

UNIVERSIDADE FEDERAL DE SÃO CARLOS
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

**UMA ABORDAGEM OTIMIZADA PARA
RECONHECIMENTO DE LOGOMARCAS DE
VEÍCULOS USANDO SURF**

CRISTIANO MACEDO

ORIENTADOR: PROF. DR. MARCIO MERINO FERNANDES

São Carlos - SP
Dezembro/2015

UNIVERSIDADE FEDERAL DE SÃO CARLOS
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

**UMA ABORDAGEM OTIMIZADA PARA
RECONHECIMENTO DE LOGOMARCAS DE
VEÍCULOS USANDO SURF**

CRISTIANO MACEDO

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de São Carlos, como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação, área de concentração: Processamento de Imagens e Sinais: Algoritmos e Arquitetura.
Orientador: Prof. Dr. Marcio Merino Fernandes.

São Carlos - SP
Dezembro/2015

Ficha catalográfica elaborada pelo DePT da Biblioteca Comunitária UFSCar
Processamento Técnico
com os dados fornecidos pelo(a) autor(a)

M141a Macedo, Cristiano
Uma abordagem otimizada para reconhecimento de
logomarcas de veículos usando SURF / Cristiano
Macedo. -- São Carlos : UFSCar, 2016.
108 p.

Dissertação (Mestrado) -- Universidade Federal de
São Carlos, 2015.

1. Processamento de Imagens. 2. SURF. 3.
Reconhecimento de logomarcas de veículos. I. Título.

Universidade Federal de São Carlos

Centro de Ciências Exatas e de Tecnologia

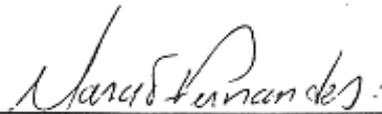
Programa de Pós-Graduação em Ciência da Computação

“Uma Abordagem Otimizada para Reconhecimento de Logomarcas de Veículos usando SURF”

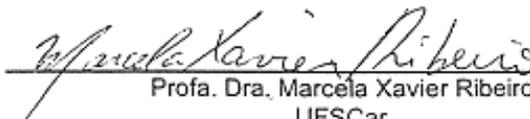
Cristiano Macedo

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de São Carlos, como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação.

Membros da Banca:



Prof. Dr. Márcio Merino Fernandes
UFSCar



Profa. Dra. Marcela Xavier Ribeiro
UFSCar



Prof. Dr. Joaquim Cezar Felipe
USP

São Carlos - SP
Dezembro/2015

Dedico este trabalho a minha família, amigos, professores e sobre tudo a Deus.

AGRADECIMENTOS

Agradeço a Deus primeiramente por me dar força para lutar para que eu pudesse concluir mais uma etapa importante da minha vida.

Em memória de meu Pai, Sr. Rubens Macedo (1954-2004).

Ao Prof. Dr. Marcio Merino Fernandes, meu orientador, por ter me escolhido, acreditado em mim, pelos ensinamentos e orientação.

A minha Mãe, Sra. Aparecida Tereza Batista Macedo.

Ao meu Tio, Sr. Pedro Macedo e família.

A minha Irmã, Sra. Patrícia Aparecida Macedo Ribeiro e família.

Ao meu Irmão, Sr. André Luiz Macedo e ao meu Tio, Sr. Antônio Batista Aparecido.

A minha namorada, Sra. Edna Cristina Moreira Bardeli.

Ao Sr. Antônio José Cerri e família por me receberem como um membro da família, pelos ensinamentos e amizade.

Ao Sr. João Pereira da Costa e família pelas orações e amizade.

A Pra. Maria Aparecida Gomes pelas orações e amizade.

Aos professores e membros da minha banca de qualificação.

Em fim, a todos meus familiares e amigos por terem acompanhado toda minha trajetória.

“A persistência é o caminho do êxito.” (Charles Chaplin).

RESUMO

Executar o reconhecimento rápido e preciso de objetos ainda é uma tarefa desafiadora em muitas aplicações. Nesse sentido as técnicas baseadas em detecção e descrição de características de imagens e em correspondência (matching) tem se configurado como uma importante evolução tecnológica no desenvolvimento de sistemas de reconhecimento de objetos. O procedimento comum da maioria dos sistemas de reconhecimento de objetos é que, dado algum conhecimento sobre a aparência de certos objetos, uma ou mais imagens são examinadas para avaliar se o objeto está presente e a localização do mesmo. Reconhecimento de objetos planos, como imagens e logomarcas insere-se nesse contexto bem como as abordagens atualmente propostas para o problema de reconhecimento de logomarcas de veículos. Este problema tem sido um dos temas de domínio em soluções de identificação de veículos. Porém realizar o reconhecimento de logomarcas de veículos em imagens de forma rápida mantendo a precisão do método empregado ainda é um problema. Por outro lado há também o problema de viabilizar a aplicação do método Speeded Up Robust Features (SURF) em soluções de reconhecimento de objetos. Este método possibilita identificar pontos de interesse em imagens de forma rápida e robusta. O presente trabalho propõe o desenvolvimento de uma abordagem baseada em técnicas de processamento de imagens e visão computacional para realizar o reconhecimento de logomarcas de veículos. O método proposto faz uso do algoritmo SURF para realizar o reconhecimento de logomarcas de veículos, combinando o algoritmo com outras técnicas, tais como o método de Canny, Conci, Carvalho e Rauber, ROI, KNN e RANSAC para solução do problema, realizando o reconhecimento de logomarcas de veículos entre 45 % a 100 % das imagens analisadas em um bom tempo de execução. O desenvolvimento deste trabalho de pesquisa possibilitou demonstrar uma opção alternativa para o problema de reconhecimento de logomarcas de veículos através de um modelo de utilização eficiente do método SURF.

Palavras-chave: Processamento de Imagens, SURF, Reconhecimento de Logomarcas de Veículos.

ABSTRACT

Perform fast and accurate recognition of objects is still a challenging task in many applications. In this sense the techniques based on detection and description of image features and in correspondence (matching) has been set up as an important technological evolution in the development of object recognition systems. The most common procedure for object recognition systems is that, given some knowledge about the appearance of some objects, one or more images are examined to determine whether the object is present and the location thereof. Recognition of flat objects, such as images and logos is part of that context today and the approaches proposed for vehicle logos recognition problem. This problem has been one of the domain issues in automatic identification solutions. But realize the recognition of logos vehicles in quick way to images while maintaining the precision of the method employed is still a problem. On the other hand there is also the problem of enabling the application of the method speeded Up Robust Features (SURF) in recognition solutions objects. This method makes it possible to identify points of interest in images quickly and robustly. This paper proposes the development of an approach based on image processing and computer vision techniques to carry out reconnaissance vehicles of logos. The proposed method makes use of the SURF algorithm to perform recognition logos vehicles, combining algorithm with other techniques, such as the Canny method, Conci, Carvalho and Rauber, ROI, KNN and RANSAC to solve the problem, performing recognition the logos of vehicles from 45% to 100% of the images analyzed in a good runtime. The development of this research demonstrate a possible alternative choice for the logos of recognition problem of vehicles through efficient use of a model SURF method.

Keywords: Image Processing, SURF, Logos Recognition of Vehicles.

LISTA DE FIGURAS

Figura 2.1: Representação de uma imagem digital. Extraído de (GONZALEZ; WOODS, 2000).	21
Figura 2.2: Vizinhança-4.	22
Figura 2.3: Vizinhança-8.....	22
Figura 2.4: Máscaras de Sobel na direção x (esquerda) e direção y (direita). Extraído de (PEDRINI; SCHWARTZ, 2008).	25
Figura 2.5: (a) imagem original; (b) aplicação do operador de Sobel para obtenção de G_x ; (c) para obtenção de G_y ; (d) magnitude do gradiente da imagem obtido pela combinação dos resultados de (b) e (c). Extraído de (PEDRINI; SCHWARTZ, 2008).	25
Figura 2.6: (a) imagem original; (b) aplicação do operador de Canny. Extraído de (PEDRINI; SCHWARTZ, 2008).	27
Figura 2.7: Exemplos de elementos estruturantes. Extraído de (GONZALEZ; WOODS, 2010).	29
Figura 2.8: Exemplo de erosão. Extraído de(GONZALEZ; WOODS, 2010).....	30
Figura 2.9: Exemplo de dilatação. Extraído de (GONZALEZ; WOODS, 2010).	31
Figura 2.10: Exemplo de abertura e fechamento. O elemento estruturante é o pequeno círculo mostrado em varias posições em (b). Extraído de (GONZALEZ; WOODS, 2010).....	32
Figura 2.11: Representação hierárquica de contornos . Extraído de (MARANA; BREGA, 2008).....	34
Figura 2.12: Representação de uma ROI na imagem. Extraído de (MOESLUND, 2012).	35
Figura 2.13: Integral da Imagem. Extraído de (BAY et al., 2008).	38
Figura 2.14: Derivada parcial de segunda ordem da gaussiana na direção y e xy , e suas respectivas aproximações. Extraído de (BAY et al., 2008).	38
Figura 2.15: Processo de análise multiescala usando redução do tamanho da imagem (esquerda) e usando integral de imagem com os filtros em diferentes escalas (direita). Extraído de (BAY et al., 2008).	39
Figura 2.16: Filtros em escalas diferentes, 9x9(esquerda) e 15x15(direita). Extraído de(BAY et al., 2008).	39
Figura 2.17: Detecção de pontos de interesse em uma imagem. Extraído de (BAY et al., 2008).	40

