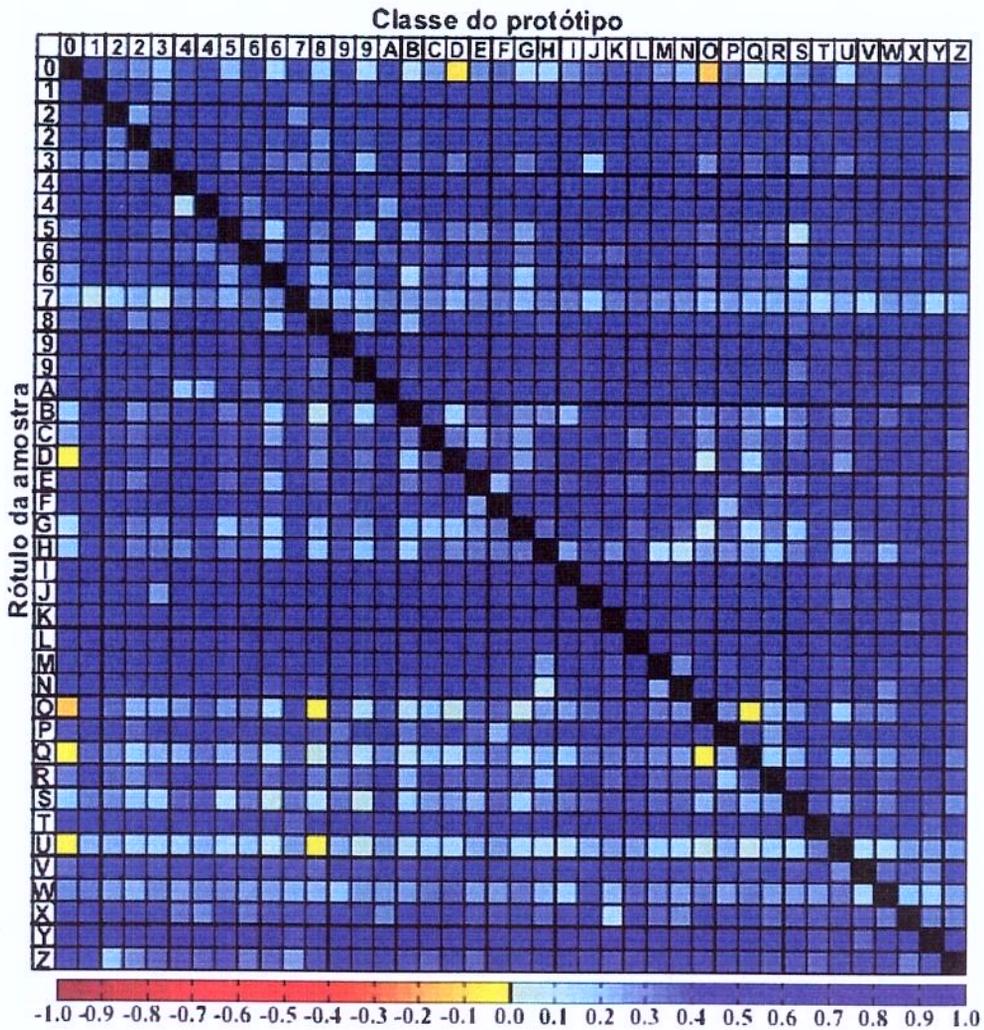


Fig. 4.71: Representação dos fatores de confiabilidade no reconhecimento entre os protótipos.

