

UNIVERSIDADE DE SÃO PAULO
ESCOLA DE ENGENHARIA DE SÃO CARLOS

LEONARDO AUGUSTO DE OLIVEIRA

Localização e reconhecimento de caracteres
em placas de automóveis

São Carlos
2010

LEONARDO AUGUSTO DE OLIVEIRA

**LOCALIZAÇÃO E RECONHECIMENTO
DE CARACTERES EM PLACAS
DE AUTOMÓVEIS**

Trabalho de Conclusão de Curso
apresentado à Escola de Engenharia
de São Carlos, da Universidade
de São Paulo.

Curso de Engenharia Elétrica
com ênfase em Eletrônica.

Orientador:

Prof. Dr. Adilson Gonzaga

São Carlos

2010

AUTORIZO A REPRODUÇÃO E DIVULGAÇÃO TOTAL OU PARCIAL DESTE TRABALHO, POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

Ficha catalográfica preparada pela Seção de Tratamento
da Informação do Serviço de Biblioteca – EESC/USP

0481

Oliveira, Leonardo Augusto de
Localização e reconhecimento de caracteres em placas
de automóveis / Leonardo Augusto de Oliveira ;
orientador Adilson Gonzaga -- São Carlos, 2010.

Monografia (Graduação em Engenharia Elétrica com
ênfase em Eletrônica) -- Escola de Engenharia de São
Carlos da Universidade de São Paulo, 2010.

1. ANPR. 2. LPR. 3. OCR. 4. Reconhecimento de placas
de automóveis. 5. Processamento de imagens. 6. Visão
computacional. I. Título.

FOLHA DE APROVAÇÃO

Nome: Leonardo Augusto de Oliveira

Título: "Localização e Reconhecimento de Caracteres em Placas de Automóveis"

Trabalho de Conclusão de Curso defendido e aprovado
em 29/12/2010,

com NOTA 9,0 (nove, zero), pela comissão julgadora:

Maria Stela Veludo de Paiva
Profa. Dra. Maria Stela Veludo de Paiva - EESC/USP

Maximiliam Luppe
Prof. Dr. Maximiliam Luppe - EESC/USP

Homero Schiabel
Prof. Associado Homero Schiabel
Coordenador da CoC-Engenharia Elétrica
EESC/USP

*Dedico essa conquista aos meus pais,
João Nazareno e Solange, que sempre me
apoiam e me amaram incondicionalmente.*

AGRADECIMENTOS

Aos meus pais, pelo apoio e suporte, superando a distância e a saudade.

Às minhas irmãs, Larissa, Lívia e Ludmila, pela amizade e união que me fortalecem sempre.

À minha namorada, Júlia, pelo amor e amizade constantes em nossa vida.

Aos meus amigos, em Goiânia e em São Carlos, em especial aos parceiros de república, pelo companheirismo nos estudos e nas festas, nos momentos de dificuldade e de alegria.

Ao meu professor, Adilson Gonzaga, pela correta orientação e constante presença na minha formação acadêmica.

A todos os professores e funcionários de Engenharia Elétrica da USP São Carlos: cada um tem sua parcela de contribuição para a formação dos novos engenheiros.

RESUMO

OLIVEIRA, L. A. Localização e reconhecimento de caracteres em placas de automóveis. 2010. 86 p. Dissertação (Graduação) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2010.

Este trabalho visa a discutir técnicas de visão computacional aplicadas ao processo de reconhecimento automático de placas de automóveis, bem como a apresentar uma solução simples e eficiente para localização, segmentação e reconhecimento dos caracteres. O sistema apresentado é dividido em três módulos principais: Localização, Segmentação e Reconhecimento. Imagens em escala de cinza são as entradas do sistema e uma sequência de sete caracteres constitui a saída. A etapa de Localização é baseada na busca de regiões da imagem de maior concentração de mudanças bruscas de intensidade, regiões em que o contraste claro-escuro é constantemente observado, ou seja, a porção da imagem em que se encontram os caracteres da placa do automóvel. A etapa de Segmentação consiste em processar a imagem, aplicando filtros espaciais e operações morfológicas, de modo a obter uma imagem binária sem ruído e com os sete objetos bem definidos. A etapa de Reconhecimento consiste em analisar cada objeto em comparação com um conjunto de máscaras que indicarão a identificação dele entre os 36 caracteres alfanuméricos. Neste trabalho, apresentamos uma discussão aprofundada acerca da importância e aplicabilidade dos sistemas de reconhecimento automático de placas de automóveis. Os resultados obtidos com o sistema proposto são satisfatórios tanto em termos de eficiência quanto em termos de velocidade de processamento, apesar de este não ser o foco do trabalho. Baseamos os objetivos deste trabalho no equilíbrio simplicidade-eficiência, buscando a formulação de um algoritmo simples e intuitivo que apresente, ao mesmo tempo, alta taxa de acerto. Separadamente, cada módulo apresenta taxa de acerto próxima de 90%. Foram utilizados cinco diferentes bancos de imagens, com características próprias, como qualidade da fotografia e posição/inclinação da placa na imagem. A taxa de acerto do sistema completo é de mais de 80% em alguns testes. Apresentamos, nas discussões finais, pontos positivos e negativos deste trabalho, indicando sugestões de pesquisas que devem elevar a taxa de acerto obtida.

Palavras-chave: ANPR. LPR. OCR. Reconhecimento de placas de automóveis. Processamento de imagens. Visão computacional.

ABSTRACT

OLIVEIRA, L. A. **Localization and recognition of characters in car license plates.** 2010. 86 p. Dissertação (Graduação) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2010.

This work discuss techniques of computer vision applied to the process of automatic recognition of license plates, and provides a simple and efficient solution for localization, segmentation and character recognition. The system presented is divided into three main modules: Localization, Segmentation and Recognition. Grayscale images are the inputs for the system and a sequence of seven characters is the output. The Localization stage is based on the search of image regions of highest concentration of abrupt changes in intensity, areas where the light-dark contrast is observed constantly, i.e. the portion of the image where the characters of the license plate are found. The Segmentation step consists in processing the image by applying spatial filtering and morphological operations in order to obtain a binary image without noise and the seven objects well defined. The stage of Recognition consists in examining each object in comparison with a set of masks that indicate his identification between the 36 alphanumeric characters. In this paper we present a detailed discussion about the importance and applicability of the automatic recognition of license plates. The results obtained with the proposed system are satisfactory both in terms of efficiency and in terms of processing speed, although this is not the focus of the work. The present study aimed at balancing simplicity and efficiency, by formulating a simple and intuitive algorithm that submits, at the same time, high success rate. Each single module has an accuracy rate around 90%. Five different image banks, with their own characteristics, such as picture quality and position/tilt of the license plate in the image. The accuracy rate of the whole system is over 80% in some tests. We present, in the final discussions, positive and negative aspects of this work, suggesting guidelines of research that should increase the hit rate obtained.

Keywords: ANPR. LPR. OCR. Plate recognition. Image processing. Computer vision.

LISTA DE FIGURAS

1.1	Diagrama de blocos das etapas de um processo de visão computacional	21
1.2	A entrada e a saída de um sistema de ANPR	22
1.3	Diagrama de blocos básico de um processo de ANPR	23
2.1	Esquema simples de um sensor de imagem digital	27
2.2	Representação matricial de uma imagem	28
2.3	Exemplo de imagem e seu respectivo histograma de níveis de cinza	30
2.4	Exemplos de operações lógicas entre imagens	32
2.5	Representações dos principais modelos de representação de cores	33
2.6	Esquematização das operações morfológicas de erosão e dilatação, abertura e fechamento	38
3.1	Exemplos de localização de placas com sucesso	45
3.2	Imagen-exemplo e sua respectiva projeção vertical	46
3.3	Atributos propostos por Polidório e Borges para análise sintática	50
3.4	Exemplo de atribuição dos nós da letra E	50
3.5	Mudança do padrão de placas de automóveis no Brasil nos anos 90	53
3.6	Fontes de placas de automóveis utilizadas nos últimos anos no Brasil	53
3.7	Exemplo de caracteres do sistema holandês de emplacamento	54
3.8	Exemplos de placas de alguns estados norte-americanos	54
3.9	Exemplos de imagens dos bancos de dados utilizados no trabalho	56
4.1	Esquema de um sistema completo de ANPR, com destaque para o núcleo do sistema	58
4.2	Diagrama de blocos da modularização do sistema de ANPR	58
4.3	Figuras-exemplo dos bancos de imagens utilizados no trabalho	60
4.4	Fotos promocionais do computador e da câmera fotográfica utilizados no trabalho	61
4.5	Esquema detalhado das etapas de um sistema de ANPR	62
4.6	Esquema do processo de carregamento das imagens	63
4.7	Esquema de correção de rotação da imagem	63
4.8	Imagens do processo de obtenção do gradiente de uma imagem	64
4.9	Esquema do processo de varredura da imagem por janelas	65
4.10	Resultado do processo de LPL e seu respectivo histograma de níveis de cinza	67
4.11	Esquema de procedimento de binarização da placa	68
4.12	Processo de eliminação de pequenos objetos não-interessantes ao sistema .	69
4.13	Eliminação de objetos acima ou abaixo dos caracteres	70
4.14	Exemplificação de correlações entre conjuntos de dados	74
5.1	Representação simples de um <i>outlier</i> : o ponto fora da curva	80
5.2	Esquema proposto de realimentação do algoritmo	83

LISTA DE TABELAS

2.1	Tabela de coeficientes t_{ij} para transformações espaciais	32
2.2	Esquema dos processos de convolução e correlação de matrizes	34
4.3	Análise de correlação de acordo com o CCP apresentada por Áurea Sousa .	74
4.4	Outras análises de correlação de acordo com o CCP	74
5.5	Resultados do processo de Localização	77
5.6	Resultados do processo de Segmentação por placa inteira	78
5.7	Resultados do processo de Segmentação por caractere	79
5.8	Resultados do processo de Reconhecimento por caractere	79
5.9	Tempos de execução, em segundos, do sistema de ANPR	81
5.10	Resultados do sistema de ANPR por caractere	81

LISTA DE SIGLAS

ABNT	Associação Brasileira de Normas Técnicas
ANPR	Automatic Number Plate Recognition
ASCII	American Standard Code for Information Interchange
BDx	Banco de Dados x (1, 2, 3, 4 ou 5)
BGT	Basic Global Thresholding
BLT	Basic Local Thresholding
BMx	Banco de Máscaras x (1 ou 2)
CBPF	Centro Brasileiro de Pesquisas Físicas
CCA	Connected Component Analysis
CCP	Coeficiente de Correlação de Pearson
CMY	Cyan Magenta Yellow
CMYK	Cyan Magenta Yellow blacK
CONTRAN	Conselho Nacional de Trânsito
COPS	Constrained Optimization Problem Solver
HT	Hough Transform
HSI	Hue Saturation Intensity
IPT	Image Processing Toolbox
LPDSI	Laboratório de Processamento Digital de Sinais e Imagens
LPD	License Plate Detection
LPL	License Plate Localization
LPR	License Plate Recognition
NBR	Norma Brasileira
OCR	Optical Character Recognition
RAM	Random Access Memory
RGB	Red Green Blue
RNA	Redes Neurais Artificiais
ROI	Region of Interest
SCW	Sliding Concentric Windows
SPIV	Sistema de Placas de Identificação de Veículos
SE	Structuring Element
USC	University of Southern California
USP	Universidade de São Paulo

SUMÁRIO

1 INTRODUÇÃO	21
1.1 Reconhecimento automático de placas de automóveis	22
1.2 Objetivos	23
1.3 Organização do trabalho	24
2 FUNDAMENTOS E APLICAÇÕES	27
2.1 Aquisição de imagens digitais	27
2.2 Fundamentos de imagens digitais	28
2.2.1 Representação	28
2.2.2 Resolução	29
2.2.3 Histograma de níveis de cinza	29
2.2.4 Imagens binárias	29
2.2.5 Binarização	30
2.2.6 Vizinhança	31
2.2.7 Operadores lógicos	31
2.2.8 Transformações espaciais	31
2.2.9 Imagens coloridas	32
2.3 Filtros espaciais	34
2.3.1 Filtro de suavização linear	35
2.3.2 Filtro de suavização não-linear	35
2.3.3 Filtro de <i>sharpening</i>	35
2.4 Operações morfológicas	36
2.4.1 Erosão e dilatação	36
2.4.2 Abertura e fechamento	37
2.5 Segmentação de imagens	37
2.5.1 Conectividade de pixels	37
2.5.2 Detectores de bordas	39
2.5.3 Transformada Hough	40
3 TÉCNICAS DE PROCESSAMENTO DE IMAGENS PARA ANPR	41
3.1 Localização e extração da placa	42
3.2 Segmentação de caracteres	46
3.3 Reconhecimento de caracteres	49
3.4 Características das placas de automóveis	52
3.5 Restrições dos algoritmos	55
3.6 Uma visão geral do sistema	55

4 METODOLOGIA E MATERIAIS	57
4.1 Foco do trabalho: ANPR	57
4.2 Uma visão geral do trabalho	58
4.3 Bancos de dados	58
4.3.1 Primeiro banco de imagens	59
4.3.2 Segundo banco de imagens	59
4.3.3 Terceiro banco de imagens	59
4.3.4 Quarto banco de imagens	59
4.3.5 Quinto banco de imagens	59
4.3.6 Exemplos dos bancos de imagens	59
4.4 <i>Software e hardware</i>	60
4.5 Uma visão geral do algoritmo	61
4.6 O sistema de ANPR desenvolvido	61
4.6.1 Carregamento de informações de entrada	62
4.6.2 Correção de rotação	62
4.6.3 Localização da região da placa	64
4.6.4 Processamento de imagem da placa	67
4.6.5 Ajuste da inclinação	69
4.6.6 Segmentação e separação de caracteres	69
4.6.7 Reconhecimento de caracteres	72
4.6.8 Obtenção dos dados de saída	75
5 RESULTADOS E CONCLUSÃO	77
5.1 Resultados de Localização	77
5.2 Resultados de Segmentação	78
5.3 Resultados de Reconhecimento	79
5.4 Resultados de processamento computacional	80
5.5 Conclusão	81
5.6 Trabalhos futuros	82

Capítulo 1

INTRODUÇÃO

O uso de imagens digitais remontam aos anos 20, com o uso do sistema de transmissão de imagens via cabo chamado Bartlane. O campo do processamento digital de imagens, porém, surge somente no começo dos anos 60, com o desenvolvimento dos computadores a ponto de suportarem armazenamento e processamento de grandes quantidades de dados, como é o caso de imagens. Técnicas de processamento de imagens começaram a surgir no fim dos anos 60 e início dos anos 70, com grande desenvolvimento nas áreas espacial e médica (GONZALEZ; WOODS, 2008, p. 3-5).

Desde então, o campo do processamento de imagens tem crescido consideravelmente, abrangendo diversas áreas de interesse e contando com técnicas e equipamentos de suporte cada vez melhores e mais sofisticados.

Visão computacional é um campo mais amplo, que engloba todas as etapas do ato de “enxergar” dos computadores. O processo inicia-se pela aquisição da imagem, passando pelo processamento, reconhecimento de características e finalmente a interpretação pela máquina das características da imagem. A Figura 1.1 é um esquema do processo.

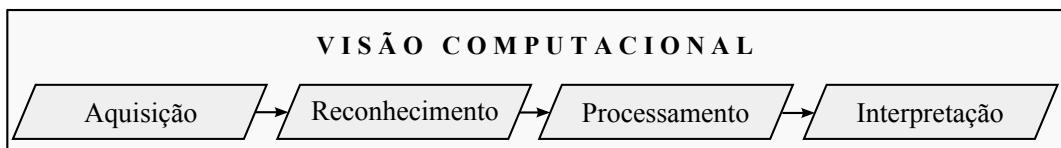


Figura 1.1: Diagrama de blocos das etapas de um processo de visão computacional

O crescente interesse pela automatização de processos e introdução de novas tecnologias nos campos de trabalho é fator motivador para diversas pesquisas no ramo de reconhecimento de padrões em uma imagem. O reconhecimento automático de placas de automóveis é um dos assuntos clássicos abordados em visão computacional desde o seu surgimento. Vários nomes são utilizados para identificar o assunto, sendo os mais comuns *Automatic Number Plate Recognition* (ANPR) e *License Plate Recognition* (LPR).

As técnicas utilizadas são igualmente variadas: reconhecimento ótico, por sistemas digitais, utilizando redes neurais e algoritmos genéticos. Na década de 70, quando iniciaram-se os trabalhos no assunto, fatores computacionais eram limitantes, não permitindo aplicabilidade direta das técnicas desenvolvidas. Com o avanço da microeletrônica e das arquiteturas de computadores, técnicas digitais passaram a ser mais empregadas, sendo

fortemente difundidas e ainda estudadas nos dias de hoje. Atualmente, métodos computacionais inteligentes permitem a abordagem do problema utilizando Lógica Fuzzy, Redes Neurais Artificiais (RNA) e Algoritmos Genéticos (CAMPOS, 2001).

Como não poderia ser diferente, as aplicações dos sistemas de reconhecimento de placas de automóveis são muitas: controle de tráfego, vigilância policial e particular, cobrança em pedágios e parques de estacionamento.

Lado a lado com os sistemas de reconhecimento de placas de automóveis, caminham pesquisas ligadas ao reconhecimento de caracteres como um todo. Alguns na área de digitalização de textos impressos, outros em reconhecimento de caligrafia manual, etc. Essa gama de assuntos incluídos na grande área “Reconhecimento de caracteres” permite uma constante e variada evolução tecnológica. Como já foi dito, não existe uma única direção a ser seguida pelos pesquisadores, e comumente surgem abordagens diferentes que trazem novos ares às pesquisas. Essa constante renovação é benéfica e bem-vinda, permitindo grande evolução dos sistemas e cada vez maior utilização prática.

1.1 Reconhecimento automático de placas de automóveis

O objetivo de um sistema de reconhecimento automático de placas de automóveis é fornecer uma saída simples para uma entrada complexa. A entrada é uma imagem estática, contendo diversos elementos desnecessários e o elemento essencial: a placa do automóvel. Computacionalmente, essa entrada é uma matriz numérica que contém informações dos pixels da imagem. A saída pode ser, por exemplo, uma sequência de caracteres, uma *string*, com os números e letras que compõe a placa.



Figura 1.2: A entrada e a saída de um sistema de ANPR

A situação descrita e exemplificada pela Figura 1.2 é o núcleo do sistema. Há que se considerar sistemas de suporte e apoio que tornam o sistema aplicável. As atividades que se encaixam nesse perfil são: obtenção da imagem estática ou vídeo, seleção de frames adequados (para o caso em que se usa vídeo), validação da saída obtida, reação do sistema à saída. São atividades que tornam o procedimento básico num sistema completo, que pode ser utilizado para os diversos fins já citados e serão mais amplamente discutidas no decorrer deste trabalho.

O grande interesse das pesquisas é o núcleo do sistema. As outras atividades são desenvolvidas especificamente para cada aplicação, sendo importantes apenas como motivação e exemplificação da aplicabilidade e importância do sistema. Portanto, o foco é basicamente fornecer a sequência de caracteres a partir de uma imagem estática.

O processo de reconhecimento de placas de automóveis segue um ciclo básico que pouco difere entre uma pesquisa (ou aplicação) e outra. Veja a Figura 1.3.

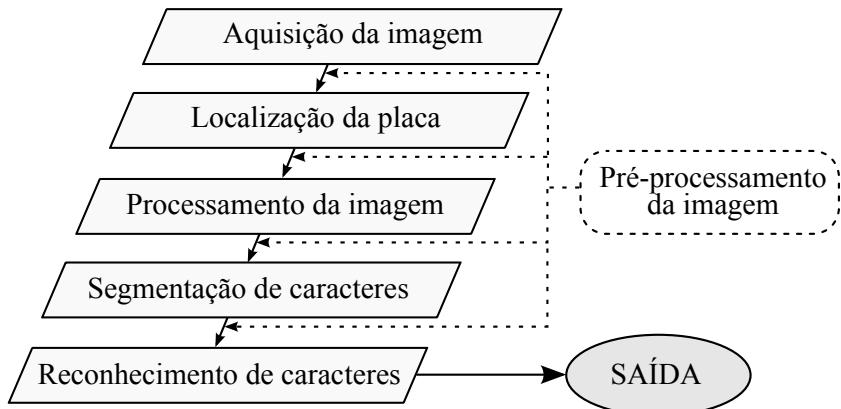


Figura 1.3: Diagrama de blocos básico de um processo de ANPR

Vale observar que podem existir etapas de pré-processamento da imagem entre as etapas intermediárias. Trata-se da utilização de técnicas de processamento que auxiliem o bom andamento das etapas subsequentes. Por exemplo, é comum binarizar as imagens antes da etapa de segmentação de caracteres. Existem, porém, exemplos de pesquisas que desconsideram algumas das etapas do diagrama. Em Zheng et al. (2007) desenvolve-se um algoritmo que não realiza a segmentação de caracteres. O padrão, seguido pela maioria das pesquisas, é o descrito pelo diagrama da Figura 1.3, o que não torna, obviamente, exemplos diferentes melhores ou piores.