Figura 2.18: Filtros Haar Wavelets para calcular a resposta em x (esquerda) e y (direita). Extraído de (BAY et al., 2008).....	40
Figura 2.19: Resposta após aplicação dos filtros. Extraído de (BAY et al., 2008)..	41
Figura 2.20: Janelas de descrição em diferentes escalas. Extraído de (BAY et al., 2008).....	41
Figura 2.21: Representação do vetor com as sub-regiões e suas respectivas respostas na janela utilizada. Extraído de (BAY et al., 2008).....	42
Figura 2.22: Regra do vizinho mais próximo com $k = 1$. Extraído de (TAN; STEINBACH; KUMAR, 2006).....	43
Figura 2.23: Regra do vizinho mais próximo com $k = 3$. Extraído de (TAN; STEINBACH; KUMAR, 2006).....	44
Figura 2.24: Regra do vizinho mais próximo com $k = 2$. Extraído de (TAN; STEINBACH; KUMAR, 2006).....	44
Figura 2.25: Estimação de uma reta ajustada a um conjunto de pontos 2D usando RANSAC. Extraído e adaptado de (HARTLEY ; ZISSERMAN, 2003)....	46
Figura 2.26: Avanço do poder computacional de CPUs e GPUs. Extraído de (KIRK; HWU, 2010).....	47
Figura 2.27: Arquitetura das CPUs e GPUs. Extraído de (KIRK; HWU, 2010).....	48
Figura 2.28: Dimensões para placas de identificação de veículos. Extraído de (CONTRAN, 2007).	50
Figura 4.1: Etapas do método proposto	66
Figura 4.2 : Imagem capturada	67
Figura 4.3: Imagem em tons de cinzas	67
Figura 4.4 : Imagem suavizada	68
Figura 4.5 : Imagem com as bordas destacadas.....	68
Figura 4.6 : Placa do veiculo localizada na imagem.....	69
Figura 4.7 : Imagem com a aplicação do operador de Canny	69
Figura 4.8 : Imagem com fechamento.....	70
Figura 4.9 : Imagem com abertura mínima	70
Figura 4.10 : Imagem com abertura máxima.....	71
Figura 4.11 : Imagem subtraída.....	71
Figura 4.12 : Imagem com eliminação de largura mínima.....	72
Figura 4.13 : Imagem dilatada.....	72
Figura 4.14 : ROI definida acima da placa	73

Figura 4.15 : ROI.....	73
Figura 4.16 : Pontos de interesse identificados na ROI	74
Figura 4.17 : Pontos de interesse identificados nas logomarcas.....	74
Figura 4.18 : Correspondência de pontos de interesse.....	74
Figura 4.19 : Extração de outliers e consistência	75
Figura 5.1 : Desempenho de processamento em uma imagem.....	80
Figura 5.2 : Método proposto parte 1	82
Figura 5.3 : Método proposto parte 2	83
Figura 5.4 : Método de Conci, Carvalho e Rauber (2009).....	84
Figura 5.5 : Desempenho de processamento base I.....	86
Figura 5.6 : Desempenho de processamento por imagem base I.....	87
Figura 5.7 : Desempenho de processamento base II.....	89
Figura 5.8 : Desempenho de processamento por imagem base II.....	90
Figura 5.9 : Desempenho de processamento base III.....	92
Figura 5.10 : Desempenho de processamento por imagem base III.....	93
Figura 5.11 Desempenho de processamento base IV	95
Figura 5.12: Desempenho de processamento por imagem base IV.....	96

LISTA DE TABELAS

Tabela 2.1: Marcas de dez fabricantes de veículos particulares do mundo que estão presentes no Brasil.....	51
Tabela 2.2: Tabela de continência. Extraído e adaptado de (FAWCETT, 2005).....	52
Tabela 3.1: Abordagens propostas e métodos empregados.....	59
Tabela 4.1: Características das bases de imagens usadas	65
Tabela 5.1 : Experimentos de validação	78
Tabela 5.2 : Método de Conci, Carvalho e Rauber (2009)	83
Tabela 5.3 : Experimentos de aplicação base I.....	85
Tabela 5.4 : Experimentos de aplicação base II.....	88
Tabela 5.5 : Experimentos de aplicação base III.....	91
Tabela 5.6: Experimentos de aplicação base IV	94

LISTA DE ABREVIATURAS E SIGLAS

ALU – *Arithmetic Logic Unit*

CPU – *Central Processing Unit*

CONTRAN – Conselho Nacional de Transito

FLANN – *Fast Library for Approximate Nearest Neighbors*

GPU – *Graphics Processing Unit*

GPGPU – *General Purpose Computing GPU*

ROI – *Region of Interest*

RANSAC – *Random Sample Consensus*

SURF – *Speeded Up Robust Feature*

KNN – *K-Neest Neighbors*

SUMÁRIO

CAPÍTULO 1 - INTRODUÇÃO	16
1.1 Contexto e motivação.....	16
1.2 Objetivo	18
1.2.1 Objetivo Geral	18
1.2.2 Objetivos Específicos	18
1.3 Método de Trabalho	18
1.4 Contribuições	19
1.5 Organização do Trabalho.....	19
CAPÍTULO 2 - LEVANTAMENTO BIBLIOGRÁFICO	20
2.1 Imagem Digital	20
2.1.1 Representação de uma Imagem Digital	20
2.1.2 Vizinhança.....	21
2.1.3 Conectividade.....	22
2.2 Detecção de Características de Borda.....	23
2.2.1 Detecção de Borda.....	23
2.2.2 Operador de Sobel	24
2.2.3 Operador de Canny	26
2.3 Morfologia Matemática	28
2.3.1 Erosão.....	29
2.3.2 Dilatação	30
2.3.3 Abertura e Fechamento.....	31
2.4 Detecção de Contorno	33
2.5 Região de Interesse	34
2.6 Descritores Locais de Imagens	35
2.6.1 Identificação de Pontos de Interesse	35
2.6.1.1 SURF.....	37
2.7 Reconhecimento de Padrões	42
2.8 RANSAC	45

2.9 Arquitetura de Processamento	47
2.9.1 CPUs e GPUs	47
2.10 Placa de Identificação de Veículos.....	49
2.11 Logomarcas de Fabricantes de Veículos	50
2.12 Avaliação de desempenho	52
2.13 Considerações Finais	53
CAPÍTULO 3 - TRABALHOS CORRELATOS	54
3.1 Considerações Iniciais.....	54
3.2 Abordagens	54
3.3 Considerações finais	58
CAPÍTULO 4 - MATERIAL E MÉTODOS	63
4.1.1 Considerações iniciais.....	63
4.1.2 Recursos de desenvolvimento	63
4.1.3 Base de imagens.....	64
4.2 Projeto e desenvolvimento do método proposto	65
4.2.1 Descrição das etapas	66
4.3 Considerações finais	75
CAPÍTULO 5 - RESULTADOS E DISCUSSÕES	76
5.1 Considerações iniciais.....	76
5.2 Validação do método.....	76
5.3 Aplicação do método	84
5.4 Considerações finais	97
CAPÍTULO 6 - CONCLUSÕES.....	98
6.1 Trabalhos futuros	99
CAPÍTULO 7 – PUBLICAÇÕES	100
REFERÊNCIAS	101

Capítulo 1

INTRODUÇÃO

Este capítulo apresenta o contexto em que este trabalho está inserido e a motivação que deu origem a este projeto de pesquisa. Serão discutidos os objetivos e o método de trabalho utilizado, principais contribuições e a descrição da organização desta dissertação.

1.1 Contexto e motivação

Entre as diversas oportunidades e lacunas de pesquisas relacionadas ao problema de reconhecimento de logomarcas de veículos, destaca o que chamam de reconhecimento rápido e robusto. Este problema tem sido um dos temas de domínio em soluções de identificação de veículos. Porém realizar o reconhecimento de logomarcas de veículos em imagens de forma rápida mantendo a precisão do método empregado ainda é um problema.

Algumas propostas para resolver esse problema são apresentadas no Capítulo 3. Os autores (BADURA; FOLTAN, 2010; PSYLLOS; ANAGNOSTOPOULOS; KAYAFAS, 2011; LIPIKORN et al., 2014; AN; THAKOOR; BHANU, 2012; LIU et al. 2014; NAN, 2012; ZHANG; SHEN; CHANG, 2013; PAN; ZHANG, 2013; SAM; TIAN, 2012) dessas propostas salientam sobre a dificuldade em resolver esse problema, isso se dá devido a complexidade envolvida para atender um alto nível de detalhes, pois a logomarca do fabricante do veículo pode ser apresentada em diferentes ângulos, possuir diferentes cores, formatos e

tamanhos. Estar em imagens com escalas diferentes, rotacionadas, com baixo contraste e interferência devido a fatores de ambiente não controlado. Uma questão destacada pelos autores é a necessidade de pesquisa devido a importância do problema em sistemas de vigilância, devido ao interesse na área de transporte inteligente e gestão de tráfego, interesse de autoridades policiais para aplicação da lei e interesse dos fabricantes de veículos no mercado regional. Por outro lado, o algoritmo Speeded Up Robust Feature (SURF) (BAY et al., 2008) tem se destacado (HEDAYATI et al. 2013; SIDLA; KOTTMANN; BENESOVA, 2011) frente a outros métodos de identificação de pontos de interesse, como sendo o que tem uma execução mais rápida não descuidando, contudo, da robustez essencial para o reconhecimento de objetos. No entanto devido a importância do algoritmo SURF e de o mesmo não ser rápido o suficiente para melhor atender a exigência de um sistema em tempo real, esforços (HUIJUAM; QIONG, 2011; XING; JUNG-HAE; BYUNG-JAE, 2012; SCHWEIGER, et al., 2013; HERAKLEOUS; PAULLIS, 2013; CHO; CHOI, 2012; CHATURVEDI; SETHI, 2013) têm sido realizados para facilitar sua adoção no desenvolvimento de sistemas de reconhecimento de objetos. Alguns visam otimizar o tempo de execução do algoritmo realizando modificações do mesmo, o que acaba na maioria das vezes alterando as suas características, outras propõem novos algoritmos ou combina com outras técnicas, visando uma aplicação específica, apresentando novas opções de contornar o problema. Apesar de novos algoritmos serem importantes, para que a proposta de um novo algoritmo seja efetivamente estabelecida é necessário que uma grande parte da comunidade científica adote tal algoritmo. Assim, é importante que sejam pesquisadas abordagens alternativas, não apenas para explorar um caminho diferente ou modificar um algoritmo, mas para facilitar a adoção do mesmo em potenciais aplicações de modo a preservar suas características originais e que a sua adoção seja adequada para compor a solução de um problema. O método SURF oferece a possibilidade de integração com outras técnicas de modo a resolver essas questões.

Assim, este trabalho, faz uso do algoritmo SURF em uma abordagem para reconhecimento de logomarcas de veículos, combinando o algoritmo com outras técnicas para compor um método alternativo para solução do problema.

1.2 Objetivo

1.2.1 Objetivo Geral

Desenvolver um método que possa realizar o reconhecimento de logomarcas de veículos de forma rápida e robusta usando o algoritmo SURF.