Classe do protótipo mais próximo

|   | 0    | 1    | 2    | 3    | 4    | 5    | 6    | 7    | 8    | 9    | A    | B    | C    | D    | E    | F    | G    | H    | I    | J    | K    | L    | M    | N    | O    | P    | Q    | R    | S    | T    | U    | V    | W    | X    | Y    | Z    |      |      |      |      |      |      |
|---|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| 0 | 0.0  | 42.0 | 36.6 | 31.0 | 24.9 | 38.2 | 44.1 | 24.6 | 42.4 | 15.7 | 42.6 | 14.0 | 43.3 | 11.2 | 26.3 | 6.3  | 28.1 | 35.5 | 15.0 | 16.7 | 28.6 | 36.3 | 38.5 | 39.4 | 30.9 | 24.6 | -13  | 31.4 | 7.4  | 20.3 | 30.2 | 50.1 | 17.0 | 40.8 | 33.2 | 44.4 | 48.8 | 38.5 |      |      |      |      |
| 1 | 45.0 | 0.0  | 45.3 | 47.7 | 29.7 | 43.1 | 49.8 | 44.4 | 51.7 | 48.1 | 33.7 | 45.9 | 51.3 | 41.6 | 50.7 | 47.2 | 53.5 | 47.7 | 54.8 | 57.5 | 47.2 | 48.9 | 48.7 | 42.5 | 53.9 | 60.0 | 49.1 | 47.1 | 46.4 | 53.7 | 44.2 | 46.2 | 44.5 | 46.3 | 46.9 | 46.9 | 47.5 | 46.1 |      |      |      |      |
| 2 | 45.1 | 53.0 | 0.0  | 25.1 | 38.2 | 56.5 | 58.2 | 50.5 | 54.2 | 45.2 | 30.1 | 41.6 | 42.7 | 44.1 | 54.6 | 44.3 | 48.5 | 46.1 | 40.7 | 48.7 | 48.5 | 50.5 | 52.2 | 50.9 | 54.1 | 51.5 | 51.4 | 32.7 | 47.0 | 43.4 | 48.0 | 43.9 | 46.4 | 51.3 | 48.8 | 51.2 | 50.1 | 49.0 | 50.5 | 23.2 |      |      |
| 3 | 41.6 | 50.5 | 32.2 | 0.0  | 39.0 | 49.3 | 53.7 | 50.1 | 48.5 | 40.1 | 48.3 | 39.2 | 42.7 | 38.1 | 50.3 | 40.3 | 45.8 | 46.9 | 42.3 | 50.2 | 45.1 | 50.4 | 47.0 | 54.1 | 54.5 | 55.5 | 53.0 | 53.1 | 43.9 | 41.8 | 43.2 | 44.0 | 42.5 | 55.2 | 50.0 | 51.8 | 46.3 | 50.9 | 53.9 | 42.2 |      |      |
| 4 | 32.4 | 32.0 | 43.2 | 43.3 | 4.0  | 47.9 | 35.8 | 42.9 | 36.3 | 32.3 | 42.7 | 22.7 | 30.5 | 35.2 | 42.8 | 32.6 | 46.8 | 49.3 | 34.4 | 40.7 | 47.0 | 52.2 | 51.2 | 48.2 | 46.1 | 30.9 | 45.2 | 34.8 | 41.6 | 29.4 | 37.4 | 32.9 | 43.4 | 46.3 | 49.0 | 46.0 | 37.7 | 40.4 | 46.0 | 37.7 |      |      |
| 5 | 66.3 | 65.6 | 71.8 | 67.6 | 68.0 | 0.0  | 51.7 | 68.8 | 61.4 | 66.1 | 70.9 | 63.5 | 70.8 | 68.4 | 64.2 | 68.2 | 69.4 | 71.1 | 72.0 | 72.7 | 68.2 | 68.0 | 65.6 | 70.6 | 69.0 | 75.2 | 69.4 | 67.7 | 66.9 | 71.7 | 65.9 | 67.8 | 68.7 | 72.2 | 69.9 | 70.3 | 64.2 | 66.1 | 71.2 | 69.2 |      |      |
| 6 | 46.3 | 51.0 | 54.5 | 49.1 | 50.9 | 11.6 | 0.0  | 48.7 | 32.9 | 43.6 | 55.1 | 45.3 | 54.0 | 50.5 | 26.3 | 39.3 | 48.3 | 30.7 | 52.0 | 53.1 | 45.4 | 48.7 | 49.3 | 51.3 | 48.1 | 53.7 | 50.7 | 47.5 | 45.4 | 53.9 | 44.0 | 37.7 | 41.9 | 50.3 | 38.0 | 49.1 | 42.9 | 50.1 | 51.1 | 46.8 |      |      |
| 7 | 39.6 | 47.1 | 45.8 | 44.3 | 36.7 | 36.3 | 47.9 | 0.0  | 40.2 | 20.7 | 48.3 | 37.7 | 49.1 | 18.6 | 49.6 | 22.3 | 38.4 | 34.3 | 30.4 | 39.0 | 27.2 | 41.5 | 46.3 | 49.3 | 44.1 | 43.4 | 48.3 | 44.3 | 45.5 | 43.6 | 39.3 | 40.7 | 41.9 | 50.3 | 38.0 | 49.1 | 42.9 | 50.1 | 51.1 | 46.8 |      |      |
| 8 | 30.2 | 54.9 | 47.9 | 39.0 | 41.2 | 48.9 | 49.3 | 27.8 | 41.1 | 0.0  | 52.3 | 23.0 | 51.1 | 30.5 | 53.0 | 14.8 | 37.1 | 33.7 | 23.7 | 40.5 | 20.2 | 38.9 | 44.8 | 53.0 | 46.6 | 44.4 | 48.1 | 45.3 | 33.8 | 43.2 | 35.4 | 36.5 | 20.6 | 55.1 | 37.7 | 52.8 | 45.0 | 54.0 | 56.1 | 46.2 |      |      |
| 9 | 22.8 | 6.2  | 16.4 | 24.0 | 6.5  | 28.7 | 31.9 | 24.8 | 32.5 | 28.7 | 0.0  | 25.5 | 22.1 | 22.6 | 32.4 | 26.4 | 28.3 | 30.7 | 32.4 | 23.8 | 31.5 | 32.8 | 21.4 | 33.4 | 37.2 | 29.7 | 28.6 | 22.2 | 27.4 | 22.4 | 22.2 | 22.7 | 24.2 | 22.2 | 22.1 | 14.8 | 24.1 | 14.8 | 26.7 | 27.9 | 14.9 | 22.2 |
| A | 35.5 | 53.7 | 44.1 | 30.4 | 36.7 | 52.1 | 55.9 | 44.8 | 47.3 | 23.9 | 54.1 | 0.0  | 50.9 | 27.5 | 55.6 | 23.9 | 48.9 | 43.9 | 51.5 | 41.6 | 39.4 | 46.3 | 52.9 | 32.9 | 56.5 | 48.3 | 46.8 | 41.2 | 46.3 | 43.7 | 38.7 | 36.1 | 60.7 | 44.8 | 55.0 | 45.5 | 55.1 | 57.2 | 47.4 |      |      |      |
| B | 68.7 | 72.6 | 60.8 | 62.9 | 66.2 | 71.9 | 73.4 | 69.3 | 72.4 | 69.0 | 57.7 | 65.9 | 0.0  | 65.6 | 69.3 | 67.0 | 69.9 | 65.4 | 69.5 | 70.2 | 72.1 | 79.5 | 71.0 | 73.8 | 67.7 | 70.8 | 68.6 | 68.1 | 70.2 | 69.3 | 68.5 | 68.1 | 68.9 | 71.7 | 66.8 | 67.0 | 67.6 | 64.2 | 67.8 |      |      |      |
| C | 48.0 | 49.3 | 49.2 | 47.1 | 47.9 | 20.9 | 22.0 | 49.8 | 36.4 | 46.3 | 48.8 | 44.1 | 48.2 | 49.8 | 0.0  | 46.9 | 51.8 | 51.6 | 52.8 | 51.5 | 48.7 | 47.9 | 46.6 | 48.4 | 45.2 | 54.8 | 46.1 | 47.2 | 48.2 | 50.0 | 48.1 | 46.3 | 48.1 | 51.4 | 51.7 | 49.3 | 42.8 | 35.5 | 51.2 | 45.3 |      |      |
| D | 20.4 | 48.0 | 41.4 | 29.7 | 34.5 | 43.1 | 50.4 | 33.9 | 45.5 | 18.9 | 49.0 | 8.9  | 45.3 | 16.3 | 48.2 | 0.0  | 41.1 | 16.4 | 36.2 | 44.1 | 29.3 | 30.5 | 24.3 | 49.3 | 43.6 | 47.7 | 41.0 | 35.5 | 29.4 | 26.0 | 24.0 | 32.2 | 56.1 | 33.4 | 47.3 | 36.5 | 48.5 | 52.6 | 43.0 |      |      |      |
| E | 27.9 | 52.6 | 40.1 | 37.3 | 42.5 | 47.1 | 37.3 | 42.5 | 48.3 | 38.8 | 40.8 | 25.8 | 46.2 | 34.3 | 45.3 | 38.7 | 46.8 | 33.9 | 0.0  | 31.6 | 24.7 | 36.7 | 26.2 | 43.9 | 45.8 | 47.1 | 41.6 | 43.2 | 47.8 | 44.4 | 28.7 | 41.9 | 28.8 | 38.8 | 33.3 | 48.3 | 35.1 | 45.9 | 45.7 | 46.1 |      |      |
| F | 1.6  | 53.9 | 46.2 | 39.5 | 34.7 | 52.7 | 53.9 | 39.3 | 49.3 | 37.7 | 51.7 | 31.0 | 52.2 | 34.1 | 34.2 | 20.3 | 35.9 | 0.0  | 42.1 | 47.9 | 27.6 | 40.0 | 49.5 | 44.5 | 51.2 | 43.0 | 46.4 | 42.8 | 7.9  | 43.1 | 16.8 | 40.8 | 40.6 | 56.2 | 26.2 | 59.9 | 50.8 | 36.7 | 57.3 | 49.0 |      |      |
| G | 40.7 | 53.9 | 39.3 | 32.9 | 47.3 | 47.5 | 47.3 | 36.7 | 41.5 | 24.3 | 46.9 | 35.9 | 42.0 | 39.8 | 45.9 | 33.0 | 38.9 | 43.2 | 0.0  | 24.2 | 38.2 | 47.1 | 39.2 | 35.7 | 45.2 | 44.0 | 42.8 | 38.5 | 43.6 | 36.3 | 35.8 | 46.6 | 44.2 | 47.1 | 39.5 | 42.4 | 48.1 | 38.0 |      |      |      |      |
| H | 45.2 | 60.0 | 48.1 | 42.3 | 51.9 | 53.8 | 52.7 | 42.4 | 49.2 | 38.2 | 46.6 | 43.5 | 39.3 | 45.7 | 51.0 | 41.0 | 41.0 | 45.4 | 42.6 | 50.3 | 56.9 | 41.8 | 40.8 | 46.3 | 46.6 | 46.6 | 24.0 | 50.3 | 38.0 | 44.1 | 49.1 | 47.9 | 50.2 | 43.4 | 48.2 | 50.7 | 46.6 |      |      |      |      |      |
| I | 15.4 | 44.2 | 38.3 | 33.6 | 28.2 | 39.3 | 38.3 | 19.4 | 32.2 | 15.9 | 40.7 | 23.3 | 40.0 | 28.1 | 40.1 | 20.8 | 21.1 | 19.0 | 27.2 | 33.0 | 0.0  | 35.7 | 43.2 | 37.7 | 41.8 | 34.4 | 41.5 | 37.0 | 7.5  | 34.6 | 15.0 | 31.7 | 18.9 | 48.7 | 24.4 | 39.9 | 38.1 | 46.2 | 45.2 | 40.8 |      |      |
| J | 19.7 | 43.1 | 40.1 | 32.9 | 34.1 | 31.2 | 38.8 | 31.8 | 38.8 | 20.2 | 44.1 | 18.8 | 39.1 | 21.2 | 40.1 | 18.7 | 38.0 | 29.1 | 31.7 | 32.7 | 32.2 | 0.0  | 27.5 | 36.9 | 31.0 | 39.0 | 15.4 | 12.2 | 27.0 | 30.0 | 27.3 | 19.7 | 33.9 | 51.2 | 20.4 | 36.1 | 23.3 | 36.0 | 45.1 | 41.1 |      |      |
| K | 62.2 | 64.2 | 68.5 | 62.8 | 66.8 | 59.0 | 68.0 | 65.5 | 67.8 | 58.3 | 69.3 | 59.2 | 67.8 | 58.8 | 69.6 | 66.8 | 67.3 | 71.0 | 66.4 | 64.1 | 0.0  | 72.6 | 67.6 | 72.8 | 65.1 | 63.4 | 65.7 | 69.0 | 63.9 | 64.2 | 66.0 | 73.1 | 66.9 | 69.5 | 58.1 | 69.3 | 71.3 | 68.2 |      |      |      |      |
| L | 43.6 | 42.4 | 44.5 | 48.0 | 27.9 | 47.1 | 48.9 | 47.3 | 46.4 | 48.2 | 45.2 | 46.1 | 51.4 | 42.5 | 51.3 | 47.5 | 51.0 | 45.4 | 55.7 | 57.9 | 56.2 | 47.4 | 54.2 | 0.0  | 55.0 | 55.7 | 51.0 | 49.4 | 43.2 | 54.9 | 43.4 | 51.4 | 45.7 | 55.2 | 35.4 | 46.9 | 49.6 | 51.4 | 47.3 |      |      |      |
| M | 55.9 | 60.9 | 57.2 | 54.9 | 59.0 | 51.7 | 50.4 | 55.9 | 51.2 | 48.4 | 57.1 | 52.1 | 56.1 | 54.9 | 51.1 | 49.9 | 54.8 | 57.3 | 46.2 | 47.7 | 54.1 | 47.4 | 50.8 | 62.5 | 0.0  | 52.0 | 47.4 | 43.4 | 57.0 | 51.2 | 56.6 | 43.7 | 52.7 | 58.3 | 54.8 | 55.2 | 41.6 | 35.7 | 53.9 | 52.0 |      |      |
| N | 55.7 | 70.5 | 59.8 | 58.5 | 63.7 | 65.5 | 62.1 | 58.0 | 59.1 | 55.3 | 64.5 | 61.2 | 62.3 | 61.1 | 62.6 | 56.9 | 57.9 | 55.3 | 44.2 | 51.3 | 54.0 | 59.1 | 66.5 | 64.8 | 56.6 | 0.0  | 61.6 | 60.0 | 54.9 | 56.5 | 56.7 | 58.2 | 57.0 | 64.9 | 53.0 | 58.7 | 60.8 | 60.3 | 63.1 | 60.2 |      |      |
| O | 43.2 | 52.0 | 53.9 | 50.1 | 49.1 | 47.8 | 51.7 | 44.3 | 50.9 | 38.3 | 54.1 | 40.4 | 52.3 | 38.8 | 51.7 | 40.3 | 51.7 | 45.7 | 48.0 | 47.4 | 45.8 | 8.3  | 41.9 | 51.8 | 38.6 | 52.7 | 26.5 | 0.0  | 46.4 | 44.4 | 43.9 | 45.8 | 32.9 | 46.9 | 58.0 | 39.2 | 50.8 | 30.0 | 47.8 | 51.5 | 55.0 |      |
| P | 24.2 | 56.8 | 42.7 | 37.1 | 47.3 | 52.3 | 53.6 | 46.7 | 50.5 | 49.4 | 41.8 | 38.2 | 33.8 | 44.6 | 50.7 | 32.4 | 42.5 | 41.0 | 34.4 | 23.1 | 44.3 | 41.0 | 51.0 | 55.3 | 41.6 | 45.1 | 43.2 | 47.0 | 41.0 | 0.0  | 43.9 | 25.9 | 45.5 | 53.4 | 45.4 | 46.8 | 43.0 | 49.1 | 50.0 | 42.4 |      |      |
| Q | 2.8  | 34.7 | 28.6 | 15.7 | 23.0 | 22.5 | 32.9 | 24.6 | 30.2 | 10.2 | 37.0 | 2.2  | 31.7 | 11.1 | 30.5 | 9.4  | 12.1 | 15.4 | 25.1 | 33.9 | 15.0 | 23.7 | 19.9 | 32.5 | 44.0 | 33.0 | 30.1 | 26.2 | -1.8 | 27.2 | 0.0  | 18.5 | 23.1 | 44.2 | 20.4 | 31.0 | 26.5 | 37.3 | 40.6 | 31.9 |      |      |
| R | 33.6 | 40.1 | 34.8 | 30.9 | 37.9 | 45.9 | 46.9 | 40.3 | 48.6 | 35.4 | 38.8 | 31.6 | 32.3 | 34.8 | 44.1 | 24.8 | 41.0 | 34.8 | 34.7 | 34.4 | 36.5 | 24.8 | 45.1 | 49.5 | 34.9 | 45.3 | 37.0 | 34.6 | 35.9 | 19.4 | 36.3 | 0.0  | 37.6 | 48.3 | 39.9 | 43.9 | 36.4 | 41.9 | 48.1 | 40.0 |      |      |
| S | 19.3 | 33.4 | 28.4 | 19.3 | 18.6 | 36.9 | 40.1 | 10.1 | 32.2 | 5.3  | 34.3 | 12.3 | 31.5 | 3.3  | 4.3  | 15.8 | 27.8 | 26.4 | 23.7 | 35.2 | 16.3 | 29.6 | 34.1 | 36.4 | 39.9 | 38.6 | 35.4 | 35.8 | 23.4 | 22.9 | 27.9 | 0.0  | 39.5 | 26.5 | 38.7 | 32.7 | 40.9 | 40.2 | 30.2 |      |      |      |
| T | 34.7 | 49.3 | 49.6 | 53.1 | 50.6 | 55.4 | 53.5 | 52.6 | 52.0 | 54.6 | 37.1 | 55.9 | 50.0 | 57.0 | 45.5 |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |

Um fator de confiabilidade de 100% somente é obtido se o padrão a ser reconhecido for idêntico a uma das amostras rotuladas ( $d_s = 0$ ) e um fator de confiabilidade de 0% indica o máximo conflito entre dois caracteres ( $d_s = d_n$ ).

A figura 4.71 ilustra a situação da fronteira das classes, de modo que podemos ter uma idéia do conflito entre algumas classes. Para isto, calculamos o fator de confiabilidade no reconhecimento para cada amostra rotulada em relação aos protótipos de cada classe, mostrando o menor valor encontrado, o qual indica a maior proximidade da fronteira de decisão. As linhas desta imagem indicam o rótulo das amostras e as colunas indicam o protótipo da classe de caractere utilizado para este cálculo. Os valores estão representados na barra de cores, onde quanto maior o valor, menor é a possibilidade da ocorrência de erro no reconhecimento do caractere, sendo que um valor negativo indica que uma amostra ultrapassou a fronteira de decisão de sua classe.

Os valores da figura 4.71 podem ser visualizados na tabela 4.3, a qual utiliza a mesma representação que a figura.

### Classificador Hierárquico de Distância Mínima

Este classificador utiliza a técnica de “*dividir para conquistar*”, tratando o problema em dois níveis, o que permite uma simplificação da tarefa. Nós elaboramos este classificador da seguinte maneira:

1. Um *classificador de distância mínima* ao protótipo, tendo como medida de similaridade a *distância euclidiana*, é utilizado para determinar os 3 protótipos mais semelhantes ao padrão que desejamos reconhecer;
2. Os 3 protótipos, resultantes do nível anterior, são combinados 2 a 2 para efetuar um torneio da seguinte maneira:
  - (a) Para cada par de protótipos é efetuada uma nova classificação do padrão, porém os vetores de características do protótipo e do padrão são ponderados pela diferença entre os protótipos, visando dar mais importância às características com maior diferença, ou seja, é utilizada a *distância euclidiana ponderada* como medida de similaridade;
  - (b) Para cada protótipo é computada a quantidade de vitórias e o menor fator de confiabilidade de reconhecimento obtidos nas vitórias;
  - (c) O padrão é classificado ao protótipo com maior quantidade de vitórias;
  - (d) Caso ocorra um empate, o padrão será classificado ao protótipo com menor fator de confiabilidade de reconhecimento.

Com base na sintaxe da placa (“AAA-9999”), é possível aplicar o classificador apenas em letras ou em números, minimizando o problema de conflito entre as classes e melhorando a performance no reconhecimento dos caracteres.

No cálculo da *distância euclidiana ponderada*, nós utilizamos a diferença entre os protótipos, como vetor de pesos, devido ao baixo custo computacional. Porém, é possível utilizar a variância das amostras pertencentes à mesma classe de um dos dois protótipos. Atualmente, existem várias pesquisas, as quais estudam a viabilidade de utilizar técnicas de computação evolutiva, tais como algoritmo genético, para definir estes pesos. Também estudamos esta possibilidade, sendo descrita em Dias (2004).

A figura 4.72 ilustra a classificação do padrão de um caractere '8', sendo reconhecido pela quantidade de vitórias. Na figura 4.73 é ilustrada a classificação do padrão de um caractere 'D' pelo critério de desempate.

```

1° Nível (0123456789)
      689
2° Nível (689)
      6x8      6x9      8x9
vencedor   8      9      8
classificação:
caractere   vitórias   Cfator (mín.)
      8           2       0,53
      9           1       0,01
      6           0       -
caractere reconhecido como (8)

```

**Fig. 4.72:** Classificador hierárquico reconhecendo o caractere '8'.

```

1° Nível (ABCDEFGHIJKLMNPOQRSTUVWXYZ)
      DOQ
2° Nível (DOQ)
      DxO      DxQ      OxQ
vencedor   O      D      Q
classificação:
caractere   vitórias   Cfator (mín.)
      D           1       0,12
      O           1       0,06
      Q           1       0,03
caractere reconhecido como (D)

```

**Fig. 4.73:** Classificador hierárquico reconhecendo o caractere 'D'.

### 4.6.3 Resultado do Módulo de Reconhecimento dos Caracteres

Como resultado do módulo de *Reconhecimento dos Caracteres*, obtemos o código ASCII de cada caractere no formato da sintaxe informada anteriormente (“AAA-9999”).

A tabela 4.4 ilustra o resultado obtido no reconhecimento dos caracteres da figura 4.65. Nesta tabela constatamos que todos os caracteres foram reconhecidos corretamente, tendo um bom fator de confiabilidade nos resultados.

|                                |          |          |          |          |          |          |          |          |
|--------------------------------|----------|----------|----------|----------|----------|----------|----------|----------|
| <i>Reconhecido</i>             | <b>B</b> | <b>P</b> | <b>T</b> | <b>-</b> | <b>1</b> | <b>8</b> | <b>2</b> | <b>2</b> |
| <i>Fator de Confiabilidade</i> | 33%      | 59%      | 75%      | 100%     | 39%      | 61%      | 57%      | 53%      |
| <i>Próximo Caractere</i>       | G        | R        | Y        | -        | 3        | 9        | 7        | 7        |

**Tab. 4.4:** Resultado do módulo de reconhecimento dos caracteres.

# Capítulo 5

## Análise dos Resultados

Neste capítulo, são analisados os resultados obtidos com nossa metodologia. Adotamos o mesmo critério para a análise que a maioria dos autores, visando comparar nossa proposta com as demais. Este critério consiste em obter a porcentagem de sucesso na localização da placa e no correto reconhecimento da cadeia de caracteres.

Para efetuar os testes, utilizamos uma base de dados contendo 1200 imagens de veículos com placa de licenciamento no formato brasileiro, tendo resolução de 320x240 em 24 *bits* de cor, gentilmente cedidas pela empresa *SDC Engenharia e Sistemas*. Estas imagens foram obtidas em estacionamentos particulares com a câmera posicionada na lateral da passagem do veículo, em diferentes condições de iluminação. Portanto, apresentam distorção de perspectiva, dificultando a separação e o reconhecimento dos caracteres. Algumas imagens foram obtidas em más condições de iluminação, prejudicando o processo de binarização. A figura 5.1 ilustra algumas imagens desta base de dados.

Para ilustrar as dificuldades apresentadas pelas 1200 imagens da base de dados, mostramos nas figuras 5.2, 5.3, 5.4 e 5.5 a região promissora retornada pelo módulo de *Localização da Placa* de 128 imagens escolhidas aleatoriamente. Também mostramos o resultado do módulo de *Enquadramento da Placa* das mesmas 128 imagens nas figuras 5.6 e 5.7, nas quais podemos constatar a eficiência no enquadramento com correção da distorção de perspectivas obtidos na maioria dos casos, o que facilita a performance da separação e reconhecimento dos caracteres.

Inicialmente, nas seções 5.1 e 5.2, analisamos algumas configurações de nossa metodologia. Em seguida, na seção 5.4, comparamos nossa metodologia com duas demonstrações de sistemas comerciais disponíveis na internet: *SeeCar* (Solutions, 2003) e *SLAV 3.0* (Pixon, 2005).

Em todas as análises foram utilizadas as 1200 imagens do banco de dados mencionado.



Fig. 5.1: Algumas imagens do banco de dados com 1200 imagens utilizadas na análise.



Fig. 5.2: Amostras de 128 regiões promissoras obtidas aleatoriamente das 1200 imagens (1 a 32).



Fig. 5.3: Amostras de 128 regiões promissoras obtidas aleatoriamente das 1200 imagens (33 a 64).



Fig. 5.4: Amostras de 128 regiões promissoras obtidas aleatoriamente das 1200 imagens (65 a 96).



Fig. 5.5: Amostras de 128 regiões promissoras obtidas aleatoriamente das 1200 imagens (97 a 128).



Fig. 5.6: Amostras de 128 placas enquadradas com correção da distorção de perspectiva (1 a 64).



Fig. 5.7: Amostras de 128 placas enquadradas com correção da distorção de perspectiva (65 a 128).

Verificamos que, na maioria dos casos, o enquadramento com correção da distorção de perspectiva da placa foi bem sucedido.