1.2 Objetivos

Este trabalho descreve um método de localização de regiões de interesse (ROI, do inglês *Region of Interest*) e reconhecimento de padrões aplicado à identificação de placas de automóveis. Os objetivos do trabalho são: **(a)** descrever e analisar diversos métodos bem-sucedidos de localização de ROI e reconhecimento de padrões, principalmente aqueles

com a mesma temática do presente estudo, visando a estabelecer relações entre métodos, aplicações e resultados e (b) desenvolver algoritmo que utilize das conclusões de (a) para obter resultados satisfatórios, buscando aperfeiçoamento ou mesmo combinação das técnicas estudadas.

Cinco banco de imagens foram utilizados para desenvolvimento e teste do algoritmo. O primeiro é um banco de imagens fornecido pelo Laboratório de Processamento Digital de Sinais e Imagens (LPDSI) do Centro Brasileiro de Pesquisas Físicas (CBPF). Além de fornecer as imagens, o LPDSI fornece resultados obtidos em trabalhos desenvolvidos com o mesmo banco de dados, que serão comparados com os resultados deste trabalho. As imagens foram obtidas por uma câmera em posto de pedágio no estado do Rio de Janeiro e tem como características marcantes baixa resolução, má qualidade e proporção distorcida. Os outros quatro bancos de imagens são próprios, com fotografias obtidas pelo autor no estacionamento do *campus* de São Carlos da Universidade de São Paulo (USP). Dois destes bancos são compostos de imagens frontais da placa e outros dois com imagens obtidas diagonalmente, que necessitam de rotacionamento ou outro tratamento geométrico. São imagens de boa qualidade e sem distorção.

1.3 Organização do trabalho

O texto está organizado em cinco capítulos. O Capítulo 1 faz uma introdução do assunto, apresentando objetivos e motivação do trabalho e dando uma visão geral do assunto desenvolvido.

O Capítulo 2 apresenta fundamentos de visão computacional e processamento de imagens, com ênfase em temas recorrentes no trabalho e demonstração de aplicações. São detalhados métodos e/ou algoritmos gerais para cada um dos seguintes pontos acerca de imagens digitais:

- Aquisição
- Fundamentos
- Filtragem
- Morfologia
- Segmentação

No capítulo 3 cumpre-se o primeiro objetivo do trabalho: são apresentadas e analisadas diversas técnicas aprofundadas de processamento digital de imagens aplicadas ao ANPR. Neste capítulo, serão discutidas diversas abordagens de outras pesquisas acerca do tema, com descrição e análise de resultados. Especificam-se as motivações para o algoritmo desenvolvido neste trabalho, buscando fundamentação e apoio aos métodos aqui utilizados.

Finalizando o capítulo, é dada uma visão geral a respeito das regras e padrões estabelecidos para emplacamento de carros e, ainda, é feita uma discussão a respeito de características restritivas dos algoritmos de ANPR.

No Capítulo 4 cumpre-se o segundo objetivo: é apresentado o algoritmo de ANPR desenvolvido. São apresentados metodologia e materiais utilizados no trabalho e pseudo-códigos descrevendo cada etapa do sistema. Cada ponto abordado é amplamente discutido e justificado, com base nas referências, nos estudos e em conclusões matemáticas.

No Capítulo 5 são apresentados os resultados e conclusões do trabalho. Os resultados serão detalhados principalmente em forma de tabelas, buscando sempre a comparação e a análise crítica dos números, de tal forma a embasar as conclusões. A análise completa se dá com a discussão de perspectivas futuras para a continuidade deste trabalho, apresentada ao fim do texto.

Capítulo 2

FUNDAMENTOS E APLICAÇÕES

Neste capítulo, fundamentos de processamento de imagens digitais e visão computacional são apresentados, com foco em tópicos abordados pelo presente trabalho. O conhecimento desta teoria é imprescindível para correto entendimento do restante do trabalho. O presente capítulo usa como referência bibliográfica constante o livro *Digital Image Processing*, de Gonzalez e Woods (2008).

2.1 Aquisição de imagens digitais

A unidade básica para aquisição de imagens digitais é um sensor ótico, esquematizado na Figura 2.1.

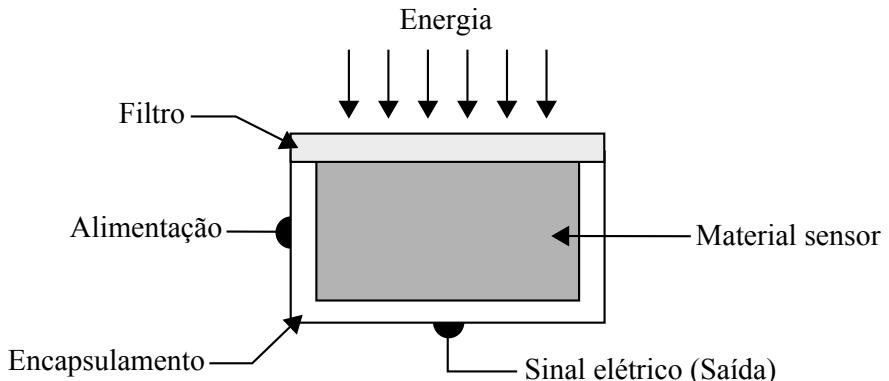


Figura 2.1: Esquema simples de um sensor de imagem digital

Basicamente, ocorre a transformação de uma energia de entrada em uma tensão elétrica na saída, obtida pela combinação da alimentação elétrica do dispositivo e a resposta do material sensor àquele tipo particular de energia. A energia pode ser raios-X, infravermelho, ultrassom e, mais comumente, luz visível. Dependendo da aplicação, utiliza-se o material sensor adequado ao tipo de energia esperado.

Outros tipos de dispositivos são meras variações deste sensor básico descrito. Por exemplo, pode-se ter uma matriz de sensores de modo se a obter vários pontos da imagem simultaneamente, ou mesmo dispositivos mecânicos podem fazer com um único sensor a varredura de um espaço unidimensional ou bidimensionalmente, obtendo vários pontos da imagem.

Uma imagem digital é uma função bidimensional da forma $f(x, y)$. O valor de f em cada ponto (x, y) é um escalar positivo cujo significado físico está ligado à fonte de luz e ao objeto. A função $f(x, y)$ será caracterizada pela multiplicação de dois fatores: **(a)** iluminação (a quantidade de luz incidente na cena) e **(b)** refletância (a quantidade de luz refletida pelos objetos na cena). Os valores de refletância variam de 0 a 1, ou de 0% a 100%, e os valores de iluminação, na prática, são positivos e finitos.

A função $f(x, y) = \text{iluminação} \times \text{refletância} = i(x, y) \cdot r(x, y)$ é contínua em relação às coordenadas x e y e também em relação à amplitude. Para a conversão para a forma digital, é preciso discretizar a função f em relação às coordenadas e também à amplitude. O processo de digitalização das coordenadas é chamado amostragem e o de digitalização da amplitude é chamado de quantização. Dada área em forma de quadrado no espaço que contém infinitos pontos, cada um com sua intensidade definida por $f(x, y)$, corresponderá a um pixel na imagem digital, com um único valor de amplitude. Quanto menor for este quadrado, e maior a quantidade de pixels, menos perceptível ao olho humano será a digitalização e melhor a qualidade da imagem.

Nas próximas seções, serão apresentados fundamentos e técnicas de processamento de imagens digitais. Obviamente, não convém expor aqui toda a teoria envolvendo visão computacional, mas sim a teoria focada à pesquisa e discussão do presente trabalho.

2.2 Fundamentos de imagens digitais

2.2.1 Representação

A principal maneira de representação das imagens digitais é a forma matricial, em que cada elemento da matriz representa um pixel, sendo os índices do elemento os indicadores da posição e o valor da sua amplitude o indicador da intensidade do pixel. Em razão dessa representação matricial, o elemento origem da imagem se localiza no canto superior esquerdo, e não no canto inferior esquerdo, como é comum em representação de funções. Veja a Figura 2.2.

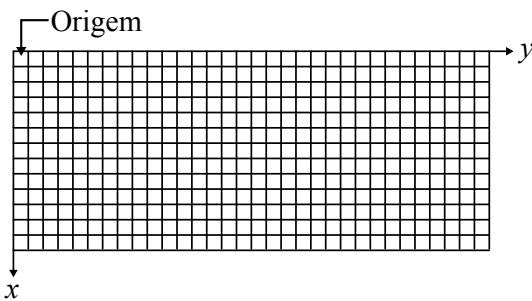


Figura 2.2: Representação matricial de uma imagem

Uma outra forma de representação, não tão usual nem intuitiva, é a indexada. Nessa forma, o valor da amplitude do elemento é um ponteiro para uma outra matriz. A ampli-

tude do elemento apontado na segunda matriz corresponde à intensidade do pixel localizado na primeira. Esta segunda matriz é usualmente chamada de mapa.

2.2.2 Resolução

A resolução espacial de uma imagem é a quantidade de pontos, ou *pixels*, representados por unidade de comprimento. A unidade de medida típica utilizada é *dpi*, do inglês *dots per inch*. Como o próprio nome diz, indica quantos pontos a imagem possui a cada unidade de polegada. Uma imagem com 150dpi terá melhor potencial de qualidade visual do que uma imagem de 72dpi , pois num mesmo espaço pode-se representar maior quantidade de valores intensidade diferentes.

A resolução de intensidade, por outro lado, corresponde à quantidade de níveis discretos diferentes para representar uma grandeza contínua. Se um pixel é representado por uma palavra de 8 bits, ou 1 byte, serão 256 níveis de cinza diferentes. Se a palavra por pixel for de 4 bits, ou 1 nibble, serão 16 níveis de cinza apenas. Quanto maior a quantidade de bits por pixel, maior a resolução de intensidade.

A questão de melhoria de qualidade de resolução espacial e de intensidade deve ser tratada com cuidado. Obviamente, maiores resoluções espacial e de intensidade implicam em maior espaço de memória utilizado para alocar uma imagem, assim como mais lento será o seu processamento pelo computador. Ainda, a percepção do olho humano deve ser considerada. Não existe a necessidade de representação de 1000 níveis de cinza se o olho humano consegue distinguir não mais do que 50 níveis. É uma questão de custo-benefício: cada aplicação tem as suas exigências e as suas limitações, cabe ao desenvolvedor definir o equilíbrio ideal entre o custo das melhorias e os reais benefícios que elas trazem.

2.2.3 Histograma de níveis de cinza

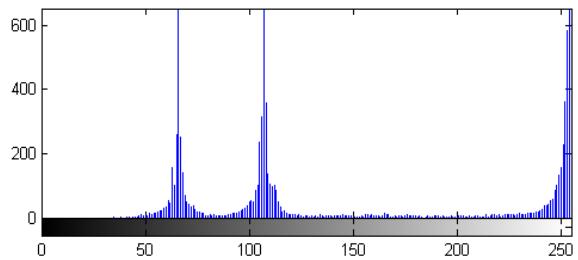
Histograma é uma representação gráfica da distribuição de frequências de uma massa de medições, normalmente um gráfico de barras verticais. O histograma é um gráfico composto por retângulos justapostos em que a base de cada um deles corresponde ao intervalo de classe e a sua altura à respectiva frequência. O histograma dos níveis de cinza de uma imagem consiste em levantar a quantidade de pontos presentes (distribuição de frequências) de cada nível de cinza (as classes). Veja as Figuras 2.3a e 2.3b. O histograma é útil para avaliar diversas características de uma imagem, como contraste, cor predominante e também limiar para binarização, como veremos mais adiante.

2.2.4 Imagens binárias

Imagens binárias são aquelas que possuem, para cada pixel, dois possíveis valores de intensidade: 1 ou 0, branco ou preto. A conversão de uma imagem em escala de cinza para uma imagem binária é o processo conhecido como binarização. Basicamente, estabelece-se



(a) Imagem-exemplo



(b) Histograma

Figura 2.3: Exemplo de imagem e seu respectivo histograma de níveis de cinza

um valor de limiar T entre 0 e 1 (sendo o limite inferior correspondente à cor preta e o limite superior à cor branca) e para cada pixel é feita a seguinte análise: se o valor de intensidade dele, numa escala de 0 a 1, for menor ou igual a T , ele é saturado para baixo; se for maior, ele é saturado para cima.

O grande segredo das técnicas de binarização está na correta escolha do valor de limiar. Uma escolha inadequada pode resultar em perda de informação relevante ou ruído indesejado em informação interessante. Existem inúmeros métodos de binarização, desde os mais simples, que definem em poucos passos um valor T para toda a imagem, até os mais complexos, que calculam um valor de T para cada pixel, levando em consideração medidas estatísticas locais.

Alguns métodos de binarização serão abordados mais profundamente na próxima seção.

2.2.5 Binarização

O processo de binarização tem como principal parte a definição do valor de limiar. Existem processos que definem um valor de limiar para toda a imagem, enquanto outros o definem por regiões ou até mesmo pixel a pixel. Abaixo, alguns métodos gerais adotados:

- (a) Método do vale - A partir da análise do histograma, pode-se determinar um vale entre dois picos, referentes ao tom claro e ao tom escuro predominantes em uma imagem. O limiar de binarização deve estar definido neste vale de modo que a binarização destaque esta polarização já existente neste tipo de imagem. Obviamente, há imagens em que não há essa clara distinção de um vale e dois picos no histograma, não sendo adequado, portanto, o uso deste método.
- (b) Binarização Básica Global (BGT, do inglês *Basic Global Thresholding*) - É um método iterativo apresentado em Gonzalez e Woods (2008). O algoritmo é composto por 5 passos: **(i)** Definir um valor estimado inicial T_0 . Recomenda-se utilizar a média das intensidades de todos os pixels da imagem. **(ii)** Dividir os pixels em dois grupos: G_1 , com todos aqueles com intensidade $\leq T$, e G_2 , com os restantes. **(iii)** Calcular as médias m_1 e m_2 dos pixels de G_1 e G_2 . **(iv)** Computar o novo valor

de limiar por $T = \frac{m_1 + m_2}{2}$. (v) Caso a diferença entre T e T_0 seja menor que um valor pré-definido ΔT , finalizar. Caso contrário, T_0 assume o valor de T e retorna-se ao segundo passo. Os autores analisam que o método funciona bem quando, assim como no método (a), há uma clara definição entre os picos e o vale no histograma da imagem.

- (c) Binarização Básica Local (BLT, do inglês *Basic Local Thresholding*) - Este método é uma variação do BGT. Consiste em dividir a imagem em partes e aplicar o método BGT em cada uma delas, obtendo um valor de limiar para cada. É um método bastante útil quando a imagem tem características variadas em partes diferentes, como iluminação não-homogênea.
- (d) Binarização variável baseado em propriedades locais - É uma abordagem mais específica do que a do BLT. Aqui, cada pixel terá um valor específico de limiar baseado em propriedades estatísticas de sua vizinhança. Seja σ_{xy} e μ_{xy} o desvio-padrão e a média dos pixels em uma vizinhança S_{xy} de um ponto (x, y) . O valor de limiar em cada pixel será $T_{xy} = a\sigma_{xy} + b\mu_{xy}$ ou $T_{xy} = a\sigma_{xy} + b\mu_G$, em que μ_G é a média global da imagem e a e b são constantes não-negativas.

2.2.6 Vizinhança

Um pixel $p(x, y)$ possui 4 vizinhos verticais e horizontais cujas coordenadas são dadas por $(x + 1, y)$, $(x - 1, y)$, $(x, y + 1)$ e $(x, y - 1)$. Diz-se que esse pixels são de vizinhança-4, e seu conjunto é denotado por $N_4(p)$.

Um pixel $p(x, y)$ possui outros 4 vizinhos diagonais cujas coordenadas são dadas por $(x + 1, y + 1)$, $(x - 1, y + 1)$, $(x + 1, y - 1)$ e $(x - 1, y - 1)$. Esse conjunto de pixels é denotado por $N_D(p)$.

Os pixels de vizinhança-8 ao pixel $p(x, y)$ são aqueles que pertencem ao conjunto $N_8(p) = N_4(p) \cup N_D(p)$.

2.2.7 Operadores lógicos

As operações lógicas nada mais são que aplicação da lógica binária pixel a pixel. São aplicadas em imagens já binarizadas. Os quatro operadores lógicos básicos são: **(a)** NOT **(b)** AND, **(c)** OR e **(d)** XOR (Exclusive-OR). A Figura 2.4 exemplifica claramente cada uma dessas operações.

2.2.8 Transformações espaciais

Operações de transformação espacial de uma imagem são basicamente mudanças de coordenadas que gerem como resultado uma imagem em escala, rotacionada, translacionada ou inclinada. A transformação de coordenadas pode ser expressa por $(x, y) = T[(v, w)]$,

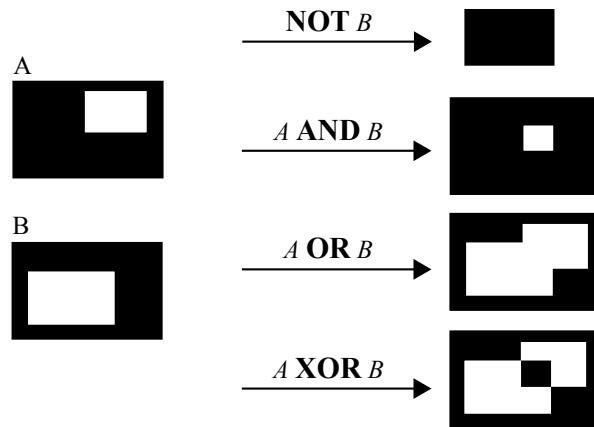


Figura 2.4: Exemplos de operações lógicas entre imagens

mas comumente o cálculo é feito de forma inversa, buscando-se para cada ponto na imagem de saída a sua transformação a partir da entrada. Ou seja, podemos expressar a transformação por $(v, w) = T^{-1}[(x, y)]$. Gonzalez e Woods (2008) citam os resultados obtidos por G. Wolberg em *Digital Image Warping* (1990). Este apresenta a *affine transform*, cuja forma geral é:

$$[x \ y \ 1] = [v \ w \ 1] \mathbf{T} = [v \ w \ 1] \begin{bmatrix} t_{11} & t_{12} & 0 \\ t_{21} & t_{22} & 0 \\ t_{31} & t_{32} & 1 \end{bmatrix}$$

Para cada transformação espacial, utiliza-se diferentes valores de t_{ij} , segundo a tabela a seguir.

Tabela 2.1: Tabela de coeficientes t_{ij} para transformações espaciais

Transformação	t_{11}	t_{12}	t_{21}	t_{22}	t_{31}	t_{32}
Escala	c_x	0	0	c_y	0	0
Rotação	$\cos \theta$	$\sin \theta$	$-\sin \theta$	$\cos \theta$	0	0
Translação	1	0	0	1	t_x	t_y
Inclinação	1	s_h	s_v	1	0	0

2.2.9 Imagens coloridas

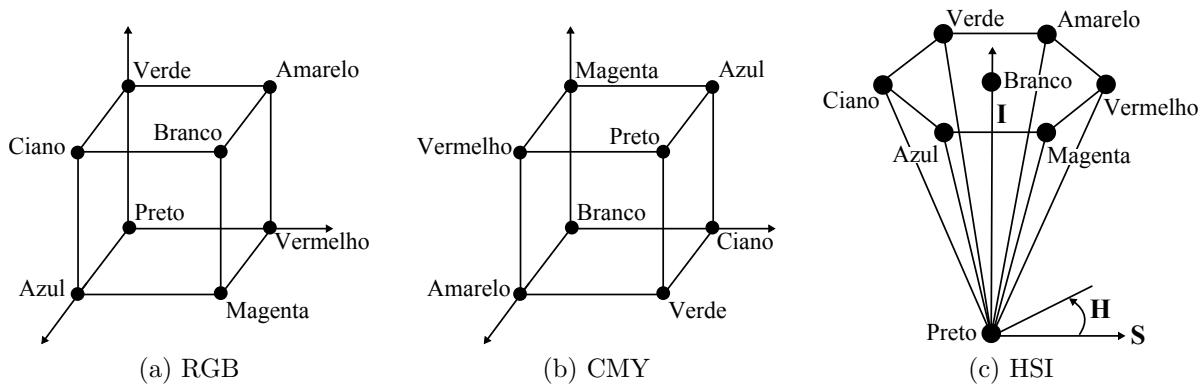
A representação de imagens coloridas nada mais é do que a representação, em diferentes matrizes, de características da cor da imagem. São vários os modelos de representação da cor de uma imagem. Destacamos aqui o mais famoso (RGB, de *Red Green Blue*), um amplamente utilizado em impressão (CMYK, de *Cyan Magenta Yellow black*) e um que aproxima a teoria matemática à maneira de percepção de cor do ser humano (HSI, de *Hue Saturation Intensity*).

- (a) RGB - Neste modelo, cada pixel será representada pelas três cores (vermelho, verde e

azul) que correspondem às componentes espectrais primárias. Tomando um exemplo simples: a cor amarela corresponde a 100% das componentes vermelha e verde e 0% da componente azul. Uma imagem digital representada pelo modelo RGB é composta de três matrizes em escala de cinza, cada qual correspondendo a uma das componentes. O valor de cinza em cada matriz indica a porcentagem da cor primária, e a análise dos três valores correspondentes à mesma posição do pixel permite a representação da cor desejada.