1.2.2 Objetivos Específicos

- Estudo de técnicas de processamento de imagens e visão computacional aplicadas ao domínio do problema.
- Identificação de uma sequencia de métodos que seja adequada para o reconhecimento de logomarcas de veículos, tomando-se como base o algoritmo SURF.
- Avaliar o método proposto.

1.3 Método de Trabalho

O método de trabalho empregado no desenvolvimento deste projeto de mestrado começou com o levantamento da literatura existente. Um estudo para verificar quais são as formas correntes usuais para se resolver o problema analisando diferentes propriedades de cada uma das abordagens para definição do método proposto, estudo de técnicas de processamento de imagens e visão computacional, tecnologias relacionadas aos recursos de software, hardware, bibliotecas e especificação dos elementos necessários à abordagem.

Em seguida o projeto foi feito em etapas para testes de validação da proposta do método, cada etapa do projeto teve uma implementação em OpenCV CPU e depois uma possível solução usando o algoritmo SURF e a maioria dos métodos em OpenCV com CUDA GPU foi desenvolvida. Em seguida foi avaliado o desempenho em taxa de acerto e tempo de processamento bem como a utilização do Excel para cálculos estatísticos e gráficos.

1.4 Contribuições

As principais contribuições deste trabalho são:

- Uma revisão bibliográfica detalhada do estado da arte.
- Um estudo dos métodos de detecção e descrição de pontos de interesse.
- Uma investigação para facilitar a adoção do algoritmo SURF no desenvolvimento de sistemas de reconhecimentos de objetos que poderá servir como referência em outras áreas de aplicação.
- Uma abordagem alternativa para o problema de reconhecimento de logomarcas de veículos através de um modelo de utilização eficiente do método SURF.
- Um estudo da possibilidade de reconhecimento de objetos a partir de métodos de identificação de pontos de interesse em sistemas embarcados.

1.5 Organização do Trabalho

Este trabalho está organizado da seguinte forma: no Capítulo 2 é apresentado o levantamento bibliográfico acerca dos conceitos fundamentais relacionados ao trabalho. O Capítulo 3 descreve algumas abordagens que se relacionam com este trabalho de pesquisa. O Capítulo 4 apresenta o material, o projeto e as etapas para o desenvolvimento do método proposto. O Capítulo 5 apresenta os resultados e as discussões sobre os experimentos realizados para a avaliação do método desenvolvido. O Capítulo 6 apresenta as conclusões e trabalhos futuros.

Capítulo 2

LEVANTAMENTO BIBLIOGRÁFICO

Este capítulo descreve os principais conceitos relacionados ao trabalho desenvolvido. São apresentados os conceitos fundamentais aplicáveis ao processo, dentre eles o conceito de imagem digital, detecção de borda e contorno, morfologia matemática, região de interesse, identificação de pontos de interesse, reconhecimento, homografia, arquitetura das CPUs e GPUs, placa de licenciamento veicular, logomarcas de fabricantes de veículos e avaliação de desempenho.

2.1 Imagem Digital

2.1.1 Representação de uma Imagem Digital

A representação de uma imagem digital monocromática é definida por Gonzalez e Woods (2000), como sendo uma função bidimensional $f(x, y)$, da intensidade da luz, onde x e y representam a coordenada espacial, e o valor f em qualquer ponto é proporcional a intensidade ou nível de cinza.

Uma imagem digital é composta por uma quantidade finita de *pixels*. O *pixel* é caracterizado como detalhe da imagem ou menor elemento da imagem, sendo que, cada um dos *pixels* possui uma localização específica e um determinado valor de intensidade. A Figura 2.1 representa a convenção geralmente adotada para a origem dos eixos em uma imagem e um ponto.

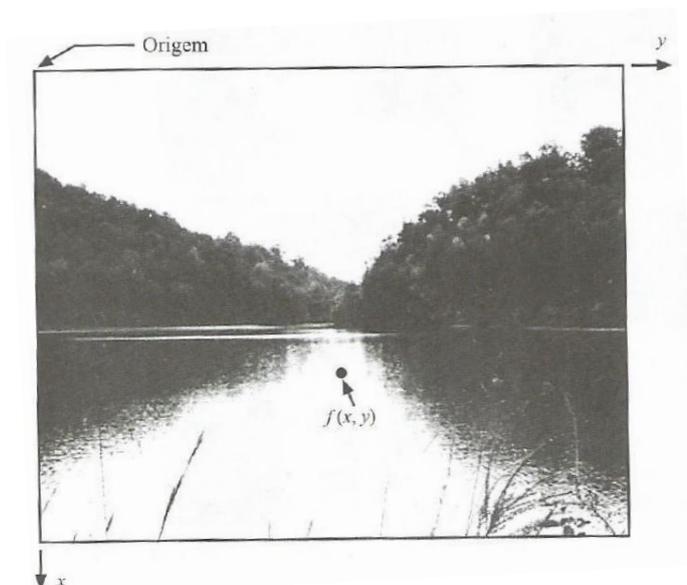


Figura 2.1: Representação de uma imagem digital. Extraído de (GONZALEZ; WOODS, 2000).

Em imagens binárias o *pixel* podem assumir valores 0 ou 1. Para imagens em tons de cinzas os valores dos *pixels* variam de 0 a 255 e no caso de imagens coloridas cada *pixel* é representado por um conjunto de três valores no modelo RGB (*red, green, blue*).

2.1.2 Vizinhança

A imagem digital é composta por *pixels*, como foi explicado na seção anterior. Espacialmente, um *pixel* de referência P localizado nas coordenadas (x, y) , tem quatro *pixels* vizinhos nas posições verticais e horizontais. Estes *pixels* podem ser endereçados da seguinte maneira (GONZALES; WOODS, 2000):

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

Os vizinhos horizontais e verticais formam um conjunto de *pixels* ao redor de P denominado de vizinhança-4 conforme ilustrado na Figura 2.2.

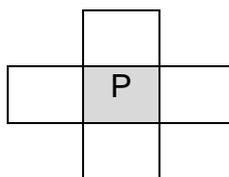


Figura 2.2: Vizinhança-4.

Uma característica importante nesta vizinhança é a distância unitária que cada *pixel* vizinho tem em relação ao *pixel P*. Pensando em uma distância de um *pixel*, pode-se ter também vizinhos diagonais ao *pixel P* cujas coordenadas serão (GONZALES; WOODS, 2000):

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

Esta vizinhança em conjunto com de vizinhança-4 forma a vizinhança-8 do *pixel P* conforme lustrado na Figura 2.3.

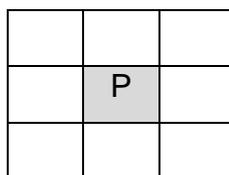


Figura 2.3: Vizinhança-8.

2.1.3 Conectividade

O conceito de conectividade entre *pixels* é usado no estabelecimento de bordas de objetos, definindo limites entre os mesmos e componentes de regiões de uma imagem (GONZALES; WOODS, 2000). Para determinar se dois *pixels* estão conectados, deve-se determinar se eles são adjacentes em algum sentido, como por exemplo, com vizinhança-4 e se a sua amplitude, como nível de cinza, respeita algum critério de similaridade, ou se temos uma imagem binária, onde os *pixels* podem ter os valores 0 ou 1, define-se um modelo, como por exemplo vizinhança-8, se estamos em um *pixel P* em (x, y) . Considerando a redondeza especificada:

$$\begin{aligned} & (x + 1, y), (x - 1, y), (x, y + 1), (x, y - 1), (x + 1, y + 1), (x + 1, y - 1), \\ & (x - 1, y + 1), (x - 1, y - 1) \end{aligned} \quad (2.3)$$

caso algum *pixel* dentro desta vizinhança tenha o mesmo valor de P , estese será considerado conectado ao *pixel* P .

2.2 Detecção de Características de Borda

Em uma imagem, as bordas caracterizam os contornos dos objetos, sendo muito uteis no processo de segmentação e localização de regiões candidatas. A seguir serão apresentados os conceitos de detecção de borda e dois operadores que se destacam na literatura.

2.2.1 Detecção de Borda

Uma borda é o limite entre duas regiões com propriedades razoavelmente distintas de nível de cinza em imagens de tons de cinzas (PEDRINI; SCHWARTZ, 2008; GONZALEZ; WOODS, 2010), sendo a detecção de bordas, essencialmente, a operação de identificação de mudanças locais significativas nos níveis de cinza da imagem, podendo ser descritas por meio de derivadas, onde o operador gradiente é geralmente utilizado para essa diferenciação (PEDRINI; SCHWARTZ, 2008).

O gradiente ∇f é um vetor cuja direção indica os locais nos quais os níveis de cinza sofrem maior variação na posição (x, y) de uma imagem f (PEDRINI; SCHWARTZ, 2008; GONZALEZ; WOODS, 2010) e pode ser expresso como:

$$\nabla f = \text{grad}(f) = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \quad (2.4)$$

onde $\frac{\partial f}{\partial x}$ e $\frac{\partial f}{\partial y}$ são as derivadas parciais da imagem na direção x e y .

Na detecção de bordas, a magnitude do vetor gradiente é uma medida importante e é denotado por $M(x, y)$, no qual:

$$M(x, y) = \text{mag}(\nabla f) = \sqrt{g_x^2 + g_y^2} \quad (2.5)$$

Outra medida importante é a direção do vetor gradiente. A direção do vetor gradiente é dada pelo ângulo $\alpha(x, y)$ medido em relação ao eixo x e expresso como:

$$\alpha(x, y) = \text{tg}^{-1} \left[\frac{g_x}{g_y} \right] \quad (2.6)$$

A diferença entre os valores de *pixels* adjacentes pode ser utilizada para detectar mudanças de intensidade. Bordas horizontais são detectadas pela diferença vertical entre os pontos adjacentes da imagem, enquanto que bordas verticais são detectadas pela diferença horizontal entre os pontos.

O cálculo da magnitude do vetor gradiente é geralmente implementado por máscaras de convoluções de tamanho 3 x 3.

Diversos métodos foram propostos ao longo dos anos para detectar bordas em imagens digitais, dentre os muitos operadores disponíveis destacam-se os operador de Sobel (1970) e Canny (1986). Ambos os operadores são apresentados a seguir.

