

CENTRO UNIVERSITÁRIO FEEVALE

ALEXANDRE LUÍS DAMBROS

SISTEMA DE RECONHECIMENTO DE PLACAS DE VEÍCULOS
AUTOMOTORES

Novo Hamburgo, junho de 2008.

ALEXANDRE LUÍS DAMBROS

SISTEMA DE RECONHECIMENTO DE PLACAS DE VEÍCULOS
AUTOMOTORES

Centro Universitário Feevale
Instituto de Ciências Exatas e Tecnológicas
Curso de Ciência da Computação
Trabalho de Conclusão de Curso

Professor Orientador: Marta Rosecler Bez

Novo Hamburgo, junho de 2008.

AGRADECIMENTOS

Gostaria de agradecer a todos os que, de alguma maneira, contribuíram para a realização desse trabalho de conclusão, em especial:

A minha família e namorada que me deram todo o apoio necessário e souberam compreender a minha ausência nesses meses de estudo. Não poderia esquecer também da Professora Marta, que me orientou, por ter acreditado no tema escolhido e por mostrar muitas vezes o caminho por onde deveria seguir, dando sua contribuição para a realização desse trabalho de conclusão.

Ao amigo Idemar, que trabalhou junto no desenvolvimento de algumas etapas.

A meu sócio Paulo, que na minha ausência cuidou muito bem da nossa empresa, a DK Suprimentos.

RESUMO

Mesmo que um ser humano consiga identificar a placa de um veículo automotor eficientemente, ele não consegue fazer isso quando possui um grande volume de informação em um curto espaço de tempo. Sem falar nos erros que podem ocorrer com o excesso de trabalho. Com o constante crescimento da frota de veículos em todo o país, cria-se a necessidade da busca de soluções mais apropriadas ao assunto, como a utilização de sistemas de reconhecimento de placas de veículos automotores, agilizando o fluxo dos automóveis, evitando engarrafamentos. A área a ser estudada trata sobre processamento de imagens digitais. O presente projeto visa o estudo das etapas necessárias para o desenvolvimento de um sistema de reconhecimento, que tem como objetivo identificar as letras e números da placa de um veículo automotor. Para tanto, serão estudadas e aplicadas técnicas de segmentação, detecção de bordas, rotulação, rotação e reconhecimento de padrões.

Palavras-chave: Processamento de Imagens. Reconhecimento de Placas. Visão Computacional. Visualização Científica. Reconhecimento de Padrões.

ABSTRACT

Even if a person could do efficiently motor vehicle plate recognition; it will be hard to recognize when there is too much information in a short period of time. Without mentioning when errors appear because of this hard work. With a constant incensement of vehicles in the country, there is a necessity of new and appropriate solutions for this matter, like motor vehicle plate recognition systems. These systems can increase traffic flow and prevent traffic jam. The matter for studying is about digital image processing. This project treats about necessary stages to develop a recognition system; its objective is identifying automotive vehicles plate's letters and numbers. For doing this will be study and apilly segmentation techniques, edge detection, labeling, rotation and pattern recognition.

Key words: Image Processing. Recognition of Plates. Computational Vision. Scientific Visualization. Pattern Recognition.

LISTA DE FIGURAS

| | |
|--|----|
| Figura 1.1 – Modelo de Placa conforme CONTRAN. _____ | 17 |
| Figura 1.2 – Caracteres na fonte Mandatory. _____ | 17 |
| Figura 1.3 – Caractere O - Fonte não Mandatory. _____ | 17 |
| Figura 1.4 – Caractere Q - Fonte não Mandatory. _____ | 17 |
| Figura 1.5 – Sistema Kapta. _____ | 20 |
| Figura 1.6 – Sistema Lookar. _____ | 21 |
| Figura 1.7 – Imagem da Versão de Demonstração do Sistema SeeCar. _____ | 21 |
| Figura 1.8 – Imagem sendo capturada. _____ | 22 |
| Figura 1.9 – Placa sendo identificada em tempo real. _____ | 23 |
| Figura 2.1 – Diagrama com os Módulos necessários do Estudo Proposto. _____ | 26 |
| Figura 2.2 – Imagem original em ambiente externo. _____ | 27 |
| Figura 2.3 – Imagem em ambiente externo com ruído. _____ | 28 |
| Figura 2.4 – Histograma da Figura 2.2. _____ | 28 |
| Figura 2.5 – Imagem com Equalização do Histograma. _____ | 29 |
| Figura 2.6 – Histograma da Imagem apresentada na Figura 2.5. _____ | 29 |
| Figura 2.7 – Imagem Equalizada e com contraste. _____ | 30 |
| Figura 2.8 – Histograma da Figura 2.7. _____ | 30 |
| Figura 2.9 – Suavização pela Média, utilizando uma máscara 3x3. (a) Imagem Original; (b) Imagem Filtrada; (c) Máscara 3x3. _____ | 31 |
| Figura 2.10 – Aplicação do filtro da mediana. (a) Imagem Original; (b) Imagem corrompida por um ruído aleatório; (c) Imagem com Filtro por Mediana com Máscara 3x3; (d) Imagem com Filtro por Mediana com Máscara 5x5. _____ | 32 |
| Figura 2.11 – Placa do Veículo Localizada. _____ | 32 |
| Figura 2.12 – Técnica Prewitt – Detecção de bordas. _____ | 35 |
| Figura 2.13 – (a) Placa do Veículo Enquadrada; (b) Placa com Detector de Bordas Prewitt. _____ | 35 |

| | |
|--|----|
| Figura 2.14 – Imagem Binária da Placa Veicular. _____ | 36 |
| Figura 2.15 – Caracteres Enquadrados. _____ | 36 |
| Figura 2.16 – Vizinhança do Pixel. (a) Vizinhança-4; (b) Vizinhança-8. _____ | 37 |
| Figura 2.17 – Componentes conexos em uma imagem. _____ | 37 |
| Figura 2.18 – Caracteres Separados formando sete novas imagens. _____ | 38 |
| Figura 2.19 – Semelhanças da Tipografia de Caracteres Diferentes. _____ | 38 |
| Figura 2.20 – Placa com Tipografia do mesmo caractere diferente. (a) Placa que contém o caractere “6” com características bem arredondadas; (b) Placa que contém o caractere “6” bem diferente da placa anterior dessa imagem. _____ | 39 |
| Figura 3.1 – Imagem sendo Capturada. _____ | 41 |
| Figura 3.2 – Imagem Monocromática obtida em um Ambiente Externo. _____ | 42 |
| Figura 3.3 – Imagem Original obtida no Estacionamento do Centro Universitário Feevale. _____ | 42 |
| Figura 3.4 – Imagem Monocromática da Versão de Demonstração do Software SeeCar. _____ | 43 |
| Figura 3.5 – Imagem Monocromática obtida em um banco de imagens na Internet. _____ | 44 |
| Figura 3.6 – Imagem Monocromática obtida em um banco sem a placa do veículo. _____ | 44 |
| Figura 3.7 – Imagem Monocromática obtida em um banco faltando uma parte da placa. _____ | 44 |
| Figura 3.8 – Imagem Monocromática obtida em um banco sem qualidade ou visibilidade. _____ | 45 |
| Figura 3.9 – Técnicas de Detecção de bordas. (a) Canny; (b) Roberts; (c) Sobel; (d) Prewitt. _____ | 46 |
| Figura 3.10 – Imagem com quatro objetos encontrados e etiquetados. _____ | 47 |
| Figura 3.11 – Imagem de um veículo automotor com um objeto rotulado, a placa. _____ | 49 |
| Figura 3.12 – Objeto com região da placa encontrada e a placa sendo recortada na imagem original do veículo. _____ | 52 |
| Figura 3.13 – Imagem de veículo que não foi possível encontrar a placa automaticamente. _____ | 53 |
| Figura 3.14 – Imagem de veículo que não foi possível encontrar a placa automaticamente devido as bordas da mesma não estarem bem definidas. _____ | 54 |
| Figura 3.15 – Imagem da versão demo do programa SeeCar, que localiza a placa de um veículo a uma distância maior e ainda reconhece seus caracteres. _____ | 55 |
| Figura 3.16 – Região da placa recortada na etapa de Localização. _____ | 56 |
| Figura 3.17 – Placa rotacionada. _____ | 57 |
| Figura 3.18 – Placa rotacionada recortada através das suas bordas. _____ | 58 |
| Figura 3.19 – Imagem com detector de bordas Canny aplicado e seus 18 objetos rotulados. _____ | 59 |
| Figura 3.20 – Sete novas imagens, provenientes da placa IOB 8158. _____ | 59 |
| Figura 3.21 – Imagem onde a rotulação identificou como apenas um objeto, porém, na verdade, deveriam ser dois. _____ | 60 |

| | |
|--|----|
| Figura 3.22 – Mapa de bits - Imagem binária da letra M e redimensionada para 16 x 16 pixels. | 62 |
| Figura 3.23 – Amostra do número 8 rotulada e contado seus objetos | 63 |
| Figura 3.24 – Projeção vertical de uma imagem | 64 |
| Figura 3.25 – Imagem salva do protótipo realizando o reconhecimento da placa | 66 |

LISTA DE TABELAS

| | |
|---|----|
| Tabela 3.1 – Tabela resultado da média de objetos encontrados em 50 imagens de placas de veículos automotores com os detectores de bordas já aplicados. _____ | 48 |
| Tabela 3.2 – Tabela resultado da localização da placa no protótipo. _____ | 52 |
| Tabela 3.3 – Tabela resultado da separação dos caracteres da placa das imagens dos estacionamentos. _____ | 60 |
| Tabela 3.4 – Tabela resultado da separação dos caracteres do banco de imagens da Internet. _____ | 61 |
| Tabela 3.5 – Tabela resultado da separação dos caracteres das imagens da versão de demonstração do programa SeeCar. _____ | 61 |
| Tabela 3.6 – Tabela resultado do reconhecimento de caracteres. _____ | 65 |

LISTA DE ABREVIATURAS E SIGLAS

| | |
|------------|---|
| RENAVAM | Registro Nacional de Veículos Automotores |
| CONTRAN | Conselho Nacional de Trânsito |
| mm | Milímetros |
| SAE | Society of Automotive Engineers |
| ASCII | American Standard Code for Information Interchange |
| UFRGS | Universidade Federal do Rio Grande do Sul |
| UFRJ | Universidade Federal do Rio de Janeiro |
| UNICAMP | Universidade Estadual de Campinas |
| MATLAB | Matrix Laboratory |
| DLL | Dynamic-link Library |
| OCR | Reconhecimento de Caractere Óptico |
| O | Objeto |
| A_i | Área |
| \Re | Números Reais |
| H_i | Altura |
| L_i | Largura |
| P_i | Perímetro |
| OB | Relação de Objetos |
| OBJ | Relação de Objetos |
| OBJE | Objeto Resultante |
| AP | Altura da Placa |
| LP_{MAD} | Linha do pixel da placa mais alto, localizado na direita da imagem |
| LP_{MBD} | Linha do pixel da placa mais baixo, localizado na direita da imagem |

| | |
|-------------|---|
| GL | Graus para cada linha da placa |
| LC | Total de linhas que a placa está rotacionada |
| LP_{BDMA} | Linha do pixel da borda direita da placa mais alta |
| LP_{BEMA} | Linha do pixel da borda esquerda da placa mais alta |
| AG | Ângulo para rotação |

SUMÁRIO

| | |
|--|-----------|
| INTRODUÇÃO | 13 |
| 1 SISTEMAS DE IDENTIFICAÇÃO DE PLACAS EM VEÍCULOS AUTOMOTORES | 15 |
| 1.1 Regulamentação das Placas | 15 |
| 1.2 Utilização | 17 |
| 1.3 Sistemas Desenvolvidos | 18 |
| 1.3.1 Sistemas Acadêmicos | 19 |
| 1.3.2 Sistemas Comerciais | 21 |
| 2 TÉCNICAS NORMALMENTE EMPREGADAS EM SISTEMAS DE RECONHECIMENTO DE PLACAS | 26 |
| 2.1 Imagem de Entrada | 27 |
| 2.2 Pré-processamento | 27 |
| 2.3 Localização da Placa | 32 |
| 2.4 Enquadramento da Placa | 35 |
| 2.5 Separação dos Caracteres | 36 |
| 2.6 Reconhecimento de Caracteres | 38 |
| 3 DESENVOLVIMENTO DO PROTÓTIPO | 40 |
| 3.1 Captura da Imagem | 40 |
| 3.1.1 Captura em Estacionamentos | 42 |
| 3.1.2 Captura nas Ruas | 43 |
| 3.1.3 Banco de Imagens | 43 |
| 3.2 Localização da Placa | 45 |
| 3.2.1 Metodologia Escolhida | 45 |
| 3.2.2 Localização de Objetos | 47 |
| 3.2.3 Testes de Localização no Protótipo e versão de demonstração do SeeCar | 52 |
| 3.3 Rotação da Placa | 56 |
| 3.4 Separação dos Caracteres | 58 |
| 3.4.1 Técnicas Utilizadas | 58 |
| 3.4.2 Testes de Separação de Caracteres no Protótipo | 59 |
| 3.5 Reconhecimento dos Caracteres | 62 |
| 3.5.1 Criando vetor de Características para Letras e Números | 62 |
| 3.5.2 Efetuando o Reconhecimento dos Caracteres | 64 |
| 3.5.3 Testes de Reconhecimento dos Caracteres | 65 |
| CONSIDERAÇÕES FINAIS | 67 |
| REFERÊNCIAS BIBLIOGRÁFICAS | 71 |

INTRODUÇÃO

Os centros urbanos do Brasil estão sofrendo cada vez mais com graves problemas no trânsito, como congestionamentos, fiscalizações e demora nos trajetos a serem percorridos, decorrentes principalmente do grande crescimento da frota de veículos automotores. Hoje em dia existe uma necessidade da engenharia de tráfego em obter informações de maneira rápida, segura e confiável na questão do reconhecimento de placas destes veículos. Através delas, pode-se aumentar a eficiência do controle e monitoramento, localização de carros roubados, aplicação de multas, controle de estacionamentos, cobrança de pedágios, acessos restritos, dentre outros.

Conforme Dias (2005), um ser humano demora cerca de um segundo para comparar o código do veículo com cada registro de uma lista, porém, com o avanço da computação, essas listas passaram a ser armazenadas em bancos de dados. Logo, um usuário treinado demora cerca de cinco segundos para identificar o código do veículo e digitá-lo no sistema.

Com o aumento da frota veicular nas ruas do país, necessita-se de um tempo de resposta de um sistema bem inferior a cinco segundos, o que não é possível por intermédio de um ser humano, motivando muitas pesquisas e desenvolvimento de sistemas de identificação automática de veículos.

Aparentemente esse trabalho de reconhecimento de placas pode parecer de fácil solução e implementação, até porque a placa possui caracteres bem definidos. Porém, existem diversos fatores que prejudicam os sistemas com esse propósito, sendo que a cada dia empresas que já desenvolveram esse tipo de sistema, procuram implementar novos algoritmos capazes de resolvê-los. Podem ser destacados alguns problemas, como: tamanho da placa e dos caracteres sem padrão, letras e números apagados ou sujos, dificultando toda a leitura da imagem, pintura reflexiva que alguns condutores utilizam (mesmo sendo proibido por lei), além de toda a questão meteorológica como iluminação precária ou chuva.

O principal objetivo do trabalho é estudar as técnicas de processamento de imagens, que aplicadas, permitirão identificar as letras e os números contidos em uma placa de veículo automotor, sem intervenção de qualquer tipo por um ser humano.

Faz parte do objetivo, também, aplicar as técnicas mostrando os resultados de cada etapa necessária para sistemas de reconhecimento de placas, através de um protótipo que é proposto.

Com relação a organização deste trabalho, no segundo capítulo é apresentada a legislação pertinente à questão das placas de veículos automotores. Além disso, é explicado onde os sistemas de reconhecimento podem ser úteis e para quais situações podem ser aplicadas, bem como são apresentados sistemas já desenvolvidos, tanto na área comercial, quanto na acadêmica, destacando suas características particulares.

O capítulo três demonstra como os sistemas de identificação de placas em veículos automotores funcionam e quais as etapas necessárias normalmente empregadas em sistemas já desenvolvidos. Em cada fase, são apresentadas as técnicas mais utilizadas, assim como alguns testes realizados.

No quarto capítulo são descritas todas as etapas realizadas e as técnicas empregadas para o desenvolvimento do protótipo, visando o reconhecimento automático das placas dos veículos automotores sem intervenção humana. Também são apresentados e explicados todos os resultados dos testes no protótipo proposto.

Para finalizar o TCII, apresentam-se às considerações finais de todo o estudo realizado e as perspectivas de futuros trabalhos provenientes dessa monografia, em busca de melhorias e continuidade dessa pesquisa.