- (b) CMYK - As cores ciano, magenta e amarelo (CMY) são consideradas cores secundárias, por serem formadas, cada uma, por duas componentes espectrais primárias. São consideradas, porém, cores primárias dos pigmentos. A conversão de RGB para CMY é muito simples: $[C\ M\ Y] = [1\ 1\ 1] - [R\ G\ B]$. Enquanto a soma das componentes primárias de RGB resulta em branco, a soma das componentes primárias de CMY resulta em preto. A mistura dos pigmentos, no entanto, gera uma cor preta com pouco vigor, sem muito intensidade. Por isso os fabricantes de material de impressão utilizam o formato CMYK, em que a cor preta não será produzida pela mistura das outras três durante a impressão, mas previamente na fábrica.
- (c) HSI - Os formatos RGB e CMY são de fácil entendimento e computacionalmente fáceis de se trabalhar. Não são adequados, porém, para o ato de descrever uma cor de forma prática, de acordo com a percepção humana. Ninguém descreveria a cor de um objeto indicando a porcentagem de cada componente. Neste modelo, a componente H é a matiz, que indica puramente a cor de um objeto; a componente S é a saturação, que indica o grau de diluição dessa cor pura em luz branca (quanto maior a saturação, mais vívida é a imagem); e a componente I é a intensidade, que indica a noção acromática de brilho.

As Figuras 2.5a, 2.5b e 2.5c são representações gráficas de cada modelo descrito anteriormente, o que facilita o entendimento de como eles funcionam na representação das imagens.



As técnicas de processamento de imagens coloridas são usualmente aplicadas a cada componente separadamente, tratando-a como uma imagem em escala de cinza. Neste trabalho serão aplicadas operações em imagens em escala de cinza, e convém detalhar a apenas a transformação de uma imagem RGB para escala de cinza. Não existe uma teoria definida para tal transformação, existem adoções. Usualmente, a transformação ocorre segundo a fórmula $g(x, y) = 0,2989 r(x, y) + 0,5870 g(x, y) + 0,1140 b(x, y)$. Este é o padrão adotado, por exemplo, pelo *software* MATLAB.

2.3 Filtros espaciais

A filtragem espacial de uma imagem ocorre por composição de dois elementos: **(a)** uma vizinhança e **(b)** uma operação pré-definida. A filtragem gera um novo pixel de mesmas coordenadas do pixel central da região de vizinhança com valor de amplitude referente aos resultados propostos pela operação espacial. A forma como esta operação é aplicada em vizinhanças de todos os pixels de uma imagem exige dois conceitos: correlação e convolução espaciais.

Correlação é o processo de movimentação de uma máscara de filtro através da imagem computando a soma dos produtos em cada localidade. Convolução corresponde ao mesmo processo, mas com a máscara do filtro rotacionada de 180 graus. O uso de correlação ou convolução para aplicação de um filtro espacial é uma questão de preferência. É comum o uso do termo convolução na literatura de processamento de imagens se referindo a qualquer uma das duas operações. A Tabela 2.2 esquematiza os processos de convolução e correlação graficamente.

Tabela 2.2: Esquema dos processos de convolução e correlação de matrizes

Imagen $f(x, y)$	Máscara $w(x, y)$	Correlação					Convolução				
		0	0	0	0	0	0	0	0	0	0
0 0 0 0 0	1 2 3	0	9	8	7	0	0	1	2	3	0
0 0 1 0 0	4 5 6	0	6	5	4	0	0	4	5	6	0
0 0 0 0 0	7 8 9	0	3	2	1	0	0	7	8	9	0
0 0 0 0 0		0	0	0	0	0	0	0	0	0	0

Filtros de suavização de imagem (conhecidos como *smoothing filters*) são de grande utilidade para redução de ruído e eliminação de pequenos detalhes em detrimento de destaque a grandes objetos de uma imagem. São largamente utilizados em processamento de imagens para localização e reconhecimento de placas de automóveis.

Filtros de *sharpening* são úteis para avaliar regiões de transição de intensidade, homogeneizando regiões de transições pequenas e destacando regiões de transições bruscas, inclusive ruídos. São filtros que conferem nitidez à imagem (*to sharpen* significa tornar

nítido). São largamente utilizados nas etapas de segmentação de objetos dos algoritmos de reconhecimento automático de placas de automóveis.

2.3.1 Filtro de suavização linear

Filtro linear é aquele que realiza uma operação linear sobre os pixels da imagem. A resposta de um filtro de suavização linear é simplesmente a média dos pixels contidos na vizinhança da máscara do filtro. Estes filtros são comumente chamados de filtros de média ou filtros passa-baixa. Pela ação do filtro, mudanças bruscas de intensidade de um pixel para seu vizinho serão suavizadas. Dessa forma, ruídos indesejáveis serão eliminados, bem como bordas de porções pequenas da imagem. Dessa forma, a escolha do tamanho da máscara do filtro definirá as características que serão suavizadas e as que serão mantidas. Em um algoritmo de localização de placas de automóveis, deseja-se que objetos do tamanho de um caractere da placa não sejam afetados pelo filtro, mas sim o ruído que prejudique o seu reconhecimento.

Um dos efeitos negativos do filtro de média é o borramento da imagem. Para evitar esse problema, o filtro de mediana (analisado a seguir) é mais recomendado.

Um exemplo básico de máscara de filtro de média é

$$\frac{1}{9} \cdot \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

O coeficiente que multiplica a matriz realiza a normalização da máscara, de modo que a soma de todos os elementos seja unitária e a operação realizada durante a convolução ou correlação seja realmente uma média.

2.3.2 Filtro de suavização não-linear

A filtragem não-linear consiste em substituir o pixel analisado por um pixel escolhido a partir de um *ranking* feito com os pixels da máscara. O mais conhecido e utilizado é o filtro de mediana. Neste caso, os pixels da vizinhança são ordenados e o pixel analisado assume o valor que representa a mediana neste *ranking*. Estes filtros tem a grande vantagem de eliminar ruído sem causar tanto borramento como causa o filtro linear. São filtros notadamente eficazes para eliminar ruídos *salt-and-pepper*, cuja característica é a presença de pontos brancos e pretos (por isso o nome remete a sal e pimenta) na imagem.

2.3.3 Filtro de *sharpening*

Enquanto os filtros de suavização comumente realizam uma média da imagem, o que pode ser associado como análogo à integração, os filtros de *sharpening* realizam a derivação de uma imagem. Um dos mais conhecidos e utilizados é o Laplaciano: sua formulação é

de segundo grau e objetiva a obtenção de uma máscara que pode ser aplicada nos mesmos moldes dos filtros de suavização: por convolução ou correlação. Derivações de segundo grau são mais eficientes para o destaque de bordas e são de fácil implementação.

Para uma operação isotrópica, ou seja, invariante à rotação, a formulação da derivada segunda de uma função $f(x, y)$, que representa a imagem, é definida por $\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$, em que $\frac{\partial^2 f}{\partial x^2} = f(x+1, y) + f(x-1, y) - 2f(x, y)$ e $\frac{\partial^2 f}{\partial y^2} = f(x, y+1) + f(x, y-1) - 2f(x, y)$. A fórmula geral é

$$\nabla^2 f(x, y) = f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1) - 4f(x, y)$$

e, a partir da formulação acima, a máscara obtida para aplicação computacional do filtro é

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

2.4 Operações morfológicas

A morfologia matemática em imagens é uma ferramenta para extrair componentes que são úteis para a representação e descrição da forma da imagem. Discute-se aqui algumas das técnicas de processamento morfológico de imagens utilizadas em métodos de ANPR.

É importante saber de antemão o conceito de elemento estruturante (SE, do inglês *Structuring Element*). Ele é para as operações morfológicas o que as máscaras são para os filtros espaciais. Basicamente, são pequenos conjuntos ou subimagens utilizados como prova em uma operação morfológica. A operação terá como uma de suas características o elemento estruturante utilizado para realizá-la.

2.4.1 Erosão e dilatação

A erosão de uma imagem pode ser descrita como o conjunto dos pontos tais que um SE, transladado através da imagem, está contido nesta. Matematicamente, tomando a imagem A e o SE B (ambos partes do espaço Z^2), a erosão de A por B , denotada por $A \ominus B$, é definido como

$$A \ominus B = \{z | (B)_z \subseteq A\}$$

Graficamente, os resultados são exemplificados pela Figura 2.6c.

A operação de erosão pode ser vista como um filtro morfológico em que porções de uma imagem menores do que o SE serão eliminados, filtrados. É uma operação muito utilizada para afinar (no sentido de tornar mais finos) objetos na medida que o SE definir.

A dilatação de uma imagem pode ser descrita como o conjunto de pontos em que a imagem A e o SE refletido em relação à sua origem \hat{B} , transladado pela imagem, sobrepõe-

se em pelo menos um ponto. Matematicamente, a dilatação de A por B , denotada por $A \oplus B$, é definida como

$$A \oplus B = \{z | (\hat{B})_z \cap A \neq \emptyset\}$$

Graficamente, os resultados são exemplificados pela Figura 2.6d.

A dilatação de uma imagem é a operação dual da erosão: é utilizada para engrossar ou aumentar objetos.

2.4.2 Abertura e fechamento

As operações de abertura geralmente arredondam bordas de objetos e tendem a eliminar protusões estreitas saindo do objeto. Matematicamente, a operação de abertura, denotada por $A \circ B$, é uma operação de erosão seguida de uma operação de dilatação, utilizando o mesmo SE, ou seja

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

Graficamente, os resultados são exemplificados pela Figura 2.6e.

As operações de fechamento tendem a arredondar as bordas dos objetos mas, ao contrário da abertura, tende a preencher espaços entre objetos ou partes de um objeto e pequenos orifícios, eliminando-os. Matematicamente, como era de se esperar, a operação de fechamento, denotada por $A \bullet B$, corresponde a uma operação de dilatação seguida de uma operação de erosão, utilizando o mesmo SE, ou seja

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

Graficamente, os resultados são exemplificados pela Figura 2.6f.

2.5 Segmentação de imagens

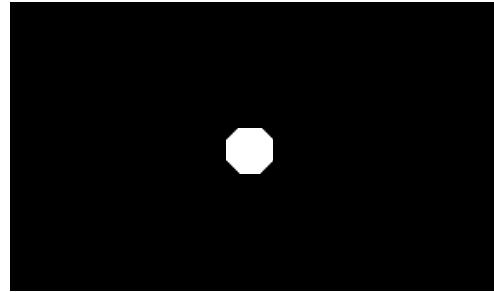
O processo de segmentação de imagens é extremamente importante para a Visão Computacional. O processamento das imagens deve culminar numa situação favorável à segmentação das regiões de interesse, para fins de análise e interpretação da imagem. Por exemplo, em um processo de ANPR, deve-se segmentar a região da placa do restante da imagem. Em seguida, costuma-se segmentar cada caractere, para reconhecimento ótico em separado. Veremos a seguir algumas técnicas de segmentação de imagens com foco nas suas aplicações em sistemas de ANPR.

2.5.1 Conectividade de pixels

Dada uma imagem com diferentes objetos, estes podem ser segmentados a partir da análise da conectividade dos pixels. É um processo simples e rápido: basta formar grupos



(a) Imagem de entrada A



(b) Elemento Estruturante B



(c) Erosão $A \ominus B$



(d) Dilatação $A \oplus B$



(e) Abertura $A \circ B = (A \ominus B) \oplus B$



(f) Fechamento $A \bullet B = (A \oplus B) \ominus B$

Figura 2.6: Esquematização das operações morfológicas de erosão e dilatação, abertura e fechamento

de pixels que sejam conexos de alguma forma. Por exemplo, um objeto será o grupo de pixels que tenham conectividade por vizinhança-8.

Um algoritmo simples para realizar este tipo de tarefa é o de rotulação da imagem. Considere uma imagem binarizada em que os pixels de valor 1 correspondem aos objetos e os pixels de valor 0 correspondem ao plano de fundo. O algoritmo possui 4 passos: **(a)** Realizar uma varredura da imagem em busca de um pixel de valor 1 e rotulá-lo com o valor $M = N + I$, sendo $I = 0$. **(b)** Rotular com o valor M todos os pixels que estejam conectados a este utilizando, por exemplo, vizinhança-8. **(c)** Incrementar I , buscar outro pixel de valor 1 e rotulá-lo com o valor $M = N + I$. **(d)** Repetir os passos **(b)** e **(c)** até que toda a imagem tenha sido varrida.

Como resultado, temos os pixels de plano de fundo com valor 0 e os pixels correspondentes aos X objetos com valor de N a $N + X - 1$.

2.5.2 Detectores de bordas

A detecção de bordas em imagens é uma das principais técnicas de segmentação utilizadas em sistemas de ANPR, seja na etapa de localização da placa, separação ou de reconhecimento de caracteres. Compreende-se por borda a região limítrofe em que há mudança brusca de intensidade em uma imagem. A detecção dessas mudanças de intensidade pode ser alcançada com o uso de derivadas de primeira e segunda ordem. Discute-se aqui as técnicas de primeira ordem.

O gradiente da função é a ferramenta utilizada para definir a borda em um dado ponto (x, y) . Seja o gradiente da imagem f definido como o vetor $\nabla f = [g_x \ g_y] = \begin{bmatrix} \delta f / \delta x & \delta f / \delta y \end{bmatrix}$, a aproximação digital das derivadas parciais em uma vizinhança próxima ao ponto é $g_x = f(x+1, y) - f(x, y)$ e $g_y = f(x, y+1) - f(x, y)$.

Para realizar a detecção de bordas de uma imagem, é preciso realizar um processo semelhante ao processo de filtragem. Basta definir corretamente os operadores matriciais que implementem o gradiente da função e realizar a convolução deste com a imagem. Os operadores de Prewitt e Sobel, apresentados por J. M. S. Prewitt e I. E. Sobel, ambos em 1970, são simples e muito utilizados. Os operadores horizontal e vertical de Prewitt são obtidos diretamente da análise das equações de g_x e g_y em uma região 3×3 :

$$G_x = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} \quad G_y = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

Já os operadores de Sobel são uma leve modificação dos operadores de Prewitt, atribuindo peso 2 à porção central da região. Esta variação dá como resultado uma leve suavização da imagem, minimizando o nível de ruído do resultado, o que é importante quando se lida com operadores derivativos.

$$G_x = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad G_y = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

Há ainda as variações de Prewitt e Sobel para detecção de bordas diagonais. As máscaras de Sobel, neste caso, são:

$$G_x = \begin{bmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{bmatrix} \quad G_y = \begin{bmatrix} -2 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 2 \end{bmatrix}$$

Obtidas as derivadas parciais g_x e g_y em cada ponto, o vetor gradiente pode ter sua magnitude e direção definidos por $M(x, y) = \sqrt{g_x^2 + g_y^2}$ e $\alpha(x, y) = \tan^{-1} \left(\frac{g_y}{g_x} \right)$. Vale ressaltar que o vetor gradiente terá direção normal à direção da borda no ponto (x, y) .

A magnitude, por sua vez, é frequentemente aproximada para $M(x, y) \approx |g_x| + |g_y|$, pois ganha-se em eficiência computacional sem grande prejuízo à correta análise das bordas.

O cálculo da magnitude do vetor gradiente é importante para distinção entre bordas com maior ou menor realce. É comum a utilização de um filtro detector de bordas combinando com a binarização. A definição das magnitudes dos vetores gradiente em cada ponto será importante para definir entre a manutenção ou eliminação das bordas com o processo de binarização.

2.5.3 Transformada Hough

A Transformada Hough (HT, do inglês *Hough Transform*) é um método proposto por P. V. C. Hough, em 1962, para análise de linhas retas em um plano xy . Trata-se de um método eficaz comumente utilizado para localizar retas em uma imagem e extrair características, como seu ângulo de inclinação. Num processo de ANPR, a HT pode ser aplicada para localização da placa ou mesmo identificação da inclinação da placa, para processo de correção de rotação. A seguir, a fundamentação teórica da HT.

Considerando um ponto $I = (x_i, y_i)$ no plano xy , existem infinitas retas que passam por este ponto, cuja fórmula geral será $y_i = ax_i + b$. Escrevendo a equação como $b = -x_i a + y_i$ e considerando o plano ab , chamado de espaço paramétrico, tem-se a equação de uma única reta para dado ponto (x_i, y_i) . Para dado ponto $J = (x_j, y_j)$ existirá uma intersecção (a, b) com o ponto I no espaço paramétrico. Dessa forma, as linhas correspondentes a todos os pontos (x_k, y_k) podem ser plotadas e, então, as linhas principais do plano xy são identificadas pelos pontos em que há intersecção de várias retas no plano ab . O uso do espaço paramétrico é incoveniente pelo fato de a inclinação da reta tender ao infinito à medida que a linha se aproxima da direção vertical. Por essa razão, adota-se o espaço paramétrico trigonométrico $\rho\theta$, cuja fórmula é $x \cos \theta + y \sin \theta = \rho$.

O uso da HT em uma imagem gera um gráfico de formato senoidal, em que os pontos de maior intersecção de senóides indicam o comprimento e a inclinação das retas mais longas identificadas na imagem. Obviamente, a imagem exige prévia binarização utilizando um filtro detector de bordas, de forma que a análise seja feita com os pixels que realmente interessam sobre um fundo preto.

Em um processo de ANPR, espera-se que a HT indique as linhas que delimitam a região retangular da placa do automóvel na etapa de LPL e a inclinação dessas linhas definem o ângulo para correção de rotação da imagem.

Capítulo 3

TÉCNICAS DE PROCESSAMENTO DE IMAGENS PARA ANPR

Visão computacional é um tema abrangente e foco de uma infinidade de pesquisas mundo afora. Em um *website*¹ da *University of Southern California* (USC), está divulgada uma bibliografia que contabilizava, em setembro de 2010, mais de 110 mil papers publicados, com referência a mais de 270 mil autores.

Uma busca por patentes no Google Patents² retorna para o verbete “OCR”, sigla referente a Reconhecimento Ótico de Caracteres (OCR vem do termo em inglês *Optical Character Recognition*), mais 500 resultados. Utilizando o verbete “*computer vision*”, são mais 450 patentes registradas.

Utilizando o Google Acadêmico³, temos mais de 1 milhão e 800 mil resultados para o verbete “*computer vision*” e mais de 1 milhão e 210 mil resultados para “*plate recognition*”.

O tema apresentado é, sem dúvida, motivação de uma infinidade de pesquisas e também diversas aplicações práticas. Apresentaremos aqui algumas pesquisas e suas principais características e resultados que serviram de referência direta para o presente trabalho, ao lado da fundamentação teórica inestimável de Gonzalez e Woods (2008), amplamente caracterizada no Capítulo 2.

São três etapas básicas no processo de reconhecimento de uma placa de automóvel:

- Localização e extração da placa
- Segmentação de caracteres
- Reconhecimento dos caracteres

O capítulo será dividido em três partes correspondendo às etapas citadas. Ao final, uma análise geral será feita buscando integração de conhecimentos e síntese das técnicas mais promissoras.

¹<http://iris.usc.edu/Vision-Notes/bibliography/contents.html>

²<http://www.google.com/patents>

³<http://scholar.google.com>

3.1 Localização e extração da placa

A localização e extração da placa consiste em realizar um rastreamento em toda a imagem a fim de identificar a exata região que contém somente a placa do automóvel, onde estão contidos os caracteres a serem reconhecidos, e extrair essa região, eliminando o restante da imagem.

Em Belvisi et al. (1999), é feita uma abordagem simples. A imagem adquirida é prontamente binarizada e utiliza-se um método baseado em algoritmos do tipo *chain-code*⁴ para buscar por uma área retangular fechada com o tamanho aproximado ao esperado das placas de automóveis. Não é recomendado fazer a binarização nesse ponto proposto pelos autores. Características diversas, como cor da placa, cor do carro, cor de outros objetos presentes na cena e iluminação muito forte ou muito fraca, podem ter uma influência negativa na definição do limiar de binarização, prejudicando a localização da placa e consequentemente todo o processo de reconhecimento de caracteres. Os próprios autores admitem a sujeição a grande número de erros do método e assinalam que o sistema ainda estava em desenvolvimento.

Em Brandão, Sequeira e Albuquerque (2004), os autores definem seu algoritmo como “baseado em operações morfológicas multi-estágio”. A busca por candidatos a placa em uma imagem considera três características básicas, para as quais foram desenvolvidos estágios independentes: **(a)** são regiões de alto contraste, **(b)** possuem sequência alinhada de objetos (os caracteres) e **(c)** são retangulares. O algoritmo percorre os três estágios em busca de regiões de interesse que obedeçam aos critérios. Os resultados indicam taxa de sucesso de 89,15% com o uso do estágio *(a)*, 94,21% com o uso dos estágios *(a)* e *(b)* e 97,18% de sucesso com o uso dos três estágios.