2.2.2 Operador de Sobel

O operador de Sobel (SOBEL, 1970; PEDRINI; SCHWARTZ, 2008) efetua uma medição de gradiente espacial 2D de uma imagem. Ele é utilizado para calcular a magnitude absoluta aproximada do gradiente em cada ponto de uma imagem em tons de cinza de entrada. Este operador utiliza duas máscaras de convolução 3x3, uma na direção x e outra na direção y para responder maximamente a bordas verticais e horizontais com relação a grade de *pixels*. A Figura 2.4 mostra as máscaras de Sobel.

-1	0	1
-2	0	2
-1	0	1

-1	-2	-1
0	0	0
1	2	1

Figura 2.4: Máscaras de Sobel na direção x (esquerda) e direção y (direita). Extraído de (PEDRINI; SCHWARTZ, 2008).

As máscaras podem ser aplicadas para produzir medidas separadas de componentes do gradiente em cada orientação chamadas G_x e G_y , que pode ser combinados para encontrar a magnitude absoluta do gradiente em cada ponto e a orientação desse gradiente. A Figura 2.5 mostra um exemplo da aplicação do operador de Sobel para detecção de bordas em uma imagem.

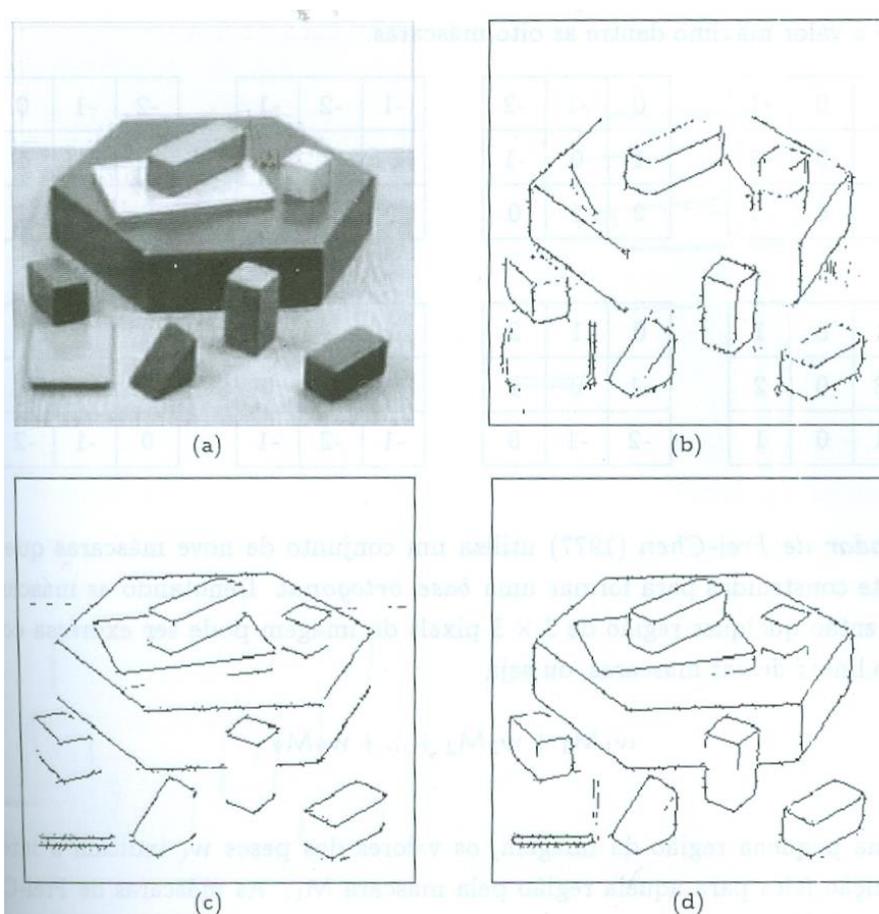


Figura 2.5: (a) imagem original; (b) aplicação do operador de Sobel para obtenção de G_x ; (c) para obtenção de G_y ; (d) magnitude do gradiente da imagem obtido pela combinação dos resultados de (b) e (c). Extraído de (PEDRINI; SCHWARTZ, 2008).

Como pode ser visto na Figura 2.5, as regiões de alto gradiente que correspondem as bordas foram destacadas.

2.2.3 Operador de Canny

Canny (1986) propôs um detector de bordas ótimo que procura otimizar a localização de pontos da borda na presença de ruído visando três objetivos básicos: (a) detectar bordas com baixa taxa de erro, devendo encontrar todas as bordas quanto possível, (b) as bordas detectadas devem ser o mais próximo possível das bordas verdadeiras, estando bem localizadas e (c) cada borda na imagem deve ser marcada uma única vez, não permitindo que o ruído crie bordas falsas (CANNY, 1986; GONZALEZ; WOODS, 2010; PEDRINI; SCHWARTZ, 2008).

O operador de Canny usa como entrada uma imagem em tons de cinza em um processo multi-estágio. O processo inicia-se suavizando a imagem com um filtro gaussiano, onde $f(x, y)$ é a imagem de entrada e $G(x, y)$ a função gaussiana:

$$G(x, y) = e^{-\frac{x^2+y^2}{2}} \quad (2.7)$$

Formando a imagem suavizada, $f_s(x, y)$, por convolução de G e f :

$$f_s(x, y) = G(x, y) * f(x, y) \quad (2.8)$$

Em seguida, a magnitude e a direção do gradiente são calculados:

$$M(x, y) = \sqrt{g_x^2 + g_y^2} \quad (2.9)$$

e

$$\alpha(x, y) = \text{tg}^{-1} \frac{g_y}{g_x} \quad (2.10)$$

Após o cálculo do gradiente, a borda é localizada tomando-se apenas os pontos cuja magnitude seja localmente máxima na direção do gradiente, mais especificamente, as bordas dão origem a picos na imagem da magnitude do gradiente e o algoritmo então rastreia ao longo do topo desses picos e seta para zero todos os *pixels* que não estão no topo dos picos, de forma a gerar linhas finas na saída, esse processo é conhecido como supressão não-máxima (MARANA; BREGA, 2008). No entanto, a borda pode ainda conter certos fragmentos espúrios que podem ser removidos realizando uma limiarização com histerese durante esse processo. Essa operação é controlada por dois limiares: T_1 e T_2 , *pixels* da borda que possuem gradiente maior que T_2 são mantidos e qualquer outro *pixel* conectado a esses *pixels* da borda é considerado como pertencente à borda se a magnitude de seu gradiente estiver acima de T_1 , considerando sempre os maiores valores na direção do gradiente. Esta histerese ajuda a garantir que bordas com ruído não sejam quebradas em múltiplos fragmentos, por fim é produzido como saída uma imagem mostrando as posições das discontinuidades de intensidade. A Figura 2.6 mostra um exemplo da aplicação do operador de Canny para detecção de bordas em uma imagem.

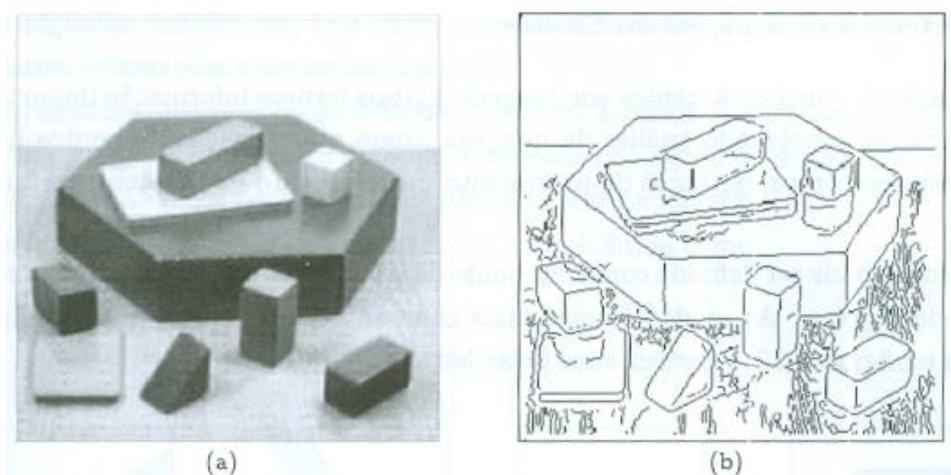


Figura 2.6: (a) imagem original; (b) aplicação do operador de Canny. Extraído de (PEDRINI; SCHWARTZ, 2008).

Como pode ser visto na Figura 2.6 a direita, as bordas da imagem foram destacadas com eficiência.

2.3 Morfologia Matemática

Segundo Gonzalez e Woods (2010), a palavra morfologia geralmente denota uma área da biologia que trata a forma e a estrutura de animais e plantas. Utiliza-se a mesma palavra no contexto de morfologia matemática, como sendo uma ferramenta utilizada para a extração de componentes de imagens que sejam úteis na representação e descrição da forma de uma região.

Os fundamentos da morfologia matemática encontram-se na teoria dos conjuntos. A morfologia oferece uma abordagem unificada e poderosa para vários problemas de processamento de imagens. De acordo com Gonzalez e Woods (2010), os conjuntos em morfologia matemática representam os objetos encontrados em uma imagem. Em imagens binárias, o conjunto de pixels da imagem é definido no espaço bidimensional dos números inteiros Z^2 , em que cada elemento do conjunto é um vetor bidimensional com coordenadas (x, y) de um pixel branco ou preto de uma imagem.

Um operador morfológico corresponde a um mapeamento entre o conjunto que define uma imagem e outro conjunto, denominado elemento estruturante, também definido em Z^2 (PEDRINI; SCHWARTZ, 2008). Ao trabalhar com imagens, é necessário acrescentar o menor número possível de elementos de fundo para converter os elementos estruturantes em arranjos retangulares (GONZALES; WOODS, 2010). A Figura 2.7 apresenta alguns elementos estruturantes geralmente utilizados, onde a origem ou ponto central é destacado na imagem com um ponto preto e os elementos de fundo são mostrados sem sombreamento.