1 SISTEMAS DE IDENTIFICAÇÃO DE PLACAS EM VEÍCULOS AUTOMOTORES

Todos os veículos automotores no Brasil precisam estar licenciados no órgão competente para trafegar, existindo duas maneiras para isso, sendo único para cada veículo. A primeira é um código chamado RENAVAM, mas que é somente marcado na documentação veicular e a outra, que é denominada placa de licenciamento veicular, de domínio público, pois essa informação pode ser identificada visualmente por qualquer pessoa.

Nesse capítulo será apresentado o funcionamento e a legislação sobre as placas de licenciamento veicular, bem como sistemas acadêmicos e comerciais existentes no mercado que trabalham com essa informação para diversos fins.

1.1 Regulamentação das Placas

Conforme Dias (2005), todos os países possuem uma legislação para padronizar a placa de licenciamento veicular. No Brasil, por exemplo, o órgão competente para essa tarefa é o CONTRAN (2007) que efetua a coordenação do Sistema Nacional de Trânsito. De acordo com a resolução número 45 de 1998 e de forma resumida, seguem as principais características contidas em uma placa:

- O veículo deverá ser identificado por placas dianteira e traseira.
- Deve conter na placa dois conjuntos de caracteres, sendo o primeiro composto por três letras e o segundo conjunto por quatro números.
- As placas deverão conter, gravados em tarjetas, a sigla identificadora da Unidade da Federação e o nome do município onde o veículo foi registrado, exceto para veículos oficiais.

- O tamanho da placa de veículos particulares, de aluguel, oficial, de experiência, de aprendizagem e fabricante devem ter as seguintes dimensões: altura = 130 mm e comprimento = 400 mm. Os caracteres têm 63 mm de altura e 10 mm de espessura do traço, porém, são toleradas variações de até 10% nas dimensões das placas e caracteres alfanuméricos das mesmas.
- A placa tem que ser retangular com fundo uniforme.
- O material utilizado na confecção das placas de identificação de veículos automotores poderá ser chapa de ferro laminado a frio, bitola 22, SAE I 008, ou em alumínio (não galvanizado) bitola 1 mm, enquanto que nas tarjetas, poderá ser em chapa de ferro, bitola 26, SAE 1008, ou em alumínio bitola 0,8.
- Os caracteres alfanuméricos são em alto relevo.

Essas características são fundamentais para o desenvolvimento de um sistema com a finalidade de identificação de placas de veículos automotores.

Cabe destacar que o CONTRAN tem uma nova resolução sobre as placas de veículos automotores, número 231, de março de 2007, onde acontecem várias modificações no padrão de placa. Esta deveria entrar em vigor a partir do dia primeiro de agosto de 2007, porém foi alterada pela resolução número 241 de junho de 2007, publicada em julho do mesmo ano, estabelecendo algumas mudanças. As diferenças pertinentes entre a resolução número 45 de 1998 e a que está vigorando no Brasil até o presente momento são:

- As dimensões da placa e dos caracteres são os mesmos da resolução número 45/98, porém não são mais tolerados os 10% de variação no tamanho da placa ou dos caracteres alfanuméricos. O modelo aprovado pode ser observado na Figura 1.1.
- Todos os carros zero quilômetro e os que forem trocadas as placas, ou seja, emplacados em outra cidade, deverão se enquadrar nas novas características das mesmas.
- A tipologia dos caracteres das placas e das tarjetas deverá seguir o modelo exposto na Figura 1.2, mais especificamente na fonte Mandatory.



Figura 1.1 – Modelo de Placa conforme CONTRAN.
Fonte: CONTRAN (2007)

1234567890
ABCDEFGHIJKLM
NOPQRSTUVWXYZ

Figura 1.2 – Caracteres na fonte Mandatory.
Fonte: CONTRAN (2007)

Essas mudanças irão beneficiar muito os sistemas de reconhecimento de placas, principalmente por ter um tamanho definido e caracteres com fonte capazes de serem identificados de forma mais fácil do que outros com fontes em que as características de determinadas letras se aproximam muito, como é o caso da letra O e a letra Q. Para elucidar melhor, basta observar a Figura 1.3 e a Figura 1.4.



Figura 1.3 – Caractere O - Fonte não Mandatory.



Figura 1.4 – Caractere Q - Fonte não Mandatory.

1.2 Utilização

Uma vez desenvolvida toda a tecnologia necessária para a utilização do reconhecimento de placas de veículos automotores, a destinação do produto ou aplicação do mesmo pode se dar para inúmeros fins, como:

- **Estacionamentos:** alguns estacionamentos, principalmente aqueles que cobram tarifa, necessitam saber quando o veículo entrou e quando saiu, o tempo que ficou lá dentro, entre outras, para que seja depois emitido um relatório ou realizada a cobrança de um determinado período.
- **Controle de Acesso a Áreas Restritas:** muitas empresas possuem controle de acesso a determinadas áreas e só aceitam a passagem de veículos se o mesmo já estiver previamente cadastrado e autorizado, porém, na maioria dos casos, essa tarefa só é realizada por intermédio de guardas.
- **Pedágios:** pode controlar quem tem permissão para passar ou então barrar, sem perda de tempo de um usuário ficar procurando no sistema se está liberado ou não.
- **Identificação de Carros Roubados:** quando um carro é roubado, sua placa vai parar em um banco de dados da polícia. Neste caso, poderiam ser instalados sistemas em qualquer ponto do país para monitorar o trânsito na busca por carros roubados e sempre que encontrados, a polícia mais próxima receber a informação para tomar as devidas providências.
- **Fiscalização:** saber, de forma ágil, se tem algum carro com infração, multa ou pendência.
- **Monitoração de Tráfego:** os responsáveis pela engenharia do tráfego podem saber qual a velocidade média de um veículo em determinado trecho da estrada, bem como, acompanhar a trajetória e até localização se, por ventura, tiver algum problema e não conseguir entrar em contato com o socorro.

1.3 Sistemas Desenvolvidos

Hoje em dia diversos centros acadêmicos e universidades do país estão desenvolvendo seu conhecimento sobre processamento digital e técnicas utilizadas para sistemas de reconhecimento de placas, a fim de ajudar a diminuir os engarrafamentos nesse trânsito caótico em que se encontra o Brasil. Além dessas instituições, empresas privadas já descobriram que é uma área pouco explorada e com potencial para expansão dos seus negócios.

Esses sistemas podem ser divididos em acadêmicos, sendo alguns com apoio de empresas externas e outros por empresas não vinculadas a nenhum grupo de estudo. Serão citados, a seguir alguns produtos disponíveis no mercado.

1.3.1 Sistemas Acadêmicos

SIIV: de acordo com Rodrigues (2003), o sistema chamado SIIV foi desenvolvido pela empresa Automatiza, com cooperação da UFRGS. Esse sistema pode ser aplicado em controle de estacionamentos, verificação de carros roubados e em segurança.

KAPTA: foi outro sistema criado em universidade, mas que teve um suporte financeiro do Departamento de Trânsito do Estado do Rio de Janeiro - DETRAN-RJ. No momento, esse sistema está para virar um produto, sendo chamado de KAPTA (KAPTA, 2007). Esse é realizado no âmbito do Laboratório de Inteligência Computacional da UFRJ.

O software foi separado em três módulos: Captura, Reconhecimento e Aplicação. No módulo de captura, transforma a imagem real em um padrão digital. No segundo módulo, são realizadas quatro etapas: Localização da Placa, Segmentação, Extração das Características e o Reconhecimento. O último módulo tem a função de pegar os caracteres da placa lidos e utilizar para os mais diversos fins, de acordo com o que os usuários desejam.

O software localiza, extrai e reconhece automaticamente os caracteres e dígitos da placa de um veículo e, segundo informações dos criadores, o sistema tem um índice de acerto aproximado de 95%. Pode ser utilizado para diversas áreas como: controle e segurança de acesso, controle de tráfego, pontos de controle, segurança e controle de estacionamento, registro de infração e blitz eletrônica. A Figura 1.5 apresenta uma tela do sistema KAPTA.



Figura 1.5 – Sistema Kapta.

Fonte: KAPTA (2007)

LOOKAR: na UNICAMP, em São Paulo, o Dr. Roberto de Alencar Lotufo e o aluno de mestrado Eng. Fábio Gaiotto Dias iniciaram um projeto com objetivo de elaborar uma dissertação de mestrado (DIAS, 2005) baseado no desenvolvimento de um sistema para reconhecimento de placas de veículos automotores. Porém, perceberam, no decorrer do projeto, o potencial comercial que teria um produto com essa finalidade e acabaram patentando a principal etapa do processo que, segundo eles, é chamada de "Método de Localização de Cadeias Alfanuméricas Tipográficas em Superfícies de Fundo Uniforme para um Sistema de Reconhecimento Baseado em Imagem". Foi então colocado o nome no projeto de LooKar (LOOKAR, 2007).

De acordo com os autores, foi desenvolvido um protótipo que utiliza técnicas de processamento de imagens digitais e de inteligência artificial, tendo como principal ferramenta, a morfologia matemática. Sua plataforma de desenvolvimento foi o MATLAB versão 6.5 e a toolbox de morfologia matemática SDC Morphology Toolbox versão 1.2. Conforme os autores, foi iniciada uma versão comercial, chamada LooKar, desenvolvido em Python e as duas versões passaram por testes em 1200 imagens por eles selecionadas. Os resultados mostram que na versão inicial do projeto a taxa de acerto ficou em 69%, enquanto que nessa versão comercial, atingiram 87,3%. A Figura 1.6 apresenta uma tela do protótipo do sistema Lookar.

O sistema foi separado em algumas etapas: pré-processamento, localização da placa, enquadramento da placa, separação dos caracteres e reconhecimento dos caracteres.



Figura 1.6 – Sistema Lookar.

Fonte: LOOKAR (2007)

1.3.2 Sistemas Comerciais

SEECAR: um dos sistemas mais conhecidos no mundo é o SeeCar (SEECAR, 2007), desenvolvido pela empresa Hi-Tech Solutions de Israel e que é utilizado em vários países como Estados Unidos, Portugal, Colômbia, África do Sul, dentre outros. Esse sistema pode ser empregado para controle de estacionamentos, monitoração de tráfego e para segurança. A Figura 1.7 apresenta uma tela do programa SeeCar, mais especificamente da versão de demonstração disponibilizada no site da empresa.

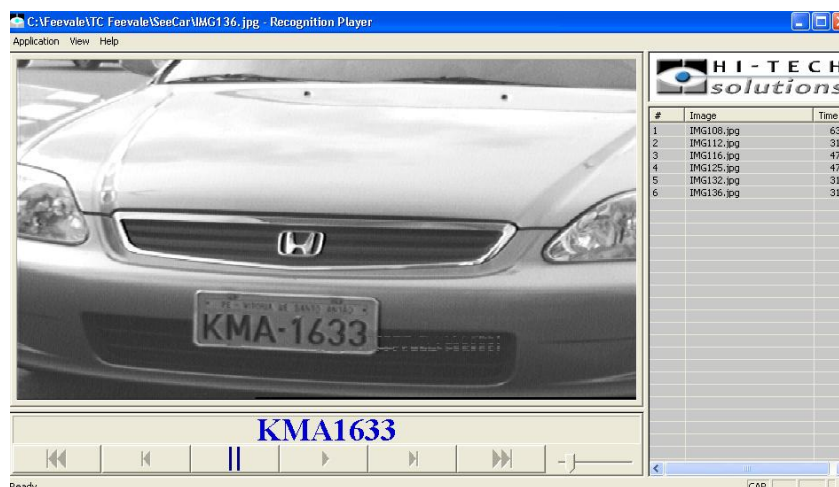


Figura 1.7 – Imagem da Versão de Demonstração do Sistema SeeCar.

Fonte: SEECAR (2007)

PONFAC: a PONFAC/SA (PONFAC, 2007), localizada no pólo de Informática de São Leopoldo, no Rio Grande do Sul, possui um sistema de leitura de placas que é utilizada para controle de acesso em estacionamentos, que pode ser utilizado em estacionamentos rotativos, o que permite o registro do tempo do veículo no estabelecimento, bem como, controle sobre os automóveis mensalistas. Para pedágios, o software consegue ajudar a calcular o valor devido no pedágio de acordo com a classificação do veículo. Já nos condomínios, shoppings centers, hipermercados e áreas de acesso restrito, pode-se controlar o tempo no estabelecimento, horários de visitação, monitoramento de quem entra e sai. Outra possibilidade é o controle de tráfego em rodovias: mesmo para veículos que estão em movimento, ajuda a encontrar veículos roubados ou com alguma irregularidade.

De acordo com a empresa, o sistema detecta a presença do veículo no ponto desejado através de sensores, capturando a imagem frontal do veículo, como pode ser visto na Figura 1.8. Uma vez capturada a imagem, esta é enviada ao computador, digitalizada e interpretada para identificação da placa.



Figura 1.8 – Imagem sendo capturada.

Fonte: PONFAC (2007)

A PONFAC/SA destaca alguns benefícios que o seu sistema possui como a segurança, que quando combinada com outra tecnologia, tipo cartão magnético, evita o portador do cartão sair com um veículo diferente do que entrou. Dado que o reconhecimento é feito automaticamente, não existe a necessidade de interação com condomínios ou mensalistas. Se for comparado com sistemas de cartão ou emissão de tickets, além do custo fixo ser menor, a manutenção é inferior também.

A empresa salienta que o sistema é uma ferramenta de marketing para estabelecimentos comerciais, pois pode controlar quais clientes estão comparecendo no estabelecimento, bem como, o tempo de permanência. Enfim, podem ser gerados vários relatórios em função das informações adquiridas.

Segundo a PONFAC/SA, o tempo de resposta do sistema para reconhecimento da placa é de apenas dois segundos e todo o software foi desenvolvido dentro da empresa em C++, com utilização de redes neurais para identificação dos números.

TECNIMA: a empresa TECNIMA Imagem & Automação Ltda (TECNIMA, 2007) possui dois produtos que são baseados em reconhecimento de placas através de leitura automatizada. Um deles se chama EVA Leitor de Placas e é utilizado para controle de acesso de veículos em condomínios e estacionamentos e o outro é o EVA Trânsito, como pode ser visto na Figura 1.9, que identifica em tempo real a placa de veículos em movimento para compará-la com banco de dados de veículos irregulares.



Figura 1.9 – Placa sendo identificada em tempo real.

Fonte: TECNIMA (2007)

O EVA Leitor de Placas realiza o controle de acesso de veículos em condomínios e estacionamentos, efetuando o seu reconhecimento por leitura automatizada de sua placa. Uma vez lida essa informação, são apresentados na tela os dados e fotos cadastrais a ele associados, permitindo que o porteiro comande a abertura do acesso ou que o próprio sistema, através de sensores e de módulo de controle dedicado, comande a abertura e o fechamento de cancelas e portões.

A empresa salienta que com um sistema desses implantado, não é necessário mais de porteiros, diminuindo o custo direto para condôminos ou mensalistas. Outro benefício é o baixo custo de implantação por integrar as funções de controle e registro do acesso, além da questão da segurança por dispensar a abertura da janela por parte do motorista e permitir a comparação visual com as fotos cadastradas dos veículos e motoristas autorizados.

A EVA recebe de um sensor ou através de comando de um operador a informação de que um veículo aguarda a autorização para entrar ou sair da área controlada. Após realiza-se a captura da imagem do veículo enquadrado de forma a destacar a sua placa frontal, facilitando a leitura da mesma. Em menos de meio segundo o sistema fornece o resultado da leitura e da pesquisa no banco de dados de veículos cadastrados, apresentando-o em sua tela.

O software opera automaticamente com imagens capturadas nas resoluções de 640x480, 640x240 e 320x240 pixels.

O EVA Trânsito é um sistema que permite identificar em tempo real a placa dos veículos em movimento e compará-la com banco de dados de veículos irregulares. Serve principalmente para monitoramento de trânsito em busca de carros roubados ou com qualquer outra irregularidade, mas deve ser ressaltado que ele só será útil caso a pesquisa da placa seja realizada em banco de dados de veículos com qualquer irregularidade.

A TECNIMA ainda oferece a possibilidade de uma DLL para outras empresas criarem suas próprias aplicações, integrando a tecnologia nos seus próprios produtos.

COMPULETRA: a Compuetra Ltda (COMPULETRA, 2007) possui um sistema de identificação de veículos em movimento com finalidade de auxiliar a fiscalização de veículos automotores em tempo real. No caso de ser constatada uma irregularidade do veículo após pesquisa em banco de dados da fiscalização, essa feita através da placa, o sistema avisa o agente de fiscalização de forma visual e sonora. A empresa não disponibiliza nenhuma outra informação e nem imagem do seu software.

Nenhum dos sistemas comerciais apresenta em seu material um índice aproximado de acertos.

Foram feitos contatos com as empresas desenvolvedoras dos sistemas anteriormente apresentados, com o intuito de obter maiores informações sobre os mesmos, ou cópias demonstrativas para testes. Apesar da insistência, segundo as empresas, não disponibilizam

maiores informações e o motivo é que estariam abrindo mão do seu diferencial em relação à concorrência. Outro argumento é que se fornecessem informações relativas a técnicas utilizadas, poderiam facilitar o desenvolvimento de novos softwares ou melhorias dos já existentes, que não estão com um nível satisfatório de reconhecimento. Dessa maneira, não foi possível fazer um comparativo sobre os softwares.