A conclusão é óbvia: combinar algoritmos é vantajoso. Não só nesta pesquisa. Observa-se que existem muitas abordagens diferentes⁵ para a Localização de Placas de Automóveis (LPL, do inglês *License Plate Localization*; ou LPD, do inglês *License Plate Detection*). São diversas as características que destacam uma placa de automóvel dentro da imagem, e analisar várias delas é a saída para obter melhores resultados. Ainda, há que se considerar as imagens de entrada e as restrições feitas. Se as fotografias são obtidas de grandes distâncias, as placas terão tamanho menor e serão mais susceptíveis a ruídos, prejudicando métodos de detecção de bordas. Por outro lado, se todas as imagens analisadas são de placas com cor amarela, por exemplo, pode-se utilizar uma análise baseada em

⁴ *Chain-code* é um algoritmo para representação de fronteira por um sequência de segmentos de retas conectados, tipicamente por vizinhança-8, sendo a direção de cada segmento codificada por um esquema de numeração: 0 para horizontal-direita, 1 para diagonal-superior-direita, etc. O código referente à fronteira formado pela sequência de números correspondentes às direções é conhecido como *Freeman chain-code* (GONZALEZ; WOODS, 2008).

⁵ Em Ganapathy e Lui (2008), os autores afirmam que técnicas simples por si só não são suficientes para atender às necessidades dos sistemas modernos. Um sistema inteligente de localização e reconhecimento de placas é requerido para operar robustamente em situações diversas e complicadas, como são as situações de aplicação real.

cor aproveitando essa característica. De acordo com a característica típica da placa que a pesquisa se dispõe a analisar, são estabelecidas restrições a serem consideradas no algoritmo. Uma análise posterior, neste trabalho, será feita a respeito das restrições (e a necessidade de se impô-las) no campo da ANPR.

Grande parte dos trabalhos, no que diz respeito a LPL, baseia-se em uma característica das placas que a destacam na imagem de um carro: alta frequência de mudança de brilho na região da placa. É uma característica que resiste à presença ou não de ruído, ao fato de ser uma imagem colorida ou em tons de cinza, inclinação inadequada e diversos outros fatores que dificultam a localização da placa.

Em Anagnostopoulos et al. (2005), é feito um grande estudo acerca das diversas abordagens do tema. Os autores reiteram a observação de que métodos baseados em detecção de bordas são altamente susceptíveis a ruídos indesejados. Eles analisam que os ruídos considerados não são somente devido à má qualidade da imagem ou má iluminação, mas a objetos que tem características semelhantes às placas. A utilização do método de detecção é motivada pela alta frequência de mudança de brilho nas regiões das placas de automóveis. Dessa forma, grades frontais presentes em alguns carros, ou mesmo adesivos com inscrições (podemos citar os adesivos ligados às propagandas políticas, tão comuns no Brasil em ano eleitoral) podem ser detectados erroneamente. Os autores, porém, analisam que o uso de técnicas morfológicas para eliminar bordas indesejadas previamente à aplicação do método de detecção de bordas resulta em índices de sucesso relativamente altos e computacionalmente rápidos se comparados a outros métodos.

Ainda em Anagnostopoulos et al. (2005), discute-se o uso da Transformada Hough, reiterando a afirmação de que ela exige muito esforço computacional. Mais uma vez é proposto a combinação de dois métodos: aplicação de algoritmos de detecção de bordas aliados à HT. O uso daquela técnica torna esta menos dispendiosa computacionalmente.

Os autores, em seu trabalho, propõe um método de análise estatística. A técnica de “Janelas Concêntricas Deslizantes” (do inglês *Sliding Concentric Windows* - SCW) foi desenvolvida para identificar irregularidades locais na imagem utilizando de medidas estatísticas como desvio padrão e média. Duas janelas retangulares concêntricas *A* e *B* (de tamanhos diferentes) deslizam pela figura, obtendo valores estatísticos que denunciam a característica irregular (a citada alta frequência de mudança de brilho) na localidade: quando a razão das medidas estatísticas das janelas *A* e *B* for maior que um valor definido *T*, a janela retangular *A* é considerada uma ROI, ou seja, candidata a placa. Para definir os tamanhos das janelas *A* e *B* e o valor de *T* os autores utilizam de um artifício interessante: realizar testes e buscar os melhores resultados. Para os tamanhos de *A* e *B*, espera-se que janelas de proporção igual à proporção das placas de automóveis são as melhores, como os testes comprovaram. Em relação ao parâmetro *T*, os autores assumem que não há evidência de como obter o valor e portanto a melhor maneira é obtê-lo por tentativa-e-erro.

Após a escolha de candidatos a placas de automóveis, os autores propõe um método de seleção do melhor candidato combinado ao processo de segmentação dos caracteres. Este método, portanto, será discutido mais adiante.

A maioria das pesquisas a respeito de LPL baseia-se no uso de técnicas de detecção de bordas, inclusive o presente trabalho. Como já foi discutido, tal técnica não apresenta imunidade a ruído. O sucesso dos métodos propostos, consequentemente, se dá pelo correto processamento da imagem a fim de eliminar ruídos que possam prejudicar a localização da placa. Basicamente, espera-se que nas regiões de alta frequência de mudança de brilho das imagens ocorram maior detecção de bordas do que em outras regiões.

Em Mahini et al. (2006), o sistema de LPL considera três características das placas⁶: a área contém várias bordas verticais, é composta de caracteres escuros em um fundo claro e os pixels têm tonalidade cinza. Para buscar as bordas verticais, aplica-se um filtro de Sobel com máscara vertical. Para buscar regiões de fundo claro, aplica-se uma operação morfológica de fechamento com elemento estruturante S_{3x3} . Para buscar as regiões de tonalidade cinza, basta que as intensidades em cada componente de cor (RGB) sejam aproximadamente iguais. Após a integração das três características, a região que obter resultado maior que um valor de limiar pequeno será considerada uma candidata a placa. Em seguida, regiões candidatas que não obedecerem a alguns critérios básicos são eliminadas. Os critérios são: **(a)** não pode ser uma região muito pequena, **(b)** deve ser retangular e de proporções semelhantes às de uma placa, **(c)** deve ter, em média, intensidade clara o bastante, **(d)** não pode estar conectada à margem da imagem e **(e)** devem ser praticamente horizontais. Os autores relatam taxa de sucesso acima de 96,5%.

Em Khalifa et al. (2006), calcula-se a projeção horizontal (vetor com soma dos valores de intensidade dos pixels em cada linha) e os picos do gráfico indicam a provável posição vertical da placa (a altura em que ela se encontra na imagem). Agora, janelas deslizantes buscam pelas regiões de maior densidade de bordas em projeções verticais. Dessa forma, temos a posição vertical e horizontal da placa na imagem. Os resultados obtidos foram de localização das placas com sucesso a uma taxa de 92,1%.

Em Araújo et al. (2006), antes de se aplicar um filtro detector de bordas, os autores restringem a busca a regiões da imagem que indicam a cor cinza, uma vez que a pesquisa é feita para aplicação com placas de automóveis particulares brasileiros. Em seguida, o filtro de Canny é aplicado e a busca pela ROI se dá pela análise dos polígonos existentes (“objetos” fechados após aplicação do detector de bordas): a ROI será aquela em que há polígonos alinhados, com altura semelhante e que obedeçam a proporção típica das placas de automóveis. Os testes foram feitos com um pequeno banco de dados (20 imagens) e apresentaram taxa de sucesso em relação à LPL de 90%.

Em Martinsky (2007), é proposto um algoritmo semelhante ao proposto no trabalho de Khalifa et al. (2006), em que primeiro encontra-se uma faixa horizontal no qual a ROI deve

⁶Os autores restringem a pesquisa às placas características do Irã.

estar contida. Dentro desta faixa, calcula-se a projeção vertical e análise semelhante é feita para identificar a posição da placa dentro daquela faixa. Com as candidatas selecionadas, o autor faz uma análise heurística de algumas características que validarão a melhor candidata. São consideradas 4 características: **(a)** a altura da região, sendo preferidas as de menor altura, **(b)** a altura do pico da projeção horizontal, sendo preferidas as regiões onde foram identificados maior quantidade de bordas verticais, **(c)** valor da área sob o gráfico deste pico e **(d)** proporção do retângulo. O autor atribui pesos a cada característica, de acordo com critérios empíricos: 15, 25, 40 e 40 para cada uma das características *(a)*, *(b)*, *(c)* e *(d)*, respectivamente.

Em um trabalho recente Ganapathy e Lui (2008), os autores analisam as quatro abordagens mais comuns para a LPL, ou LPD: **(a)** detecção de bordas, **(b)** análise de histograma, **(c)** operações morfológicas e **(d)** Transformada Hough. Os autores verificaram que a análise por detecção de bordas é simples e rápida, mas susceptível a ruído. A análise de histograma não é capaz de reconhecer placas em imagens com considerável nível de ruído e rotacionadas ou inclinadas, o que ocorre quando a imagem não é obtida frontalmente. A abordagem por operadores morfológicos não é susceptível a ruído, mas de execução muito lenta. Finalmente, o uso da Transformada Hough para detecção de linhas, assumindo que as placas são feitas por linhas retas, dá bons resultados, mas exige que as linhas que delimitam a placa estejam bem definidas, grande quantidade de memória computacional e tempo de execução alto.

Observa-se que são diversas técnicas e os resultados geralmente são satisfatórios. De fato, a localização das placas é uma das etapas mais simples dentro do sistema completo de ANPR. No entanto, deve-se atentar ao fato de que a localização do exato local de interesse é importante para abrir caminho para o sucesso no processo de segmentação de caracteres, uma das etapas mais complexas. Veja as Figuras 3.1a e 3.1b.



(a) LPL excelente



(b) LPL boa

Figura 3.1: Exemplos de localização de placas com sucesso

Ambas são consideradas exemplos de sucesso, mas a LPL mostrada na Figura 3.1a apresenta melhor resultado e provavelmente apresentará menos problemas ou dificuldades na etapa de segmentação dos caracteres.

3.2 Segmentação de caracteres

A segmentação de caracteres consiste em separar cada objeto contido na placa correspondente a um caractere para análise e reconhecimento em etapa posterior.

Existem vários métodos para segmentação de caracteres de uma placa de automóvel já localizada. O uso de métodos mais simples ou mais complexos, bem como a taxa de sucesso deles, depende muito de como foi feita a localização da placa e da qualidade dos algoritmos de pré-processamento das imagens, principalmente do algoritmo de binarização.

Na seção anterior, as Figuras 3.1a e 3.1b são exemplos de diferentes resultados em questão de qualidade dos algoritmos de LPL. Na Figura 3.1b, existirá a preocupação de não se considerar o furo destinado a fixação da placa no automóvel, na parte inferior, como caractere a ser enviado ao processo de reconhecimento. Na Figura 3.1a, não haverá a necessidade desse tipo de preocupação. Pode-se pensar ainda na questão do ruído. O adequado pré-processamento e uma técnica de binarização que elimine ruídos indesejáveis são interessantes para não haver engano nesta etapa. Ainda, após a seleção dos objetos que serão enviados à etapa de reconhecimento de caracteres, técnicas de análise de características típicas devem ser aplicadas para verificar se o objeto é realmente um candidato a caractere.

Em Guingo, Thomé e Rodrigues (2002), é feita uma análise da projeção vertical dos caracteres. A Figura 3.2 é um exemplo de aplicação do método com uma imagem de uma placa de dados próprio. O gráfico da projeção vertical indica em seus vales os pontos de separação entre um caractere e outro. Os autores não especificam o algoritmo utilizado, alertando apenas para a taxa de acerto de 76,61% em fotos de baxíssima qualidade e 100% de acerto em fotos de boa qualidade.

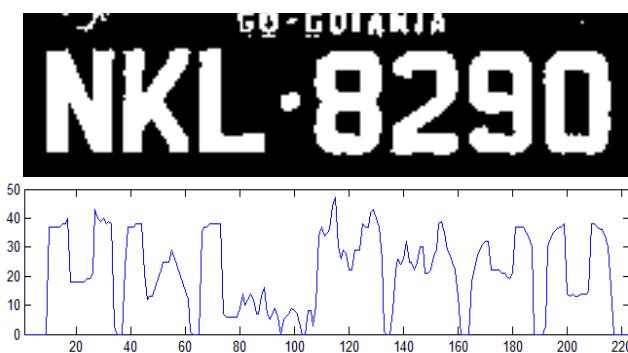


Figura 3.2: Imagem-exemplo e sua respectiva projeção vertical

O uso da projeção vertical é interessante pois se ocorrer união de dois caracteres por alguns pixels, pode-se considerá-los desde já como dois objetos separados, pois a região, por mais que tenha algum valor no gráfico da projeção, ainda é um vale. A dificuldade existe, porém, em como definir valores para delimitar uma vale ou um pico. Analisando objetivamente: tomemos um objeto que representa a letra **T**. As laterais terão projeções que possuem valor baixo que podem se confundir com um pequeno objeto que

não representa nenhum caractere, um simples ruído. Fica difícil definir em que ponto está o vale. Uma outra abordagem, largamente utilizada, consiste em agrupar pixels conectados entre si e analisar cada grupo para definir se é ou não um objeto candidato a caractere. Para essa abordagem, utiliza-se a técnica de rotulação de pixels, descrita na Seção 2.5.1.

Em Chang et al. (2002), assim como diversos trabalhos, realiza-se o agrupamento de pixels pela rotulação. Para definir quais objetos interessam e quais são descartáveis, o algoritmo faz diversas considerações. Primeiramente, exclui os objetos que tenham proporção (*aspect ratio*) fora de uma faixa pré-estabelecida, que é típica dos caracteres das placas analisadas. Em seguida, os objetos restantes devem estar alinhados. Para este cálculo, utiliza-se a Transformada Hough com os centróides⁷ para determinar o alinhamento dos objetos. Finalmente, se o número de objetos for maior que um número pré-estabelecido, elimina-se um a um a partir do de menor tamanho. Os autores, porém, consideram diversas possibilidades de erros, como um caractere ser composto de dois objetos ou dois caracteres formarem um único objeto. Esses dois fatores prejudicam a segmentação de caracteres com frequência, devido a problemas de ruído, iluminação ou mesmo má qualidade das placas. Considerá-los torna o projeto extremamente robusto. Os autores propõe um algoritmo que realiza sequencialmente operações de exclusão de objetos, junção de objetos e separação de um objeto em dois. A cada sequência, faz-se as análises já citadas: proporção, alinhamento e quantidade de objetos. O processo é repetido até que se chegue a uma solução satisfatória. É um algoritmo complexo, mas valioso se considerarmos a taxa de 95,6% de sucesso para a identificação de caracteres com uso de imagens de entrada complexas. Os autores não disponibilizam resultados isolados para a segmentação dos caracteres, mas percebe-se que ela se encontra acima dos 95%.

Em Conci e Monteiro (2004), após a rotulação dos pixels e obtenção dos diversos objetos contidos na imagem, um algoritmo estima, a partir das características dos objetos, os limites superior e inferior dos caracteres na placa. Dessa forma, todos os objetos que não obedecam a esse critério são eliminados e apenas os sete caracteres interessantes restarão. Os critérios para escolha de tais limites e de seleção dos objetos que correspondam a caracteres não é exposto pelos autores.

Em Anagnostopoulos et al. (2005), os autores propõe um método interessante de localização de candidatos a placas: SCW, ou “Janelas Concêntricas Deslizantes”, já descrito na Seção 3.1. Aos candidatos a placa, aplica-se um método de binarização utilizando um valor de limiar localmente adaptivo chamado de método de Sauvola, que calcula um valor de limiar para cada pixel considerando média e variância locais. O objetivo é eliminar problemas relacionados a iluminação não-homogênea. Em seguida, os autores propõe o método de Análise de Componentes Conectados (CCA, do inglês *Connected Component*

⁷Centróide é o centro de gravidade do objeto, considerando cada valor do pixel como um valor de massa naquele local.

Analysis), que nada mais é do que o método de rotulação já descrito neste trabalho. Com os pixels rotulados e agrupados, aplica-se um algoritmo de seleção dos objetos de interesse. Os autores provocam, propositalmente, que a placa seja um único objeto com diversos orifícios (os caracteres), para então fazer a seguinte análise: o objeto desejado deve ter orientação horizontal (inclinação menor do que 35 graus), *aspect ratio* entre 2 e 6 e *Euler Number*⁸ maior do que 3. Considera-se, ainda, o caso de a binarização resultar em caracteres como objetos e a placa como plano de fundo. Dessa forma, a condição do número de Euler será desobedecida, a figura terá seus valores de pixels invertidos e o processo de seleção será realizado novamente.

Com a imagem invertida, tem-se agora os caracteres como objetos. De forma semelhante à realizada anteriormente, aqueles objetos que não satisfizerem condições de orientação angular e altura são eliminados e o processo de segmentação é finalizado. Os autores indicam taxa de sucesso de 96,5%, o que confere grande valor do algoritmo desenvolvido, visto que utilizaram um banco de dados extenso (1334 entradas) e composto de imagens de características complexas.

Em Khalifa et al. (2006), é aplicado novamente um método de segmentação pelos componentes conectados. Após pré-processamento simples e binarização da imagem, os objetos rotulados corresponderão aos caracteres. Os autores desconsideram ruídos e problemas de iluminação. Têm a vantagem, porém, de restringirem o trabalho à análise de automóveis da Malásia, cujas placas são de grande contraste (fundo preto e caracteres brancos); as imagens serem de boa qualidade; e a placas terem sido extraídas manualmente, permitindo facilmente obter resultados como descrito na Figura 3.1a. A taxa de sucesso foi de 90,5%.

Em Draghici (2007), é feita inicialmente uma projeção horizontal de uma imagem binária, de modo a identificar grupos de objetos que estejam em uma mesma reta horizontal. Neste grupo, é feita a projeção vertical da imagem para identificar cada objeto separadamente. Feita a segmentação dos caracteres, os resultados serão validados. Caso um erro seja reportado, o algoritmo retorna ao ponto de binarização da imagem, buscando outro valor de limiar e repetindo o processo *projeção horizontal - projeção vertical - segmentação* até que sejam encontrados objetos que sejam validados como caracteres de uma placa de automóvel.

Em Ganapathy e Lui (2008), as imagens de entrada já estão binarizadas, uma vez que o processo de binarização já foi realizado antes da etapa de localização e extração da placa. O algoritmo de segmentação dos caracteres consiste em rotular os pixels, agrupando-os, e em seguida extrair parte da imagem em formato de um retângulo cujos lados são as bordas do agrupamento de pixels. Dentro desta área retangular, eliminam-se os menores

⁸*Euler Number* é definido como o número de objetos menos o número de orifícios. Para um único objeto, como é o caso, ele provê o número de curvas fechadas que o objeto contém, ou seja, o número de orifícios.

agrupamentos, restando apenas aquele que tem maior área e é considerado o caractere. Este processo complica algo que era simples: se os pixels estavam rotulados, cada objeto é um grupo de pixels de mesmo rótulo, não sendo necessária a extração da porção retangular. O processo é falho, ainda, por considerar que apenas objetos correspondentes a caracteres estavam presentes na figura. Qualquer objeto proveniente de ruído seria analisado como se fosse um dos caracteres da placa. Conclui-se que as etapas de LPL e pré-processamento da imagem devem ser executadas com primor para que a etapa de segmentação tenha sucesso.

Observa-se que constantemente os autores impõe restrições ao algoritmo desenvolvido. Claramente, uma das restrições mais comuns é quanto à “nacionalidade” das placas. Cada país tem sua característica de emplacamento, e grande parte dos autores desenvolvem as suas pesquisas considerando as características inerentes às placas de seu país. Uma análise mais profunda será feita na Seção 3.5.

3.3 Reconhecimento de caracteres

O reconhecimento de caracteres é a etapa em que um objeto é analisado e associado a um único caractere alfanumérico.

As técnicas de OCR por si só representam grande parte das pesquisas na área de visão computacional. As aplicações são inúmeras, desde o reconhecimento dos caracteres das placas de automóveis bem como de qualquer outro tipo de placa ou aviso, passando pela digitalização de documentos em formato texto (e não simplesmente em uma imagem) e finalmente reconhecimento de caligrafias manuais. O uso das técnicas de OCR em reconhecimento de caracteres em placas de automóveis é relativamente simples, pois normalmente não são consideradas diferentes fontes para as letras. Por outro lado, as imagens são bem mais ruidosas do que imagens de documentos digitalizados via *scanner*. No fim das contas, a relativa simplicidade deve favorecer no quesito velocidade de processamento, pois sistemas de ANPR normalmente exigem resposta em tempo real para a devida aplicabilidade.

Diversos autores utilizam de redes neurais para o processo de OCR. O objetivo básico de sistemas baseados em redes neurais é realizar um treinamento prévio do programa com um banco de dados específico para o treinamento, e então fazer a análise das placas de um outro banco de dados, que seria a placa que se deseja reconhecer. As Redes Neurais Artificiais, como o próprio nome sugere, são sistemas computacionais estruturados baseados em ligações como aquelas feitas pelos neurônios de um sistema nervoso. Não é, porém, tema abrangido pelo presente trabalho. Consideramos as RNA como uma ferramenta poderosíssima e que sem dúvida movimenta grande parte das pesquisas em visão computacional, com resultados excelentes. O presente trabalho foca-se, no entanto, em soluções mais simples, baseadas em morfologia e características estatísticas das imagens,

mas igualmente interessantes.