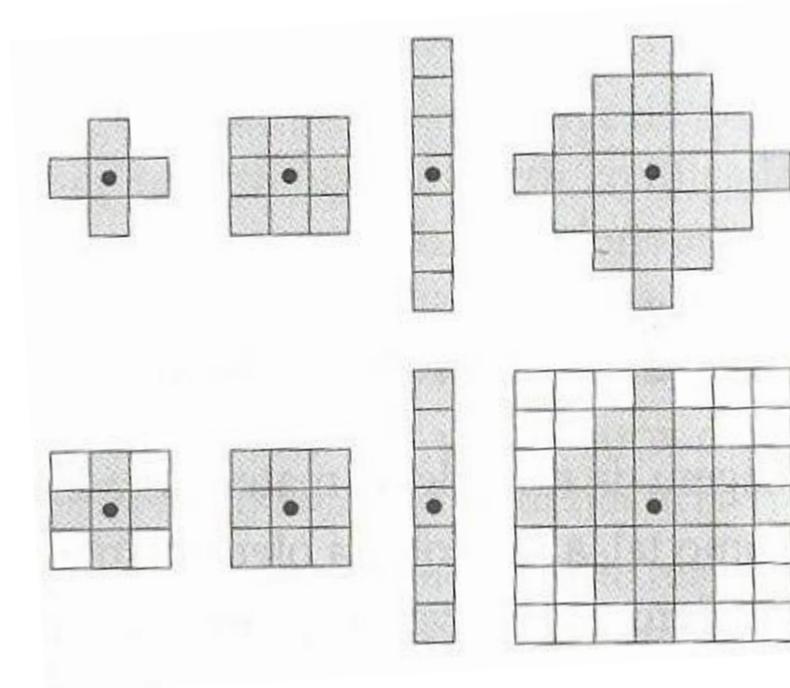


Figura 2.7: Exemplos de elementos estruturantes. Extraído de (GONZALEZ; WOODS, 2010).

Nas operações morfológicas, o elemento estruturante é deslocado sobre a imagem, sendo esses deslocamentos relacionados com a origem do elemento e com o seu formato.

2.3.1 Erosão

Segundo Gonzalez e Woods (2010) a erosão é uma operação de diminuição ou afinamento. Considerando A e B como conjuntos de Z^2 , a erosão de A por B , indicada por $A \ominus B$, é definida como:

$$A \ominus B = \{z \mid (B)_z \subseteq A\} \quad (2.11)$$

O qual indica que a erosão de A por B é o conjunto de todos os pontos de z tal que B , transladado por z , esta contido em A . A Figura 2.8 mostra um exemplo de erosão.

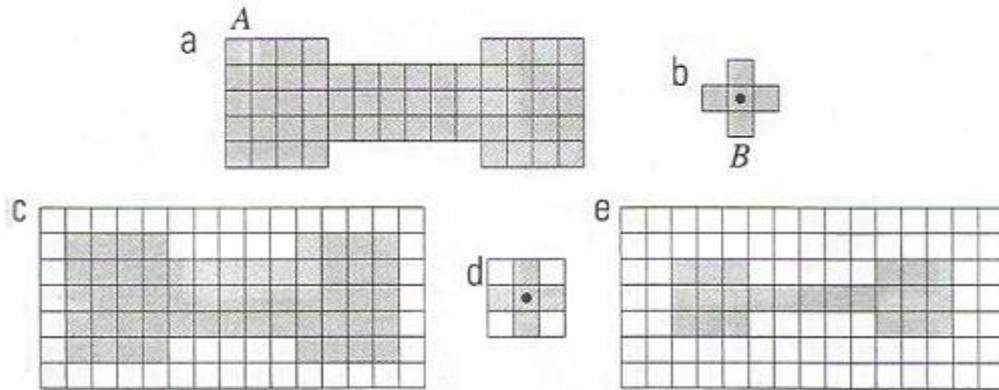


Figura 2.8: Exemplo de erosão. Extraído de(GONZALEZ; WOODS, 2010).

Erodir a imagem binária A pelo elemento estruturante B consiste em mover a origem de B sobre a imagem A , tal que, caso B esteja totalmente contido em A , o pixel da imagem sob a origem de B pertencerá a erosão entre A e B (PEDRINI; SCHWARTZ, 2008).

2.3.2 Dilatação

Segundo Gonzalez e Woods (2010) a dilatação aumenta ou engrossa os objetos em uma imagem binária. Considerando A e B como conjuntos de Z^2 , a dilatação de A por B , indicada por $A \oplus B$, é definida como:

$$A \oplus B = \{z \mid \overset{\wedge}{(B)}_z \cap A \neq \emptyset\} \tag{2.12}$$

O qual indica que a dilatação de A por B é o conjunto de todos os deslocamentos, z , tal que B e A se sobreponham pelo menos por um elemento. A Figura 2.9 mostra um exemplo de dilatação.

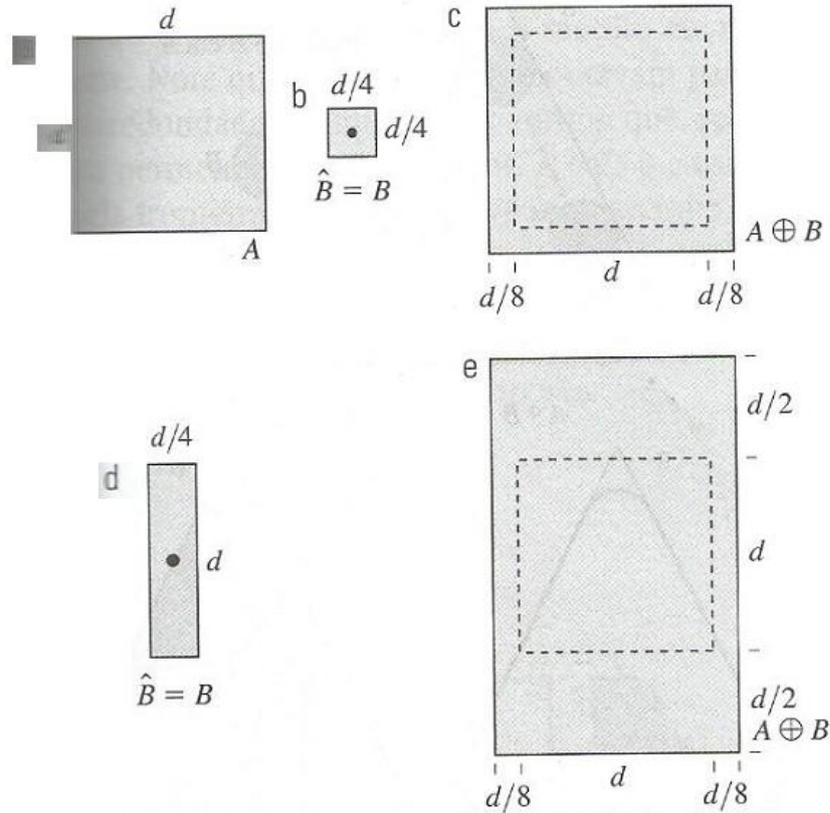


Figura 2.9: Exemplo de dilatação. Extraído de (GONZALEZ; WOODS, 2010).

Dilatar a imagem binária A pelo elemento estruturante B consiste em mover a origem de B sobre a imagem A , atribuindo o valor 1 a cada posição que é sobreposta por B (PEDRINI; SCHWARTZ, 2008).

2.3.3 Abertura e Fechamento

Segundo Gonzalez e Woods (2010) a abertura geralmente suaviza o contorno de um objeto, rompe os istmos e elimina as saliências finas. A abertura do conjunto A pelo elemento estruturante B , indicado por $A \circ B$, é definido como:

$$A \circ B = (A \ominus B) \oplus B \tag{2.13}$$

A abertura da imagem binária A pelo elemento estruturante B é a erosão de A por B , seguida de uma dilatação do resultado por B .

Já o fechamento segundo Gonzalez e Woods (2010) e Pedrini e Schwartz (2008) geralmente funde separações estreitas entre objetos, elimina pequenos buracos e preenche as lacunas no contorno.

O fechamento do conjunto A pelo elemento estruturante B , indicado por $A \cdot B$, é definido como:

$$A \cdot B = (A \oplus B) \ominus B \tag{2.14}$$

O fechamento da imagem binária A pelo elemento estruturante B é a dilatação de A por B , seguida de uma erosão do resultado por B .

A Figura 2.10 ilustra as operações de abertura e fechamento.

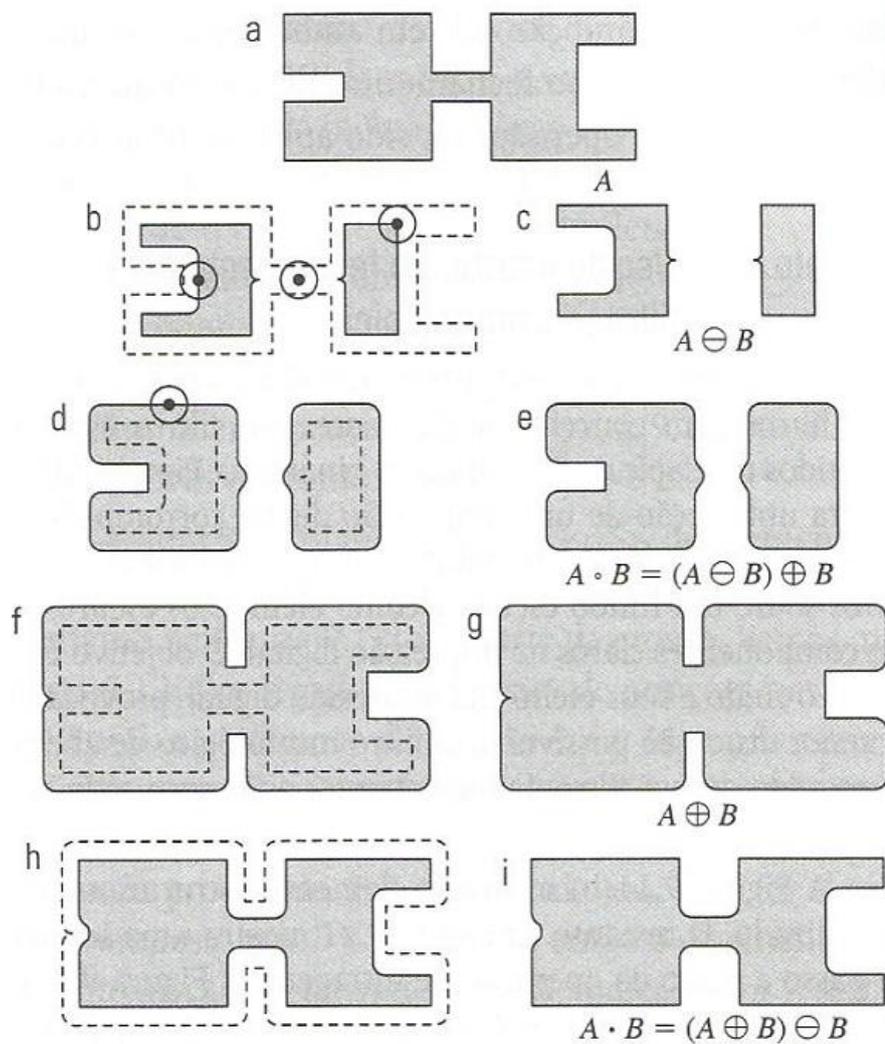


Figura 2.10: Exemplo de abertura e fechamento. O elemento estruturante é o pequeno círculo mostrado em varias posições em (b). Extraído de (GONZALEZ; WOODS, 2010).