2 TÉCNICAS NORMALMENTE EMPREGADAS EM SISTEMAS DE RECONHECIMENTO DE PLACAS

Os sistemas de reconhecimento de placas possuem várias etapas, que não são necessariamente utilizadas por todos os desenvolvedores. Cada uma delas pode ser desenvolvida através de diferentes técnicas, definidas exclusivamente por cada desenvolvedor/pesquisador, que julgam ser mais apropriadas para chegar ao resultado, neste caso, o reconhecimento das letras e números contidos na placa de veículos automotores. Nesse capítulo serão abordados, de forma sucinta, os módulos necessários de acordo com o estudo já realizado e com o que será aplicado no protótipo, como pode ser visto na Figura 2.1. Mais adiante será estudada cada etapa detalhadamente.

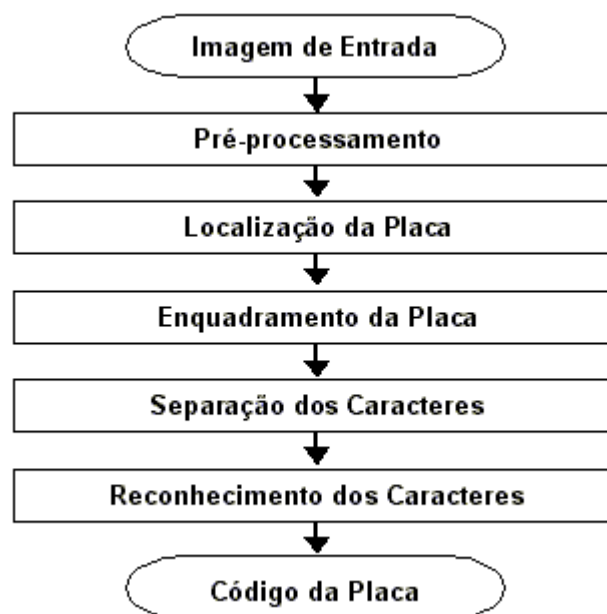


Figura 2.1 – Diagrama com os Módulos necessários do Estudo Proposto.
Fonte: DIAS (2005)

2.1 Imagem de Entrada

“O primeiro passo no processo é a aquisição da imagem, isto é, adquirir uma imagem digital” (GONZALEZ e WOODS, 2000, p. 5). Essa imagem pode ser obtida por câmera filmadora ou por foto.

2.2 Pré-processamento

Realizada a captura da imagem, necessita-se pré-processar para melhorá-la através de técnicas de realce de contraste e eliminação de ruídos. “O realce tem por objetivo destacar detalhes da imagem que são de interesse para análise ou que tenham sofrido alguma deterioração” (CONCI, AZEVEDO e LETA, 2008, p. 55). “Imagens reais frequentemente sofrem degradações durante seu processo de aquisição, transmissão ou processamento. Essa degradação é normalmente chamada de ruído” (PEDRINI e SCHWARTZ, 2008, p.26).

A Figura 2.2 apresenta uma imagem em um ambiente externo e que servirá como base para os primeiros testes realizados, que serão explicados nesse capítulo.



Figura 2.2 – Imagem original em ambiente externo.

A melhoria nas imagens é realizada através de técnicas para extração de ruídos, além de alterações de brilho e contraste. A intenção é realçar a imagem e eliminar detalhes não desejados. “A função chave no pré-processamento é melhorar a imagem de forma a aumentar as chances para o sucesso dos processos seguintes” (GONZALEZ e WOODS, 2000, p. 6). Na Figura 2.3, pode-se observar a mesma imagem da Figura 2.2, contendo ruídos que devem ser suavizados.



Figura 2.3 – Imagem em ambiente externo com ruído.

Pode ser utilizado o histograma nesta etapa. “O histograma possibilita equilibrar a curva de níveis de cinza realçando o contraste da imagem” (RODRIGUES, 2003, p. 28). O histograma permite fazer uma leitura da imagem, pois para cada nível de cinza, consegue fornecer o número de pixels correspondentes na imagem. Um pixel é cada um dos pontos que formam uma imagem digital, sendo caracterizado por sua posição na imagem e pelo seu valor (DAVIS, 1992). “Várias medidas estatísticas podem ser obtidas a partir do histograma de uma imagem, tais como os valores mínimo e máximo, o valor médio, a variância e o desvio padrão dos níveis de cinza da imagem” (PEDRINI e SCHWARTZ, 2008, p.104). A Figura 2.4 apresenta o Histograma da Figura 2.2. A Figura 2.5 apresenta uma imagem com o histograma equalizado e na Figura 2.6, é apresentada a representação desse histograma.

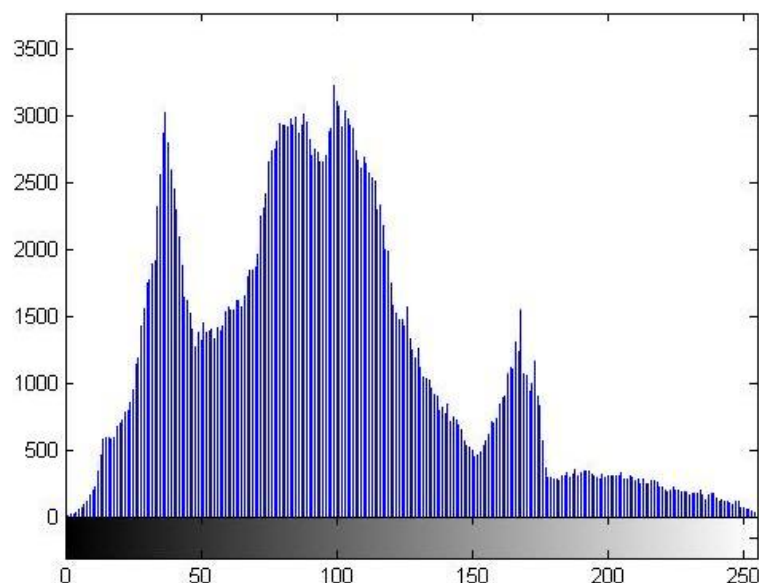


Figura 2.4 – Histograma da Figura 2.2.



Figura 2.5 – Imagem com Equalização do Histograma.

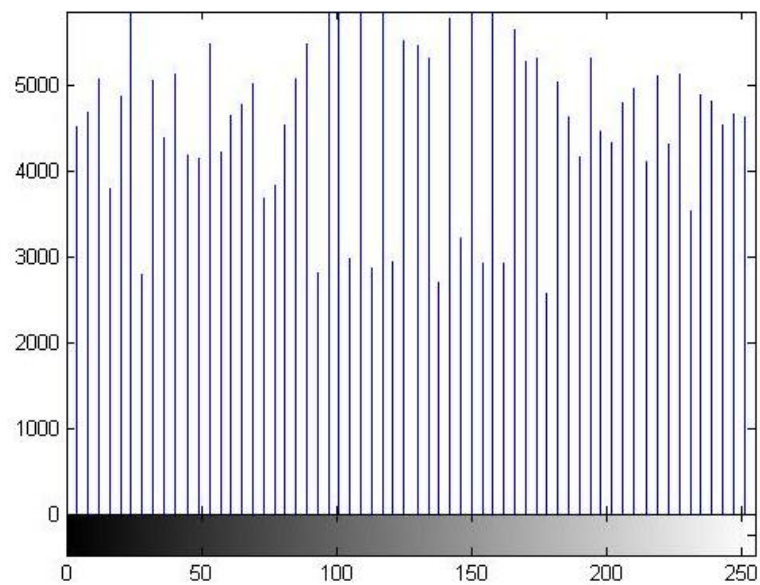


Figura 2.6 – Histograma da Imagem apresentada na Figura 2.5.

A Figura 2.7 apresenta uma imagem onde o seu histograma foi equalizado e, além disso, alterado o contraste da mesma. A Figura 2.8 apresenta o histograma da Figura 2.7.



Figura 2.7 – Imagem Equalizada e com contraste.

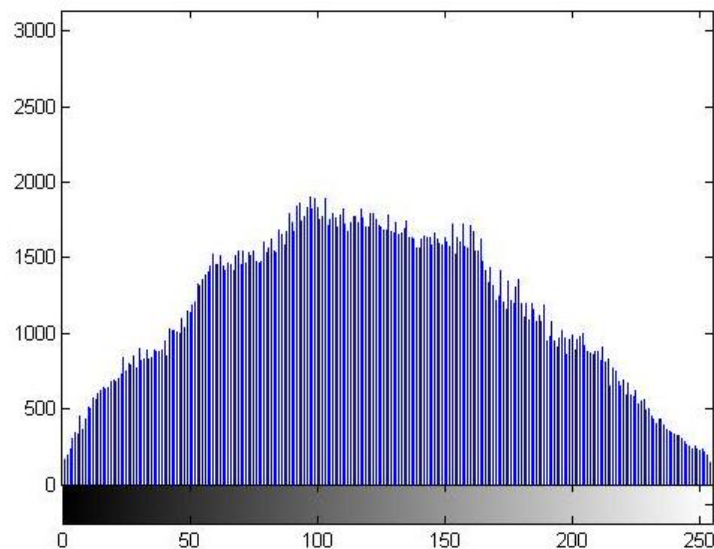


Figura 2.8 – Histograma da Figura 2.7.

Pode-se também suavizar a imagem através de uma função de transformação que distribui os níveis de cinza. O histograma acaba modificando os pixels da imagem completa para realçá-la, mas não garante que o local desejado seja realçado. Em alguns casos, existe a necessidade de efetuar esse procedimento em pequenas regiões, mais especificamente na vizinhança de cada pixel da imagem.

Existem técnicas de filtragem que efetuam a transformação da imagem pixel a pixel, levando em consideração não apenas o nível de cinza de um pixel escolhido, mas também de acordo com o valor dos níveis dos pixels de seus vizinhos. Esse processo utiliza matrizes aplicadas na imagem, as quais são denominadas máscaras.

Pode-se aplicar um filtro que suavizará a imagem atenuando as altas frequências (passa-baixa), mas que produzirá um efeito de borramento, diminuindo a nitidez da imagem. Conforme Rodrigues (2003), o processo de filtragem é feito utilizando matrizes denominadas máscaras, que são aplicadas sobre a imagem. “A aplicação da máscara consiste na substituição do valor do pixel por um novo valor que depende dos valores dos pixels vizinhos e dos pesos da máscara, gerando uma nova imagem” (RODRIGUES, 2003, p. 35). As máscaras podem ser de vários tamanhos, dependendo do tamanho da imagem e do ruído. A Figura 2.9 apresenta esse filtro aplicado na média da vizinhança.

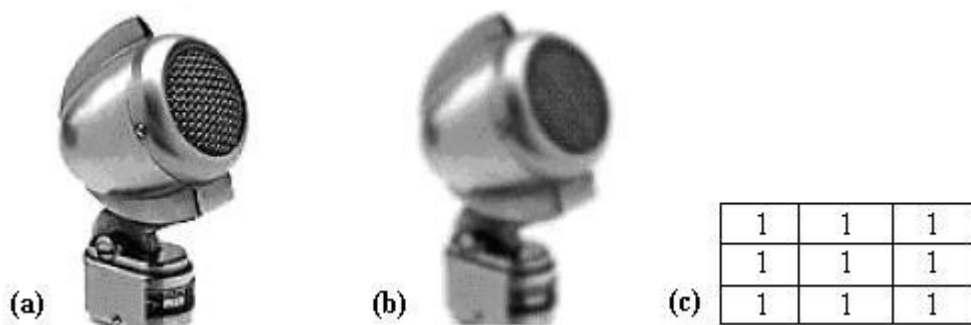


Figura 2.9 – Suavização pela Média, utilizando uma máscara 3x3. (a) Imagem Original; (b) Imagem Filtrada; (c) Máscara 3x3.

Fonte: RODRIGUES (2003)

Pode-se observar que as bordas dessa imagem ficaram borradas, assim como outros detalhes, mas obteve-se uma imagem mais uniforme. Caso o desejo seja diminuir o ruído em vez de borrar, pode-se aplicar o filtro da mediana, onde cada pixel é substituído pela mediana dos níveis de cinza da imagem. Utilizando a mediana, consegue-se melhorar as bordas da imagem.

A Figura 2.10 apresenta uma suavização pelo filtro da mediana.

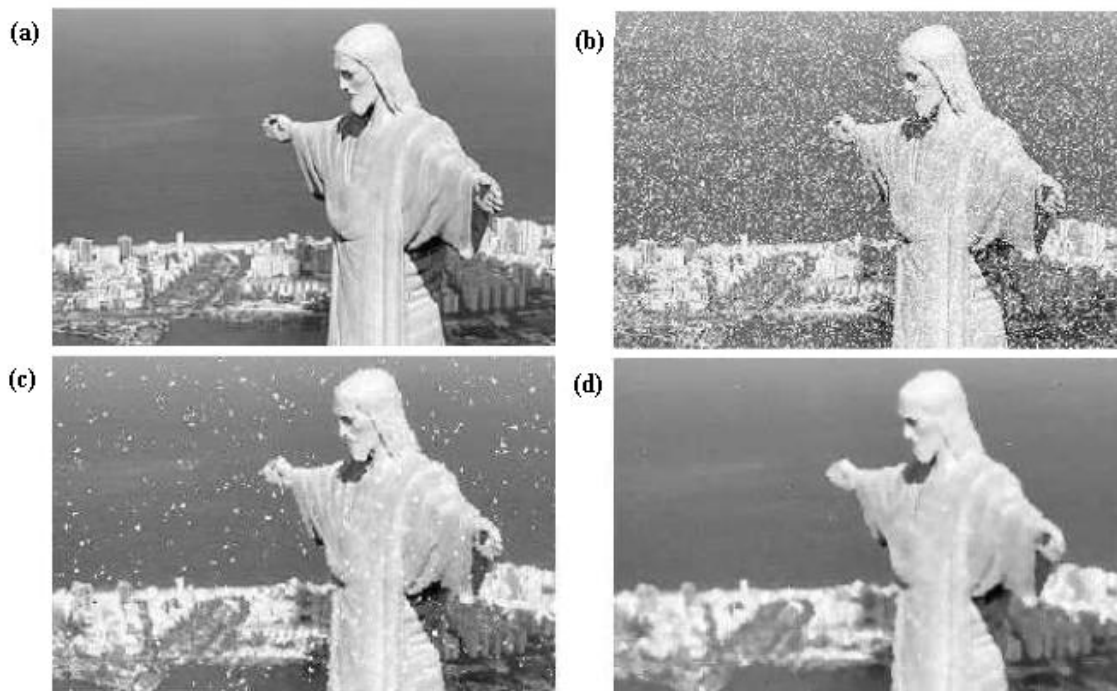


Figura 2.10 – Aplicação do filtro da mediana. (a) Imagem Original; (b) Imagem corrompida por um ruído aleatório; (c) Imagem com Filtro por Mediana com Máscara 3x3; (d) Imagem com Filtro por Mediana com Máscara 5x5.

Fonte: BARBOSA (2004)

2.3 Localização da Placa

Uma vez trabalhada a imagem, é necessário localizar as regiões candidatas a placa para realizar os testes necessários até o isolamento da parte que a contém, retirando do restante da imagem, para, a partir daí, só trabalhar em um pequeno pedaço de imagem. Se o sistema não conseguir localizar a placa, todas as demais etapas de nada servirão. A Figura 2.11 é um exemplo de uma placa de veículo que foi localizada, mas que ainda contém parte do carro, além da distorção que deverá ser corrigida na próxima etapa, que é a do enquadramento.



Figura 2.11 – Placa do Veículo Localizada.

Existem alguns métodos que podem ser aplicados para localização da placa, apresentados a seguir:

Método baseado na Morfologia dos Caracteres: leva em consideração vários fatores como os caracteres da placa que possuem espaçamento e altura regular. Uma grande dificuldade, nesse caso, é a existência nos carros de outras letras como marca do fabricante, nome do carro, dentre outras. Outro fator é a cor dos caracteres que é diferente do fundo da placa. A idéia é efetuar uma varredura em toda a imagem em busca de uma variação tonal, ou transição da cor, o que significaria letra e fundo da placa. A dificuldade está em localizar regiões que não seriam da placa, como a grade do radiador e paralelepípedos.

Método baseado na Morfologia do Fundo da Placa: leva em consideração que a placa tem o formato retangular e fundo uniforme. Pode ser utilizado um detector de bordas e depois um de retas. A idéia é efetuar a segmentação da imagem através de uma técnica de detecção de bordas, como pode ser observado na Figura 2.12. Logo após, utilizar uma técnica para encontrar as arestas da placa do veículo. Sabe-se que todas as placas no país possuem quatro arestas, sendo duas horizontais e duas verticais e ainda paralelas entre si. Uma vez encontradas essas retas, pode-se traçar a região na imagem onde existe a placa, efetuando o recorte dessa parte e salvando como outra imagem.