A análise de histogramas e de características morfológicas constitui tema de diversas pesquisas, inclusive deste trabalho. Trata-se de um modo simples e rápido de realizar o OCR. Exige, porém, boa segmentação e processamento prévio da imagem de modo a aproximar ao máximo o caractere desconhecido de sua máscara comparativa. Esta máscara comparativa nada mais é do que um objeto correspondente a um caractere do alfabeto que é previamente armazenado e serve para comparação com o objeto desconhecido. É o gabarito do sistema. Os métodos descritos abaixo tratam basicamente de maneiras de se comparar o objeto desconhecido a várias máscaras a fim de encontrar aquela com a qual ele mais se assemelha.

Em Polidório e Borges (1996), propõe-se um método sintático de análise dos objetos. Alguns atributos dos caracteres constituem a gramática, como denominam os autores. São as máscaras de comparação dos caracteres. Os autores propõe um primeiro grupo de atributos referentes aos nós existentes em um caractere: (a) um final de linha, (b) um encontro de duas linhas, (c) uma mudança suave de direção em 90 graus, (d) uma mudança brusca de direção em 90 graus e (e) um cruzamento de duas linhas. A Figura 3.6 apresenta os tipos de nós graficamente e a Figura 3.4 é um exemplo de atribuição dos nós da letra **E**. O segundo grupo de atributos trata do quadrante que está sendo analisado: I, II, III ou IV.

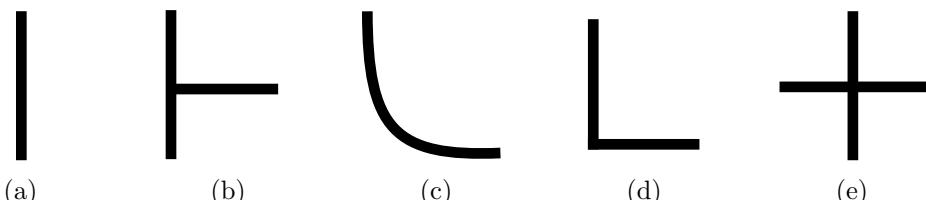


Figura 3.3: Atributos propostos por Polidório e Borges para análise sintática

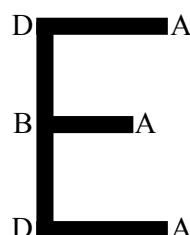


Figura 3.4: Exemplo de atribuição dos nós da letra E

Considerando os dois grupos de atributos, um caractere terá uma palavra associada a ele, na seguinte forma: *1-atributos-2-atributos-3-atributos-4-atributos*, em que *atributos* é substituídos pelas letras de (a) a (e) de acordo com o(s) atributo(s) de nó(s) que está(ão) contido(s) naquele quadrante. A comparação da palavra obtida com as palavras referentes às máscaras permite definir a qual caractere o objeto analisado se refere. Este é um sistema interessante e apresenta resultados satisfatórios: 91% de acerto contra 0,8% de erro e 8,2%

de rejeição. Rejeição ocorre quando não há acerto nem erro no reconhecimento. Vale destacar a baixíssima taxa de erro obtida e principalmente o tempo de processamento: de 2,5 a 3 segundos com o uso de um microcomputador DX4 1000MHz com 8MB de memória RAM. O uso de microcontroladores modernos (executando a vários MHz de frequência e com memória muito maior que 8MB) deve provocar diminuição do tempo de processamento deste algoritmo. O uso de microcomputadores, executando em GHz e memória em GB, deve diminuir drasticamente este tempo.

Em Belvisi et al. (1999), propõe-se um método de análise das projeções vertical e horizontal dos objetos. Aqui a comparação é feita não pelas funções $f(x, y)$ e $w(x, y)$ do objeto e da máscara, mas pelas suas projeções verticais e horizontais. A medida de igualdade entre as projeções do objeto e da máscara é dada através de um limite máximo percentual de pontos diferentes. Os autores não especificam um valor, mas assinalam uma alta porcentagem de acerto do sistema.

Em Conci e Monteiro (2004), aplica-se o método dos Momentos Invariantes de Hu. O momento de uma imagem é uma média ponderada das intensidades dos pixels que define alguma característica da imagem, como área, centróide, orientação, etc. O momento de ordem $(p+q)$ de uma imagem cuja função é $f(x, y)$ é definido como $M_{ij} = \sum_x \sum_y x^i y^j I(x, y)$. Assim, a soma das intensidades de uma imagem em escala de cinza será o momento M_{00} . Chega-se também à posição do centróide (\bar{x}, \bar{y}) por $(M_{10}/M_{00}, M_{01}/M_{00})$. Os autores avançam no sentido de buscar momentos invariantes a rotação, translação e escala, chegando aos momentos definidos por M. K. Hu em 1961 e amplamente difundidos e utilizados até os dias de hoje⁹. Espera-se, então, que cada objeto, ou seja, cada caractere, tenha um valor definido para os todos momentos que o diferem dos demais caracteres. Definidos os valores de momentos para cada caractere de **0** a **9** e de **A** a **Z**, e calculados os momentos do caractere desconhecido, basta estabelecer com qual deles há maior aproximação e está identificado o caractere desconhecido. O banco de dados de entrada utilizado pelos autores é simples, mas obteve-se o excelente resultado de 99% de acerto dos caracteres. O grande ponto negativo deste método se dá pelo longo tempo de processamento, já que os Momentos de Hu são computacionalmente dispendiosos. Dessa forma, torna-se de difícil aplicabilidade o sistema proposto por não responder em tempo real.

É amplamente utilizado o número de Euler (que indica o número de orifícios de um objeto) para agrupar caracteres de placas de automóveis em 3 classes: **(a)** grupo de caracteres com dois orifícios, composto pelos caracteres **8** e **B**, **(b)** grupo de caracteres com um orifício, composto por **6**, **9**, **A**, **P**, **R** e eventualmente, dependendo da fonte utilizada, o número **4** e **(c)** grupo de caracteres com nenhum orifício, composto pelos demais. O método, porém, não vai além desta simples separação em três grupos. Em Araújo et al. (2006), aplica-se uma técnica interessante que vai além dessa classificação:

⁹Os cálculos dos momentos invariantes utilizados estão esquematizados em Conci e Monteiro (2004).

seis grupos de caracteres são definidos, e em seguida os autores analisam os objetos dentro de cada grupo a fim de decidir por 1 entre os 36 possíveis como o caractere identificado. Os grupos são: **(a)** objetos com proporção de largura muito pequena em relação à altura (**1** e **I**), **(b)** objetos com duplo orifício, composto por **8** e **B**, **(c)** objetos com orifício Norte, em que o orifício está predominantemente na parte superior (**9**, **P** e **R**), **(d)** objetos com orifício Sul (**6** e **A**), **(e)** objetos com orifício completo, em que o orifício situa-se nas duas partes da imagem (**0**, **D**, **O** e **Q**) e **(f)** objetos de classe simples, que inclui todos os demais. Em seguida, é feita uma análise de quatro regiões da imagem: superior-esquerda, superior-direita, inferior-esquerda e inferior-direita, buscando características que indiquem diferenças entre os objetos. Por exemplo, se numa primeira análise temos um objeto do grupo **(c)**, e sabemos que trata-se de uma letra, a análise da porção inferior-direita indicará se é a letra **P** ou a letra **R**. Numa última análise, um algoritmo genético testa algumas possibilidades para chegar a uma conclusão final, para os casos de objetos com várias características semelhantes. Traços virtuais testam se o objeto que, por exemplo, supõe-se ser um **H**, tem as regiões preenchidas onde devem ser preenchidas e vazias onde devem ser vazias. Os autores assinalam taxa de sucesso maior do que 90% com o uso deste algoritmo.

Em Sancho (2006), utiliza-se a técnica de correlação-cruzada (*cross-correlation*). Basicamente, a imagem de um caractere $f(x, y)$ é comparada a uma máscara padrão $w(x, y)$ através da fórmula de correlação, conforme teorizado na Seção 2.3. Grosso modo, o valor de saída da função de correlação-cruzada $c(x, y) = \sum_s \sum_t f(s, t)w(x + s, y + t)$ será maior de acordo com a maior semelhança entre a máscara e o objeto analisado. O autor admite que símbolos semelhantes (por exemplo, as letras **O** e **D**) não podem ser avaliadas com segurança pelo método proposto. Para este problema, aplica-se um Solucionador de Problemas com Restrições Otimizado (COPS, do inglês *Constrained Optimization Problem Solver*). O COPS têm o objetivo de atingir os seguintes objetivos: **(a)** os candidatos tem ser placas de automóveis válidas na Espanha, **(b)** apenas os três símbolos com maior valor de correlação-cruzada serão considerados, **(c)** sequências maiores de caracteres terão precedência em relação às menores e **(d)** a soma dos valores de correlação-cruzada serão considerados. O autor indica uma taxa de 90% de acerto com menos de um segundo de tempo de processamento.

3.4 Características das placas de automóveis

As placas automóveis, em geral, seguem um padrão definido em lei pelo governo a que está submetido o dono do veículo. É costume usar diferentes cores para distinguir tipos de carros, estabelecer regras de nomenclatura, definir uma fonte típica, etc.

No Brasil, o Conselho Nacional de Trânsito (CONTRAN), no uso da competência conferida em lei, estabeleceu o Sistema de Placas de Identificação de Veículos (SPIV), por

meio da Resolução 231, de 15 de março de 2007 (CONTRAN, 2007).

O CONTRAN definiu tamanhos e materiais padrão para a confecção da placa, bem como a regra de cores e, principalmente, a fonte padrão utilizada para escrever os 7 caracteres alfanuméricos das placas nacionais. No início dos anos 90, institui-se uma grande mudança das placas de automóveis no Brasil, fortemente caracterizada pela substituição das placas amarelas pelas placas cinza. Veja os exemplos nas Figuras 3.5a e 3.5b.



(a) Padrão utilizado entre 1969 e 1990

(b) Padrão utilizado entre 1991 e 2006

Figura 3.5: Mudança do padrão de placas de automóveis no Brasil nos anos 90

Não foi feita, porém, uma padronização da fonte utilizada para os caracteres, apesar de ser comum o uso da fonte *DIN Mittelschrift*, exemplificada na Figura 3.6a. O CONTRAN, por meio da Resolução 231, dá um passo a frente estabelecendo como regra o uso da fonte *Mandatory*, exemplificada na Figura 3.6b.



(a) DIN Mittelschrift



(b) Mandatory

Figura 3.6: Fontes de placas de automóveis utilizadas nos últimos anos no Brasil

A padronização das placas de automóveis é importante para os sistemas de ANPR. Pode-se dizer que a eficiência de um sistema de ANPR depende de três pilares: **(a)** qualidade da placa, **(b)** qualidade da imagem e **(c)** qualidade do algoritmo.

A atitude do CONTRAN visa a não somente evitar fraudes, como clonagem de placas, bem como a auxiliar sistemas de ANPR, tão importantes em termos de segurança nos dias atuais. Nota-se que o sistema novo estabelece diferenças marcantes para os caracteres **D** e **9**, buscando evitar identificações incorretas destes com os caracteres **O** e **7**, respectivamente. É uma maneira eficiente de melhorar os resultados dos sistemas de reconhecimento.

Outros países vão mais além. Na Holanda, estabeleceu-se um sistema padrão das fontes de tal forma que houvessem características marcantes que distinguissem uma letra de outra. Observe na Figura 3.7 a característica das letras **R** e **P**. No Reino Unido, mudanças recentes estabelecem que, entre outras coisas, as letras **I** e **Q** não mais serão utilizadas nas placas dos veículos, dado o alto grau de ineficiência dos algoritmos de ANPR no reconhecimento destes caracteres, devido a semelhança com os caracteres **1** e **O**, respectivamente.



Figura 3.7: Exemplo de caracteres do sistema holandês de emplacamento

Na contramão das idéias holandesa e britânica, nos Estados Unidos as placas de automóveis tem uma variedade enorme de cores, desenhos, fontes. Cada estado norte-americano possui seu padrão (se é que podemos dizer que são padronizadas) e este muda constantemente, sempre valorizando alguma característica local (Veja as Figuras 3.8a, 3.8b, 3.8c, 3.8d, 3.8e e 3.8f). É uma questão cultural e há que se respeitar a decisão. Por mais que as padronizações ao estilo holandês, britânico e até mesmo brasileiro sejam boas para os sistemas de ANPR, a qualidade da imagem e do algoritmo desenvolvido são mais decisivos para que um sistema de ANPR seja eficiente.



(a) Florida '89



(b) Texas '98



(c) Arizona '97



(d) Florida '07



(e) Texas '09



(f) Arizona '09

Figura 3.8: Exemplos de placas de alguns estados norte-americanos

Neste trabalho, consideramos como regra a proporção típica das placas brasileiras (1:3 para a placa e aproximadamente 1:5 para o conjunto de caracteres) e as fontes mais comuns: *DIN Mittelschrift* e *Mandatory*. A seção seguinte trata melhor do assunto de restrições dos algoritmos.

3.5 Restrições dos algoritmos

Grande parte dos trabalhos de ANPR consideram, como já visto, algumas restrições inerentes às placas de automóveis. Por exemplo, grande parte dos pesquisadores foca, em um processo de LPL, na busca por um retângulo de proporção constante. Isso é possível porque sabemos que as placas dos automóveis, via de regra, têm forma retangular com proporção conhecida.

O prévio conhecimento da fonte padrão dos caracteres também é uma restrição aplicável aos algoritmos, aumentando eficiência e principalmente a velocidade de processamento.

Outras restrições se aplicam desde a obtenção das fotografias dos carros. É comum considerar imagens em que o objeto esteja a aproximadamente uma certa distância, sob certa inclinação, etc. Por exemplo, um algoritmo desenvolvido especificamente para identificar placas de automóveis em um cancela de um pedágio pode usar as características do posicionamento da câmera e do carro para tornar mais eficiente o sistema de ANPR.

Grande parte dos pesquisadores evita utilizar cenas de fundo complexo. É comum as imagens serem preenchidas totalmente pela frente ou traseira do carro. Imagens de cenas muito abertas dificultam o processo de reconhecimento, pois letreiros, adesivos ou qualquer outro objeto pode “confundir” o sistema de ANPR.

A discussão sobre quais restrições utilizar e quais não utilizar esbarra em uma análise básica: qual é o objetivo do sistema? Não há necessidade de desenvolver um sistema que se aplique a qualquer tipo de placa em qualquer situação, demandando enorme esforço, se ele será aplicado em situações simples e constantes.

Obviamente, o desenvolvimento de algoritmos robustos e pouco restritivos é importante para evolução dos sistemas de ANPR. É uma prática que deve ser incentivada, pois contribui para o crescimento de eficiência e importância desse ramo da visão computacional.

3.6 Uma visão geral do sistema

As três etapas distintas descritas nas Seções 3.1, 3.2 e 3.3 apresentam tanto facilidades quanto dificuldades, e estas variam conforme o banco de imagens de entrada que se tem disponível e a aplicação desejada. Por exemplo, imagens de plano de fundo complexo e poluído com outras inscrições além da placa exigirão grande cuidado na etapa de LPL, enquanto imagens frontais e próximas ao automóvel tendem a facilitar esse processo. Portanto, para aplicações com câmeras fixas em cancelas de pedágio espera-se LPL simples, enquanto sistemas de controle de tráfego necessitam de LPL complexo e robusto.

A etapa da segmentação parece ser a de maior susceptibilidade a erros. Problemas de iluminação, ruído e até mesmo na própria placa (corrosão, arranhados, etc) dificultam a correta segmentação dos caracteres e são difíceis de contornar. Além disso, as dificuldades

da etapa de LPL são acumuladas caso esta não seja executada com excelência, como já foi discutido anteriormente.

Os métodos de OCR, por sua vez, apresentam grande qualidade e ótimos resultados. Em geral, após uma segmentação com sucesso, os objetos chegam a esta etapa com uma boa qualidade e serão facilmente identificados. Costumam causar problemas apenas caracteres muito parecidos entre si, como o número **0** e a letra **O**. Neste caso, porém, a maior responsabilidade é dos órgãos reguladores de emplacamento de cada país, que devem agir no sentido de padronizar fontes e modelos de placas que eliminem este tipo de problema.

O método desenvolvido neste trabalho visa a localizar e identificar placas em imagens com fundo pouco complexo. Algumas imagens analisadas têm péssima qualidade (Figura 3.9a) e outras têm qualidade boa (Figura 3.9b). Todas as imagens são de placas de automóveis de padrão brasileiro e o desenvolvimento do algoritmo leva em consideração algumas características e restrições desse padrão.



(a) Péssima qualidade



(b) Qualidade boa

Figura 3.9: Exemplos de imagens dos bancos de dados utilizados no trabalho

Capítulo 4

METODOLOGIA E MATERIAIS

Neste capítulo, a metodologia da pesquisa será apresentada, juntamente com os materiais utilizados para desenvolvimento do sistema de Reconhecimento Automático de Placas de Automóveis.

Apresentaremos os algoritmos propostos para sistemas de ANPR. O problema foi modularizado, composto por três etapas principais: **(a)** localização e extração das placas, **(b)** segmentação dos caracteres e **(c)** reconhecimento ótico dos caracteres. Há que se expor também, como introdução, características e objetivos de um sistema de ANPR prático.

Os fundamentos teóricos apresentados no Capítulo 2 são de grande importância para entendimento dos procedimentos aplicados nos algoritmos propostos. A discussão realizada no Capítulo 3 é a base para direcionar os métodos utilizados neste trabalho. É a partir da análise e discussão de trabalhos anteriores que buscamos aprimorar os sistemas de ANPR, seja na qualidade dos resultados, em questão de robustez ou de velocidade de processamento.

4.1 Foco do trabalho: ANPR

Um sistema de ANPR deve visar sempre à execução com resposta rápida, ou seja, a imagem deve ser capturada de uma câmera de vídeo, processada e analisada pelo sistema e emitida uma resposta. Por exemplo, um sistema de controle da cancela de um estacionamento pago: o cliente tem a placa de seu carro cadastrada no momento em que entra no estacionamento. Portanto, o sistema deve capturar uma imagem do carro, executar o programa de reconhecimento da placa e armazenar aquela informação em sua memória, assim como informações de data, horário, etc. Havendo sucesso, a entrada será liberada. Na saída, o processo de reconhecimento da placa deve novamente ser executado, e o sistema deve confirmar o pagamento da tarifa para então liberar a saída do carro.

Este trabalho visa apenas a desenvolver o núcleo desse tipo de sistema: o processo de ANPR. Não será desenvolvido um sistema de captura das imagens, reconhecimento e geração de resposta. Propomos aqui apenas a discussão das técnicas e a validação dos algoritmos.

Os sistemas que usam o ANPR são diversos e cada desenvolvedor faz seu projeto de

acordo com as suas necessidades. O sistema de reconhecimento de caracteres, porém, é universal. É um simples módulo a ser inserido em todo o sistema. O esquema na Figura 4.1 exemplifica o foco deste trabalho.

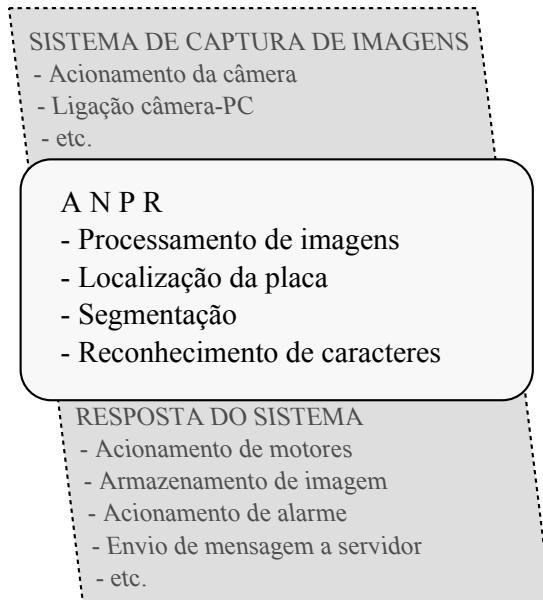


Figura 4.1: Esquema de um sistema completo de ANPR, com destaque para o núcleo do sistema

4.2 Uma visão geral do trabalho

O esquema da Figura 4.2 mostra um diagrama de blocos geral da modularização do sistema desenvolvido neste trabalho.

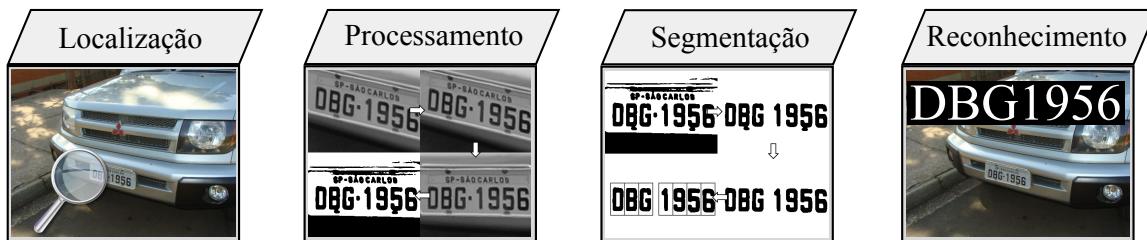


Figura 4.2: Diagrama de blocos da modularização do sistema de ANPR

4.3 Bancos de dados

Utilizou-se imagens estáticas capturadas e armazenadas para desenvolvimento e testes dos algoritmos neste trabalho. Os resultados a serem apresentados foram obtidos utilizando os bancos de imagens detalhados a seguir.