2.4 Detecção de Contorno

A detecção de contorno é empregada na etapa de segmentação. A segmentação consiste em analisar uma imagem de modo a distinguir elementos específicos. Um contorno segundo Pedrini e Schwartz (2008) é uma linha fechada formada pela união das bordas de um objeto. O processo de detecção de contornos constitui em duas etapas básicas: (a) traçado de contornos e (b) representação de contornos. Na etapa de traçado de contornos, os contornos existentes são seguidos na imagem. O método de Suzuki (SUZUKI; ABE, 1985; MARANA; BREGA, 2008) é geralmente usado para executar essa tarefa, nesse método, inicialmente o *pixel* do contorno mais acima e a esquerda é encontrado. Então a vizinhança do primeiro *pixel* é verificada no sentido horário para encontrar o próximo *pixel* do contorno. Em seguida a busca por outros *pixels* é feita no sentido anti-horário e termina quando os primeiros dois *pixels* do contorno são encontrados novamente. Na etapa de representação de contornos, os mesmos são descritos de uma forma significativa, para essa tarefa, existem diversas técnicas, por exemplo, código de cadeia, representação poligonal. Os códigos de cadeia é uma sequência de números que determinam a vizinhança de um *pixel* de contorno onde o próximo *pixel* de contorno se encontra. A representação poligonal é uma sequência de vértices que quando ligados simbolizam a essência do contorno. Os contornos obtidos a partir da imagem poderão ser organizados em uma árvore, onde um contorno C1 é pai de um contorno C2 se e somente se C1 contém C2 (MARANA; BREGA, 2008). A Figura 2.11 representa os contornos obtidos a partir de uma imagem organizados em uma árvore.

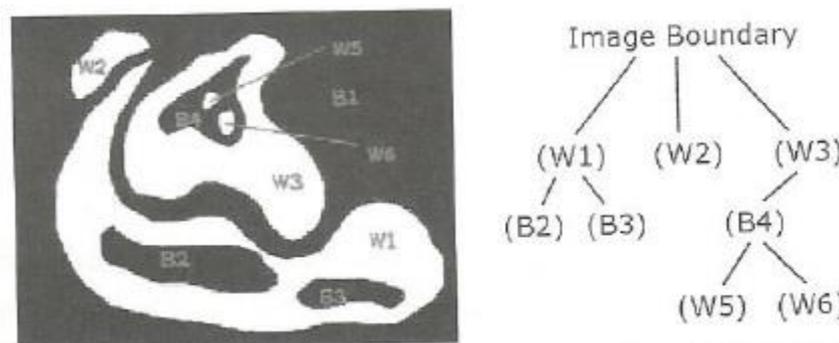


Figura 2.11: Representação hierárquica de contornos . Extraído de (MARANA; BREGA, 2008).

Os contornos poderão ser manipulados de forma a obter informações de área e aproximação poligonal. O método de Douglas-Peucker (DOUGLAS; PEUCKER, 1973; MARANA; BREGA, 2008) é geralmente empregado para realizar a simplificação no processo de aproximação e viabiliza a construção de uma árvore binária para estruturar o resultado do processamento.

2.5 Região de Interesse

Uma região de interesse ou Region of Interest (ROI) é uma região da imagem em que estamos particularmente interessados para concentrarmos o processamento e análise. É normalmente um retângulo dentro da imagem que define os pixels de interesse, sendo portanto ignorados todos os pixels que estão fora da ROI (MOESLUND, 2012). Esta técnica é geralmente usada em casos em que a carga de processamento é muito pesada para considerar a imagem inteira ou quando se está interessado em analisar somente parte da imagem, consequentemente diminuindo o tempo gasto nas operações matemáticas, uma vez que apenas uma quantidade de pixels pertencentes a ROI é considerada no processamento. A Figura 2.12 mostra a imagem com uma ROI, ou seja, apenas a parte da imagem a ser processada.



Figura 2.12: Representação de uma ROI na imagem. Extraído de (MOESLUND, 2012).

2.6 Descritores Locais de Imagens

Descritores Locais de Imagens são algoritmos que descrevem imagens por meio de elementos denominados de pontos de interesse ou pontos chave. A seguir serão apresentados os conceitos de identificação de pontos de interesse e o método SURF.

2.6.1 Identificação de Pontos de Interesse

Pontos de interesse, na maioria das vezes, são vértices de contornos ou regiões relevantes na imagem. Eles são encontrados por detectores de pontos de interesse.

A repetibilidade é a propriedade mais importante dos detectores de pontos de interesse, ou seja, um bom detector de pontos de interesse deve selecionar exatamente o mesmo conjunto de pontos sob diferentes condições de visualização da mesma imagem (BAY et al., 2008).

Detectados os pontos de interesse na imagem, descritores são usados para calcular os vetores de características para representarem cada uma das regiões que foram definidas pelos pontos de interesse. Semelhantemente um descritor deve ser invariante a diferentes condições de visualização e gerar vetores distintos.

A distinção de vetores é realizada no processo de correspondência geralmente medida pela função de distância Euclidiana, onde quanto maior for a distância entre dois vetores mais distintos eles serão entre si. Sendo assim espera-se que pontos de interesse repetidos tenham distância mínima entre os mesmos.

O método Speeded Up Robust Features (SURF) (BAY et al., 2008) é a técnica principal desta pesquisa, uma vez que pode ser adequado para o problema em questão e devido ao grande interesse da comunidade acadêmica (HUIJUAM; QIONG, 2011; CHATURVEDI; SETHI, 2013; HERAKLEOUS; PAULLIS, 2013; SCHWEIGER, et al., 2013; CHO; CHOI, 2012; XING; JUNG-HAE; BYUNG-JAE, 2012) por esse algoritmo, no qual será empregado na etapa de reconhecimento do método proposto. No entanto, é possível encontrar na literatura diversas propostas tanto de detectores e descritores de características que foram apresentadas ao longo dos anos, por exemplo, o detector de cantos de Harris (HARRIS; STEPHENS, 1988), Good Features To Track (GFTT) (SHI; TOMASI, 1994), Scale Invariant Feature Transform (SIFT) (LOWE, 1999), Maximally-Stable Extremal Region Extractor (MSER) (MATAS, 2002), Features from Accelerated Segment Test (FAST) (ROSTEN; DRUMMOND, 2005), Center Surrounded Extrema (STAR) (AGRAWAL; KONOLIGE; BLAS, 2008), Binary Robust Independent Elementary Features (BRIEF) (CALONDER; LEPETIT; FUA, 2010) e Oriented FAST and Rotated BRIEF (ORB) (RUBLEE et al., 2011).

Sendo o SURF ainda destacado por Hedayati et al. (2013) dentre a maioria dos algoritmos descritos anteriormente por obter um melhor desempenho, levando em consideração um conjunto de valores na avaliação, tais como alta invariância a transformações geométricas (como rotação e mudança de escala), alta invariância a transformações radiométricas (como ruído ou variação luminosa) na imagem e tempo de execução.

2.6.1.1 SURF

Bay et al. (2008) propuseram o Speeded Up Robust Feature (SURF), este método é um detector e descritor de pontos de interesse. O processo de detecção e descrição de pontos de interesse do SURF pode ser definido em três etapas básicas: (a) criação da integral de imagem; (b) seleção dos pontos de interesse através de matrizes Hessianas e (c) criação de descritores.

O processo inicia-se detectando os pontos de interesse através de matrizes Hessianas por meio de uma imagem auxiliar que é resultado de uma operação de integração da imagem original. Dado um ponto $x = (x, y)$ na imagem I , a matriz Hessiana $H(x, \sigma)$ em x na escala σ é definida como:

$$H(x, \sigma) = \begin{bmatrix} L_{xx}(x, \sigma) & L_{xy}(x, \sigma) \\ L_{xy}(x, \sigma) & L_{yy}(x, \sigma) \end{bmatrix} \quad (2.15)$$

onde $L_{xx}(x, \sigma)$ é a convolução da derivada de segunda ordem da Gaussiana $\frac{\partial^2}{\partial^2} g(\sigma)$ com a imagem I no ponto x , semelhantemente para $L_{xy}(x, \sigma)$ e $L_{yy}(x, \sigma)$. O determinante Hessiano para aproximação $\det(H_{approx})$ é escrito como:

$$\det(H_{approx}) = D_{xx}D_{yy} - (wD_{xy})^2 \quad (2.16)$$

sendo D_{xx} derivada parcial no eixo x , D_{yy} derivada parcial no eixo y e D_{xy} derivada parcial no eixo xy .

A integração de imagem é uma técnica em que cada ponto armazena a soma dos *pixéis* de uma área retangular entre o ponto de origem e o ponto em análise, criando uma imagem com estruturas, ou seja, gera a partir da imagem original outra imagem onde cada ponto será a soma de todos os pontos que estão acima e a esquerda do mesmo até o ponto O . A Figura 2.13 representa essa técnica.