Outra maneira é procurar na imagem regiões que possuam cor e textura semelhantes às de uma placa. Uma vez encontradas, devem ser processadas para verificar suas características até chegar a que mais se aproxima de uma placa.

Métodos Híbridos: misturam os dois métodos anteriormente mencionados para obter uma melhor localização.

Após a localização da região da placa, não será mais necessário trabalhar com toda a imagem do veículo, mas somente com a da região selecionada. Uma técnica muito utilizada para essa etapa é a de segmentação, como será exposto a seguir.

“A segmentação subdivide uma imagem em suas partes ou objetos constituintes. O nível até o qual essa subdivisão deve ser realizada depende do problema a ser resolvido” (GONZALEZ e WOODS, 2000, p. 295), ou seja, a segmentação deverá parar quando os objetos de interesse na aplicação tiverem sido isolados. “Na maioria das aplicações, a segmentação de imagens é um estágio prévio de processamento para reconhecimento ou identificação das imagens” (CONCI, AZEVEDO e LETA, 2008, p. 199).

Existem dois conceitos importantes na segmentação: a similaridade e a descontinuidade. “As descontinuidades são representadas pelas mudanças bruscas nos níveis de tons, cores e texturas” (CONCI, AZEVEDO e LETA, 2008, p. 199). “Os tipos básicos de descontinuidades normalmente detectadas em imagens digitais são pontos, segmentos de retas, junções e bordas” (PEDRINI e SCHWARTZ, 2008, p. 152). É possível efetuar, assim, a detecção de pontos isolados, bem como a detecção de linhas ou ainda as bordas da imagem, características fundamentais encontradas em placas de veículos.

Na similaridade, [...] “os métodos procuram agrupar pontos da imagem que apresentam valores similares para um determinado conjunto de características” (PEDRINI e SCHWARTZ, 2008, p. 151). As principais abordagens na similaridade [...] “baseiam-se em limiarização, crescimento de regiões e fusão de regiões” (GONZALEZ e WOODS, 2000, p. 295).

Em decorrência das imagens dos veículos automotores serem capturadas em ambientes externos, para esse tipo de aplicação, adota-se, preferencialmente, o conceito de descontinuidade, pois existem mudanças bruscas de níveis de cinza, principalmente na questão do ambiente e da placa que se distingue do restante do carro. [...] “A detecção de bordas é, de longe, a abordagem mais comum para a detecção de descontinuidades significantes nos níveis de cinza. A razão é que pontos e linhas finas isoladas não são ocorrências frequentes na maioria das aplicações práticas” (GONZALEZ e WOODS, 2000, p. 297).

Além da segmentação, é necessário utilizar a detecção de bordas para localizar a placa. “Uma borda é o limite ou a fronteira entre duas regiões com propriedades relativamente distintas de nível de cinza” (PEDRINI e SCHWARTZ, 2008, p. 153). Como existe a necessidade de se identificar e classificar os objetos encontrados em uma imagem, a detecção de bordas torna-se de suma importância nesse processo de análise de imagens.

“Os métodos para detecção de bordas têm como objetivo detectar descontinuidades na imagem. Eles são fundamentais quando se deseja fazer o reconhecimento de objetos” (CONCI, AZEVEDO e LETA, 2008, p. 237). Uma vez detectadas as bordas, pode-se localizar todos os objetos de uma imagem, que neste caso seria a placa do veículo e, ainda, se conseguiria definir propriedades como área e perímetro.

Dentre as principais técnicas de detecção de bordas existentes baseadas no gradiente, podem ser citadas: Canny (CANNY, 1986), Sobel (RUSS, 1998), Roberts (RUSS, 1998) e Prewitt (RITTER e WILSON, 1996). A seguir é apresentada uma imagem onde foi aplicada a detecção de bordas Prewitt.



Figura 2.12 – Técnica Prewitt – Detecção de bordas.

2.4 Enquadramento da Placa

O próximo estágio é do enquadramento da placa. Este estágio é necessário porque as imagens normalmente são obtidas em ângulos que dificultam o reconhecimento dos caracteres, ocorrendo uma distorção, devido a câmera não ficar posicionada de modo ortogonal à placa. Em função disso, a imagem adquirida na etapa anterior contém, além da placa, partes do veículo. Com a correção da distorção, através das quatro arestas da placa, consegue-se obter uma imagem contendo somente a placa.

Através do ângulo formado pelas quatro arestas da placa é possível corrigir a perspectiva da mesma, acertando a rotação da placa e após, corrigindo a inclinação dos seus caracteres, como pode ser visto na Figura 2.13.



Figura 2.13 – (a) Placa do Veículo Enquadrada; (b) Placa com Detector de Bordas Prewitt.

2.5 Separação dos Caracteres

Após essa correção, é necessário separar da imagem sete novas imagens com cada letra e número da placa para, posteriormente, ser trabalhada a etapa do reconhecimento de cada caractere ASCII encontrado.

Caso a placa do veículo não tenha ruído e nenhuma das letras esteja se encostando, é uma das etapas mais simples. A primeira tarefa a ser executada é pegar a imagem da etapa anterior, a do enquadramento, e transformá-la para binária, como na Figura 2.14.

Os pixels das imagens podem ter uma variação de tons de cinza, variando sempre da cor preta que possui a menor intensidade para cor branca que possui maior intensidade. Em uma imagem, pode-se ter vários níveis de cinza, mas se ela for binária, então é uma imagem que possui apenas dois valores, que podem ser 0 ou 1, ou seja, preto ou branco. “Uma imagem binária é um *array* lógico de zeros e uns” (GONZALEZ, WOODS e EDDINS, 2002, p. 25).



Figura 2.14 – Imagem Binária da Placa Veicular.

O próximo passo é enquadrar cada caractere para que seja possível, na próxima etapa, efetuar o reconhecimento do mesmo, que pode ser realizado baseando-se nos componentes conexos, porém os caracteres não podem estar rompidos. “A conectividade entre pixels é muito importante para estabelecer bordas de objetos e componentes de regiões de uma imagem” (CONCI, AZEVEDO e LETA, 2008, p. 214).



Figura 2.15 – Caracteres Enquadrados.

“Para verificar se dois elementos são conexos é necessário determinar se eles são vizinhos segundo o tipo de vizinhança adotado e se os elementos satisfazem determinados critérios de similaridade, tais como intensidade de cinza, cor ou textura” (PEDRINI e SCHWARTZ, 2008, p. 31). Conforme Shapiro e Stockam (2000), em muitos algoritmos não basta ter somente o valor de um determinado pixel, mas é importante saber os valores dos

seus pixels vizinhos para processar aquele pixel. Comentam ainda que as definições mais comuns para vizinhos são: vizinhança-4 e vizinhança-8. De acordo com GONZALEZ e WOODS (2000), um pixel p nas coordenadas (x, y) possui quatro vizinhos horizontais e verticais, representadas pelas coordenadas:

$$(x + 1, y), (x - 1, y), (x, y + 1), (x, y - 1)$$

Os autores dizem que esse conjunto de pixels é chamado de vizinhança-de-4 de p , e que é representado por $N_4(p)$, onde cada um dos pixels está a uma unidade de distância de (x, y) . A Figura 2.16 apresenta a relação de vizinhança de um pixel.

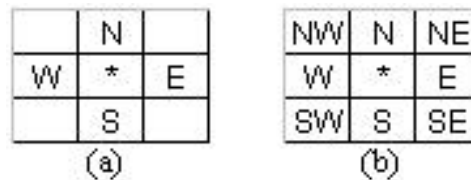


Figura 2.16 – Vizinhança do Pixel. (a) Vizinhança-4; (b) Vizinhança-8.

Fonte: SHAPIRO e STOCKAMN (2000)

“Uma vez definida uma conectividade entre os pixels, é possível definir um caminho conexo. Dois pixels são conexos se existir um caminho entre eles. Um objeto na imagem segmentada é usualmente definido como um componente conexo” (CONCI, AZEVEDO e LETA, 2008, p. 214). A Figura 2.17 apresenta uma imagem bidimensional, que contém três componentes conexos, considerando a vizinhança-4. Se for considerada a vizinhança-8, então a mesma imagem têm dois componentes conexos.

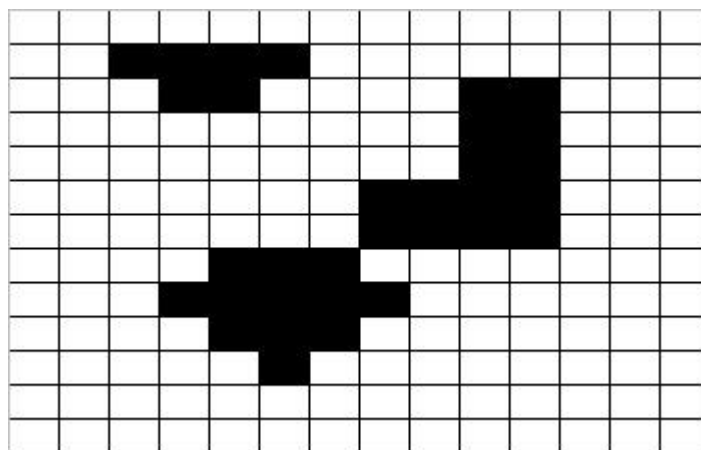


Figura 2.17 – Componentes conexos em uma imagem.

Fonte: PEDRINI e SCHWARTZ (2008)

A última etapa é de separar cada caractere em uma nova imagem, formando um total de sete pequenas imagens, como pode ser observado na Figura 2.18.

I M S 8 8 7 9

Figura 2.18 – Caracteres Separados formando sete novas imagens.

2.6 Reconhecimento de Caracteres

Para finalizar, junta-se o resultado da segmentação de cada caractere para obter-se o conteúdo da placa, ou seja, sua identificação, para que um sistema posteriormente consiga trabalhar com essa informação em busca dos objetivos por ele definidos.

A placa veicular é composta por letras do alfabeto, que são 26, além dos números que vão de 0 à 9, que formam mais 10 caracteres, compondo, no total, 36 possibilidades. Cada possibilidade deverá ter várias amostras, que serão inseridas em um vetor de características. A amostra corresponde a um código ASCII.

O reconhecimento é dado pela comparação do caractere candidato, que neste caso é uma das sete imagens da etapa de separação, com o vetor de características. A que mais se aproximar da candidata será o correspondente ASCII.

As dificuldades dessa etapa estão relacionadas as semelhanças da tipografia dos caracteres diferentes, como a letra “O” e o número “0”, além da letra “D” e a letra “Q”. Isso pode ser observado na Figura 2.19.



Figura 2.19 – Semelhanças da Tipografia de Caracteres Diferentes.

Fonte: SEECAR (2007)

Também são encontrados problemas relacionados a diferentes tipos de tipografia do mesmo caractere, que pode ser acompanhado na Figura 2.20, o que deverá ser corrigido, como foi visto na nova resolução do CONTRAN, que agora prevê só um tipo de fonte para placas, bem como, as diferenças significativas das características dos caracteres que estarão bem definidas com o tipo de fonte Mandatory.



Figura 2.20 – Placa com Tipografia do mesmo caractere diferente. (a) Placa que contém o caractere “6” com características bem arredondadas; (b) Placa que contém o caractere “6” bem diferente da placa anterior dessa imagem.

Fonte: SEECAR (2007)

O reconhecimento poderá ser realizado por técnicas estatísticas, utilizando cálculo de distâncias, onde se procura por um padrão que mais se aproxima das amostras que foram inseridas no vetor, ou então, por redes neurais que utilizam um conjunto de amostras para treinar uma rede de elementos.

Outra opção é utilizar um programa OCR, porém, conforme RODRIGUES (2003), os programas de OCR nos testes realizados apresentaram bons resultados a partir de recortes perfeitos de placas com boa qualidade, mas quando tinham bordas e um ruído elevado, falhavam. O mesmo autor comenta ainda que esses programas não são hábeis para localizar e extrair as placas quando submetidas a partir de imagens completas do veículo.

Também existe a possibilidade de efetuar o reconhecimento através de redes neurais. Neste caso, deve-se criar uma rede de treinamento para diminuir o erro, através de amostras de cada caractere. Após, consegue-se ter um padrão para que seja comparado com as imagens candidatas.

O reconhecimento é dado pela comparação do caractere candidato, que neste caso é uma das sete imagens da etapa de separação, com o vetor de características. A que mais se aproximar da candidata, será o correspondente ASCII.

No capítulo seguinte será apresentado o desenvolvimento do protótipo. As definições e os testes existentes para cada etapa do processo serão explicados e demonstrados.

3 DESENVOLVIMENTO DO PROTÓTIPO

Partindo do exposto no segundo capítulo, sobre softwares acadêmicos, percebe-se que as etapas necessárias para o desenvolvimento de um software que faça o reconhecimento de placas de veículos automotores são várias. Porém, deve ser ressaltado que apesar dos nomes para as etapas ou técnicas empregadas ser diferentes entre cada software, as fases não diferem do diagrama apresentado no terceiro capítulo.

Esse capítulo abordará as fases empregadas no protótipo, que são: captura da imagem, localização da placa, rotação da placa, separação dos caracteres e o reconhecimento dos mesmos. Para tanto, serão apresentadas as técnicas utilizadas e os testes realizados.

O ambiente de desenvolvimento deste protótipo foi o seguinte:

- a) Software para desenvolvimento das etapas do projeto e seus testes: Matlab R2007a.
- b) Computador: Notebook Acer Aspire 5570-2758 com processador Intel Pentium dual-core, 1.60 GHz, HD 100 GB e com 1GB de memória DDR2.
- c) Sistema Operacional: Windows XP Professional, versão 5.1 com service pack 2.

3.1 Captura da Imagem

Essa é a primeira etapa necessária em um sistema de identificação de placas em veículos automotores. Cabe dizer que as imagens obtidas para esse tipo de sistema são capturadas normalmente em ambientes externos, praticamente excluindo a possibilidade de se conseguir controlar alguns fatores que podem prejudicar no reconhecimento, como luminância, chuva, reflexos, dentre outros. Um ambiente dificilmente será igual ao outro, pois foi visto que um sistema com a finalidade de efetuar a identificação de uma placa pode ser

usado em diversos locais e áreas. Assim, um sistema não pode ser baseado em apenas um ambiente. CONCI, AZEVEDO e LETA (2008) comentam que em ambientes controlados e com grandes contrastes, a tendência seria facilitar a interpretação de imagens, mas devido aos motivos explicados referente aos ambientes onde normalmente são capturadas as imagens, não se consegue isso.

Essa peculiaridade é um fator de suma importância, bem como o tamanho do carro, da placa e a posição em que foi obtida a imagem. Tudo isso deve ser levado em consideração e, por esse motivo, a captura da imagem é sempre realizada não só da placa e sim do veículo também, bem como, na maioria dos casos, do ambiente ao seu redor.

Na Figura 3.1 observa-se a captura de uma imagem sendo realizada em uma rodovia. As imagens podem ser obtidas de várias maneiras, como: câmera filmadora através de vídeo, por fotografias digitais ou ainda fotografias analógicas. A Figura 3.2 apresenta uma imagem monocromática sendo obtida em um ambiente externo.



Figura 3.1 – Imagem sendo Capturada.
Fonte: GUNGO, RODRIGUES, THOME (2002)

Se uma imagem não for obtida com boa qualidade, o sistema de reconhecimento poderá ter seu desempenho prejudicado. A captura da imagem tem como objetivo transformá-la em um padrão digital. A imagem digital monocromática pode ser considerada como uma matriz cujos índices de linhas e de colunas identificam um ponto na imagem, e o correspondente valor do elemento da matriz identifica o nível de cinza naquele ponto (GONZALEZ e WOODS, 2000, p. 4).



Figura 3.2 – Imagem Monocromática obtida em um Ambiente Externo.
Fonte: SEECAR (2007)

As imagens utilizadas nos testes do protótipo foram adquiridas de três fontes diferentes: fotos obtidas pelo próprio desenvolvedor, imagens do sistema SeeCar, capturadas nas ruas e um banco de imagens obtido na internet.

3.1.1 Captura em Estacionamentos

Foram capturadas 50 imagens digitais da parte frontal dos veículos automotores em estacionamentos públicos ou privados, como no Centro Universitário Feevale, conforme Figura 3.3, que continham a placa de identificação do automóvel. Todas as imagens foram obtidas com consentimento dos proprietários dos veículos que fizeram uma autorização por escrito, onde declararam que a imagem do seu veículo poderia ser utilizada sem restrição alguma para este trabalho de conclusão. A distância entre a placa do veículo e a máquina fotográfica variou de um à três metros.



Figura 3.3 – Imagem Original obtida no Estacionamento do Centro Universitário Feevale.