4.3.1 Primeiro banco de imagens

Este banco de imagens (Banco de Dados 1, ou BD1) foi obtido junto ao LPDSI e é composto de 75 imagens em níveis de cinza, obtidas em cancelas de pedágio no estado do Rio de Janeiro. As imagens são consideradas de má qualidade, tanto em questão de iluminação do ambiente quanto pela câmera utilizada. As imagens são distorcidas e os caracteres costumam apresentar falhas que prejudicam o sistema de ANPR.

4.3.2 Segundo banco de imagens

Este banco de imagens (Banco de Dados 2, ou BD2) é composto por 79 fotografias obtidas pelo próprio autor no estacionamento do *campus* I da USP São Carlos. As imagens têm qualidade boa e sem distorção. As imagens deste banco de dados foram obtidas tanto frontalmente quanto da traseira dos automóveis, sem qualquer inclinação.

4.3.3 Terceiro banco de imagens

Este banco de imagens (Banco de Dados 3, ou BD3) tem as mesmas características do BD2, com a única diferença que as imagens que o compõe não foram em nenhum momento utilizadas para testes durante o desenvolvimento do sistema, mas apenas para obtenção dos resultados com a execução do sistema pronto. Para os testes e ajustes durante o desenvolvimento utilizou-se o BD2. O BD3 é composto de 127 imagens.

4.3.4 Quarto banco de imagens

Este banco de imagens (Banco de Dados 4, ou BD4) é composto por 17 fotografias obtidas pelo próprio autor no mesmo local e com uso de mesmo equipamento do BD2 e BD3. Neste caso, porém, as imagens foram obtidas pela diagonal dos carros, o que exige correção de rotação e inclinação pelo sistema de ANPR antes da etapa de OCR.

4.3.5 Quinto banco de imagens

Este banco de imagens (Banco de Dados 5, ou BD5) tem as mesmas características do BD4, com a diferença, assim como o BD3 tem para o BD2, de não ter sido utilizado para testes durante o desenvolvimento do sistema. O BD5 também é composto de 17 imagens.

4.3.6 Exemplos dos bancos de imagens

As Figuras 4.3a, 4.3b, 4.3c, 4.3d e 4.3e são exemplos de cada um dos bancos de imagens utilizados.



Figura 4.3: Figuras-exemplo dos bancos de imagens utilizados no trabalho

4.4 Software e hardware

Os algoritmos propostos foram executados utilizando MATLAB versão R2008b 64-bit (7.7.0.471, de 17 de setembro de 2008), do fabricante The Mathworks, utilizando a linguagem própria dessa ferramenta. Esta é uma linguagem simples, de fácil entendimento e a ferramenta dispensa compilação para realizar testes, o que é importante quando há a necessidade de constantes pequenas mudanças no código. Ainda, o MATLAB oferece diversas funções matemáticas prontas, sem necessidade de adição de pacotes ou cabeçalhos, o que facilita e agiliza o processo de programação. Indo mais além, são oferecidas diversas extensões do MATLAB, entre elas o *Image Processing Toolbox* (IPT), que nada mais é do que um conjunto de funções direcionadas ao processamento digital de imagens. Concluindo: a ferramenta MATLAB oferece facilidades e suporte que o qualificam como ideal para desenvolvimento de sistemas para testes, que é o caso do presente trabalho.

A ferramenta MATLAB exige muita memória do computador e grande investimento para obtenção da licença. No caso de desenvolvimento de um produto final de um sistema de ANPR, o recomendado é a utilização da linguagem C com o uso do pacote OpenCV, que traz uma biblioteca de funções direcionadas à visão computacional. Dessa forma, o código pode ser compilado e executado sem a necessidade de qualquer *software* de suporte, como exige o MATLAB.

O *software* está instalado em um computador pessoal portátil da marca Hewlett-Packard, modelo HP Pavillion dv4-1225dx (Veja a Figura 4.4a). O computador possui processador AMD Turion X2 64-bit de 2,1GHz com 4GB de memória RAM e utiliza o sistema operacional Windows 7 Professional operando em modo 64-bit.

Para obtenção das imagens dos bancos de imagens BD2, BD3, BD4 e BD5 foi utilizada uma câmera da marca Sony, modelo Cyber-shot DSC-H10 (Veja a Figura 4.4b). A câmera foi utilizada com todas as configurações automáticas, sem flash e com resolução mínima ($640 \times 480\ px$).



(a) HP Pavillion dv4

(b) Sony Cyber-shot H10

Figura 4.4: Fotos promocionais do computador e da câmera fotográfica utilizados no trabalho

4.5 Uma visão geral do algoritmo

O esquema da Figura 4.5 é um detalhamento das etapas de programação do sistema de ANPR desenvolvido. A seguir, apresentaremos cada um desses blocos, detalhando entradas e saídas, procedimentos realizados e resultados esperados.

Os blocos de contorno tracejado na Figura 4.5 representam os três principais módulos deste e de tantos outros sistemas de ANPR: **(a)** Localização, **(b)** Segmentação e **(c)** Reconhecimento. São estes módulos que devem ter resultados analisados e comparados, se possível separadamente. No Capítulo 3, analisamos e discutimos propostas de outros trabalhos acerca de cada um desses três módulos. Neste capítulo apresentamos o nosso trabalho e no capítulo seguinte a análise se estende aos resultados de cada módulo, com as devidas comparações e conclusões necessárias.

4.6 O sistema de ANPR desenvolvido

As seções a seguir descrevem detalhadamente todo o procedimento a ser realizado com uma imagem para reconhecimento dos caracteres da placa do automóvel.

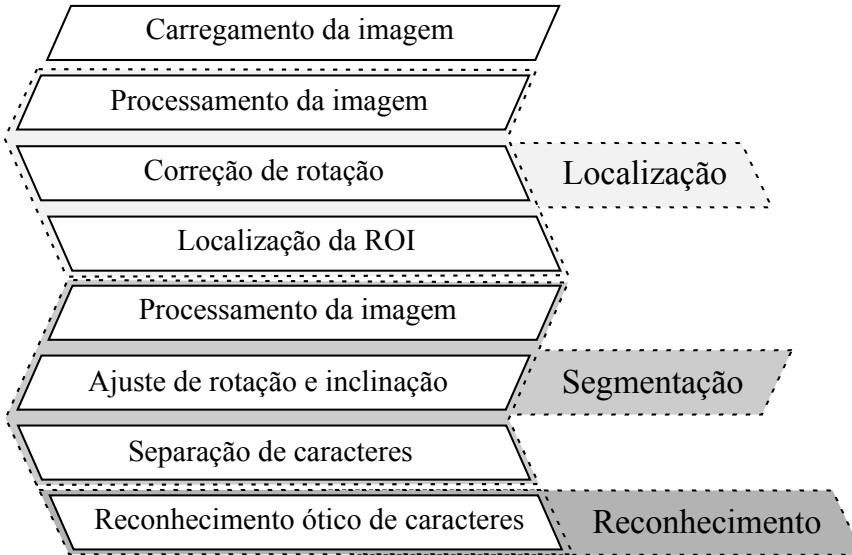


Figura 4.5: Esquema detalhado das etapas de um sistema de ANPR

4.6.1 Carregamento de informações de entrada

A primeira etapa consiste em carregar as informações de entrada: a própria imagem e informações a respeito de suas restrições. A imagem de entrada, quando colorida, será prontamente convertida para escala de cinza, de acordo com o método padrão do MATLAB, descrito na Seção 2.2.9. Esta imagem fornece algumas informações que serão úteis futuramente: tamanho em pixels (altura e largura) e banco de dados a que pertence. Esta informação remete à idéia das restrições, já discutidas na Seção 3.5. Cada banco de dados possui algumas características restritivas ao código: tamanho esperado da placa e proporção típica. O tamanho esperado é definido pela distância do objeto no momento da obtenção da imagem. A proporção de largura por altura de uma imagem (conhecido como *aspect ratio*) é, na prática, sempre a mesma. No entanto, dois motivos provocam a alteração dessa proporção: **(a)** distorção da imagem pela própria câmera fotográfica (este caso ocorre no BD1) e **(b)** pequenas rotações da placa. Portanto, para cada banco de dados, haverá uma faixa de tamanhos e proporções esperadas.

A Figura 4.6 é um esquema desses primeiros passos do sistema.

4.6.2 Correção de rotação

Imagens da placa do automóvel obtidas frontalmente apresentam pequena rotação e podem ser consideradas alinhadas, prontas para a etapa de localização. Aquelas obtidas na diagonal, porém, necessitam de correção de rotação antes de seguir adiante. Veja o esquema da Figura 4.7.

É importante notar que as imagens que necessitam de correção de rotação também necessitam de correção de inclinação dos caracteres. No entanto, não há necessidade de tratar desta questão agora.

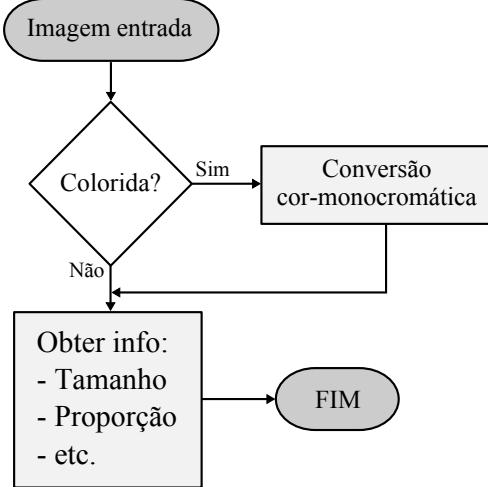


Figura 4.6: Esquema do processo de carregamento das imagens

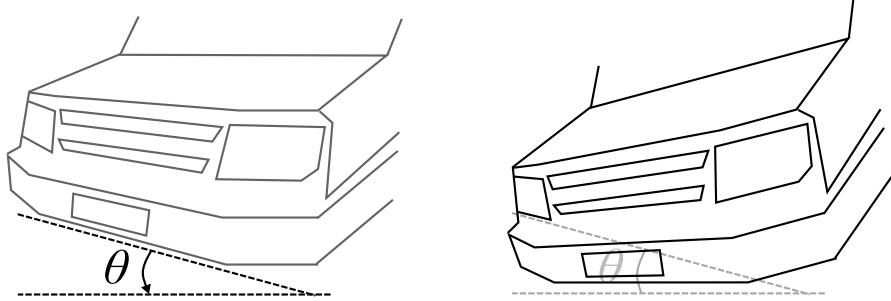


Figura 4.7: Esquema de correção de rotação da imagem

A correção de rotação será feito com o uso da HT, descrita na Seção 2.5.3. A seguir, os passos do processo:

1. Aplicar detector de bordas de Sobel na imagem;
2. Aplicar HT ($\rho_j = x_j \cos \theta_j + y_j \sin \theta_j$) em cada pixel J da imagem;
3. Obter uma matriz H cujos índices indicam o par (ρ_j, θ_j) e o valor de cada elemento indica a quantidade de pontos do plano xy que residem na reta indicada em $\rho\theta$;
4. Elementos grandes de H indicam as linhas retas da imagem, pois indica que vários pontos residem naquele par (ρ_j, θ_j) . Selecionar o maior elemento de H ;
5. Encontrar θ correspondente ao H_{max} no gráfico $\rho = x \cos \theta + y \sin \theta$;
6. O valor do ângulo é tal que $0 \leq \theta \leq \pi$. Caso $\theta > \pi/2$, definir $\theta' = \theta - \pi$, que indica mudança no sentido de rotação. Caso contrário, $\theta' = \theta$. Armazenar valor θ' ;
7. Rotacionar a imagem de θ' radianos conforme método descrito na Seção 2.2.8.

Após essa correção de rotação, a imagem pode ser analisada nas etapas seguintes como se a fotografia tivesse sido obtida frontalmente à placa do automóvel.

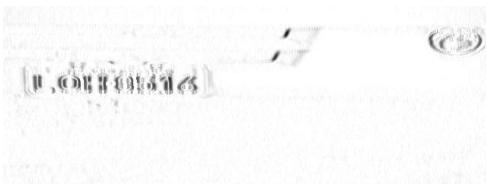
4.6.3 Localização da região da placa

O processo de LPL é uma das principais etapas de um sistema de ANPR. Já vimos no Capítulo 3 que algumas pesquisas dedicam-se exclusivamente a desenvolver sistemas de LPL. O método para localização e extração da placa desenvolvido se baseia na busca por regiões da imagem onde o gradiente horizontal tem valor mais significativo. A característica de alta frequência de mudança de intensidade das placas dos automóveis devido aos caracteres e o fundo da placa terem grande contraste entre si promove maior concentração de grandes magnitudes do gradiente naquela região.

O gradiente horizontal de uma função é dado por $\nabla f_y = \frac{\delta f}{\delta y}$. Vale lembrar que o eixo horizontal de uma imagem é o eixo y , conforme mostra a Figura 2.2. O vetor gradiente terá os valores correspondentes às diferenças entre os pixels vizinhos na direção horizontal. Uma imagem resultante dessa operação (considerando o módulo do gradiente) acusa regiões de maior mudança brusca de intensidade onde o gradiente tiver maior valor, ou seja, em regiões mais próximas da cor branca. E as regiões de maior mudança brusca de intensidade são as ROI: as placas dos automóveis. Veja as figuras a seguir. Note que na Figura 4.8b a escala de cinza foi invertida para destacar a ROI. Portanto, os pixels escuros indicam maior mudança de intensidade na imagem original.



(a) Imagem original



(b) Imagem do gradiente

Figura 4.8: Imagens do processo de obtenção do gradiente de uma imagem

O método proposto baseia-se em uma varredura da imagem do gradiente feita por janelas retangulares, buscando regiões de maior intensidade do gradiente, ou seja, regiões de pixels com maior valor numérico. Num dado ponto, a janela retangular calcula a média dos pixels daquela região, comparando sempre com a maior média já encontrada. Ao fim da varredura, a região com a maior média entre todas será a ROI, ou seja, a placa do automóvel. Obviamente, a porção será extraída da imagem original a partir das coordenadas obtidas pela imagem do gradiente. A Figura 4.9 é um esquema do processo de varredura da imagem.

Em qualquer etapa de análise de uma imagem, é indispensável realizar um pré-processamento de forma a evitar erros por má qualidade da imagem. O processo desenvolvido exige uma filtragem de suavização para retirar ruído e eliminar a possibilidade

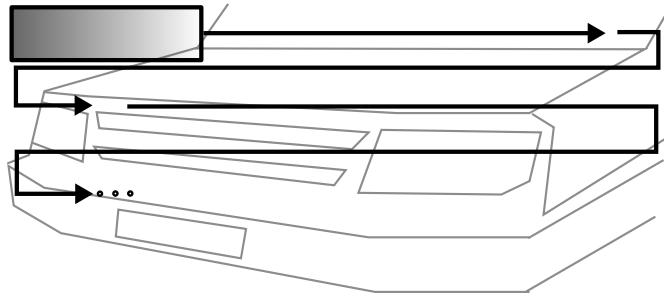


Figura 4.9: Esquema do processo de varredura da imagem por janelas

de detecção de elementos indesejados. A grade frontal presente em alguns modelos de carros é um exemplo de elemento indesejado. Estes, porém, costumam apresentar frequência de mudança de intensidade maior do que os caracteres da placa. O filtro de suavização aliado à operação morfológica de abertura¹⁰ promovem a eliminação dessas altíssimas frequências, fazendo com que a região de maior densidade de magnitude do gradiente seja a região da placa. O pseudo-código a seguir esquematiza o processo descrito nessa seção.

```

Receber Imagem_Monocromática
Imagem_Filtrada <= Correlação (Imagem_Monocromática , Máscara MxN)
Imagem_Filtrada <= Abertura (Imagem_Filtrada , Elemento PxQ)

Para L=1 até L=Ultima_Linha
  Para C=2 até C=Ultima_Coluna
    Imagem_Gradiente <= Imagem_Filtrada(L,C) - Imagem_Filtrada(L,C-1)
    Imagem_Gradiente <= Módulo(Imagen_Gradiente)
  Fim Para
Fim Para

S_Max <= Zero
Para L=1 até L=Ultima_Linha
  Para C=1 até C=Ultima_Coluna-Largura_Maior_Janela
    S <= Soma(Imagen_Gradiente(L , C : C+Largura_Maior_Janela))
    Se S > S_Max
      S_Max <= S
      L_Salvo <= L
    Fim Se
  Fim Para
Fim Para

```

¹⁰A operação de abertura em imagens em escala de cinza realiza operação semelhante à explicada na Seção 2.4.2, valorizando regiões maiores que o elemento estruturante e eliminando regiões menores.

```

L_Min <= L_Salvo - Altura_Maior_Janela
L_Max <= L_Salvo + Altura_Maior_Janela

```

As constantes N da máscara de filtro e M do elemento estruturante devem ser escolhidas convenientemente. O conhecimento prévio do tamanho dos caracteres na imagem ajuda em uma primeira estimativa (o elemento estruturante não deve ser mais largo que a largura da letra **I** e a máscara de filtro não pode ser grande a ponto de suavizar os próprios caracteres), mas o valor ideal só deve ser encontrado utilizando a técnica de tentativa e erro. Para as imagens dos bancos BD2, BD3, BD4 e BD5 utilizou-se $M \times N = 8 \times 8$ e $P \times Q = 7 \times 4$. Para o banco BD1, $M \times N = 5 \times 5$ e $P \times Q = 5 \times 5$.

No segundo laço duplo **Para** do pseudo-código anterior, os valores de L e C foram incrementados de 10 por passo. Dessa forma, houve grande ganho de velocidade de processamento sem prejudicar o procedimento, visto que esta é um etapa de localização de uma faixa que contém a ROI. Temos, portanto, uma faixa compreendida entre L_{min} e L_{max} . Nessa faixa faz-se uma varredura de várias janelas de diferentes tamanhos. A janela que apresentar maior média indicará a região escolhida. A porção a seguir indica a operação feita para cada janela.

```

S_Max <= Zero
M_Max <= Zero
Para L=L_Min até L=L_Max
  Para C=1 até C=Ultima_Coluna-Largura_Janela
    S <= Soma(Imagen_Gradients(L:L+Altura_Janela , C:C+Largura_Janela))
    Se S > S_Max
      S_Max <= S
      L_Salvo <= L
      C_Salvo <= C
    Fim Se
  Fim Para
Fim Para
M_Max <= S_Max/Largura_Janela

```

No laço **Para** descrito acima, os valores de L e C foram incrementados de 5 por passo. Neste momento, a busca realizada é mais fina que a busca na etapa anterior, em que se utilizou incremento de 10. Em ambos os casos estes valores foram ajustados à medida que o código era desenvolvido e testado, buscando um equilíbrio entre o sucesso na localização e a velocidade de processamento.

Após executar essa operação com todas as janelas, aquela que apresentar maior M_{max} será a escolhida e teremos

```
Imagen_Placa <= Imagen_Monocromática(Limites_Janela_Escolhida)
```

Quanto ao tamanho das janelas, sabe-se que vários tamanhos devem ser considerados, uma vez que a distância de obtenção da fotografia, inclinação e distorção da câmera fotográfica podem alterar os valores de altura e largura. Mas o conhecimento prévio dos bancos de imagens permite a restrição a um grupo pequeno de janelas que atendam às necessidades daquelas imagens sem prejudicar o programa no que diz respeito a velocidade de processamento.

Para o bancos BD1 as janelas utilizadas têm valores de altura $17px$, $19px$, $21px$, $23px$, $25px$, $27px$ e $29px$ combinados com valores de largura pelas proporções $1:9$, $1:10$, $1:11$ e $1:12$.

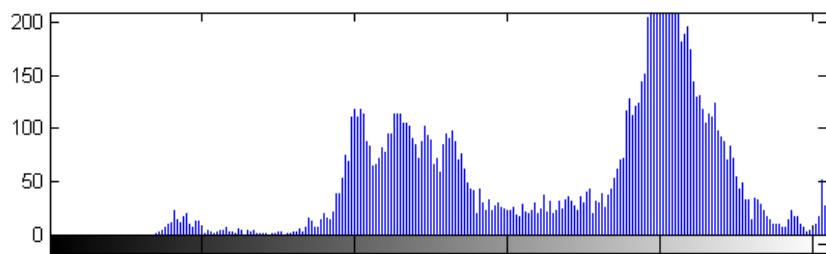
Para os bancos BD2, BD3, BD4 e BD5 as janelas utilizadas têm valores de altura $25px$, $27px$, $29px$, $31px$, $33px$, $35px$, $37px$, $39px$, $41px$, $43px$ e $45px$ combinados com valores de largura pelas proporções $1:4,5$ e $1:5$.