Também constatamos a eficiência na utilização dos segmentos verticais “virtuais” para corrigir a inclinação dos caracteres nas seguintes placas: (localização das placas no formato coluna, linha) (3, 3), (1, 7) e (4, 14) da figura 5.6; (4, 6), (3, 7), (1, 10) e (3, 10) da figura 5.7, nas quais as partes do pára-choque do veículo são removidas pelo módulo de *Separação dos Caracteres*. Entretanto, um contra exemplo é a placa (1, 9) da figura 5.6, onde os segmentos verticais “virtuais” não ajudaram na estimativa da inclinação dos caracteres.

Algumas placas tiveram problemas em seu enquadramento, tendo como conseqüência erro na correção da distorção de perspectiva. Dentre os principais problemas podemos citar: objetos ocultando parte da placa (1, 5) da figura 5.6 e (1, 1) da figura 5.7; baixo contraste entre a placa e o local de fixação nas placas (1, 2), (3, 7), (4, 11) e (1, 15) da figura 5.7; e falha das funções-objetivo nas placas (4, 5), (1, 9), (2, 11) e (2, 14) da figura 5.6; (2, 4), (1, 11), (1, 12) e (4, 16) da figura 5.7.

É interessante notar que existem casos em que aparentemente não se vê motivos para os erros ocorridos. Salientamos entretanto que toda a filosofia do desenvolvimento do sistema foi de minimizar os erros globais e não tratar os casos individuais.

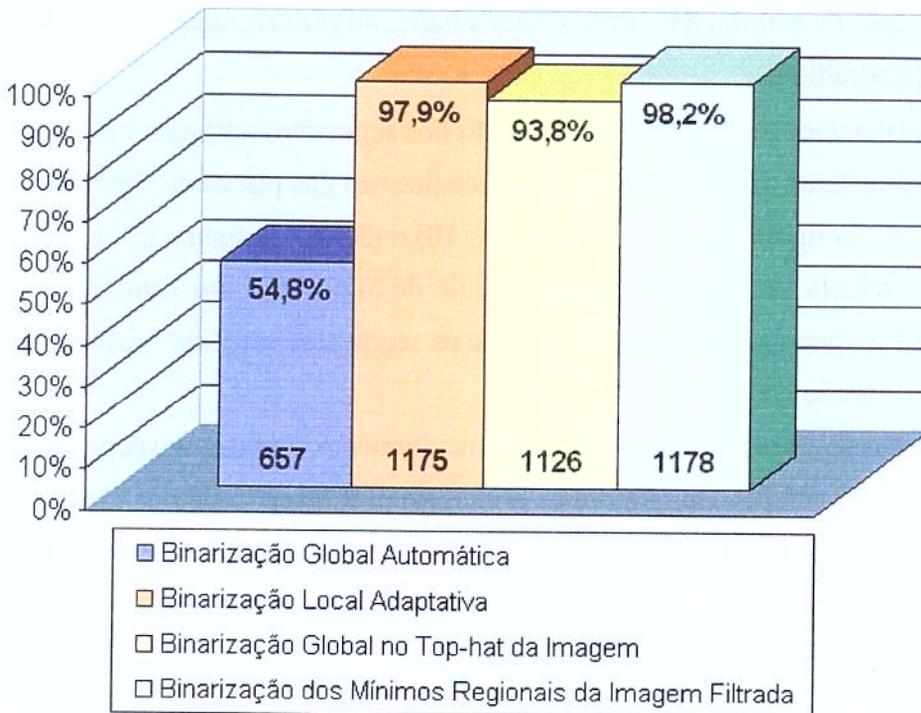
## 5.1 Análise da Localização da Placa

A principal etapa do módulo de *Localização da Placa* que propomos é a *binarização*, pois uma vez que cada caractere da placa seja definido por um componente conexo, o filtro que elaboramos com base na morfologia dos caracteres garante a localização das regiões que contenham um conjunto de objetos com características semelhantes à da placa de licenciamento (quantidade determinada de objetos de altura e espaçamento regular, alinhados horizontalmente).

Por esta razão, analisamos 4 métodos de binarização:

1. *Método Global Automático* de Otsu (1979);
2. *Método Local Adaptativo* de Niblack (1986);
3. *Método Global no Top-hat da Imagem* de Martín et al. (2000);
4. *Método dos Mínimos Regionais da Imagem Filtrada*, por nós idealizado.

A figura 5.8 ilustra o sucesso na localização da placa por proposta, na qual constatamos que o método de binarização que propomos obteve o melhor resultado.



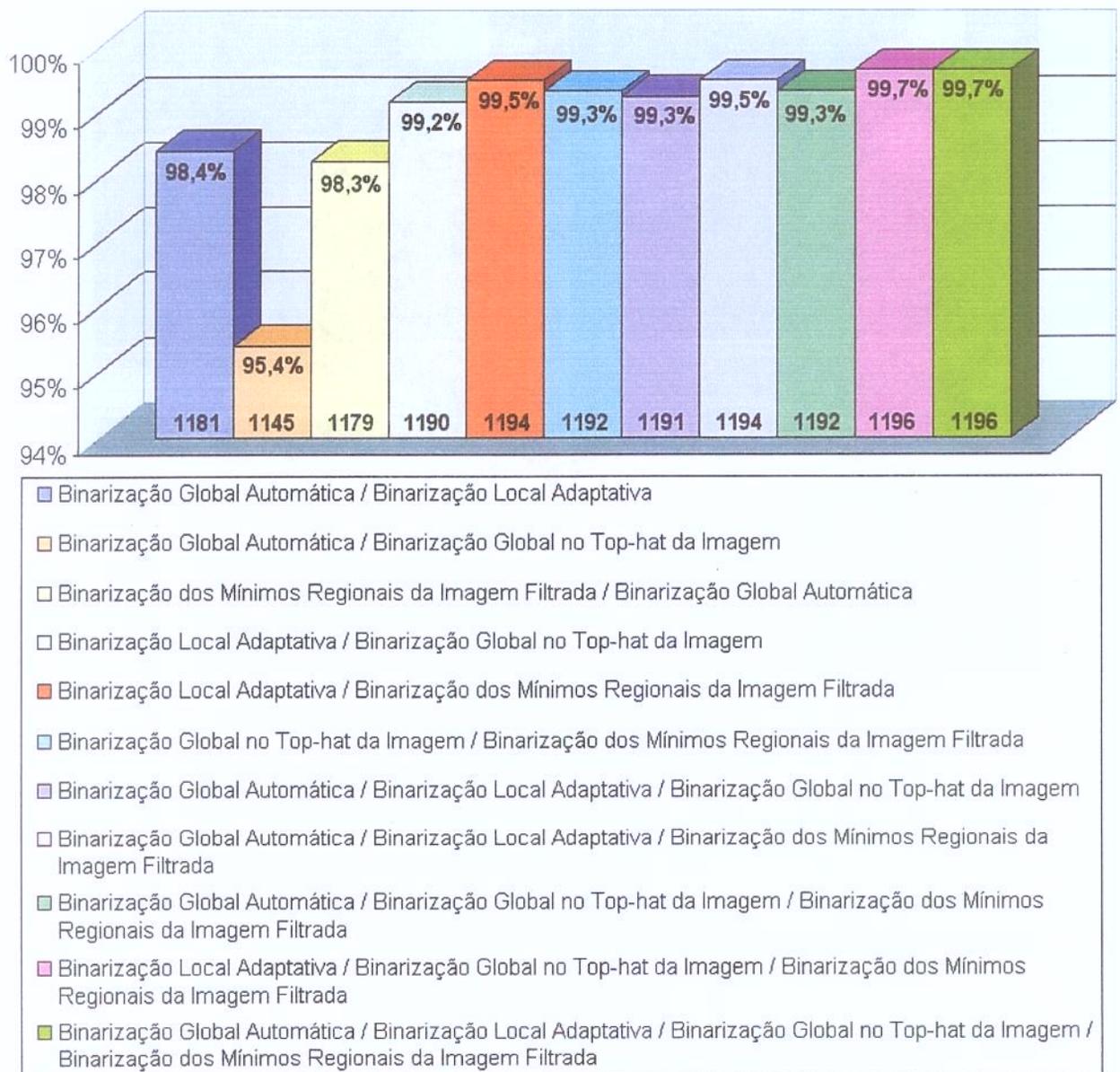
**Fig. 5.8:** Porcentagem de sucesso na localização da placa com diferentes métodos de binarização.

Cada método de binarização é mais sensível a determinadas características da imagem. Estas características dependem do ambiente no qual as imagens são obtidas (tais como: iluminação, estado de conservação das placas, etc). Portanto, optamos por adotar um método híbrido de binarização, visando minimizar o problema da aquisição de imagens em diferentes ambientes.

Analisando a porcentagem de sucesso na localização da placa com diferentes combinações de métodos de binarização, concluímos que a melhor hibridização consiste em utilizar os seguintes métodos: *Binarização Local Adaptativa*; *Binarização Global no Top-hat da Imagem*; e *Binarização dos Mínimos Regionais da Imagem Filtrada*. O uso dos 4 métodos não é vantajoso, visto que obteve resultado igual ao uso dos 3 métodos que adotamos. A figura 5.9 ilustra o resultado obtido por cada combinação dos métodos de binarização propostos.

Visando minimizar o acréscimo no custo computacional, definimos a seqüência da aplicação dos métodos de acordo com sua performance individual, a qual pode ser visualizada na figura 5.8. Deste modo, o método híbrido utiliza a seguinte seqüência de métodos:

1. Binarização dos Mínimos Regionais da Imagem Filtrada;
2. Binarização Local Adaptativa;
3. Binarização Global do Top-hat da Imagem.



**Fig. 5.9:** Porcentagem de sucesso na localização da placa com diferentes combinações de métodos de binarização.

O método híbrido consiste em aplicar o primeiro método e tentar localizar a placa, se ela não for encontrada o mesmo método é aplicado no negativo desta imagem. Em seguida, são aplicados os demais módulos em cada região promissora encontrada.

Caso nenhuma placa seja encontrada, o próximo método é utilizado até que uma placa seja encontrada ou todos os métodos sejam utilizados.