Para obtenção destas imagens digitais foi utilizada uma máquina digital Olympus Camedia – Digital Câmera D-435, com 5.1 megapixel. A resolução das imagens é de 640 x

480 pixels. As imagens são coloridas, porém passarão por uma etapa que irá transformá-las para tonalidades de cinza. Após isso, estarão aptas para as demais etapas do protótipo.

3.1.2 Captura nas Ruas

Existe uma versão de demonstração do software SeeCar que além de fornecer o programa, ainda disponibiliza um diretório com 31 imagens de veículos automotores para testes, obtidas em ambiente externo, mais especificamente, nas ruas. As imagens provenientes da versão 4.0.4.0 já estão em tonalidades de cinza, porém a resolução é de 768 x 288 pixel.

Para efetuar todos os testes no software SeeCar, é necessário mudar a resolução das imagens que foram adquiridas em estacionamentos. Todas deverão estar em tonalidades de cinza e com resolução de 768 x 288. Caso alguma não esteja nessa resolução, o software não consegue prosseguir na leitura das imagens, que ficam armazenadas nessa versão dentro de um diretório. Basta colocar qualquer imagem com essas características dentro do diretório que o programa tentará identificar a placa veicular. A Figura 3.4 é da versão de demonstração do SeeCar. A dificuldade no comparativo do protótipo com o programa SeeCar se dará na questão das etapas. Enquanto o protótipo será testado etapa a etapa, podendo ser verificados os resultados em cada uma delas, no SeeCar, só é fornecida a resposta de placa lida ou não, ou seja, se foi possível reconhecer ou não.



Figura 3.4 – Imagem Monocromática da Versão de Demonstração do Software SeeCar.

Fonte: SEECAR (2007)

3.1.3 Banco de Imagens

Para os testes via protótipo, foi adquirido um banco de imagens de placas de veículos automotores na Internet, disponível para pesquisas relacionadas a reconhecimento dos caracteres contidos na mesma, através do site <http://www.cbpf.br/cat/pdsi/lpr/lpr.html>. Neste, estão disponíveis 100 imagens, como pode ser observado na Figura 3.5.



Figura 3.5 – Imagem Monocromática obtida em um banco de imagens na Internet.

Fonte: <http://www.cbpf.br/cat/pdsi/lpr/lpr.html>

Das 100 imagens, foram selecionadas 50 para os testes, pois as demais não tinham condições de serem utilizadas. Os motivos pelas quais 50 imagens desse banco foram descartadas, deve-se ao fato de várias dessas não terem a placa, conforme Figura 3.6, outras que foram capturadas faltando um pedaço da placa, como pode ser observado na Figura 3.7 e as demais por estarem sem qualidade ou visibilidade, que é apresentado na Figura 3.8.



Figura 3.6 – Imagem Monocromática obtida em um banco sem a placa do veículo.

Fonte: <http://www.cbpf.br/cat/pdsi/lpr/lpr.html>



Figura 3.7 – Imagem Monocromática obtida em um banco faltando uma parte da placa.

Fonte: <http://www.cbpf.br/cat/pdsi/lpr/lpr.html>



Figura 3.8 – Imagem Monocromática obtida em um banco sem qualidade ou visibilidade.

Fonte: <http://www.cbpf.br/cat/pdsi/lpr/lpr.html>

3.2 Localização da Placa

Esse item aborda de que forma a placa foi localizada automaticamente e quais as técnicas utilizadas para alcançar os objetivos.

3.2.1 Metodologia Escolhida

O método pré-selecionado na trabalho de conclusão I foi o da Morfologia do Fundo da Placa, que leva em consideração que todas as placas possuem formato retangular, com quatro arestas e que uma vez detectadas, consegue-se recortar a imagem naquele ponto conforme as coordenadas verticais e horizontais encontradas.

O primeiro passo então foi aplicar uma detecção de bordas nas imagens dos veículos para analisar qual o melhor método para esse tipo de aplicação. A Figura 3.9 apresenta a imagem de um veículo onde foram aplicadas as técnicas de detecção de bordas baseadas em gradiente.

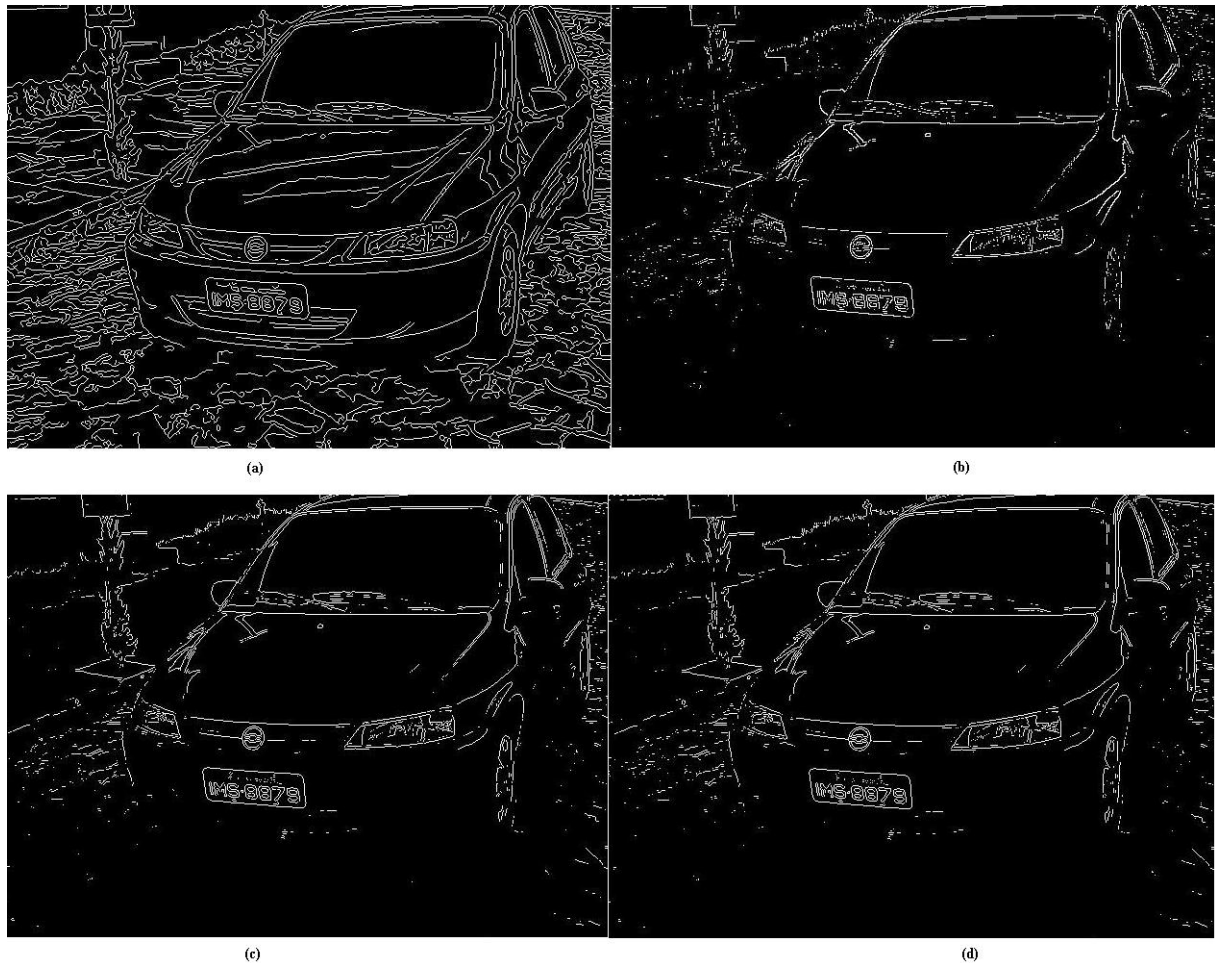


Figura 3.9 – Técnicas de Detecção de bordas. (a) Canny; (b) Roberts; (c) Sobel; (d) Prewitt.

Pelos testes apresentados, a melhor detecção de bordas foi quando aplicado Sobel (c) ou Prewitt (d), pois nestes casos conseguiu-se obter uma imagem mais limpa e com as bordas bem definidas. Quando aplicado Roberts (b), a imagem ficou limpa, porém muitas bordas acabaram sumindo em partes, ou seja, podendo comprometer a região da placa. Para finalizar, na imagem onde foi aplicada a detecção de bordas de Canny (a), percebe-se que as bordas estão muito bem definidas, mas o número de bordas é elevado, ocasionando muito processamento na hora de procurar as retas. Entre Prewitt (d) e Sobel (c), foi escolhido a técnica de Prewitt (d), pois nas imagens de veículos automotores as bordas ficaram mais definidas e contínuas do que a outra técnica.

A próxima etapa é encontrar todas as retas da placa para analisar e descobrir apenas duas retas paralelas verticalmente e duas horizontalmente. Neste ponto, ocorre uma mudança de planejamento. Em vez de percorrer a imagem procurando por todas as retas e analisar toda a imagem, buscou-se outras alternativas e os principais motivos para essa mudança são:

- a) Existem muitas retas detectadas na imagem.

b) Devido ao posicionamento da câmera na hora da captura da imagem, as retas não estão em uma única linha da imagem. Os conjuntos de pixels que formam a reta podem estar em linhas diferentes.

c) Devido ao elevado número de linhas presentes nas imagens, o processamento é lento e a possibilidade de ser selecionada uma região que não contenha a placa é grande.

Em decorrência desses fatos, buscou-se alternativas. Uma opção encontrada e aplicada com sucesso foi a de localizar todos os objetos na imagem, resultando em grandes mudanças no andamento dessa etapa.

3.2.2 Localização de Objetos

A técnica de localizar os objetos funciona como se fosse uma etiquetação que encontra todos os pixels conectados em uma mesma imagem, atribuindo um mesmo valor para cada conjunto unido. Conforme Jorge, Júnior e Mantovani (1998), a seleção de objetos atribui um índice a cada objeto encontrado, num processo de rotulação. “A rotulação é fundamental para a segmentação e contagem de regiões e objetos” (CONCI, AZEVEDO e LETA, 2008, p. 215). A Figura 3.10 apresenta quatro objetos rotulados em uma imagem exemplo. Com a rotulação de todos os objetos, pode-se trabalhar posteriormente com cada bloco encontrado, obtendo várias informações morfométricas como área, localização, perímetro, dentre outros. Utilizando o programa MatLab, o comando encontrado para fazer isso foi o *bwlabeln*.

Cada objeto é rotulado com um número seqüencial, aqui denominado O, que é composto por $\{O_1, O_2, O_3, \dots, O_n\}$

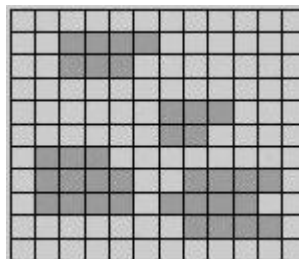


Figura 3.10 – Imagem com quatro objetos encontrados e etiquetados.

Os testes de rotulação nas imagens, obtidas dos veículos nos estacionamentos, mostrou que a quantia de objetos seria superior a 10.000 em cada imagem, o que levaria muito tempo de processamento. Porém, tem-se a possibilidade de trabalhar com rotulação nas imagens somente com objetos que possuam vizinhança-8, o que veio a reduzir para menos de 10% do total de objetos. Sendo assim, a técnica escolhida para etiquetar os objetos no protótipo leva em consideração a vizinhança-8, que foi explicada no terceiro capítulo.

Uma vez definida a técnica a ser implementada, partiu-se para os testes práticos. A Tabela 3.1 apresenta os resultados aplicados em 50 imagens de veículos automotores.

Tabela 3.1 – Tabela resultado da média de objetos encontrados em 50 imagens de placas de veículos automotores com os detectores de bordas já aplicados.

| Detector de Bordas | Canny | Roberts | Sobel | Prewitt |
|--------------------|-------|---------|-------|---------|
| Média de Objetos | 581 | 1237 | 793 | 814 |

Tendo em vista os resultados, observou-se que a técnica de detecção de bordas escolhida anteriormente, Prewitt, não é a mais indicada, devido ao grande número de objetos que se obtém em cada uma das 50 imagens analisadas, gerando assim, um processamento além do necessário.

De acordo com o que foi exposto, a melhor opção seria o uso de detecção de bordas de Canny. O fato é que, apesar de na detecção de imagens mostrar muitos detalhes com Canny, enquanto que nas outras não acontece o mesmo, os objetos ficam com seus contornos bem definidos, gerando um número menor, como pode ser comprovado pelos testes da Tabela 3.1. Observando novamente a Figura 3.9, pode-se perceber que existem menos detalhes nas outras técnicas de detecção de bordas, mas isso acaba gerando muitos objetos, pois suas bordas estão normalmente descontinuadas.

O próximo passo foi analisar as 50 imagens na busca de parâmetros que auxiliem na localização da região promissora da placa. A Figura 3.11 apresenta um objeto rotulado, neste caso, a placa de um veículo automotor. Tomando como estudo todos os objetos com a placa dos 50 veículos, observou-se que alguns parâmetros podem auxiliar para que um sistema processe automaticamente todas as etapas do sistema sem necessitar da intervenção humana.

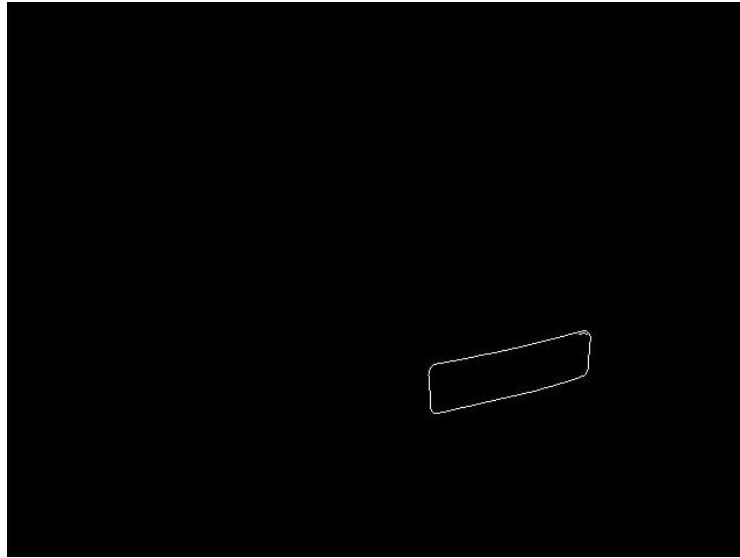


Figura 3.11 – Imagem de um veículo automotor com um objeto rotulado, a placa.

De acordo com Shah (2007), uma vez que a imagem esteja segmentada, pode-se buscar regiões contidas na mesma e obter características geométricas. “As características dimensionais são, por exemplo: área, perímetro, centro de gravidade, larguras máxima e mínima, comprimento, etc” (CONCI, AZEVEDO e LETA, 2008, p. 56). Algumas dessas características foram observadas e levadas em consideração para parametrizar as buscas pela região candidata a placa, como será descrito a seguir, sabendo que as imagens possuem tamanho de 640x480 pixels, ou seja, 640 colunas por 480 linhas:

a) Área: “a área pode ser expressa como o número de pixels que compreende o objeto ou a região de interesse” (PEDRINI e SCHWARTZ, 2008, p. 267). A média da área para as 50 imagens ficou com um valor de 375, com limiar mínimo de 300 e máximo em 450. Formalmente, a área pode ser definida como: $A_i \in \mathfrak{R} \leq 450 \wedge A_i \in \mathfrak{R} \geq 300$.

b) Altura: a média da altura do objeto da placa, que é calculada entre o pixel mais alto da imagem e o mais baixo foi de 55 linhas. O limiar de segurança, após a análise das imagens ficou definido entre 42 e 125 linhas. Formalmente, a altura pode ser definida como: $H_i \in \mathfrak{R} \leq 125 \text{ linhas} \wedge H_i \in \mathfrak{R} \geq 42 \text{ linhas}$.

c) Largura: a média da largura do objeto da placa, que é calculada entre o pixel mais a esquerda da imagem e o mais da direita foi de 150 colunas. O limiar de segurança para as imagens ficou definido entre 119 e 182 colunas. A largura pode ser definida como: $L_i \in \mathfrak{R} \leq 182 \text{ colunas} \wedge L_i \in \mathfrak{R} \geq 119 \text{ colunas}$.

d) Perímetro: “o perímetro representa o comprimento da borda de um objeto” (PEDRINI e SCHWARTZ, 2008, p. 261). Segundo os autores, essa característica pode ser obtida pela contagem dos pixels ao longo da borda do objeto da imagem. A média do perímetro para as 50 imagens ficou com um valor de 370, com limiar de segurança após análise das imagens, mínimo de 328 e máximo em 471. Formalmente, o perímetro pode ser definido como: $P_i \in \mathfrak{R} \leq 471 \wedge P_i \in \mathfrak{R} \geq 328$.

Após a rotulação dos objetos na imagem, realizou-se uma análise para localização da região promissora da placa, através das características parametrizadas.