4.6.4 Processamento de imagem da placa

Neste momento, temos uma imagem composta pela região da placa. Dependendo do desempenho do algoritmo de LPL, teremos mais ou menos informações não-interessantes nesta imagem. De qualquer forma, espera-se que a imagem tenha característica semelhante à vista na Figura 4.10a e, portanto, terá histograma semelhante ao visto na Figura 4.10b.



(a) Placa localizada



(b) Histograma da placa

Figura 4.10: Resultado do processo de LPL e seu respectivo histograma de níveis de cinza

Apesar de não haver consenso a respeito do melhor momento para se realizar a binarização da imagem (existem trabalhos que o fazem como primeira etapa, outros guardam para o final do processo), é possível identificar quando o processo de binarização produzirá bons resultados e quando não os produzirá. Entendemos que neste ponto em que estamos, com a imagem da placa bem delimitada, com pouca informação de segundo plano e histograma de níveis de cinza com dois picos bem definidos, é um bom momento para

realizar a binarização da imagem.

A partir dessas condições para realização da binarização, um processo simples de binarização global pode ser aplicado. Usamos o método BGT (vide Seção 2.2.5) para definir o valor de limiar.

A maioria das placas brasileiras possui fundo claro e caracteres escuros. Dessa forma, o processo de binarização levará a caracteres pretos em um fundo branco. Convém inverter essa situação, pois é comum tratar os bits brancos como partes de objetos e os bits pretos como partes do fundo de uma imagem binária.

É preciso, no entanto, considerar as placas cuja característica é de fundo escuro para caracteres claros, como as placas de táxis e caminhões. Como é esperada maior quantidade de pixels de plano de fundo do que de pixels de objetos na imagem, a diferença entre a quantidade de pixels brancos e de pixels pretos definirá se a imagem deve ou não ser invertida.

O esquema a seguir mostra a entrada, o procedimento e a saída desse processo.

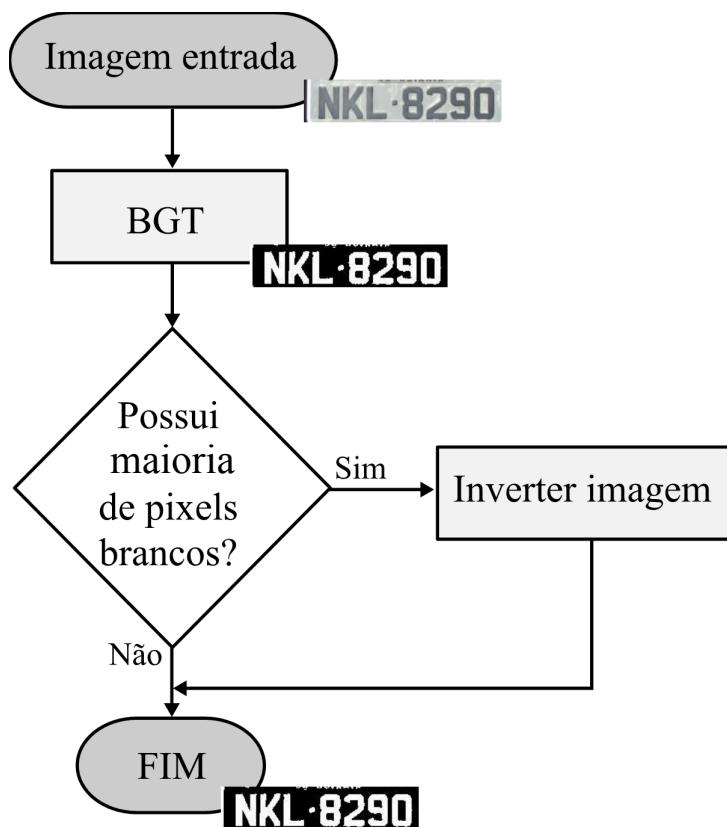


Figura 4.11: Esquema de procedimento de binarização da placa

Com a imagem binarizada, a análise é mais simples e alguns métodos interessantes poderão ser aplicados para processamento da imagem. É comum, como pode-se observar no bloco **FIM** da Figura 4.11, a imagem possuir pequenos objetos na parte superior, inferior ou mesmo na porção central da imagem. Podemos inclusive prever objetos referentes à municipalidade da placa na parte superior, o hífen separador dos grupos alfabético e

numérico na parte central e pequenos buracos utilizados para fixação da placa no veículo na parte inferior. De qualquer forma, é simples eliminar objetos que estejam somente na parte superior ou somente na parte inferior ou somente na parte central da imagem sem qualquer prejuízo aos sete caracteres alfanuméricos. Estes ocupam grande parte das linhas da imagem e não serão eliminados neste processo.

Para identificar os objetos, deve-se utilizar a técnica de rotulação de pixels conectados, descrita na Seção 2.5.1. Supondo a imagem com N objetos rotulados de 1 a N , sendo o fundo da imagem caracterizado por pixels de valor 0, basta testar cada objeto se ele pertence a somente uma dessas três porções, eliminando-o se este for o caso.

Veja o resultado desse procedimento nas figuras a seguir.

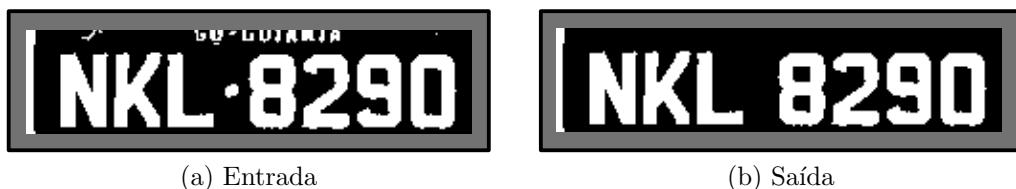


Figura 4.12: Processo de eliminação de pequenos objetos não-interessantes ao sistema

A partir de agora, trataremos por *objetos* os grupos de pixels conectados entre si por vizinhança-8, que são rotulados conforme já descrito.

4.6.5 Ajuste da inclinação

O ajuste de inclinação é praticamente a repetição de um processo já realizado: o de correção de rotação. Nesta etapa, o uso da HT auxilia na definição do ângulo de inclinação das retas verticais que definem grande parte dos caracteres alfanuméricos. Nesta etapa, serão identificadas as retas em direção aproximadamente vertical.

A correção da inclinação é feita utilizando a convolução da imagem pela matriz de transformação espacial de inclinação, apresentada na Seção 2.2.8. O valor do ângulo θ é obtido conforme o método já descrito na Seção 4.6.2.

4.6.6 Segmentação e separação de caracteres

A entrada para o processo de segmentação e separação de caracteres é uma imagem binária, com poucos objetos além dos caracteres, com rotação e inclinação já corrigidas. O processo de segmentação é constantemente citado entre os mais complexos em sistema de visão computacional. Neste trabalho, buscamos realizar as etapas anteriores com muito cuidado de forma a facilitar o processo de segmentação.

Os objetos da imagem serão analisados por heurísticas¹¹. Neste caso, o objetivo das

¹¹As pesquisas por heurísticas, de maneira geral, são aquelas realizadas por meio da quantificação de proximidade a um determinado objetivo.

heurísticas é eliminar todos os objetos que excedam ao grupo de 7 caracteres que interessam.

Num primeiro momento, a definição de uma altura esperada dos caracteres permite a eliminação de objetos que estejam fora da área central delimitada pelos caracteres. A Figura 4.13¹² mostra a idéia proposta.



Figura 4.13: Eliminação de objetos acima ou abaixo dos caracteres

Com os objetos restantes, aplica-se uma heurística para definir aqueles que melhor se encaixam nas características de um caractere de placa de automóvel. Serão realizados 4 testes com os caracteres, atribuindo valores a uma variável em cada teste. Quanto mais distante do valor esperado o objeto estiver, maior será o valor dessa variável. Ao final dos testes, as 4 variáveis serão somadas.

Realizando o procedimento com todos os objetos da imagem, aqueles 7 que apresentarem os menores valores ao fim dos testes serão definidos como os caracteres das placas.

Os testes consideram as seguintes características: **(a)** posicionamento do centróide, **(b)** altura do objeto, **(c)** proporção de largura por altura (*aspect ratio*) e **(d)** área do objeto.

Denotamos por H_A , H_B , H_C e H_D as variáveis referentes a cada teste e

$$H_{total} = H_A + H_B + H_C + H_D$$

Dessa forma, serão selecionados os 7 objetos de menor valor H_{total} . A expressão de H_{total} indica que as quatro variáveis H_N possuem pesos iguais. Obviamente, é feito um ajuste anterior nestas variáveis de forma a normalizar e atribuir-lhes pesos anteriormente. Por exemplo, a altura do objeto é fator mais determinante do que o posicionamento do centróide, pois este tem uma valor esperado menos constante do que a altura. Portanto a variável H_B deve ter maior peso do que a variável H_A . Quanto à normalização, é útil para que o algoritmo seja aplicável a qualquer tipo de entrada independente de seu tamanho. Dessa forma, uma diferença de altura entre objetos de $90px$ e $100px$ terá peso semelhante a uma diferença entre objetos de $9px$ e $10px$. A seguir os detalhes de cada teste e o peso atribuído à respectiva variável H_N .

- **Teste A** - Espera-se que o centróide de um objeto que corresponda a um caractere esteja na porção central da imagem, no sentido vertical. Portanto, neste teste

¹²As cores foram invertidas para facilitar a visualização.

obtém-se a diferença entre a posição do centróide e a metade da altura da imagem. Obviamente, interessa o módulo dessa diferença, normalizada pela altura da imagem. Atribuímos peso 50 a essa variável.

- **Teste B** - Obtendo a altura de todos os objetos da imagem, sabe-se que o valor da mediana da sequência de valores de altura corresponderá necessariamente à altura de um caractere. Isso pode ser afirmado porque são poucos objetos presentes na imagem além dos 7 de interesse. Dessa forma, essa altura será considerada como teste para cada caractere. Portanto, aqui faz-se o módulo da diferença entre a altura do objeto e a altura mediana, normalizada pela altura da imagem e com peso definido 80.
- **Teste C** - O *aspect ratio* de um objeto é definido pela divisão da largura pela altura. Entre os caracteres de uma placa de automóvel, a letra **I** ou o número **1** podem ser considerados aqueles de menor *aspect ratio* e letras como **M** ou **G** podem ser considerados os de maior *aspect ratio*. Deve-se estabelecer um valor mínimo e um valor máximo para *aspect ratio* e testar os objetos quanto a essa característica. Para os bancos BD2, BD3, BD4 e BD5, temos $AR_{min} = 0,15$ e $AR_{max} = 1,00$. Para o banco BD1, temos $AR_{min} = 0,50$ e $AR_{max} = 1,75$. Caso o objeto se enquadre entre AR_{min} e AR_{max} , temos $H_C = 0$. Caso contrário, temos:

$$\begin{aligned} - H_C &= X \cdot \frac{|AR_{min} - AR_{obj}|}{AR_{obj}} \text{ quando } AR_{obj} < AR_{min} \\ - H_C &= X \cdot \frac{|AR_{max} - AR_{obj}|}{AR_{obj}} \text{ quando } AR_{obj} > AR_{max} \end{aligned}$$

O valor de peso X foi utilizado para maximizar os valores daqueles objetos que apresentem *aspect ratio* muito distante do esperado. Estabeleceu-se, empiricamente:

$$\begin{aligned} - X &= 10 \text{ para } AR_{obj} < AR_{min} \\ - X &= 20 \text{ para } AR_{obj} < \frac{AR_{min}}{1,2} \\ - X &= 60 \text{ para } AR_{obj} < \frac{AR_{min}}{1,5} \\ - X &= 15 \text{ para } AR_{obj} > AR_{max} \\ - X &= 30 \text{ para } AR_{obj} > AR_{max} \cdot 1,2 \\ - X &= 90 \text{ para } AR_{obj} > AR_{max} \cdot 1,5 \end{aligned}$$

- **Teste D** - O último teste diz respeito à área dos objetos. Com o *aspect ratio* esperado dos caracteres e a altura mediana definida, temos

$$\text{Área} = \text{AspectRatio} \cdot \text{AlturaMediana}^2$$

Portando, pode-se considerar $A_{min} = AR_{min} \cdot AM^2$ e $A_{max} = AR_{max} \cdot AM^2$. Neste teste, estabeleceu-se, empiricamente:

- $H_D = 5$ para $A < A_{min}$ ou $A > A_{max}$
- $H_D = 10$ para $A < \frac{A_{min}}{1,5}$ ou $A > A_{max} \cdot 1,5$
- $H_D = 70$ para $A < \frac{A_{min}}{2,0}$ ou $A > A_{max} \cdot 2,0$

4.6.7 Reconhecimento de caracteres

O sistema de ANPR, até aqui, visa a preparar os caracteres para a execução da sua principal etapa: o reconhecimento dos caracteres. Após esta etapa, conclui-se um processo de transformação de uma imagem sem qualquer significado direto em uma simples sequência de valores, cada um representando um caractere alfanumérico. A partir deste ponto, esse simples código pode ser manipulado de acordo com a aplicação desejada, seja buscando em uma base de dados se refere-se a um carro roubado ou verificando se o motorista efetuou o pagamento da tarifa de estacionamento.

O processo de OCR abordado neste trabalho é muito simples: comparação direta da imagem obtida de um caractere com diversas imagens pertencentes a uma base de dados padrão. São as máscaras. Aquela com a qual a imagem mais se assemelhe indicará o caractere identificado.

As imagens utilizadas para comparação foram divididas em dois bancos, os quais chamaremos de Banco de Máscaras 1 e Banco de Máscaras 2 (BM1 e BM2). Já vimos na Seção 3.4 que é regra no Brasil a utilização da fonte *Mandatory* para placas de automóveis. O BM1 é formado por imagens de todos os 10 caracteres numéricos e 26 letras do alfabeto escritos com esta fonte. Já foi visto também que antes dessa padronização não havia uma fonte definida para se utilizar em placas brasileiras, mas era comum o uso da fonte *DIN Mittelschrift*. O banco BM2 é formado por imagens dos 36 caracteres alfanuméricos utilizando esta segunda fonte.

A análise visual das imagens dos bancos BD1, BD2, BD3, BD4 e BD5 não acusa nenhum outro tipo de fonte utilizada para emplacamento, embora não seja impossível que alguns casos ocorram. No entanto, a decisão acertada de impor regras a respeito da fonte, tamanho, cores, indica que a tendência é que os sistemas de ANPR sejam cada vez mais eficientes também por este motivo.

A comparação entre o caractere obtida na entrada e as máscaras é feita pelo processo de Correlação 2-D, também conhecido como *Template Matching*. É um processo simples, rápido e eficiente, além de ser muito utilizado em diversos ramos da visão computacional.

A melhor forma de exemplificar o processo de OCR deste trabalho é enunciando-o em uma sequência de passos, descritos a seguir.

1. Carregar as máscaras de BM1 e BM2 (As imagens tem o tamanho $50 \times 78px$);

2. Carregar o objeto a ser reconhecido;
3. Escalonar o tamanho do objeto para $78px$ de altura;
4. Completar as laterais do objeto ou escalar a largura para $50px$;
5. Aplicar a função *Template Matching* entre objeto e máscara;
6. Armazenar valor resultante da operação *Template Matching*;
7. Repetir os passos 5 e 6 com todas as máscaras do BM1 e BM2;
8. Escolher a máscara que apresentar maior valor resultante de *Template Matching*;
9. Relacionar a máscara escolhida a um código do caractere alfanumérico;
10. Salvar código em uma sequência-resposta;
11. Repetir do passo 2 ao 10 para todos os objetos da imagem.

Cabe agora detalhar o processo de correlação 2-D, ou *Template Matching*, utilizado para comparação das imagens. O método consiste em basicamente calcular o Coeficiente de Correlação de Pearson (CCP). Este coeficiente é uma medida estatística para comparação linear entre dois conjuntos de dados. Seus valores variam de -1 a 1, sendo o valor positivo indicando correlação perfeita positiva e o valor negativo indicando correlação perfeita negativa.

Analizar de forma bidimensional, como é o caso da comparação de duas imagens, nada mais é do que considerar a matriz como um grande vetor, e o coeficiente retornará 1 quando as imagens forem iguais e -1 quando forem exatamente opostas.

O CCP, denotado por ρ , entre dois vetores X e Y é:

$$\rho = \frac{\sum (x_i - \bar{x}) \cdot (y_i - \bar{y})}{\sum (x_i - \bar{x})^2 \cdot \sum (y_i - \bar{y})^2}$$

Nota-se que o CCP é a divisão da covariância de X e Y pela raiz quadrada da multiplicação das variâncias de X e Y . Em Sousa (2010), a autora apresenta uma tabela de análise da correlação entre variáveis pelo valor do CCP. Veja a Tabela 4.3.

Esse tipo de análise, porém, não é aplicável ao nosso trabalho. Buscamos aqui definir uma única máscara para dado objeto. Em Filho e Júnior (2009), os autores apresentam classificações de outros autores (Veja a Tabela 4.4), mas são enfáticos em sua declaração: “Seja como for, o certo é que quanto mais perto de 1 (independente do sinal) maior é o grau de dependência estatística linear entre as variáveis. No outro oposto, quanto mais próximo de zero, menor é a força dessa relação”.

Os gráficos das Figuras 4.14a, 4.14b e 4.14c são exemplos de correlações positiva, nula e negativa entre duas variáveis, num sentido estatístico geral. Nota-se que valores iguais,

Tabela 4.3: Análise de correlação de acordo com o CCP apresentada por Áurea Sousa

Coeficiente	Correlação
$ \rho = 1$	Perfeita
$0,8 \leq \rho < 1,0$	Forte
$0,5 \leq \rho < 0,8$	Moderada
$0,1 \leq \rho < 0,5$	Fraca
$0,0 < \rho < 0,1$	Ínfima
$\rho = 0$	Nula

Tabela 4.4: Outras análises de correlação de acordo com o CCP

Cohen, 1998		Dancey e Reidy, 2005	
Coeficiente	Correlação	Coeficiente	Correlação
$ \rho = 0,1$ até $0,3$	Fraca	$ \rho = 0,10$ até $0,29$	Pequena
$ \rho = 0,4$ até $0,6$	Moderada	$ \rho = 0,30$ até $0,49$	Média
$ \rho = 0,7$ até 1	Forte	$ \rho = 0,50$ até 1	Grande

ou próximos, de x e y levam a uma correlação positiva dos dados. É o caso de uma análise entre um caractere e sua máscara correspondente.

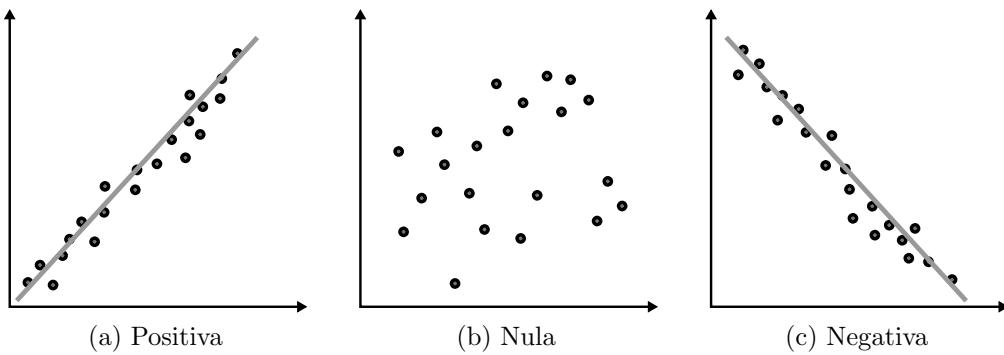


Figura 4.14: Exemplificação de correlações entre conjuntos de dados

O cálculo do CCP entre as imagens exige que elas tenham o mesmo tamanho e o resultado apresentado não é invariante a escala ou rotação. O desenvolvimento de um algoritmo que obedeça a essas condições de invariância é complexo e computacionalmente mais dispendioso. Neste trabalho, no entanto, houve um esforço de correção de rotação, inclinação e escala de tal forma que não se faz necessário considerar tais condições. A simples aplicação do método do CCP é suficiente para definir a melhor máscara para dado objeto.

Vale ressaltar que em algumas aplicações estatísticas, a correlação negativa indica maior grau de dependência estatística entre as variáveis, mas os pontos correlatos formariam um reta descendente. No caso de análise das imagens, porém, valores negativos indicam grau de similaridade entre as imagens menor do que um valor nulo. Portanto, interessa apenas a análise de valores positivos de correlação.

4.6.8 Obtenção dos dados de saída

Após a execução de todas as etapas descritas ao longo deste capítulo, basta associar a resposta do OCR a um código. Assim como a grande maioria dos trabalhos, associamos cada caractere ao código ASCII (*American Standard Code for Information Interchange*) correspondente. O processo de ANPR é, agora, um “simples” módulo de um sistema maior, uma caixa preta¹³, cuja entrada é uma imagem e cuja saída é um sequência de 7 bytes contendo o código ASCII da placa do automóvel.

¹³Caixa preta é um termo usado em computação para designar um programa em que interessa ao programador somente suas entradas e suas saídas. Não importam os procedimentos e funções executadas, mas apenas os resultados que fornecidos.