Com o hibridismo da binarização, nossa proposta obteve **99,7%** de sucesso na localização da placa. Isto indica que apenas 4 placas não foram localizadas, as quais são ilustradas pela figura 5.10.



Fig. 5.10: *Imagens cujas placas não foram localizadas.*

## 5.2 Análise do Reconhecimento dos Caracteres

O resultado do módulo de *Reconhecimento dos Caracteres* depende principalmente da qualidade da imagem binarizada do caractere a ser reconhecido. Portanto, é possível obter melhores resultados se for melhorada a qualidade desta imagem.

Na figura 5.11, podemos verificar a distribuição de caracteres errados por placa, no qual constatamos que **69%** das placas de licenciamento foram reconhecidas com todos os caracteres corretos.

Considerando que toda placa possa ter no máximo 1 caractere incorreto, obtemos 88,25% de sucesso, se a possibilidade for de 2 caracteres incorretos temos 94,67%. Portanto, podemos afirmar que nossa metodologia obteve um alto valor de sucesso com pequenas quantidades de caracteres com erro no reconhecimento.

A figura 5.12 ilustra a porcentagem de acerto no reconhecimento por caractere, sendo a quantidade de amostras por caracteres mostrada na figura 5.13.

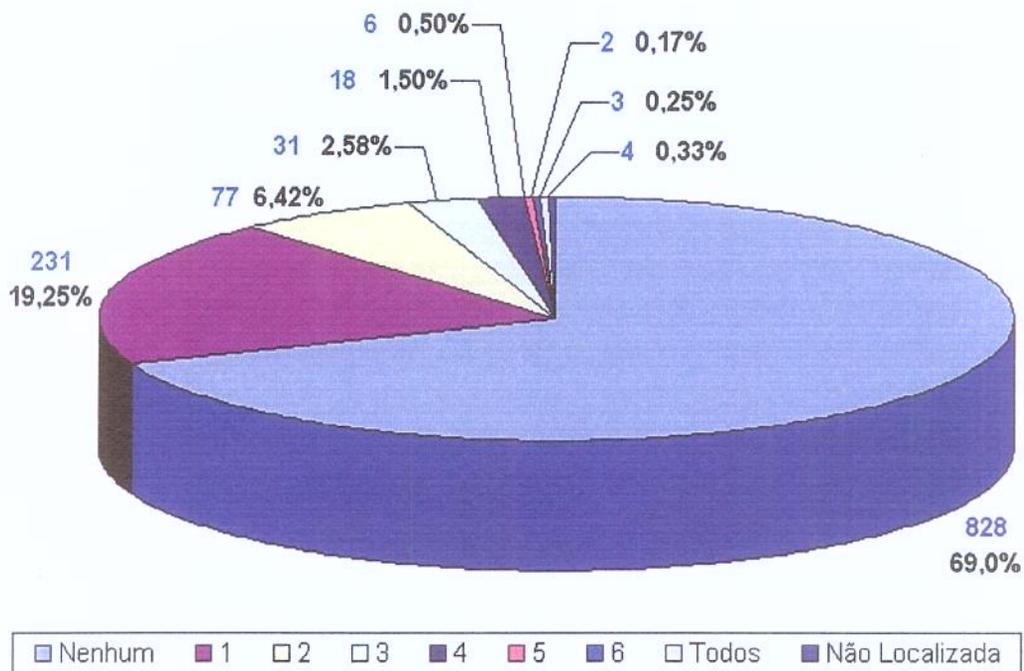


Fig. 5.11: Distribuição de caracteres errados por placa.

A matriz de confusão, tabela 5.1, ilustra a porcentagem de erro de reconhecimento dos caracteres da linha pelos caracteres da coluna. Com isto, podemos determinar com quais outros um determinado caractere é confundido, também visualizando a intensidade desta ocorrência.

Nas figuras 5.12 e 5.13, podemos constatar que a maior dificuldade está em reconhecer os caracteres: '1', 'D', 'I', 'K', 'M', 'N', 'O', 'Q', 'V' e 'W'.

Com base na matriz de confusão, tabela 5.1, podemos associar os caracteres com maior porcentagem de erros, citados anteriormente, com 3 principais causas: inclinação, onde '1' é confundido com '7', e 'I' com 'Z'; semelhança na tipografia, onde 'D', 'O' e 'Q' se confundem, e 'K' com 'X'; e resolução espacial, onde 'M' é confundido com 'H', 'N' com 'H', 'V' com 'Y', e 'W' com 'N'.

O caso do 'I' pode ser resolvido verificando a razão de aspecto (altura dividido pela largura) da imagem do caractere. Os demais casos já eram esperados, e para resolvê-los pretendemos utilizar descritores morfológicos no segundo nível do classificador hierárquico. Porém, este será um trabalho futuro.

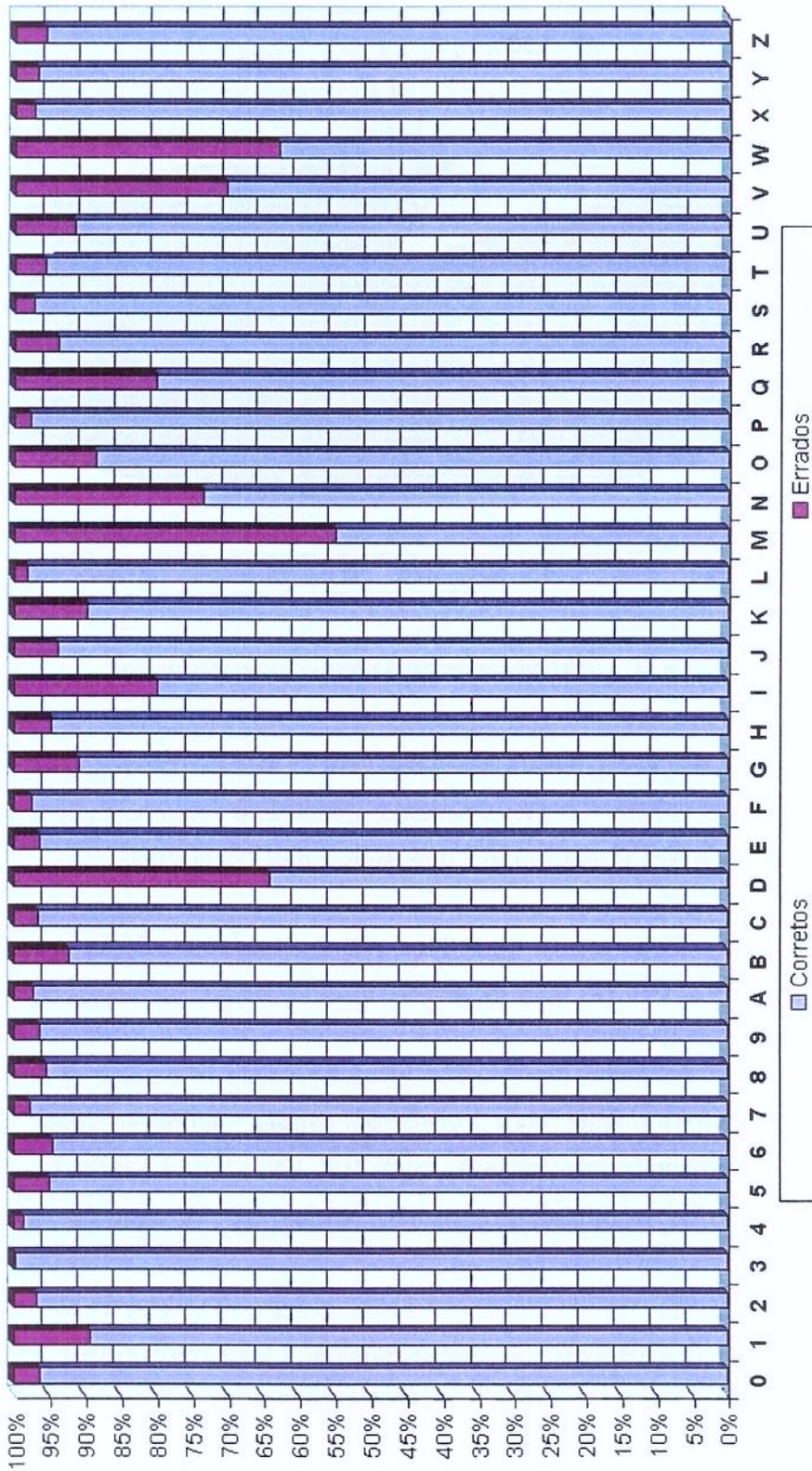


Fig. 5.12: Porcentagem de reconhecimento por caracteres.