A primeira a ser utilizada é a busca por objetos que contenham área com o limiar definido e apresentado anteriormente, que pode ser representado da seguinte forma:

$OB = O \in A_i$, onde:

OB = nova relação de objetos.

O = relação inicial de todos os objetos da imagem.

A_i = área utilizada como parâmetro na imagem inicial.

Realizada essa etapa, o número de objetos restantes com possibilidades reais de ter a placa fica restrito a cinco objetos no máximo. Esses objetos são novamente rotulados.

Após a segunda rotulação, busca-se nos objetos restantes aqueles que estão na faixa de altura e largura dentro dos parâmetros pré-definidos, a partir da seguinte equação:

$OBJ = OB \in H_i \wedge OB \in L_i$, onde:

OBJ = nova relação de objetos.

OB = relação de objetos com área com valor entre o limiar pré-estabelecido.

H_i = altura utilizada como parâmetro nos objetos restantes.

L_i = largura utilizada como parâmetro nos objetos restantes.

O Anexo A apresenta o código fonte desenvolvido no programa Matlab para as comparações de Altura e Largura. Nessa etapa, praticamente em todas as imagens, já restou apenas um objeto, ou seja, a região da placa.

Para finalizar, se ainda existir mais de um objeto, procura-se algum que tenha um perímetro que fique dentro dos valores estabelecidos na média das imagens estudadas, podendo ser formalizado a equação por: $OBJE = OBJ \in P_i$, onde:

OBJE = objeto resultante da última etapa de filtro da localização da placa, se houver necessidade, conforme mencionado anteriormente.

OBJ = relação dos objetos da imagem que possuam área, altura e largura dentro dos parâmetros pré-definidos.

P_i = perímetro utilizado como parâmetro na imagem inicial.

Trabalhando com um único objeto, obtem-se as coordenadas do mesmo e realiza-se o recorte na imagem original, salvando assim, uma nova imagem para outras etapas do processo, somente com a região da placa, como pode ser visto na Figura 3.12.

O comando a ser utilizado para o recorte no programa MatLab é o *imcrop* e para que seja possível isso, é necessário informar em que posição, ou pixel, do lado esquerdo, inicia a imagem, além da altura inicial, a sua largura e altura total. Contendo todas essas informações, basta fazer o recorte.

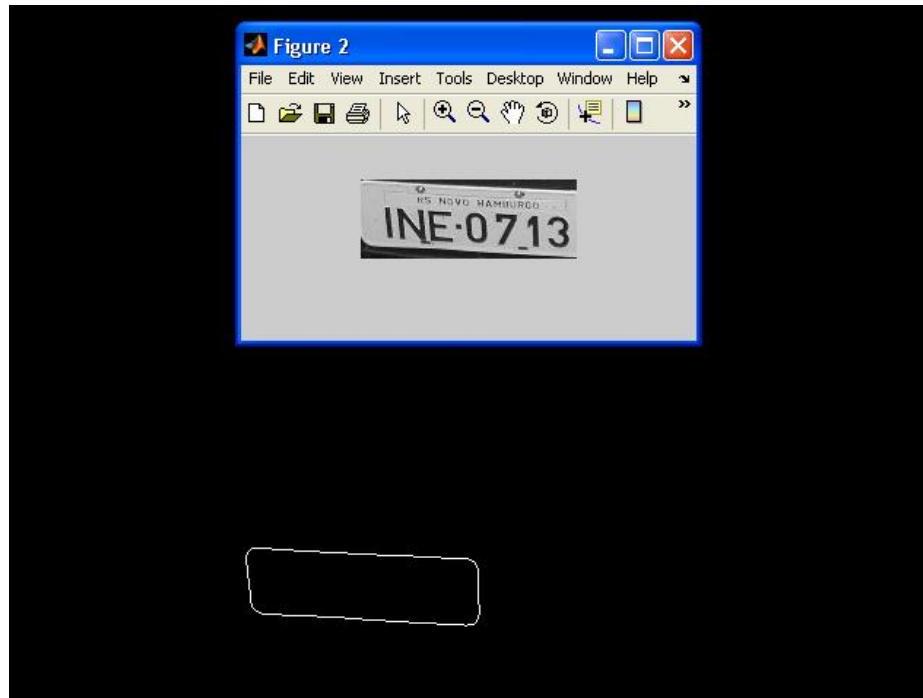


Figura 3.12 – Objeto com região da placa encontrada e a placa sendo recortada na imagem original do veículo.

3.2.3 Testes de Localização no Protótipo e versão de demonstração do SeeCar

A Tabela 3.2 apresenta os resultados da localização da placa no protótipo proposto e que serão em seguida comentados.

Tabela 3.2 – Tabela resultado da localização da placa no protótipo.

| Origem Imagens | Imagens | % Localizada só a placa | % Localizada placa com outros objetos juntos | Não localizada placa |
|-----------------|---------|-------------------------|--|----------------------|
| Estacionamentos | 50 | 98,00% | 0,00% | 2,00% |
| Internet | 50 | 70,00% | 14,00% | 16,00% |
| SeeCar | 31 | 54,84% | 19,35% | 25,81% |

O tempo de localização e recorte da placa ficou um pouco acima de um segundo para cada imagem.

Com relação as imagens obtidas nos estacionamentos, em uma única imagem, dentre as 50 testados, não foi possível localizar a placa no protótipo proposto. Percebe-se que a borda superior da região da placa não foi localizada perfeitamente, mesmo utilizando a técnica de detecção de bordas de Canny, o que ocasionou o aparecimento de outros objetos na imagem com tamanhos fora dos estabelecidos como referência para localização. Analisando a imagem original, descobriu-se que o veículo contém um friso da mesma cor da placa. Este passa exatamente na linha superior da placa, impedindo o detector de bordas de separar o mesmo.

Importante salientar que mesmo a olho nu é difícil detectar onde termina o friso e inicia a placa. A Figura 3.13 demonstra a detecção de bordas da imagem onde não foi possível extrair a placa corretamente através do método anteriormente apresentado.



Figura 3.13 – Imagem de veículo que não foi possível encontrar a placa automaticamente.

Quanto as imagens obtidas na internet, conseguiu-se encontrar e selecionar 70,00 % das mesmas, porém as características parametrizadas tiveram nova definição, conforme segue:

$$A_i \in \mathcal{R} \leq 645 \wedge A_i \in \mathcal{R} \geq 255$$

$$H_i \in \mathcal{R} \leq 48 \text{ linhas} \wedge H_i \in \mathcal{R} \geq 36 \text{ linhas}$$

$$L_i \in \mathcal{R} \leq 299 \text{ colunas} \wedge L_i \in \mathcal{R} \geq 218 \text{ colunas}$$

$$P_i \in \mathcal{R} \leq 1335 \wedge P_i \in \mathcal{R} \geq 505$$

Em 14,00 % das imagens, além da placa, outros objetos ficaram dentro dessas características, o que pode levar o protótipo a errar no objeto a ser recortado. Em 16,00 % não foi possível localizar a placa. Esses 30,00 % deve-se basicamente a alguns fatores como um limiar de área e perímetro muito grande, em decorrência das imagens que possuem uma distância muito diferente entre determinadas imagens e outras devido a qualidade, onde não foi possível detectar corretamente as bordas da placa. A Figura 3.14 (a) apresenta uma imagem de um veículo do banco de imagens da internet e a mesma sendo realizado a detecção de bordas (b). Essa imagem foi uma das que não foi possível localizar a placa devido as

bordas da mesma não estarem definidas, principalmente na parte superior da placa à esquerda como pode ser observado, o que ocasionou um objeto de tamanho inferior as características parametrizadas.

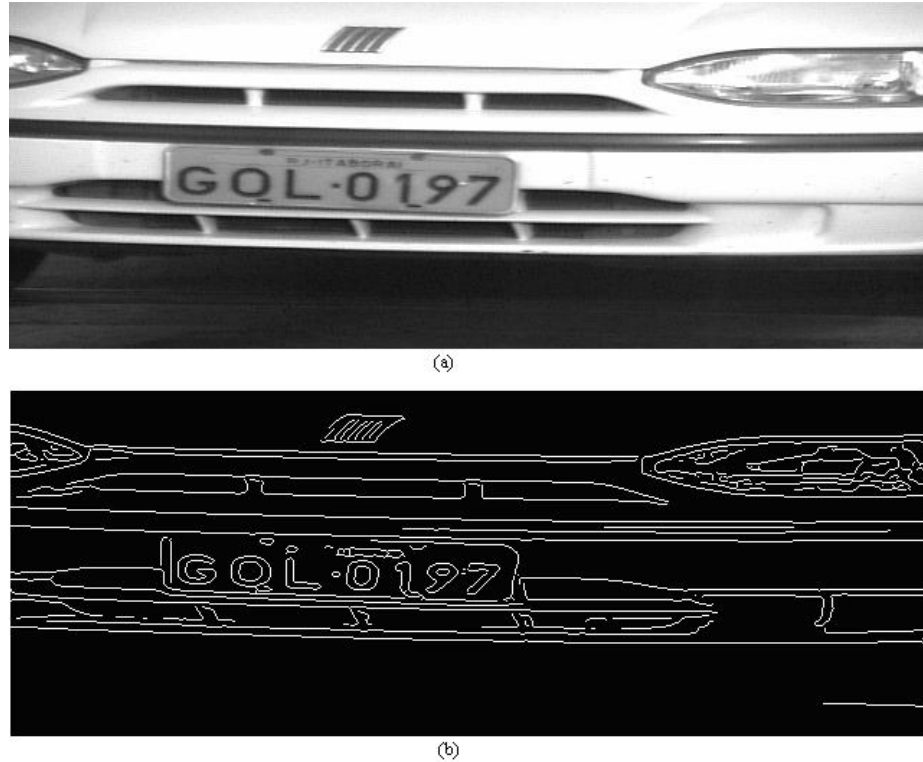


Figura 3.14 – Imagem de veículo que não foi possível encontrar a placa automaticamente devido as bordas da mesma não estarem bem definidas.

Com relação as imagens do programa de demonstração SeeCar, em 54,84 % foi possível localizar e recortar somente a placa. Em 19,35 % localizou-se a placa e outros objetos juntos e em 25,81 % das imagens não foi possível localizar a placa. Para esses testes, a formulação das características parametrizadas teve o seguinte critério:

$$A_i \in \mathfrak{R} \leq 589 \wedge A_i \in \mathfrak{R} \geq 306$$

$$H_i \in \mathfrak{R} \leq 43 \text{ linhas} \wedge H_i \in \mathfrak{R} \geq 26 \text{ linhas}$$

$$L_i \in \mathfrak{R} \leq 260 \text{ colunas} \wedge L_i \in \mathfrak{R} \geq 148 \text{ colunas}$$

$$P_i \in \mathfrak{R} \leq 966 \wedge P_i \in \mathfrak{R} \geq 337$$

Uma vez realizados os testes no protótipo, tentou-se realizar a localização no programa de demonstração SeeCar. As mesmas imagens, 50 dos estacionamentos e 50 da

internet, foram transformadas para a resolução de 768 x 288 pixel, que é a resolução exigida pelo sistema, e colocadas no diretório do programa SeeCar, porém o mesmo não conseguiu localizar a placa.

Foi procurada alguma alternativa para o problema e o que verificou-se é que o programa SeeCar possui um arquivo de configuração no diretório de instalação, onde são especificados vários parâmetros, como pode ser observado no Anexo B deste trabalho. Foram alterados alguns valores, como *min_hor_plate_length*, *max_hor_plate_length*, *min_hor_plate_height* e *max_hor_plate_height*. Porém, em nenhum caso, conseguiu-se sucesso e, parte disso, também deve ser por conta das demais variáveis que existe neste arquivo e que só o fabricante pode explicar.

Para finalizar, foi verificada a possibilidade para validar a versão demo do programa SeeCar, pois já não era possível saber se ele realmente era funcional ou não. Foram capturadas mais de 20 imagens de um dos veículos automotores de diferentes distâncias para verificar se o software conseguiria localizar a placa. E, em uma das imagens, cerca de quatro a cinco metros de distância da máquina digital, o programa SeeCar localizou a placa e inclusive reconheceu seus caracteres em menos de um segundo, como pode ser observado na Figura 3.15.

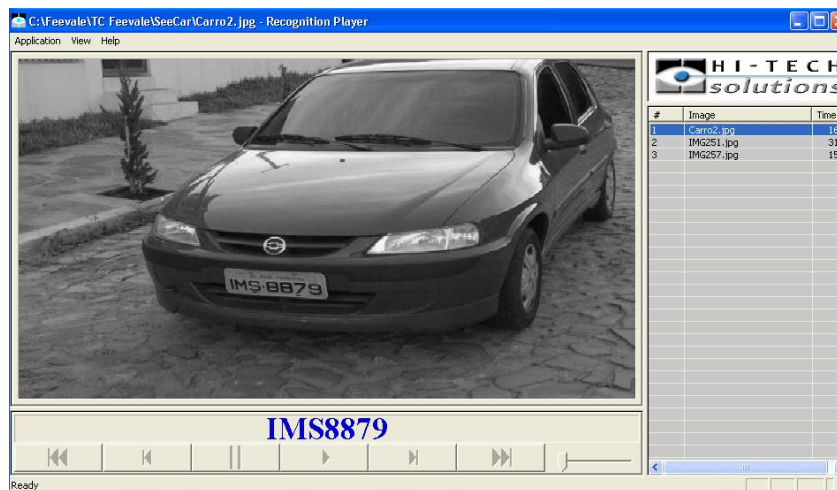


Figura 3.15 – Imagem da versão demo do programa SeeCar, que localiza a placa de um veículo a uma distância maior e ainda reconhece seus caracteres.

Fonte: SEECAR (2007)

Percebe-se assim, que o programa SeeCar também utiliza recursos morfométricos, assim como variáveis parametrizadas para localização da placa, pois caso contrário, localizaria a placa do mesmo veículo a uma distância menor, assim como dos demais veículos.

Dessa maneira, os testes práticos das demais etapas serão realizados apenas no protótipo proposto, sendo utilizadas as 101 imagens que foram localizadas a região da placa, sendo 49 provenientes dos estacionamentos, 35 da internet e 17 do SeeCar.

3.3 Rotação da Placa

Conforme já comentado neste trabalho, as câmeras que capturam a imagem do veículo sempre estarão rotacionadas com um ângulo em relação ao automóvel, ocasionando, dessa maneira, uma imagem onde a placa não estará alinhada, como pode ser observado na Figura 3.16. Nessa etapa, deve-se buscar, na imagem recortada da etapa anterior, a da localização, e corrigir o alinhamento da placa para que a mesma possa ser recortada, eliminando tudo o que estiver além das bordas da mesma. De acordo com Azevedo e Conci (2003), rotacionar a imagem significa girar a mesma.



Figura 3.16 – Região da placa recortada na etapa de Localização.

Sabe-se que as câmeras estarão sempre posicionadas no menor ângulo possível para que a imagem não tenha uma distorção elevada e que isso não excederá a 90 graus. Dessa maneira, é importante buscar algumas informações para calcular o ângulo da correção. A primeira é a altura da placa, que pode ser obtida pela diferença entre a linha do pixel da borda da placa mais alto da lateral direita, com a linha do pixel mais baixo da mesma lateral. Também poderia ser feito esse cálculo pela lateral esquerda. Essa diferença fornecerá o número de linhas total da placa, ou seja, a altura, representada por $AP = LP_{MAD} - LP_{MBD}$, onde:

AP = altura da placa

LP_{MAD} = linha do pixel da placa mais alto, localizado na direita da imagem

LP_{MBD} = linha do pixel da placa mais baixo, localizado na direita da imagem

A divisão de AP por 90 graus (ângulo máximo) fornecerá o valor em graus para cada linha que contém a placa, conforme a equação: $GL = AP / 90$, onde:

GL = graus para cada linha da placa

AP = altura da placa

90 = grau máximo para rotação

A outra informação deve ser a diferença entre a linha do pixel da borda direita superior da placa mais alto com relação a linha do pixel mais alto da borda superior esquerda, resultando assim na quantidade de linhas que a placa está rotacionada, representado por:

$LC = LP_{BDMA} - LP_{BEMA}$, onde:

LC = total de linhas que a placa está rotacionada

LP_{BDMA} = linha do pixel da borda direita da placa mais alta

LP_{BEMA} = linha do pixel da borda esquerda da placa mais alta

Para se obter o grau correto, basta utilizar o total de linhas que a placa está rotacionada, multiplicando pelos graus de cada linha. A equação fica assim: $AG = LC * GL$, onde:

AG = ângulo para rotação

LC = total de linhas que a placa está rotacionada

GL = graus para cada linha da placa

Para fazer a rotação, foi utilizado o comando *imrotate* no programa MatLab. A Figura 3.17 apresenta a imagem com a sua rotação, de acordo com essa fórmula.



Figura 3.17 – Placa rotacionada.