Capítulo 5

RESULTADOS E CONCLUSÃO

Neste capítulo apresentaremos os resultados obtidos em cada uma das três etapas principais de um processo de ANPR. É importante comparar e discutir resultados, analisar os pontos positivos e negativos dos algoritmos apresentados. Ao fim do capítulo, faremos uma discussão geral do trabalho desenvolvido e apresentaremos sugestões de trabalhos futuros.

5.1 Resultados de Localização

A análise do algoritmo de Localização consiste em confirmar se as regiões que contém a placa foram devidamente extraídas da imagem original. Não faremos análise da precisão dos resultados, mas apenas definição entre resultado positivo e negativo.

A Tabela 5.5 apresenta os resultados¹⁴ de todos os bancos de dados analisados.

Tabela 5.5: Resultados do processo de Localização

Banco de imagens	Quantidade	LPL positiva	Taxa de acerto
BD1	75	73	97,3%
BD2	79	78	98,7%
BD3	127	116	91,3%
BD4	17	14	82,3%
BD5	17	13	76,5%

Apesar da má qualidade das imagens do BD1, foram obtidos ótimos resultados. Os resultados apresentados pelo LPDSI indicam acerto de localização em 74 das 75 imagens utilizados neste trabalho. A taxa de 97,3% alcançada neste trabalho é extremamente positiva, visto que foi utilizado um método simples e computacionalmente rápido.

Os resultados referentes aos outros bancos de imagens não podem ser comparados, mas as taxas de acerto acima de 90% para BD2 e BD3 indicam um enorme sucesso, com destaque para os quase 100% no segundo teste. As taxas mais baixas nos bancos BD4 e BD5 indicam algumas falhas no processo de correção de rotação, extremamente

¹⁴Os resultados foram apresentados considerando LPL positiva quando a análise visual das imagens indicasse a predominância da placa do automóvel na região definida, estando ela inteira e bem centralizada.

importante para essas imagens. O pequeno número de imagens para testes, no entanto, não permite conclusão definitiva dos resultados destes bancos.

As imagens do banco BD1 foram obtidas por uma câmera fixa sob uma cabine de pedágio no Rio de Janeiro, o que é um grande trunfo pelo fato de minimizar efeitos negativos de iluminação e condições climáticas. Já os outros bancos de dados são de fotografias obtidas durante a tarde, sem qualquer cobertura ou posição fixa da câmera. Pode-se notar que os efeitos de sombras de árvores são prejudiciais ao processo de ANPR desde o começo. Alguns dos erros da etapa de LPL são atribuídos a este fator.

Outros erros ocorrem devido a objetos que desviam o foco do algoritmo, como adesivos colados ao carro e grades frontais com características semelhantes às das placas. De qualquer forma, os resultados apresentados com o uso do filtro de média e da operação morfológica de abertura são animadores no sentido de eliminar essa confusão no processo de LPL.

5.2 Resultados de Segmentação

A análise dos resultados de Segmentação será feita de duas maneiras: **(a)** taxa de acerto por placa inteira e **(b)** taxa de acerto por caractere.

Na primeira análise (Tabela 5.6), a placa inteira será considerada um acerto quando for composta, ao fim do processo de Segmentação, por 7 objetos referentes aos caracteres alfanuméricos bem definidos e separados. Para a segunda análise (Tabela 5.7), cada objeto será analisado, considerando como o total de objetos 7 vezes o número de placas localizadas na etapa anterior.

Vale ressaltar que a análise será feita a partir apenas das placas localizadas na etapa anterior, uma vez que seria impossível testar a segmentação de objetos que nem mesmo se referem a caracteres das placas de automóveis.

Veja a seguir as tabelas de resultados¹⁵.

Tabela 5.6: Resultados do processo de Segmentação por placa inteira

Banco de imagens	Quantidade	Segmentação positiva	Taxa de acerto
BD1	73	53	72,6%
BD2	77	67	87,0%
BD3	116	102	87,9%
BD4	14	11	78,6%
BD5	13	9	69,2%

Interessa mais, de acordo com os objetivos do presente trabalho, a análise por caractere. Num sistema prático, não interessaria a taxa de acerto por objeto, mas sim por placa, já

¹⁵Os resultados foram apresentados considerando segmentação positiva quando a análise visual das imagens indicasse o objeto de saída como bem definido, livre de ruído e separado dos demais objetos.

Tabela 5.7: Resultados do processo de Segmentação por caractere

Banco de imagens	Quantidade	Segmentação positiva	Taxa de acerto
BD1	511	455	89,0%
BD2	539	493	91,5%
BD3	812	774	95,3%
BD4	98	85	86,7%
BD5	91	78	85,7%

que o reconhecimento adequado de parte da placa do automóvel não tem grande utilidade prática. O trabalho, no entanto, se concentra na análise de procedimentos e algoritmos, buscando identificar os melhores rumos para pesquisas de ANPR.

Os resultados do BD3 são os mais animadores. É o maior banco de imagens utilizados no trabalho e apresentou taxa de acerto de 88% para placas inteiras e 95% na análise por objeto. As imagens que o compõe não foram utilizadas para testes intermediários no desenvolvimento do algoritmo, o que de certa forma confere maior validade aos resultados obtidos.

Pode-se ver que as taxas de acerto dos bancos BD4 e BD5 se aproximam das taxas dos bancos BD2 e BD3, como esperado, já que os resultados negativos do processo de correção de rotação não têm influência nesta etapa.

5.3 Resultados de Reconhecimento

O processo de Reconhecimento desenvolvido neste trabalho é um dos mais simples métodos existentes. Consiste em calcular o Coeficiente de Correlação de Pearson, que define o grau de correlação entre duas variáveis.

Os resultados, no entanto, são satisfatórios. Veja a Tabela 5.8. Note que aqui analisamos somente a taxa de acerto por objeto, uma vez que a análise por placa inteira seria prejudicada pela quantidade de imagens que seriam descartadas devido a insucesso na etapa anterior.

Tabela 5.8: Resultados do processo de Reconhecimento por caractere

Banco de imagens	Quantidade	Reconhecimento positivo	Taxa de acerto
BD1	455	428	94,1%
BD2	493	460	93,3%
BD3	774	724	93,5%
BD4	85	77	90,6%
BD5	78	70	89,7%

As altas taxas de acerto do algoritmo indicam, mais do que a qualidade do processo de Reconhecimento, a qualidade das etapas anteriores ao Reconhecimento. O método de

Template Matching, ou Correlação 2-D, não é invariante a escala de rotação e altamente sensível a pequenos desvios da reta indicativa da correlação. Em Filho e Júnior (2009), os autores afirmam que o CCP é fortemente afetado pela presença de *outliers*, que são pontos aleatórios fora da curva esperada. Veja a exemplificação de um *outlier* na Figura 5.1. O segmento sólido indica a provável reta obtida com a influência do *outlier* e o segmento tracejado indica a provável reta sem a sua influência. Analisando graficamente, diz-se que quanto maior é a distância da reta em relação às regiões mais densas de pontos, menor é o coeficiente de correlação entre os eles.

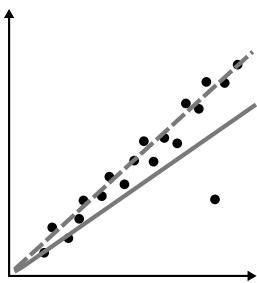


Figura 5.1: Representação simples de um outlier: o ponto fora da curva

Portanto, um caractere pode ter seu reconhecimento prejudicado por mais que ocorram apenas pequenos desvios em relação à sua máscara correspondente. As altas taxas de acerto do processo de Reconhecimento indicam, portanto, que as etapas de Localização e Segmentação foram bem executadas.

5.4 Resultados de processamento computacional

Não é objetivo deste trabalho obter resultados ótimos de velocidade de processamento. A preocupação se restringe a garantir rapidez de processamento do sistema que permita a análise em tempo real da placa do automóvel. Para obter resultados de processamento do programa, realizou-se cinco execuções do mesmo código e obtivemos os melhores e piores tempos de execução, além de um tempo médio.

Dividimos o sistema em quatro etapas para análise dos tempos: **(a)** carregamento de informações de entrada, **(b)** localização e extração da placa, **(c)** segmentação da imagem e separação dos caracteres e **(d)** reconhecimento dos caracteres e fornecimento da saída.

A Tabela 5.9 apresenta os resultados obtidos nos testes¹⁶.

O tempo médio total de execução de aproximadamente 2,2 segundos foi considerado bom. O desenvolvimento do sistema utilizando técnicas de melhoria de processamento e em linguagem compilada, como a linguagem C, deve melhorar significativamente o tempo de execução deste processo. Esse não foi o foco deste trabalho, mas deve-se reconhecer

¹⁶Utilizou-se a função *tic toc* do MATLAB para medição do tempo. Apesar de ser uma análise relativamente grosseira, já permite uma visão geral, que é o nosso objetivo, do tempo de processamento do programa.

Tabela 5.9: Tempos de execução, em segundos, do sistema de ANPR

Carregamento	Localização	Segmentação	Reconhecimento	Total
0,628071	1,413346	0,066714	0,178098	
0,681922	1,358259	0,038400	0,147666	
0,644210	1,442390	0,030106	0,145747	
0,614962	1,407800	0,030967	0,134348	
0,634131	1,343069	0,028873	0,141188	
$t_{médio}$	0,640659	1,392973	0,039012	0,149409
				2,222053

a importância desse tipo de preocupação para desenvolvimento de um sistema prático de ANPR.

5.5 Conclusão

Os objetivos deste trabalho foram alcançados com sucesso. No Capítulo 3, apresentamos diversas pesquisas de visão computacional com foco em sistemas de ANPR, discutindo resultados e tendências de inovações neste ramo. No Capítulo 4, desenvolvemos algoritmos de um sistema de ANPR pautado no equilíbrio simplicidade-eficiência. No Capítulo 5, apresentamos os resultados satisfatórios obtidos com os algoritmos propostos para processos de Localização, Segmentação e Reconhecimento.

Os resultados combinados das três etapas são apresentados na Tabela 5.10.

Tabela 5.10: Resultados do sistema de ANPR por caractere

Banco de imagens	Quantidade	Resultados positivos	Taxa de acerto
BD1	525	428	81,5%
BD2	553	460	83,2%
BD3	889	724	81,4%
BD4	119	77	64,7%
BD5	119	70	58,8%

A taxa de acerto acima de 80% é considerada um sucesso. Um sistema prático de ANPR deve atingir um taxa de acerto acima de 95%, buscando 99%. Não há dúvidas de que a busca pelos 95% demandaria a “reengenharia”¹⁷ do sistema proposto.

É preciso ressaltar, porém, que em termos de pesquisa os resultados são satisfatórios, destacando o fato de que grande parte das dificuldades encontradas estão relacionadas às imagens de entrada. Um sistema prático de ANPR demandaria o planejamento de uma forma de eliminar problemas de iluminação das fotografias. Percebemos que as taxas de

¹⁷Reengenharia é um conceito introduzido por James Champy e Michael Hammer (1993) e consiste em repensar e redesenhar radicalmente as práticas e processos nucleares da organização.

acerto do BD1 são próximas às taxas dos outros bancos, mesmo com a utilização de imagens visivelmente de menor qualidade. As imagens do BD1 não apresentam, no entanto, os problemas de iluminação e sombra que as imagens dos outros bancos apresentam. A simples combinação de qualidade de fotografia e planejamento de como obter as imagens, eliminando efeitos de sombra, deve produzir resultados extremamente satisfatórios, próximos à taxa de 95%. Fazemos essa afirmação pela percepção, ao longo do desenvolvimento do sistema, dos resultados acima da média com imagens de boa qualidade sem problemas de iluminação.

Não só sistemas de ANPR, mas a Visão Computacional como um todo, é um ramo de pesquisa ainda em forte crescimento e que constantemente apresenta inovações incríveis em diversas pesquisas por todo o mundo. O resultado final deste projeto é encorajador, visto que o desenvolvimento de um sistema prático e totalmente aplicável no mundo atual acontece de forma relativamente simples. Determinação, organização e certa dose de criatividade são importantes para desenvolver ferramentas aplicáveis em visão computacional e o incentivo a novos pesquisadores nesse ramo é importantíssimo.

5.6 Trabalhos futuros

As conlussões acerca do trabalho desenvolvido sugerem duas vertentes para trabalhos futuros: **(a)** quanto à aplicabilidade do sistema e **(b)** quanto à eficiência dos algoritmos.

Trabalhos futuros devem estabelecer um foco na qualidade das imagens obtidas para o processo de Reconhecimento Automático de Placas de Automóveis. A engenharia deve ser pensada como algo integrado, e a evolução dos equipamentos de fotografia digital deve ser aproveitada pelos pesquisadores. Não se deve utilizar imagens de má qualidade para desenvolver um sistema de ANPR, o que demanda grande esforço de pré-processamento das imagens, se é possível obter imagens com câmeras fotográficas digitais de qualidade por um custo mínimo. Não podemos ignorar tecnologias, devemos aproveitá-las.

Trabalhos com foco em desenvolvimento de sistemas práticos devem também estabelecer como objetivo velocidade de processamento. Este trabalho foi desenvolvido com a ferramenta MATLAB por motivos de facilidade para testes e modificações, mas um sistema prático exige o uso de uma linguagem de programação mais sofisticada em termos de processamento, como a linguagem C, além da preocupação com o *hardware* utilizado.

Os trabalhos de pesquisa futuros, no que concerne à eficiência dos algoritmos aqui apresentados, devem ter seu foco na questão de Inteligência Artificial. É importantíssimo desenvolver algoritmos que utilizem dos conceitos revolucionários desse ramo da ciência que, basicamente, busca modelos computacionais que simulem a capacidade do ser humano de processar informações, ou seja, objetiva ensinar o computador a pensar. O uso de Redes Neurais Artificiais tem sido primordial nas pesquisas recentes em ANPR.

Uma das falhas mais sentidas no presente trabalho é a ausência de realimentação dos

algoritmos. Por exemplo, uma região que seja erroneamente identificada como a ROI no processo de LPL certamente apresentará problemas durante a segmentação dos caracteres. Seria conveniente, neste caso, refazer o processo de LPL para aquela imagem assim que o erro fosse identificado na etapa seguinte, alterando parâmetros de filtragem ou operação morfológica que levassem à correta extração da placa e chance de sucesso no processo de segmentação. A Figura 5.2 é um esquema do processo proposto.

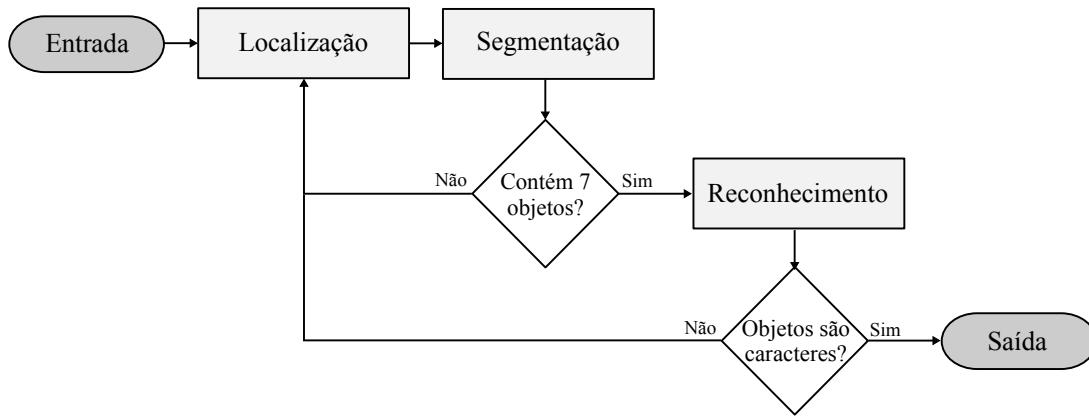


Figura 5.2: Esquema proposto de realimentação do algoritmo

O desenvolvimento de formas eficientes de análise dessa realimentação pode levar a consideráveis melhorias nos algoritmos, sem fugir do foco de simplicidade-eficiência proposto.

Por fim, constatamos que é pouco utilizada em pesquisas de ANPR as cores nas imagens. Apesar de os resultados com imagens em escala de cinza serem interessantes, a análise das imagens coloridas amplia o leque de possibilidades de abordagens para o problema. É preciso avançar com as pesquisas nesse sentido.

Pode-se concluir que o grande avanço das pesquisas em ANPR se dará sempre com o uso de diversas tecnologias para obtenção de um bom resultado. A engenharia necessita dessa visão sistêmica, da integração de diversos conhecimentos, para avançar a passos largos.

REFERÊNCIAS¹⁸

- ANAGNOSTOPOULOS, C. N. et al. **A license plate recognition algorithm for Intelligent Transportation System applications.** 16 p. — Cultural Technology & Communication Dpt., University of the Aegean, Mytilene, 2005.
- ARAÚJO, V. P. de et al. **Automatic Plate Detectin Using Genetic Algorithm.** 6 p. — Department of Computer Science, State University of Montes Claros, Montes Claros, 2006.
- BELVISI, Ricardo et al. **Um sistema de reconhecimento automático de placas de automóveis.** 3 p. — Departamento de Engenharia Elétrica, Faculdade de Engenharia Industrial, São Bernardo do Campo, 1999.
- BRADSKI, Gary; KAEHLER, Adrian. **Learning OpenCV.** First edition. Sebastopol, CA: O'Reilly Media, 2008.
- BRANDÃO, T.; SEQUEIRA, M. M.; ALBUQUERQUE, M. **Multistage morphology-based license-plate location algorithm.** 4 p. — Instituto Universitário de Lisboa, Lisboa, 2004.
- CAMPOS, Tatiane Jesus de. **Reconhecimento de Caracteres Alfanuméricas de Placas em Imagens de Veículos.** 120 f. Monografia (Mestrado) — Instituto de Informática, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2001.
- CHANG, Shyang-Lih et al. **Automatic License Plate Recognition.** 12 p. — Dept. of Information and Computer Education, National Taiwan Normal Institute, Taipei, 2002.
- CONCI, Aura; MONTEIRO, L. H. **Reconhecimento de placas de veículos por imagens.** 12 p. — Universidade Federal Fluminense, Niterói, 2004.
- CONTRAN, Conselho Nacional de Trânsito. **Resolução 231.** 11 p., Brasil, 2007.
- DRAGHICI, Sorin. **A neural network based artificial vision system for licence plate recognition.** 21 f. — Dept. of Computer Science, Wayne State University, Detroit, 2007.
- FILHO, D. B. Figueiredo; JÚNIOR, J. A. Silva. **Desvendando os Mistérios do Coeficiente de Correlação de Pearson.** 32 p. — Universidade Federal do Pernambuco, Recife, 2009.
- GANAPATHY, V.; LUI, W. L. Dennis. **Automatic License Plate Recognition.** 8 p., 2008.

¹⁸Elaborado de acordo com a ABNT, sob a norma NBR 6023.

GONZALEZ, R. C.; WOODS, R. E. **Digital Image Processing**. Third edition. New Jersey: Pearson Prentice Hall, 2008.

GONZALEZ, R. C.; WOODS, R. E.; EDDINS, S. L. **Digital Image Processing Using MATLAB**. Second edition. USA: Gatesmark, 2009.

GUINGO, B. C.; THOMÉ, A. C. G.; RODRIGUES, R. J. **Reconhecimento automático de placas de veículos automotores através de redes neurais artificiais**. 12 p. — Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2002.

JORGE, Guilherme Henrique Renó. **Arquitetura para extração de características invariantes em imagens binárias utilizando dispositivos de lógica programável complexa**. 147 f. Monografia (Mestrado) — Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2006.

KHALIFA, O. et al. **Malaysian Vehicle License Plate Recognition**. 6 p. — Kulliyyah of Engineering, International Islamic University, Malaysia, 2006.

MAHINI, Hamid et al. **An Efficient Features-Based License Plate Localization Method**. 4 p. — Computer Engineering Department, Sharif University of Technology, Tehran, 2006.

MARTINSKY, Ondrej. **Algorithmic and Mathematical Principles of Automatic Number Plate Recognition Systems**. 76 f. Monografia (Tese) — Faculty of Information Technology, Brno University of Technology, Brno, 2007.

POLIDÓRIO, A. M.; BORGES, D. L. **Um Método de Reconhecimento Sintático de Caracteres para Identificação de Placas de Veículos**. 7 p. — Universidade Estadual de Maringá e Universidade Federal de Goiás, 1996.

RODRIGUES, Fabrício Augusto. **Localização e Reconhecimento de Placas de Sinalização Utilizando um Mecanismo de Atenção Visual e Redes Neurais Articiais**. 124 f. Monografia (Mestrado) — Universidade Federal de Campina Grande, Campina Grande, 2002.

SANCHO, X. G. **A Simple License Plate Recognition System for Spanish License Plates**. 9 p. — Universitat Rovira i Virgili, Tarragona, 2006.

SOUSA Áurea. **Coeficiente de Correlação Linear de Pearson**. 10 f. — Departamento de Matemática, Universidade dos Açores, Açores, Portugal, 2010.

ZHENG, Lihong et al. **Number Plate Recognition without Segmentation**. 5 p. — Faculty of Information Technology, University of Technology, Sydney, 2007.