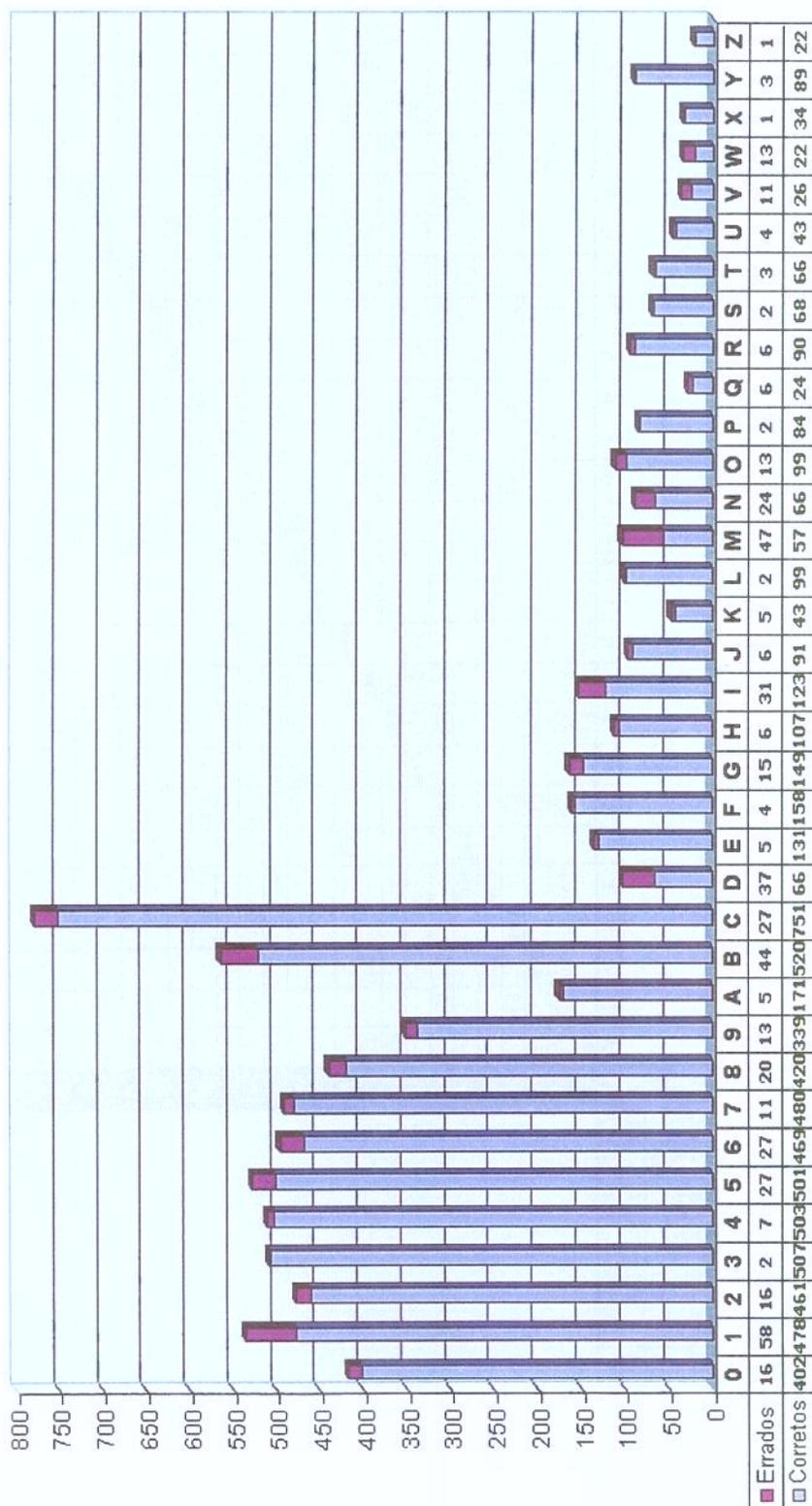


Fig. 5.13: Reconhecimento por caracteres.

|   | 0    | 1    | 2    | 3    | 4    | 5    | 6    | 7    | 8    | 9    |
|---|------|------|------|------|------|------|------|------|------|------|
| 0 | 96,2 | -    | -    | 0,5  | 1,0  | 0,2  | 0,2  | -    | 1,7  | 0,2  |
| 1 | 0,2  | 89,2 | 1,5  | 0,2  | 0,4  | -    | 0,2  | 6,5  | 0,6  | 1,3  |
| 2 | 0,4  | -    | 96,7 | -    | -    | 0,2  | 2,5  | -    | 0,2  | -    |
| 3 | -    | -    | 0,2  | 99,6 | -    | -    | -    | 0,2  | -    | -    |
| 4 | -    | -    | -    | -    | 98,6 | 0,2  | 1,0  | -    | 0,2  | -    |
| 5 | 0,2  | -    | -    | 0,2  | 0,2  | 94,9 | 4,0  | 0,4  | -    | 0,2  |
| 6 | 0,4  | -    | -    | -    | 2,2  | 0,2  | 94,6 | 0,4  | 2,0  | 0,2  |
| 7 | 0,2  | 0,2  | 0,6  | -    | 0,2  | -    | -    | 97,8 | -    | 1,0  |
| 8 | 0,7  | 0,2  | 1,4  | -    | 0,5  | -    | 0,7  | -    | 95,5 | 1,1  |
| 9 | -    | 0,3  | -    | 0,6  | 0,3  | -    | 0,3  | 0,9  | 1,4  | 96,3 |

|   | A    | B    | C    | D    | E    | F    | G    | H    | I    | J    | K    | L    | M    | N    | O    | P    | Q    | R    | S    | T    | U    | V | W | X   | Y   | Z    |      |      |
|---|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|---|---|-----|-----|------|------|------|
| A | 97,2 | -    | -    | -    | -    | -    | -    | -    | 1,1  | -    | 0,6  | 0,6  | -    | -    | -    | -    | -    | -    | -    | -    | -    | - | - | 0,6 | -   | -    |      |      |
| B | 0,4  | 92,2 | 0,2  | -    | 0,4  | -    | -    | 0,2  | -    | 0,5  | 0,2  | -    | 0,4  | -    | 1,1  | 0,2  | -    | 0,5  | 3,2  | -    | -    | - | - | 0,2 | -   | 0,4  | 0,2  |      |
| C | 0,3  | -    | 96,5 | -    | -    | 0,1  | 0,3  | -    | 0,1  | -    | 0,1  | 1,7  | -    | -    | -    | 0,1  | -    | -    | 0,1  | 0,3  | -    | - | - | 0,1 | -   | 0,1  | 0,1  |      |
| D | -    | 4,9  | -    | 64,1 | -    | -    | 1,9  | -    | 1,0  | -    | -    | -    | -    | -    | 26,2 | -    | 1,0  | -    | 1,0  | -    | -    | - | - | -   | -   | -    | -    |      |
| E | -    | -    | 0,7  | -    | 96,3 | 0,7  | -    | -    | -    | -    | -    | 0,7  | -    | -    | -    | -    | -    | -    | -    | -    | -    | - | - | -   | -   | -    | 1,5  |      |
| F | -    | -    | -    | -    | 0,6  | 97,5 | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | - | - | -   | -   | -    | -    |      |
| G | -    | 1,8  | 1,2  | -    | -    | -    | 90,9 | -    | 0,6  | -    | -    | -    | -    | -    | 1,2  | 0,6  | 0,6  | -    | 3,1  | -    | -    | - | - | -   | -   | -    | -    |      |
| H | -    | -    | -    | -    | -    | -    | -    | 94,7 | 0,9  | 0,9  | 0,9  | -    | -    | -    | 0,9  | -    | -    | 0,9  | -    | -    | -    | - | - | -   | 0,9 | -    | -    |      |
| I | 3,3  | -    | -    | -    | -    | -    | -    | -    | 79,9 | 0,7  | -    | 0,7  | -    | -    | -    | -    | -    | -    | -    | 3,9  | -    | - | - | 1,3 | 3,9 | 0,7  | 5,8  |      |
| J | 2,1  | -    | 1,0  | -    | -    | -    | -    | -    | -    | 93,8 | -    | 2,1  | -    | -    | -    | -    | -    | -    | -    | -    | -    | - | - | -   | -   | 1,0  | -    |      |
| K | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | 89,6 | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | - | - | -   | 8,3 | -    | -    |      |
| L | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | 98,0 | -    | -    | -    | -    | -    | -    | 1,0  | -    | -    | - | - | -   | 1,0 | -    | -    |      |
| M | -    | -    | -    | -    | -    | -    | -    | 23,1 | 1,0  | -    | 10,6 | -    | 54,8 | 3,9  | -    | -    | -    | -    | -    | -    | -    | - | - | 5,8 | -   | -    | -    |      |
| N | -    | -    | -    | -    | -    | -    | -    | 12,2 | 5,6  | -    | 2,2  | -    | 1,1  | 73,3 | -    | -    | -    | 1,1  | -    | -    | -    | - | - | 2,2 | 1,1 | -    | -    |      |
| O | -    | 0,9  | -    | 2,7  | -    | -    | -    | 1,8  | -    | -    | -    | -    | -    | -    | 88,4 | -    | 6,3  | -    | -    | -    | -    | - | - | -   | -   | -    | -    |      |
| P | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | 97,7 | -    | -    | -    | -    | -    | - | - | -   | -   | 1,2  | -    |      |
| Q | -    | -    | -    | -    | -    | -    | -    | 3,3  | -    | -    | -    | -    | -    | -    | -    | -    | 80,0 | -    | -    | -    | -    | - | - | -   | -   | -    | -    |      |
| R | -    | -    | -    | -    | -    | -    | -    | -    | 3,1  | -    | -    | -    | -    | -    | -    | 1,0  | -    | 93,8 | -    | 1,0  | 1,0  | - | - | -   | -   | -    | -    |      |
| S | 1,4  | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | 97,1 | -    | -    | - | - | -   | -   | -    | -    |      |
| T | 1,5  | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | 95,7 | -    | - | - | -   | -   | -    | -    |      |
| U | 2,1  | -    | -    | -    | -    | -    | -    | -    | 2,1  | 2,1  | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | 91,5 | - | - | -   | -   | -    | 1,5  |      |
| V | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | - | - | -   | -   | -    | -    |      |
| W | -    | -    | -    | -    | -    | -    | -    | 2,9  | 17,1 | -    | -    | -    | 17,1 | -    | -    | -    | -    | -    | -    | -    | -    | - | - | -   | -   | 70,3 | -    |      |
| X | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | - | - | -   | -   | -    | 62,9 |      |
| Y | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | - | - | -   | -   | -    | 97,1 |      |
| Z | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | -    | 4,4  | -    | -    | -    | -    | -    | -    | -    | -    | -    | - | - | -   | -   | -    | 3,3  |      |
|   |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |   |   |     |     |      |      | 95,7 |

Tab. 5.1: Matriz de confusão.

## 5.3 Análise do Custo Computacional

O protótipo deste sistema foi desenvolvido em MatLab v6.5 r13, utilizando também a biblioteca *SDC Morphology Toolbox V1.2* para o MatLab.

Analizamos a performance de nosso protótipo em um Pentium 4 3.2 Ghz HT com 256Mb de memória.