Observando a imagem, a placa foi rotacionada, porém ainda contém partes do carro. Para a próxima etapa do projeto, que é necessário separar os caracteres, será utilizada a mesma técnica de localização da placa, ou seja, procurar por objetos na imagem. Utilizando a imagem, da Figura 3.17, no momento que os seus objetos forem rotulados, resultará em 28 objetos. Dentre esses 28 objetos, deverão ser encontrados apenas os sete que compõem os

caracteres da placa. O número de objetos depende de cada placa, mas sempre será maior se a imagem contiver outras partes do veículo.

Outra alternativa para finalizar essa etapa, dando prosseguimento ao próximo passo de separação dos caracteres, é o de localizar novamente as bordas da placa através da área parametrizada, visto que o maior objeto na imagem agora é a placa, e então recortá-la novamente, como é apresentado na Figura 3.18.



Figura 3.18 – Placa rotacionada recortada através das suas bordas.

A vantagem de trabalhar com essa imagem está no fato de que ela passaria a conter apenas 18 objetos, ou seja, seriam menos objetos para análise na próxima etapa. A desvantagem está no fato de ter novamente um processamento para localizar as bordas da placa e fazer um novo recorte.

Todas as imagens de placas localizadas dos estacionamentos, internet e do software SeeCar foram rotacionadas com sucesso e o tempo para isso foi inferior a um segundo.

3.4 Separação dos Caracteres

Visando a separação dos sete caracteres contidos na placa, sendo os três primeiros as letras e os quatro últimos referente a números, é necessário desmembrar a imagem obtida na etapa anterior em sete novas. Deve ser comentado que cada uma dessas novas imagens deve conter somente o caractere localizado, eliminando as demais partes da placa que não tem nenhuma informação útil para esse tipo de sistema.

3.4.1 Técnicas Utilizadas

Para alcançar tal objetivo, serão utilizadas as mesmas técnicas empregadas na etapa de localização da placa, pois cada caractere é um objeto, em que deve-se buscar, através das suas bordas, as coordenadas exatas para efetuar o recorte do objeto, salvando como uma nova imagem.

A primeira função é aplicar um detector de bordas na imagem rotacionada. O detector de bordas utilizado é Canny, como já foi visto anteriormente. Após, utiliza-se a

rotulação de todos os objetos da imagem. O ideal, nessa rotulação, seria aparecer somente sete objetos, que nada mais seriam do que os caracteres, mas em todas as placas rotacionadas isso não ocorre. O número de objetos é superior devido a outras bordas que não estão bem formadas, gerando outros objetos, como letras do nome da cidade, furação da placa, etc. A Figura 3.19 é a imagem da Figura 3.18 com detector de bordas Canny aplicado, sendo rotulados, resultando em 18 objetos.



Figura 3.19 – Imagem com detector de bordas Canny aplicado e seus 18 objetos rotulados.

Uma das dificuldades nessa etapa está na quantia de objetos, pois, por exemplo, a letra “O” é equivalente a dois objetos. O primeiro objeto é da borda interna da letra e o segundo da borda externa. Como as bordas não se encostam, formam dois objetos. O número “8” e a letra “B”, geram, cada um, três objetos.

Como o número de objetos é superior aos sete que seriam correspondentes aos caracteres da placa, precisa-se restringir os que não são caracteres. Para tanto, utiliza-se as mesmas características apresentadas na etapa de localização, ou seja, área, altura, largura e perímetro de cada objeto. Cada uma dessas características tem os seus valores de limiar definidos.

Obtendo somente os sete objetos da placa localizados, basta encontrar as coordenadas de cada um deles e efetuar o recorte da imagem inicial. O algoritmo de rotulação varre a imagem da esquerda para a direita. Dessa maneira, o sistema não precisa se preocupar com a ordenação de cada objeto, ou seja, o primeiro caractere salvo será o mais da esquerda, seguindo neste sentido até o último que é o mais da direita. Os objetos são salvos na própria ordem de apresentação de uma placa. A Figura 3.20 apresenta os sete caracteres salvos em novas imagens.



Figura 3.20 – Sete novas imagens, provenientes da placa IOB 8158.

3.4.2 Testes de Separação de Caracteres no Protótipo

Para imagens provenientes de estacionamentos, os limiares definidos são:

$$A_i \in \mathfrak{R} \leq 135 \wedge A_i \in \mathfrak{R} \geq 30$$

$$H_i \in \mathfrak{R} \leq 30 \text{ linhas} \wedge H_i \in \mathfrak{R} \geq 16 \text{ linhas}$$

$$L_i \in \mathfrak{R} \leq 30 \text{ colunas} \wedge L_i \in \mathfrak{R} \geq 5 \text{ colunas}$$

$$P_i \in \mathfrak{R} \leq 275 \wedge P_i \in \mathfrak{R} \geq 35$$

A seguir é apresentada a Tabela 3.3 com os resultados da separação dos caracteres das imagens provenientes de estacionamentos.

Tabela 3.3 – Tabela resultado da separação dos caracteres da placa das imagens dos estacionamentos.

| Definição | Total | Percentual |
|---------------------------------------|------------|----------------|
| Total de Placas Localizadas | 49 | 100,00% |
| Placas com os 7 caracteres separados | 39 | 79,59% |
| Placas com outros objetos ou faltando | 10 | 20,41% |
| Total de Caracteres das Placas | 343 | 100,00% |
| Total de Caracteres Separados | 314 | 91,55% |
| Total de Caracteres Não Separados | 29 | 8,45% |

Como pode ser observado, dentre as 49 placas localizadas, em 39 delas foi possível separar todos os sete caracteres da placa automaticamente, enquanto que em dez, isso não ocorreu. Nessas dez, algumas tiveram mais objetos além dos sete caracteres selecionados ou faltaram objetos, que foi a principal ocorrência. A Figura 3.21 apresenta um exemplo deste fato. A placa tem duas letras “JJ”, mas o programa identificou como apenas um objeto e que fugiu das características parametrizadas, pois sua área e perímetro ficaram com valores elevados em função das bordas que estão juntas. Dessa maneira, o protótipo localizou apenas cinco objetos que eram os caracteres.



Figura 3.21 – Imagem onde a rotulação identificou como apenas um objeto, porém, na verdade, deveriam ser dois.

Para as 35 imagens obtidas via internet e que foram localizadas as placas, os limiares definidos para cada característica foram:

$$A_i \in \mathfrak{R} \leq 136 \wedge A_i \in \mathfrak{R} \geq 39$$

$$H_i \in \mathfrak{R} \leq 29 \text{ linhas} \wedge H_i \in \mathfrak{R} \geq 17 \text{ linhas}$$

$$L_i \in \mathfrak{R} \leq 37 \text{ colunas} \wedge L_i \in \mathfrak{R} \geq 5 \text{ colunas}$$

$$P_i \in \mathfrak{R} \leq 300 \wedge P_i \in \mathfrak{R} \geq 44$$

A seguir é apresentada a Tabela 3.4 com os resultados da separação dos caracteres das imagens provenientes do banco de imagens da internet.

Tabela 3.4 – Tabela resultado da separação dos caracteres do banco de imagens da Internet.

| Definição | Total | Percentual |
|---------------------------------------|------------|----------------|
| Total de Placas Localizadas | 35 | 100,00% |
| Placas com os 7 caracteres separados | 23 | 65,71% |
| Placas com outros objetos ou faltando | 12 | 34,29% |
| Total de Caracteres das Placas | 245 | 100,00% |
| Total de Caracteres Separados | 208 | 84,90% |
| Total de Caracteres Não Separados | 37 | 15,10% |

Com relação as 17 imagens de placas localizadas da versão de demonstração do programa SeeCar, os limiares definidos foram:

$$A_i \in \mathfrak{R} \leq 110 \wedge A_i \in \mathfrak{R} \geq 28$$

$$H_i \in \mathfrak{R} \leq 25 \text{ linhas} \wedge H_i \in \mathfrak{R} \geq 12 \text{ linhas}$$

$$L_i \in \mathfrak{R} \leq 30 \text{ colunas} \wedge L_i \in \mathfrak{R} \geq 5 \text{ colunas}$$

$$P_i \in \mathfrak{R} \leq 230 \wedge P_i \in \mathfrak{R} \geq 30$$

A seguir é apresentada a Tabela 3.5 com os resultados da separação dos caracteres das imagens provenientes da versão de demonstração do programa SeeCar.

Tabela 3.5 – Tabela resultado da separação dos caracteres das imagens da versão de demonstração do programa SeeCar.

| Definição | Total | Percentual |
|---------------------------------------|------------|----------------|
| Total de Placas Localizadas | 17 | 100,00% |
| Placas com os 7 caracteres separados | 9 | 52,94% |
| Placas com outros objetos ou faltando | 8 | 47,06% |
| Total de Caracteres das Placas | 119 | 100,00% |
| Total de Caracteres Separados | 104 | 87,39% |
| Total de Caracteres Não Separados | 15 | 12,61% |

O tempo de separação dos caracteres foi de aproximadamente dois segundos para cada placa de veículo automotor.

3.5 Reconhecimento dos Caracteres

Esse módulo tem por objetivo criar vetores de características para cada tipo de letra do alfabeto português e de números de 0 à 9. Uma vez criados os vetores, qualquer imagem segmentada correspondente a um caractere pode ser comparada com os vetores existentes em busca do seu respectivo código ASCII.

3.5.1 Criando vetor de Características para Letras e Números

Conforme explicado, os três primeiros caracteres na placa do veículo são letras e os quatro últimos números e que com o algoritmo desenvolvido, os objetos correspondentes aos caracteres são separados já na ordem da placa, da esquerda para a direita. Dessa maneira, não existe a necessidade de montar um único vetor de características para todos os caracteres e números, até porque pode ocasionar erros, devido ao número de pixels do caractere, por exemplo, serem muito parecidos. Conseqüentemente, será criado vetores de características só para letras e outros só para números, gerando uma assinatura para cada caractere.

A primeira função a ser realizada para criar esses vetores é transformar cada caractere que servirá de amostra para o vetor em uma imagem binária e redimensionar a imagem para 16 x 16 pixels, como pode ser observado na Figura 3.22 e que pode ser chamado de mapa de bits.

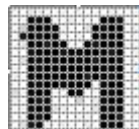


Figura 3.22 – Mapa de bits - Imagem binária da letra M e redimensionada para 16 x 16 pixels.

O mapa de bits é, na verdade, uma matriz onde cada célula representa a luminosidade de um pixel. No exemplo acima, foi adotada uma matriz 16 x 16 pixels, construída a partir de uma imagem binária, onde os bits do fundo são brancos e os do caractere são pretos. A função do mapa de bits é contar para cada caractere que servirá de amostra o número de bits pretos e guardar essa informação no Vetor de Letras. Para o de números é o mesmo procedimento, porém somente a informação será armazenada dentro do Vetor de Números.

O vetor de letras começa com a letra A e termina com a letra Z. Esse vetor é composto por três amostras de cada letra. Já o vetor de números é composto por três amostras

de cada número, iniciando pelo 0 e finalizando no 9. Os caracteres que servem como base para construção dos vetores são os das imagens das placas deste trabalho.

O total de amostras e, conseqüentemente, o tamanho do vetor de letras é de 78 e para o de números é 30.

O segundo vetor de características tanto para letras como para números baseia-se na quantidade de objetos encontrados por amostra. Utiliza a mesma função que foi empregada na etapa de localização e na de separação de caracteres. Primeiramente faz a leitura da imagem, detectando suas bordas através de Canny e depois rotula todos os objetos encontrados nela. Tem-se, assim, o número total de objetos da imagem. A Figura 3.23 faz a rotulação da amostra do número 8. Percebe-se que foram encontrados três objetos.

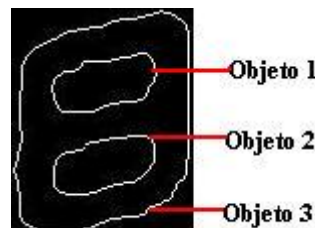


Figura 3.23 – Amostra do número 8 rotulada e contado seus objetos

Tomando como base essa informação, no momento do reconhecimento de um número 8, o software deve verificar se existem três objetos no caractere candidato. O sistema automaticamente saberá que se trata do número 8, pois não existe nenhum outro número que possua como característica, três objetos em uma imagem. Caso não seja três, ele terá que percorrer o vetor do mapa de pixels e o vetor que será explicado a seguir.

O último vetor a ser criado para levantamento das características é o da projeção vertical, que faz uso de uma matriz de bits. Essa projeção faz o somatório dos pixels em cada coluna da matriz e armazena essa informação. Cada amostra terá nesse vetor 16 somatórios que estarão guardados no vetor. A Figura 3.24 apresenta uma projeção vertical onde na coluna três tem-se o somatório de 16 pixels e na coluna quatro tem-se quatro pixels.

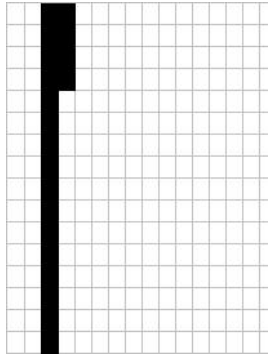


Figura 3.24 – Projeção vertical de uma imagem

Como se tem três vetores para letras, é necessário passar um índice para cada posição deles. Dessa maneira, a letra A, no vetor mapa de bits corresponde da posição um à três, assim como no vetor de objetos rotulados, enquanto na no vetor de projeção vertical vai da posição um a 48, pois são dezesseis linhas para cada amostra. A letra B segue a seqüência em todas as posições dos vetores, até a última letra. Quanto aos números, segue o mesmo funcionamento que as letras.

3.5.2 Efetuando o Reconhecimento dos Caracteres

O reconhecimento da cada caractere deve ser realizado de acordo com as suas características. O procedimento é o mesmo adotado para criar os vetores, ou seja, conta-se quantos objetos tem na imagem, que chama-se imagem candidata, soma-se a quantidade de bits total da imagem e a quantidade por cada linha, armazenando todas essas informações. Como a posição da imagem candidata na placa é conhecida, já será realizada a pesquisa dessas informações diretamente nos vetores de características corretos.

Sabendo que a imagem candidata tem um determinado número de objetos, essa quantia é comparada com todas as posições do vetor de características de objetos. Se for encontrado apenas um código ASCII correspondente com a quantia de objetos da imagem candidata, esse código já é informado e os demais vetores nem são verificados.

Caso tenha mais do que uma posição no vetor, correspondente a mais do que um caractere ou os que mais se aproximarem, deve-se percorrer o próximo vetor de mapa de bits para encontrar em qual deles se enquadra o total de bits da imagem candidata. Se tiver apenas um resultado, então basta informar que caractere é o correspondente. Caso exista mais do que um possível caractere ou que não haja, então serão selecionados os que mais se aproximarem

novamente e deve ser procurado no vetor da projeção vertical. Neste caso, serão comparadas o somatório de linhas e a que tiver o maior número de linhas iguais será o escolhido.

Para os casos onde não se encontrarem os correspondentes, devem ser pegos os que mais se aproximarem e ir percorrendo todos os vetores até o último e, no final, deve ser informado seu código ASCII.

3.5.3 Testes de Reconhecimento dos Caracteres

A Tabela 3.6, apresenta os resultados obtidos nas 626 imagens candidatas testadas, sendo 249 de letras e 377 de números.

Tabela 3.6 – Tabela resultado do reconhecimento de caracteres.

| | | |
|--------------------------------|------------|----------------|
| Total de Letras | 249 | 100,00% |
| Letras Reconhecidas - Correto | 201 | 80,72% |
| Letras Reconhecidas - Errado | 48 | 19,28% |
| Total de Números | 377 | 100,00% |
| Números Reconhecidos - Correto | 332 | 88,06% |
| Números Reconhecidos - Errado | 45 | 11,94% |

Os erros normalmente foram causados por bordas mal definidas, assim como por tamanhos de letras muito diferentes uns dos outros. O tempo de reconhecimento ficou aproximadamente em dois segundos.

Para finalizar, uma vez encontrados os caracteres, basta informar ao usuário ou sistema a placa. A Figura 3.25 apresenta uma imagem de um veículo automotor capturada, sendo lida no protótipo e o mesmo informando sua placa.



Figura 3.25 – Imagem salva do protótipo realizando o reconhecimento da placa

CONSIDERAÇÕES FINAIS

O principal objetivo deste trabalho foi estudar e aplicar aspectos relacionados a área de processamento de imagens digitais, através de técnicas de segmentação, detecção de bordas, rotulação, rotação e reconhecimento de padrões. O intuito foi definir as etapas necessárias para o desenvolvimento de um protótipo de software para reconhecimento automático de placas de veículos automotores. Depois de estabelecidas as etapas, foi criado um protótipo, que pode ser testado e que apresentou seus resultados para análise. Para tanto, buscou-se embasamento teórico desde o Trabalho de Conclusão I sobre os temas em questão através, da literatura e trabalhos já realizados, além de ter sido realizado muitos testes para comprovação das técnicas empregadas.

No primeiro capítulo, foi apresentada uma introdução sobre os problemas do trânsito no Brasil, soluções e motivação do estudo.