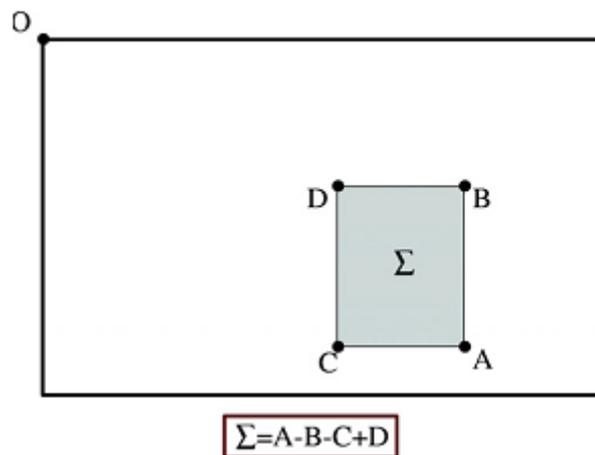


Figura 2.13: Integral da Imagem. Extraído de (BAY et al., 2008).

Este conceito é usado no SURF para calcular de modo eficiente a soma de valores em *pixels* de uma imagem. Então os filtros caixas com várias direções são usados para detectar essas estruturas. A Figura 2.14 representa os filtros e suas respectivas aproximações.

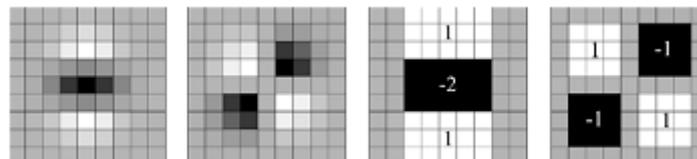


Figura 2.14: Derivada parcial de segunda ordem da gaussiana na direção y e xy , e suas respectivas aproximações. Extraído de (BAY et al., 2008).

O processo de análise multiescala é realizado através do aumento gradativo do tamanho do filtro em uma pirâmide. A Figura 2.15 demonstra esse processo. Uma pirâmide de imagens consiste em uma sequência na qual cada imagem subsequente tem um nível de resolução menor que a anterior.

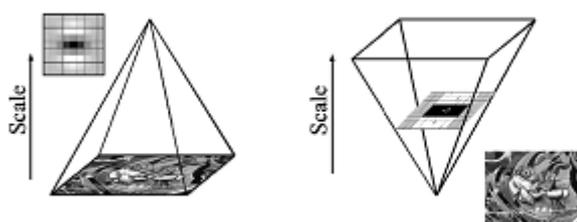


Figura 2.15: Processo de análise multiescala usando redução do tamanho da imagem (esquerda) e usando integral de imagem com os filtros em diferentes escalas (direita). Extraído de (BAY et al., 2008).

Como pode ser observado na Figura 2.15 esse processo difere de métodos usuais de análise multiescala em pirâmide que usa redução do tamanho da imagem. A Figura 2.16 representa dois filtros em escalas diferentes.

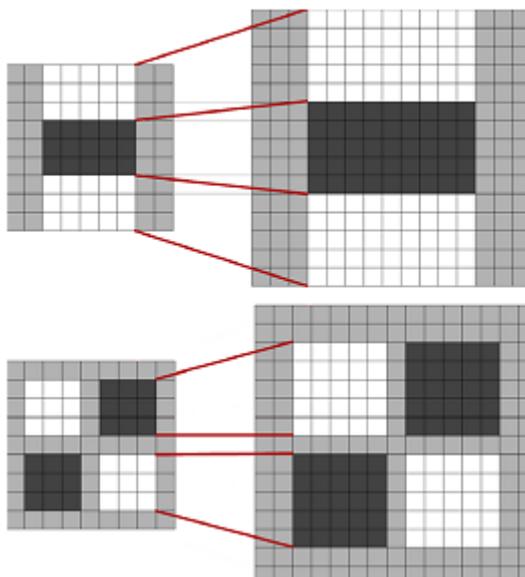


Figura 2.16: Filtros em escalas diferentes, 9x9(esquerda) e 15x15(direita). Extraído de(BAY et al., 2008).

Este método possibilita otimizar o desempenho do algoritmo em termos de processamento, uma vez que as operações envolve apenas alguns cálculos com a integração. As respostas às diferenças de escala e rotação são analisadas unicamente com a transformação dos filtros tanto na sua direção como no seu tamanho.

A aplicação do filtro em diferentes escalas esta relacionado ao conceito de oitavas, que consiste em agregar valores intermediários entre as escalas de análise. Por fim, para localizar os pontos de interesse sobre escalas, é aplicada uma

supressão não máxima em uma vizinhança 3x3x3. A supressão não máxima consiste em encontrar pontos que se sobressaem em relação a sua vizinhança. A Figura 2.17 representa o resultado obtido após a execução da primeira parte do método, onde é possível observar que há vários círculos, o centro desses círculos representam os pontos de interesse que foram detectados na imagem. O tamanho do círculo está diretamente ligado à escala em que o ponto foi encontrado.

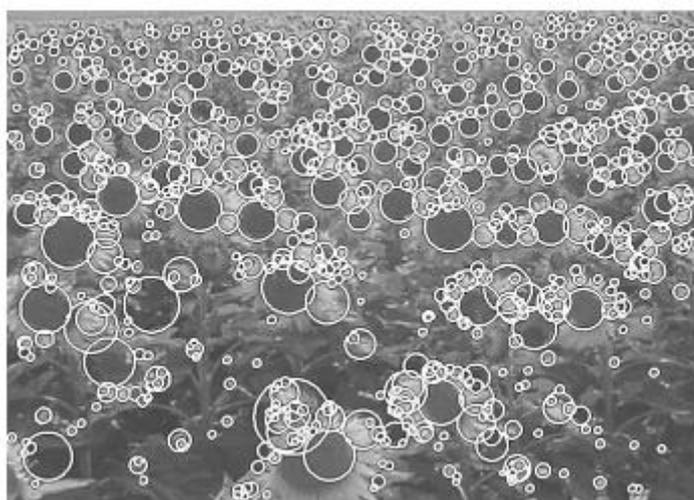


Figura 2.17: Detecção de pontos de interesse em uma imagem. Extraído de (BAY et al., 2008).

Na etapa de descrição é realizado a atribuição de uma orientação dominante com base nas informações de uma região circular em torno dos pontos de interesse detectados, o que vai tornar o algoritmo invariante a rotação. Esta orientação é computada por convolução com filtros Haar Wavelets, nas direções x e y . A Figura 2.18 representa os filtros Haar Wavelets, o resultado de aplicação pode ser observado na Figura 2.19.



Figura 2.18: Filtros Haar Wavelets para calcular a resposta em x (esquerda) e y (direita). Extraído de (BAY et al., 2008).

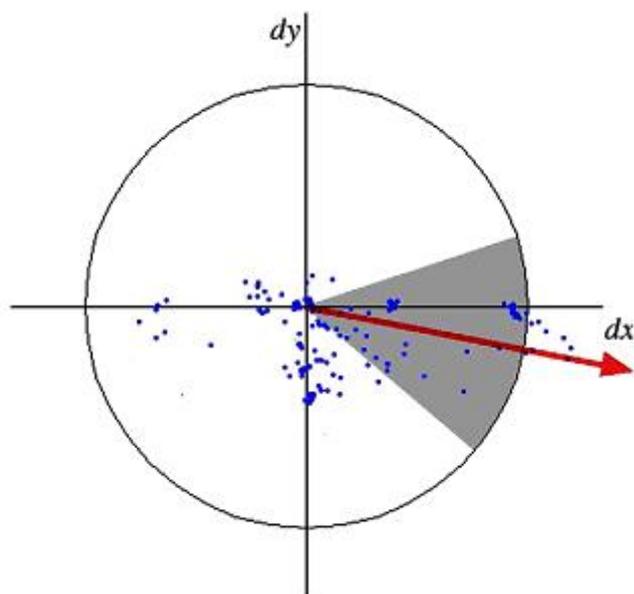


Figura 2.19: Resposta após aplicação dos filtros. Extraído de (BAY et al., 2008).

Em seguida uma janela quadrada centrada de mesma orientação é utilizada. A Figura 2.20 representa essas janelas em diferentes escalas na imagem.



Figura 2.20: Janelas de descrição em diferentes escalas. Extraído de (BAY et al., 2008).

Esta janela é segmentada em sub-regiões e respostas Haar Wavelets em direções dx horizontal e vertical dy são adicionadas em cada sub-região e os valores absolutos $|dx|$ e $|dy|$ são somados a fim de obter informação sobre a polaridade das alterações de intensidade. O resultado final é um vetor de descritores

de 64 valores (16x4) com as respectivas respostas desse processo, conforme demonstra a Figura 2.21.

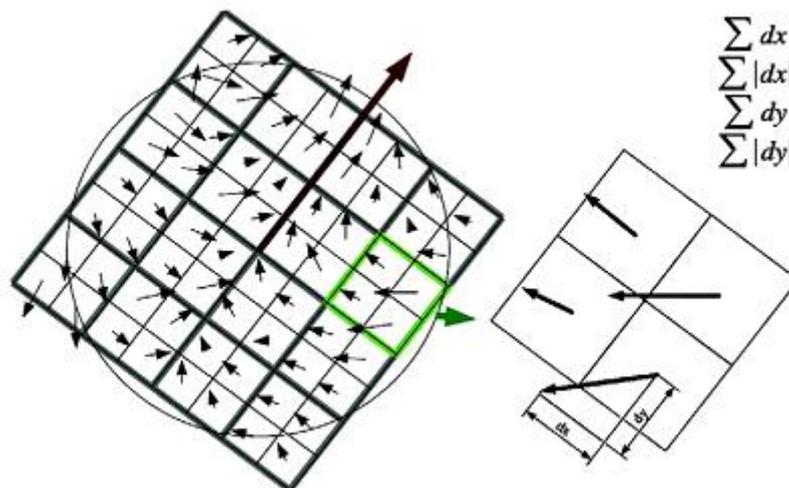


Figura 2.21: Representação do vetor com as sub-regiões e suas respectivas respostas na janela utilizada. Extraído de (BAY et al., 2008).

Assim para cada ponto de interesse haverá um vetor de 64 posições descrevendo a forma como a imagem varia nesse ponto. A correspondência entre os descritores obtidos das imagens de referência e uma imagem de consulta é realizada através da medição de distância Euclidiana.

2.7 Reconhecimento de Padrões

O reconhecimento de padrões consiste em detectar um padrão ou objeto de referência em uma imagem digital. Segundo Conci, Azevedo e Leta (2008) o reconhecimento é o processo de atribuição de um rótulo a um objeto, baseado nas características extraídas do objeto que são representadas por seus descritores.