Na média, o tempo de processamento das 1200 imagens do banco de dados que utilizamos foi de 4 segundos. Para analisar o custo computacional, nos baseamos em uma imagem que consumiu 3,7460 segundos no processamento, onde cada um dos módulos utilizou o seguinte percentual deste tempo:

- *Módulo de Localização da Placa*: 0,2030 segundos (5,42%);
- *Módulo de Enquadramento da Placa*: 3,2510 segundos (86,79%);
- *Módulo de Separação dos Caracteres*: 0,0060 segundos (0.16%);
- *Módulo de Reconhecimento dos Caracteres*: 0,1260 segundos (3,36%);
- *Interface do Sistema*: 0,1600 segundos (4,27%).

## 5.4 Análise Comparativa

Para analisar nossa proposta efetuamos um teste comparativo com duas demonstrações de sistemas comerciais disponíveis na internet: *SeeCar* (Solutions, 2003) e *SIAV 3.0* (Pixon, 2005). O sistema *SIAV 3.0* é uma versão melhorada da proposta de Souza and Susin (2000). Ambos os sistemas utilizam o método da *Variação Tonal* para localizar a placa, o método *Conexionista* (Redes Neurais) para o reconhecimento dos caracteres e não corrigem a distorção de perspectiva. Estes sistemas não possuem configurações para aplicar um ajuste fino de acordo com o ambiente.

A figura 5.14 apresenta os resultados obtidos neste comparativo, onde podemos constatar que nossa proposta foi 1,17 vezes superior na localização da placa e 1,58 vezes superior no correto reconhecimento da placa em relação ao melhor resultado entre os sistemas *SIAV 3.0* e *SeeCar*, comprovando a eficiência de nossa proposta em ambientes com ocorrência da distorção de perspectiva, comum em muitas aplicações deste sistema.

A baixa porcentagem do correto reconhecimento da placa é devido a imagens de caracteres em más condições, provenientes de: placa suja, apagada, obstruída, com superfície reflexiva, etc.

A página Lotufo and Dias (2005) da internet contém resultados detalhados e atualizados deste projeto.

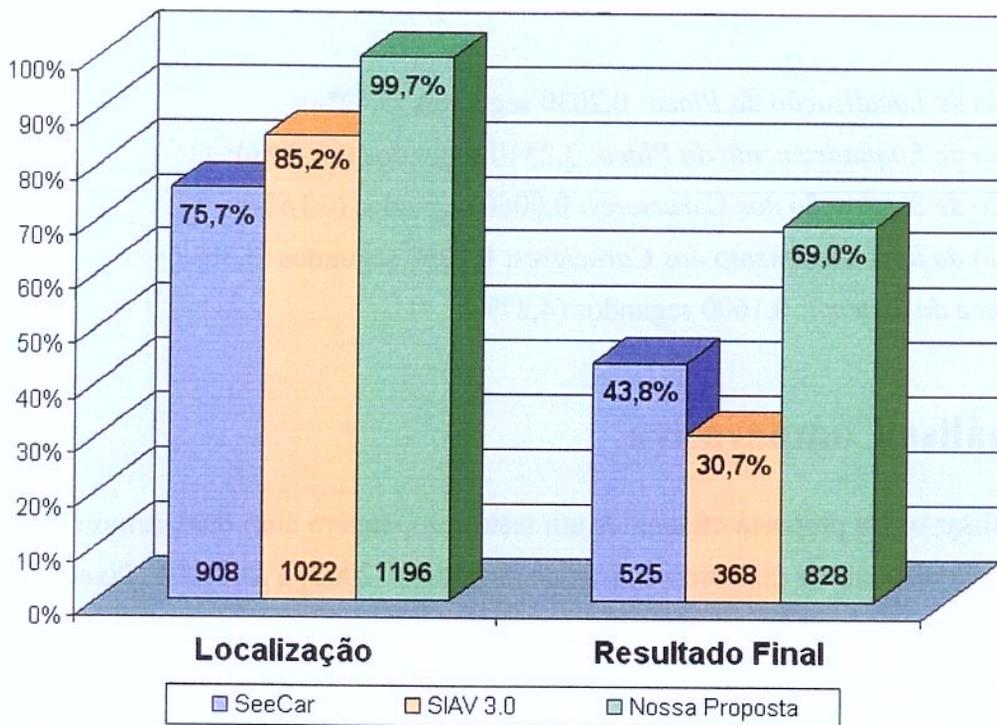


Fig. 5.14: Teste comparativo com duas demonstrações de sistemas comerciais disponíveis na internet.

# Capítulo 6

## Conclusões

Este trabalho apresentou técnicas de processamento de imagens e reconhecimento de padrões, visando o desenvolvimento de um *software* para sistemas de reconhecimento da placa de licenciamento veicular. Também elaboramos uma classificação dos métodos mais utilizados com base na análise de 42 propostas.

Comprovamos a eficiência do método aqui proposto, tanto na localização da placa como no reconhecimento dos caracteres, através de um teste comparativo com duas demonstrações de sistemas comerciais disponíveis na internet: *SeeCar* (Solutions, 2003) e *SIAV 3.0* (Pixeon, 2005), onde foram processadas 1200 imagens com resolução de 320x240.

A porcentagem de sucesso na localização da placa foi de 99,7%, significando a falha do nosso método em apenas 4 das 1200 imagens processadas. No correto reconhecimento da cadeia de caracteres da placa, a porcentagem de sucesso foi de 69%, sendo um valor alto, visto as dificuldades apresentadas pelas imagens. Considerando que toda placa possa ter no máximo 1 caractere incorreto, obtemos 88,25% de sucesso, se a possibilidade for de 2 caracteres incorretos temos 94,67%.

Nosso método de localização da placa é menos sensível aos padrões de textura (tais como: paralelepípedos, grade do radiador do veículo, galhos de árvore, etc.), obtendo um resultado 1,17 vezes superior que o sistema *SIAV 3.0*. Outro fator que contribuiu para este resultado foi a utilização de uma combinação de métodos de binarização, visando maximizar a probabilidade de segmentar os caracteres da placa.

O resultado obtido no correto reconhecimento da placa, 1,58 vezes superior que o sistema *SeeCar*, se deve ao enquadramento da placa com correção da distorção de perspectiva, pois os sistemas *SIAV 3.0* e *SeeCar* não efetuam esta correção. Deste modo, comprovamos a melhoria na separação e reconhecimento dos caracteres promovida pelo enquadramento com correção da distorção de perspectiva.

Constatamos que a maior dificuldade está em reconhecer os caracteres: 'I', 'D', 'I', 'K', 'M', 'N', 'O', 'Q', 'V' e 'W', sendo necessário elaborar métodos específicos para a resolução de conflitos conhecidos entre caracteres. Também verificamos que o conflito entre os números não é preocupante, sendo recomendável o reconhecimento em diferentes classificadores, um para os números e outro para as letras.

A maior parte do custo computacional apresentado é utilizado pelo módulo de enquadramento da placa, justamente o que forneceu uma melhoria significativa aos módulos de separação e reconhecimento dos caracteres. Deste modo, verificamos o compromisso da velocidade de processamento com a qualidade do resultado.

Os principais fatores que prejudicaram os resultados foram: baixa resolução espacial das imagens, considerável distorção de perspectiva, presença de ruídos, superfície reflexiva da placa, má qualidade de conservação da placa, iluminação heterogênea, objetos posicionados na frente da placa, diferença na tipografia de um mesmo caractere e semelhança na tipografia de caracteres diferentes.

Embora nossa proposta tenha obtido melhores resultados que os outros dois sistemas analisados, percebemos as seguintes possibilidades de melhorias:

- Utilizar um método complementar para a localização da placa;
- Estudar outras técnicas de binarização para aplicar em nosso método de localização da placa;
- Aprimorar as funções-objetivo utilizadas na identificação das arestas da placa;
- Desenvolver um método de enquadramento da placa com menor custo computacional;
- Tratar o problema da união de caracteres cortados;
- Elaborar um método de separação dos caracteres unidos mais robusto;
- Melhorar a binarização dos caracteres antes de efetuar seu reconhecimento, visando diminuir o erro com imagens danificadas por ruídos aditivos ou subtrativos;
- Utilizar outras características, sendo também necessário desenvolver uma técnica para ponderá-las, visando maximizar a confiabilidade no reconhecimento;
- Aprimorar o segundo nível do classificador hierárquico utilizando descritores morfológicos para resolver conflitos conhecidos entre caracteres.

Nossa principal conclusão é a necessidade da utilização de métodos complementares, de modo que um consiga suprir as falhas do outro.

O artigo (Dias and Lotufo, 2005), apresenta um resumo deste trabalho, o qual foi apresentado no IV Workshop de Teses e Dissertações em Computação Gráfica e Processamento de Imagens do SIBGRAPI 2005, sendo premiado na categoria de mestrado.

# Referências Bibliográficas

- M. J. Ahmed, M. Sarfraz, A. Zidouri, and W. G. Al-Khatib. License plate recognition system. In *Proceedings of IEEE 10th International Conference on Electronics, Circuits and Systems*, volume 2, pages 898–901, 2003.
- J. Barroso, A. Rafael, E. L. Dagless, and J. Bulas-Cruz. Number plate reading using computer vision. In *Proceedings of IEEE International Symposium on Industrial Electronics*, 1997.
- B. Borba, C. Vasconcelos, M. Albuquerque, M. Albuquerque, I. A. Esquef, and J. M. Seixas. Localization of brazilian vehicles plates using frequency analysis. In *Proceedings of XV Brazilian Symposium on Computer Graphics and Image Processing*, Brazil, 2002.
- M. H. Brugge, J. H. Stevens, J.A. G. Nijhuis, and L. Spaanenburg. License plate recognition using dtcnns. In *Proceedings of IEEE Fifth International Workshop on Cellular Neural Networks and their Applications*, pages 212–217, London, 1998.
- S. Chang, L. Chen, Y. Chung, and S. Chen. Automatic license plate recognition. *Proceedings of IEEE International Conference on Intelligent Transportation Systems*, 5:42–53, 2004.
- C. Coetzee, C. Botha, and D. Weber. PC based number plate recognition system. In *Proceedings of IEEE International Conference on Industrial Electronics*, 1998.
- R. Cowell. Syntactic pattern recognizer for vehicle identification numbers. *Image and Vision Computing*, pages 13–19, 1995.
- F. G. Dias. Uma proposta evolutiva para classificador de distância mínima. <http://www.dca.fee.unicamp.br/~gaiotto/ia707/projeto-final.doc>, 2004.
- F. G. Dias and R. A. Lotufo. Melhorias para sistemas de reconhecimento da placa de licenciamento veicular. In *IV Workshop de Teses de Dissertações em Computação Gráfica e Processamento de Imagens - WTDCGPI 2005*, Natal, RN, Brazil, 2005.