No capítulo dois constou a regulamentação no Brasil sobre as placas de veículos automotores, segundo o CONTRAN. Explica-se onde podem ser utilizados os programas de reconhecimento e para quais situações são destinados. Também foram citados programas acadêmicos e comerciais que existem, sendo que alguns possuem inclusive apoio público ou privado, muitas vezes fundamental para se continuar projetos de pesquisa visando inclusive um produto voltado a tecnologia. Importante ressaltar que em um dos trabalhos acadêmicos, foi apresentado até o índice de acertos no reconhecimento, sendo um dos principais motivos para desenvolvimento desse projeto, visto que os programas existentes não conseguem reconhecimento em 100,00% dos casos. Sendo assim, um programa com índices elevados de acertos pode conseguir atingir um mercado grandioso no Brasil e que não está saturado, dado a um pequeno número de softwares comerciais existentes e pelo potencial número de aplicações que podem ser criados com o domínio da tecnologia necessária para o reconhecimento de placas.

No terceiro capítulo foi apresentado como funcionam normalmente os programas que efetuam o reconhecimento de placas e as etapas necessárias. Fica claro que as etapas que são colocadas como necessárias não são obrigatórias. Elas seguem uma linha de raciocínio e alguns autores/desenvolvedores não utilizam todas, acabam colocando outros nomes ou, até mesmo, unindo fases.

Prosseguindo o trabalho, no próximo capítulo foi apresentado o protótipo do software criado, com todas as etapas necessárias desenvolvidas e seus respectivos resultados, assim como as técnicas escolhidas para cada fase.

A primeira etapa que é a da captura da imagem é fundamental para o sucesso da aplicação. No decorrer do trabalho, pode ser observado que tudo o que possui uma padronização, apresenta um resultado melhor. Primeiro, precisa-se de uma imagem com ótima qualidade, que ajudou para a detecção de bordas, tanto da placa, quanto dos caracteres. Outras questões não menos importantes, estão relacionadas a distância e as condições climáticas. Se a distância entre o equipamento que obtém a foto e a placa estiver padronizada, ou seja, praticamente estável, facilita para as definições de limiares das características da placa, que ajuda na localização da mesma e, conseqüentemente, na separação dos caracteres. Quanto as condições climáticas, elas interferem diretamente no resultado da localização ou reconhecimento, pois as imagens podem aparecer com pouca ou muita luminância, com sombras, dentre outras, que acarretam no surgimento de bordas ou o não aparecimento das mesmas. Como as imagens foram capturadas em ambientes sem controle dessa natureza, o software fica sujeito a não encontrar a placa ou alguns caracteres.

A próxima fase e que é considerada a de maior relevância científica deste trabalho, contribuindo para a área, foi a de localização da placa. Isso é devido ao algoritmo desenvolvido e aos resultados alcançados. A principal questão ficou por conta da etapa que muitos autores comentam sobre o pré-processamento que não foi preciso fazer. Outras técnicas normalmente empregadas para localização foram substituídas por uma que leva em consideração características morfométricas. A técnica utilizada partiu do princípio que todas as placas possuem um padrão de área, perímetro, altura e largura. Sendo assim, buscou-se detectar as bordas da imagem capturada, rotulando todos os seus objetos e efetuando testes para descobrir os seus extremos. Uma vez definidos os limiares, a busca pela placa foi realizada sem preocupação com outras características utilizadas por outros autores como cor de placa ou do carro, além de retas, variação tonal ou em busca por caracteres. Qualquer

metodologia escolhida tem seus respectivos problemas, mas a que foi implementada depende basicamente da detecção de bordas, onde necessita-se de uma imagem de ótima qualidade e de um padrão de distância, mas que não precisa ser exato, como é mostrado no trabalho. Observa-se nos resultados que as imagens que foram obtidas nos estacionamentos, que possuem ótima qualidade e distância entre um e três metros, ou seja, padronizadas, o percentual de localização da placa ficou em 98,00 %, enquanto que nas imagens sem esse padrão, provenientes da Internet ficaram com 70,00 % e as do SeeCar em 54,84 %. Assim, ficou claro que através da técnica empregada, se for capturada uma imagem com qualidade e manter um padrão de distância para essa captura, o resultado será praticamente o ideal, que é o de localizar todas as placas.

Em seguida, foi realizado a rotação da placa, com um algoritmo simples e com 100,00 % de eficiência nos testes realizados.

Após a rotação, iniciou-se a fase de separação dos caracteres, que parte do mesmo princípio de localização da placa, ou seja, fazer através de características padronizadas. Os resultados mostraram mais uma vez que onde se tem um padrão aliado a qualidade da imagem, obtém-se melhores índices de separação dos caracteres. Enquanto que nas imagens dos estacionamentos foi conseguido 91,55 % de sucesso, na Internet foi de 84,90 % e no SeeCar de 87,39 %.

A última etapa do protótipo foi a do reconhecimento dos caracteres, através de método estatístico. Antes de se efetuar o reconhecimento propriamente dito, foi necessário criar vetores de características para letras e números, para posteriormente ser realizada a comparação entre os vetores e o caractere candidato em busca do seu respectivo código ASCII. Os testes mostraram resultados satisfatórios. De um total de 626 caracteres, 85,14 % dos mesmos foram reconhecidos acertadamente. Dentre esses, o melhor resultado foi com os números que foram reconhecidos em 88,06 % das vezes, enquanto que as letras ficaram com um índice de acertos de 80,72 %.

Com relação às dificuldades encontradas para a elaboração do trabalho, cabe destacar as inúmeras técnicas existentes, tendo que escolher conforme a aplicação e verificando se era viável ou não para esse projeto. O processamento de imagens é caracterizado por soluções específicas. Desse modo, técnicas que funcionam bem em uma área podem se mostrar totalmente inadequadas em uma outra área (GONZALEZ E WOODS, 2000). A etapa de

maior dificuldade foi a de localização da placa, pois foram testadas técnicas normalmente empregadas que funcionavam para determinados veículos, mas para outras não comportavam bem. Do início de desenvolvimento dessa etapa no protótipo até o seu término, transcorreram quase três meses, porém o resultado obtido demonstra o interesse e a vontade em oferecer algo para contribuir ao meio acadêmico/científico.

O reconhecimento da placa possui muitos desafios, principalmente por nenhuma empresa ter conseguido até hoje obter 100,00 % de reconhecimento, devido as técnicas empregadas ou até pelas condições da imagem ou captura. O importante é que mesmo os que já estão com produto comercial continuam a testar técnicas em busca de melhorias e que existe um mercado carente nesta área.

Para futuros trabalhos, acredita-se que algumas etapas poderiam sofrer alterações ou adaptações na busca por melhores resultados. Mesmo tendo um excelente índice de acerto na localização, para ambientes onde a distância não é padronizada, existe a necessidade de se criar novos parâmetros para localizar a placa. Na etapa de separação, quando a borda não for totalmente nítida, precisa-se buscar nova alternativa para efetuar a separação dos caracteres. A etapa de reconhecimento, em vez de método estatístico poderia ser realizada através de uma rede neural ou, se continuar com o método estatístico, criar vetores com maior número de amostras. Na etapa de separação, quando a borda não for totalmente nítida, é necessário encontrar nova alternativa para efetuar a separação dos caracteres.

Conclui-se, deste trabalho, ter adquirido o conhecimento necessário para dar continuidade a este, buscando melhorias e até no âmbito de desenvolver comercialmente um software capaz de efetuar o reconhecimento de placas. Todos os objetivos deste trabalho foram alcançados e cabe ao autor dar continuidade ou compartilhar seu conhecimento adquirido com parceiros. Cabe ressaltar ainda a cooperação no desenvolvimento de parte do protótipo, até a etapa de localização do colega, Idemar Felau, que faz nessa mesma instituição seu Trabalho de Conclusão I, sobre segmentação e localização de placas de veículos.

REFERÊNCIAS BIBLIOGRÁFICAS

AZEVEDO, Eduardo; CONCI, Aura. **Computação Gráfica – Teoria e Prática**. Rio de Janeiro: Elsevier, 2003. 384p.

BARBOSA, Jorge Leonardo Lima. **Processamento de Imagens Aplicado à Monitoração de Processos**. (Dissertação de Mestrado). Universidade Federal do Rio de Janeiro. Rio de Janeiro-RJ, 2004. 154p.

CANNY, John (1986). **A computational approach to edge detection**. In IEEE: Transactions on Pattern Analysis and Machine Intelligence, 8(6):679-698.

COMPULETRA. **SIVEM – Sistema de Identificação de Veículos em Movimento**. Disponível em: <<http://www.compuletra.com.br>>. Acesso em 11 de outubro de 2007.

CONCI, Aura; AZEVEDO, Eduardo; LETA, Fabiana R.. **Computação Gráfica – Teoria e Prática**. Volume 2. Rio de Janeiro: Elsevier, 2008. 420p.

CONTRAN. **Resoluções do CONTRAN**. Disponível em: <<http://www.denatran.gov.br/contran.htm>>. Acesso em 06 de outubro de 2007.

DAVIS, Clodoveu A. Jr. **PixelWare: Um Sistema de Processamento Digital de Imagens**. Minas Gerais: 1992. (Dissertação de Mestrado em Ciência da Computação) - Departamento de Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais, 1992.

DIAS, Eng. Fábio Gaiotto. **Melhorias para Sistemas de Reconhecimento da Placa de Licenciamento Veicular**. (Dissertação de Mestrado da Faculdade de Engenharia Elétrica e de Computação). UNICAMP. Campinas – SP, 2005. 162p.

GUINGO, Bruno Clemente; RODRIGUES, Roberto José; THOME, Antonio Carlos Gay. **Reconhecimento Automático de Placas de Veículos Automotores através de Redes Neurais Artificiais**. 2º Congresso Brasileiro de Computação, Itajaí-sc, 2002.

GONZALEZ Rafael C.; WOODS, Richard E.. **Processamento de Imagens Digitais**. Editora Edgard Blucher, Brazil, 2000. 509 p.

GONZALEZ, Rafael. C.; WOODS Richard E.; EDDINS Steven L. **Digital Image processing using MATLAB**. Upper Saddle River, N.J: Prentice Hall, 2002. 609 p.

JORGE, Lúcio A. C.; JÚNIOR Pedro A. V.; MANTOVANI Evandro C. **Avaliação de Atributos em Lotes de Sementes de Milho através do SIAC 3.02**. Circular Técnica Número 9/98. Embrapa, Brasil, 1998. 23p.

KAPTA. **Kapta – Sistema de Reconhecimento Automático de Placas de Veículos Automotores**. Disponível em: <<http://www.labic.nce.ufrj.br/projetos.htm>>. Acesso em 07 de outubro de 2007.

LOOKAR. **LookKar - Sistema de Reconhecimento da Placa de Licenciamento Veicular**. Disponível em: <<http://www.dca.fee.unicamp.br/~gaiotto/projects/srplv/conceitos.html>>. Acesso em 10 outubro de 2007.

PEDRINI, Hélio; SCHWARZ, William Robson. **Análise de Imagens Digitais – Princípios, Algoritmos e Aplicações**. São Paulo: Thomson Learning, 2008. 528p.

PONFAC. **Leitor de Placas PONFAC**. Disponível em: <<http://www.ponfac.com.br>>. Acesso em 07 de outubro de 2007.

RITTER, Gerhard X.; WILSON, Joseph N.. **Handbook of Computer Vision Algorithms in Image Algebra**, 1996. 384p.

RODRIGUES, Roberto José. **Segmentação e Extração de Características para Reconhecimento Automático de Caracteres – Estudo e Propostas**. (Dissertação de Mestrado). Universidade Federal do Rio de Janeiro. Rio de Janeiro-RJ, 2003. 169p.

RUSS, John C. **The image processing handbook** - 3rd ed. Materials Science and Engineering Department North Carolina State University Raleigh - North Carolina. 1998. 984p.

SHAH, Mubarak. **Fundamentals of Computer Vision** – Computer Science Department University of Central Florida - Orlando. 1997. 133p.

SHAPIRO, Linda; STOCKMAN, George. **Computer Vision** – The University of Washington Seattle, Washington e Department of Computer Science Michigan State University East Lansing, MI. 2000. 617p.

SEECAR. **SeeCar License Plate Recognition (LPR)**. Disponível em: <<http://www.htsol.com/>>. Acesso em 04 de outubro de 2007.

TECNIMA. **Leitor de Placas e de Trânsito**. Disponível em: <<http://www.tecnima.com.br>>. Acesso em 12 de outubro de 2007.

ANEXOS

ANEXO A Parte do fonte do protótipo desenvolvido no Matlab, onde são realizadas as comparações de altura e largura dos objetos, com os valores padronizados. Esse trecho do código fonte é equivalente a procura da região da placa das imagens de estacionamentos, mas o mesmo algoritmo será utilizado para separação dos caracteres também, bastando alterar os valores das variáveis de comparação.

```
%-----Altura e Largura-----
VetMax = max(TodosPontos);%Encontra todos os valores máximos de cada objeto
VetMin = min(TodosPontos);%Encontra todos os valores mínimos de cada objeto
AltMax = 125;%Altura Máxima do objeto
AltMin = 42;%altura Mínima do objeto
LarMax = 182;%Largura Máxima do objeto
LarMin = 119;%Largura Mínima do objeto
CompAltura = 0;%Compara Altura
CompLarg = 0;%Compara Largura
Pos1 = -1;%Utilizado para buscar valor dentro do vetor de pontos do
objeto
Cont1 = 1;%Contador dos vetores de Altura e Largura

%q3=total de objetos na imagem que serão percorridos em busca de selecionar
somente os que venham a estar dentro dos padrões estabelecidos de altura e
largura.
for j2 =1:q3
    CompAltura = VetMax(2*j2) - VetMin(2*j2);
    CompLarg = VetMax(Pos1 + 2) - VetMin(Pos1 + 2);
    VetorAltura(Cont1) = CompAltura;
    VetorLargura(Cont1) = CompLarg;
    Cont1 = Cont1 + 1;

    if CompAltura >= AltMin & CompAltura <= AltMax & CompLarg >= LarMin &
CompLarg <= LarMax
%São armazenadas em variáveis os pontos necessários do objeto para efetuar
o recorte do objeto logo a seguir
        EI = VetMin(Pos1 + 2);
        AI = VetMin(2*j2);
        AA = CompAltura;
        LL = CompLarg;

%aquí é realizado o recorte do objeto da imagem caso esteja dentro dos
parâmetros de altura e largura pré-definidos
        PlacaFinal = imcrop(fg,[EI AI LL AA]);
        figure, imshow(PlacaFinal);
    end;
    Pos1 = Pos1 + 2;
end;
```

ANEXO B Arquivo de Configuração da Versão de Demonstração do
Software SeeCar – Format.ini

Fonte: SeeCar (2007)

;Brazil

[General]

```
capture_width      = 768    ; captured picture size
capture_height    = 288    ;
frame              = 0      ; whole frame = 1; even field = 0
deinterlace       = 0      ; whether to deinterlace picture
n_formats         = 2      ; number of possible formats
eliminate_separators = 1    ; send id_string without separators
show_zoom         = 0      ; show plate after identification (SeeLane)
```

[Control]

```
min_grade         = 40
id_retry          = some
```

[Locate]

```
left_margin       = 4
right_margin      = 4
top_margin        = 4
bottom_margin     = 4
num_passes        = 2
win1_width        = 45     ; window rect
win1_height       = 32
win2_width        = 40     ; window rect
win2_height       = 25
max_areas         = 15     ; maximum output areas
min_hor_plate_length = 70
max_hor_plate_length = 350
min_hor_plate_height = 15
max_hor_plate_height = 65
```

; exact locate params

```
horiz_factor      = 5      ; factor used in exact find
vert_factor       = 6
txt_space         = 5      ; space between adjacent letters, used in x & y exact find
min_slope        = 0.01   ; minimal slope
pad_x            = 3      ; final area padding
pad_y            = 3
width2height     = 7
max_roll         = 0.6
```

[Enhance]

```
win_width_limit    = 130    ; window width and height limits.
                    ; window smaller than this requires doubling
win_height_limit   = 45
gentle_filter      = 0
sep_filter_constant = 6     ; filter value for separation
id_filter_constant = 9     ; filter value for identification
refine_sides       = 1     ; refine left / right sides
min_line_width     = 4
max_line_width     = 14
;----- FORMATS-----
```

[Format_1]

```
n_levels = 1
sex      = BW
```

[Format_1_Level_1]

```
n_symbols = 8           ; maximum number of symbols in level
symbol_1  = L
symbol_2  = L
symbol_3  = L
symbol_4  = B-
symbol_5  = D
symbol_6  = D
symbol_7  = D
symbol_8  = D
```

[Format_2]

```
n_levels = 1
sex      = BW
```

[Format_2_Level_1]

```
n_symbols = 7           ; maximum number of symbols in level
symbol_1  = L
symbol_2  = L
symbol_3  = L
symbol_4  = D
symbol_5  = D
symbol_6  = D
symbol_7  = D
```