- E. R. Dougherty and R. A. Lotufo. *Hands-On Morphological Image Processing*. SPIE, Bellingham, WA, USA, 2003.
- T. D. Duan, T. L. H. Du, T. V. Phuoc, and N. V. Hoang. Building an automatic vehicle license-plate recognition system. In *Proceedings of International Conference in Computer Science*, 2005.
- R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis*. Wiley-Interscience, California, 1973.
- R. C. Gonzalez and R. E. Woods. *Processamento de Imagens Digitais*. Editora Edgard Blücher, Brazil, 2000.
- B. C. Guingo, A. C. G. Thomé, and R. J. Rodrigues. Reconhecimento automático de placas de veículos automotores através de redes neurais artificiais. In *Proceedings of II CBComp - Congresso Brasileiro de Computação*, Brazil, 2002.
- H. A. Hegt, R. J. Haye, and N.A. Khan. A high performance license plate recognition system. In *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, pages 4357–4362, 1998.
- H. Hontani and T. Koga. Character extraction method without prior knowledge on size and position information. In *Proceedings of the IEEE International Conference on Vehicle Electronics*, pages 67–72, 2001.
- C. T. Hsieh, Y. S. Juan, and K. M. Hung. Multiple license plate detection for complex background. In *Proceedings of the 19th International Conference on Advanced Information Networking and Applications*, volume 2, pages 389–392, 2005.
- Y. Huang, S. Lai, and W. Chuang;. A template-based model for license plate recognition. In *Proceedings IEEE International Conference on Networking, Sensing and Control*, volume 2, pages 737–742, 2004.
- Z. B. Huang and Y. F. Guo. Classifier fusion-based vehicle license plate detection algorithm. In *Proceedings of IEEE International Conference on Machine Learning and Cybernetics*, volume 5, pages 2984–2989, 2003.
- X. Jianfeng, L. Shaofa, and C. Zhibin. Color analysis for chinese car plate recognition. In *Proceedings of IEEE International Conference on Robotics, Intelligent Systems and Signal Processing*, volume 2, pages 1312–1316, 2003.

- V. Kamat and S. Ganesan. An efficient implementation of the hough transform for detecting vehicle license plates using dsp's. In *Proceedings of IEEE Real-Time Technology and Applications Symposium*, pages 58–59, 1995.
- K. K. Kim, K. I. Kim, J. B. Kim, and H. J. Kim. Learning based approach for license plate recognition. In *Proceedings of IEEE Processing Society Workshop on Neural Networks for Signal Processing*, volume 2, pages 614–623, 2000.
- S. K. Kim, D. W. Kim, and H. J. Kim. A recognition of vehicle license plate using a genetic algorithm based segmentation. In *Proceedings of IEEE International Conference on Image Processing*, volume 2, pages 661–664, 1996.
- A. F. Kohn. *Reconhecimento de Padrões: Uma Abordagem Estatística*. Editora da USP, São Paulo, 1998.
- V. Koval, V. Turchenko, V. Kochan, A. Sachenko, and G. Markowsky. Smart license plate recognition system based on image processing using eural network. In *Intelligent Data Acquisition and Advanced Computing Systems: Technology and , 2003. Proceedings of the Second IEEE International Workshop on Applications*, pages 123–127, 2003.
- S. Lee, Y. Seok, and E. Lee. Multi-national integrated car-license plate recognition system using geometrical feature and hybrid pattern vector. In *Proceedings of International Conference on Circuits, Systems, Computers and Communications*, pages 1256–1259, Phuket, Thailand., 2002.
- R. A. Lotufo and F. G. Dias. PI 0405256-0 - Método de localização de cadeias alfanuméricas tipográficas em superfícies de fundo uniforme para um sistema de reconhecimento baseado em imagem. INPI - Instituto Nacional de Propriedade Industrial, 2004.
- R. A. Lotufo and F. G. Dias. Sistemas de reconhecimento da placa de licenciamento veicular. <http://www.dca.fee.unicamp.br/~gaiotto/projects/srplv.html>, 2005.
- Y Lu. Machine printed character segmentation. *Pattern Recognition*, 28:67–80, 1995.
- F. Martín, M. García, and J. L. Alba. New advances in automatic reading of vlp's. In *Proceedings of SPC-2000 (IASTED)*, pages 126–131, Marbella, Spain, 2000.
- K. Miyamoto, K. Nagano, M. Tamagawa, I. Fujita, and M. Yamamoto. Vehicle license-plate recognition by image analysis. In *Proceedings of IEEE International Conference Industrial Electronic Control Instrument*, volume 3, pages 1734–1738, 1991.

- T. Naito, T. Tsukada, K. Yamada, K. Kozuka, and S. Yamamoto. License plate recognition method for inclined plates outdoors. In *Proceedings of IEEE International Conference on Information Intelligence and Systems*, pages 304–312, 1999.
- T. Naito, T. Tsukada, K. Yamada, K. Kozuka, and S. Yamamoto. Robust license-plate recognition method for passing vehicles under outside environment. *IEEE Transactions on Vehicular Technology*, 49:2309–2319, 2000.
- V. S. L. Nathan, J. Ramkumar, and S. K. Priya. New approaches for license plate recognition system. In *Proceedings of IEEE International Conference on Intelligent Sensing and Information Processing*, pages 149–152, 2004.
- W. Niblack. *An Introduction to Digital Image Processing*. Prentice Hall, 1986.
- J. A. G. Nijhuis, M. H. T. Brugge, K. A. Helmholt, J. P. W. Pluim, L. Spaanenburg, R. S. Venema, and M. A. Westenberg. Car license plate recognition with neural networks and fuzzy logic. In *Proceedings of IEEE International Conference on Neural Networks*, volume 5, pages 2232–2236, 1995.
- N. Otsu. A threshold selection method for gray level histograms. *IEEE Transactions on System, Man and Cybernetics*, 1979.
- Pixeon. SIAV 3.0. <http://www.pixeon.com.br>, 2005.
- A. M. Polidório and D. L. Borges. Reconhecimento de caracteres alfanuméricos para identificação de placas de veículos. In *Proceedings of IX Brazilian Symposium on Computer Graphics and Image Processing*, pages 363–364, 1996.
- C. A. Rahman, W. Badawy, and A. Radmanesh. A real time vehicle's license plate recognition system. In *Proceedings of IEEE Conference on Advanced Video and Signal Based Surveillance*, pages 163–166, 2003.
- A. L. Sáenz, M. I. Chacón M., and A. Z. Solís. Localización de placas en vehículos automotores. In *Proceedings of ELECTRO-2001 - Instituto Tecnológico de Chihuahua*, pages 147–152, Mexico, 2001.
- L. Salgado, J. M. Menendez, E. Rendon, and N. Garcia. Automatic car plate detection and recognition through intelligent vision engineering. In *Proceedings of IEEE 33rd Annual International Carnahan Conference on Security Technology*, pages 71–76, 1999.

- M. Sarfraz, M. J. Ahmed, and S. A. Ghazi. Saudi arabian license plate recognition system. In *Proceedings of IEEE International Conference on Geometric Modeling and Graphics*, pages 36–41, 2003.
- V. Shapiro, D. Dimov, S. Bonchev, V. Velichkov, and G. Gluhchev. Adaptive license plate image extraction. In *Proceedings of International Conference on Computer Systems and Technologies*, pages IIIA.2–1–IIIA.2–7, 2003.
- Y. S. Soh, B. T. Chun, and H. S. Yoon. Design of real time vehicle identification system. In *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, volume 3, pages 2147–2152, San Antonio, TX, USA, 1994.
- High Technologies Solutions. SeeCar. <http://www.htsol.com>, 2003.
- F. P. C. Souza and A. Susin. SIAV - Um sistema de identificação automática de veículos. In *Proceedings of XIII Congresso Brasileiro de Automática*, pages 1377–1380, SC, Brazil, 2000.
- A. Taleb-Ahmed, D. Hamad, and G. Tilmant. Vehicle license plate recognition in marketing application. In *Proceedings of Intelligent Vehicles Symposium*, pages 90–94, 2003.
- E. Trucco and A. Verri. *Introductory Techniques for 3-D Computer Vision*. Prentice Hall, NJ, USA, 1998.
- J. Tyan and C. Neubauer. US 6,473,517 - Character segmentation method for vehicle license plate recognition. *USPTO - United States Patent and Trademark Office*, 2002.
- S. Z. Wang and H. J. Lee. Detection and recognition of license plate characters with different appearances. In *Proceedings of IEEE International Conference on Intelligent Transportation Systems*, volume 2, pages 979–984, 2003.
- T. Wang, F. Ni, K. Li, and Y. Chen. Robust license plate recognition based on dynamic projection warping. In *Proceedings of IEEE International Conference on Networking, Sensing and Control*, volume 2, pages 784–788, 2004.
- D. Yan, M. Hongqing, L. Jilin, and L. Langang. A high performance license plate recognition system based on the web technique. In *Proceedings of IEEE International Conference on Intelligent Transportation Systems*, pages 14–18, CA, USA, 2001.

- M. Yu and Y. D. Kim. An approach to korean license plate recognition based on vertical edge matching. In *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics*, volume 4, pages 2975–2980, 2000.
- Y. Zhang and C. Zhang. A new algorithm for character segmentation of license plate. In *Proceedings of IEEE Intelligent Vehicles Symposium*, pages 106–109, 2003.