Um sistema de reconhecimento de padrões pode ser dividido em duas etapas: extração de características e classificação (MARQUES, 2005). A etapa de extração de características pode ser realizada por descritores locais de imagens, por exemplo, o método SURF apresentado na seção anterior realiza essa tarefa. Já a etapa de classificação pode ser realizada por técnicas de casamento de modelos,

por exemplo, o método de busca exaustiva. De acordo com Lowe (2004) é o método mais eficiente para identificar os vizinhos mais próximos em espaços de grande dimensão. Outra forma consiste no método Fast Library for Approximate Nearest Neighbors (FLANN) (MUJA; LOWE, 2009), uma biblioteca que possui um conjunto de algoritmos para realizar buscas rápidas aproximadas, utilizando o método do vizinho mais próximo.

O método K-Nearest Neighbors (k-vizinhos mais próximos ou K-NN) se encontra entre os mais simples e mais largamente utilizados métodos em classificação de padrões citados na literatura (COVER; HART, 1967; TAN; STEINBACH; KUMAR, 2006). Ele é um classificador não paramétrico, o qual não requer aprendizagem. O que diferencia de outros classificadores, por exemplo, Support Vector Machine (SVM) que requer aprendizagem. Existem duas escolhas importantes a serem feitas ao usar esse método: o valor de k e a função de distância a ser utilizada. Para o caso base, por exemplo, $k = 1$, seu funcionamento consiste em que para cada padrão apresentado, atribuir ao mesmo a classificação do seu vizinho mais próximo. A Figura 2.22 demonstra esse conceito.

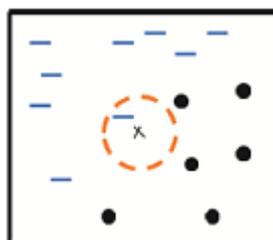


Figura 2.22: Regra do vizinho mais próximo com $k = 1$. Extraído de (TAN; STEINBACH; KUMAR, 2006).

Como pode ser observado na Figura 2.22 o padrão será classificado como sendo da classe "—" uma vez que foi definido $k = 1$ vizinho. No entanto pode-se variar o valor de k , de forma a considerar não apenas o vizinho mais próximo, mas os k vizinhos mais próximos, por exemplo, $k = 3$. Assim, cada novo padrão será classificado segundo a classificação da maioria de seus k vizinhos. A Figura 2.23 demonstra essa alteração.

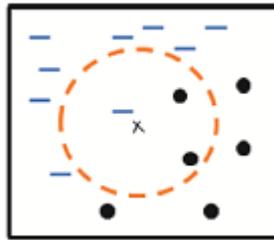


Figura 2.23: Regra do vizinho mais próximo com $k = 3$. Extraído de (TAN; STEINBACH; KUMAR, 2006).

Para evitar empates, pode-se utilizar sempre valores ímpares de k . No entanto o procedimento não é obrigatório, podendo-se utilizar valores pares de k , por exemplo, $k = 2$, decidindo de maneira aleatória em caso de empate. A Figura 2.24 demonstra essa outra alteração.

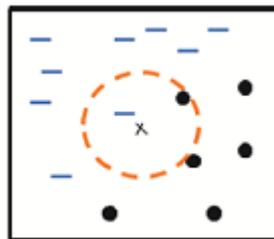


Figura 2.24: Regra do vizinho mais próximo com $k = 2$. Extraído de (TAN; STEINBACH; KUMAR, 2006).

Sendo o valor de k um problema a ser averiguado em cada aplicação, pois se k for muito pequeno, a classificação ficará sensível a ruído, já se for muito grande, a vizinhança pode incluir elementos de outras classes. Outro fator é que esse método é bastante flexível.

2.8 RANSAC

O Random Sample Consensus (RANSAC) (FISCHIER; BOLLES, 1981) é um método de estimação robusto projetado para extração dos *inliers*¹ e *outliers*² do conjunto de pontos de interesse. Um número mínimo de pontos do conjunto é usado para uma primeira estimativa e continua o processo, aumentando o conjunto de pontos de dados consistentes. O número mínimo de pontos do conjunto depende do modelo a ser estimado, caso o modelo a ser estimado seja uma reta, um número mínimo de pontos consistirá de dois elementos. A Figura 2.25 ilustra um exemplo de estimativa de uma reta ajustada a partir de um conjunto de pontos 2D usando RANSAC, onde dois pontos são selecionados de forma aleatória definindo uma reta, esta seleção aleatória se repete varias vezes e a reta com maior numero de pontos dentro de um limiar T de distância é considerada como ajuste robusto.

¹ *inliers*: pontos de dados que se ajustam com um determinado modelo desejado dentro de uma certa tolerância de erro;

² *outliers*: pontos de dados que não se ajustam ao modelo correspondente ao objeto desejado, estão fora de uma certa tolerância de erro.

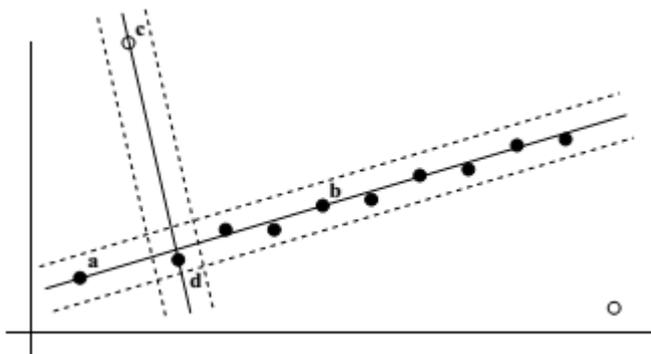


Figura 2.25: Estimação de uma reta ajustada a um conjunto de pontos 2D usando RANSAC. Extraído e adaptado de (HARTLEY ; ZISSERMAN, 2003).

Os pontos sólidos da Figura 2.25 representam os *inliers*, estes estão dentro da distância do limiar T e os pontos abertos representam os *outliers*, as linhas entre os pontos representam a distância do limiar T , a linha passando pelo ponto a e b é a linha ajustada, o ponto c é um *outlier*.

Este método tem sido muito usado para o reconhecimento de objetos (COLLET; BERENSON; SRINIVASA, 2009; OKABE; SATO, 2003), pois permite encontrar correspondências geometricamente consistentes, mesmo em condições extremas ou com algum tipo de *outlier*. Sendo muito importante a adoção do mesmo em uma solução que usa detector de ponto de interesse para geração da homografia, pois o detector pode cometer erros, tais como erros de medida e classificação. Erros de medida podem ocorrer quando o detector identifica pontos de interesse de forma correta, no entanto podem cometer erros de cálculos em um de seus parâmetros, por exemplo, sua localização na imagem. Já erros de classificação podem ocorrer quando um detector de pontos de interesse identifica uma porção da imagem de forma incorreta (FISCHIER; BOLLES, 1981).

2.9 Arquitetura de Processamento

Nesta seção é apresentado uma visão geral de arquitetura das Central Processing Units (CPUs) e Graphics Processing Units (GPUs) bem como da biblioteca OpenCV.

2.9.1 CPUs e GPUs

Os avanços tecnológicos na fabricação de processadores gráficos e o paralelismo das operações envolvidas no processamento de imagens em tempo real transformaram as Graphics Processing Units (Unidades de Processamento Gráfico ou GPUs) em recursos computacionais extremamente atraentes. Tal potencial de paralelismo das GPUs é devido ao maior número de transistores dedicados a Arithmetic Logic Units (Unidade Lógica Aritmética ou ALUs), em vez de controle de fluxo e cache, comparado com as Central Processing Units (Unidades Centrais de Processamento ou CPUs). A Figura 2.26, compara o avanço do poder computacional de CPUs e GPUs.

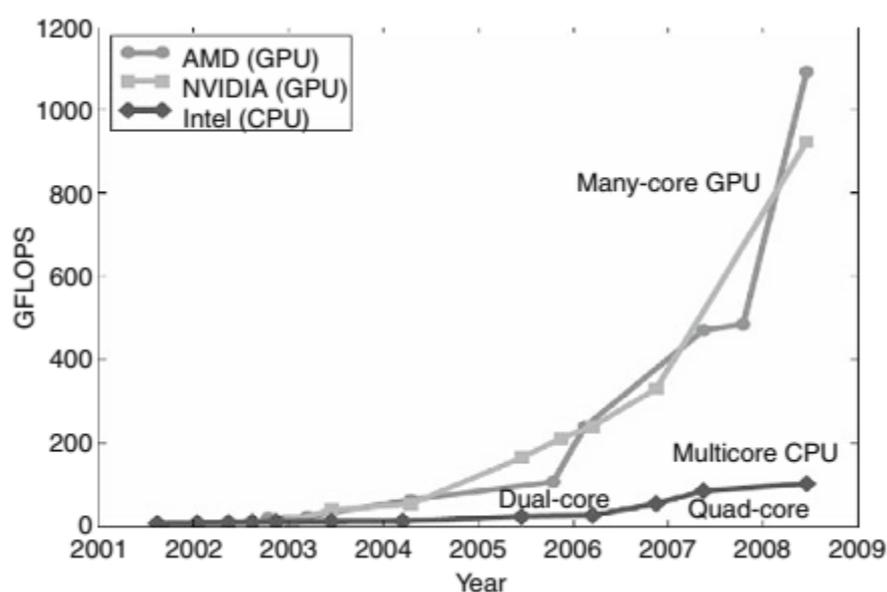


Figura 2.26: Avanço do poder computacional de CPUs e GPUs. Extraído de (KIRK; HWU, 2010).

A discrepância no desempenho entre CPUs e GPUs observado no gráfico, pode ser atribuído a diferença na arquitetura. Portanto, na maioria das vezes um algoritmo que utiliza GPU tende a atender melhor a necessidade de um sistema em tempo real que aquele que utiliza somente CPU. A Figura 2.27 apresenta uma comparação de utilização da CPU e GPU. A arquitetura de uma CPU possui mais memória principal e reserva uma maior área para o controle de fluxo e a cache, sendo portanto mais adequada para tarefas sequenciais. Já a arquitetura de uma GPU possui uma maior área dedicada para as ULAs, conseqüentemente um maior potencial para executar operações de ponto flutuante, sendo portanto mais adequada para tarefas com paralelismo de maneira Single Instruction Multiple Data (SIMD).

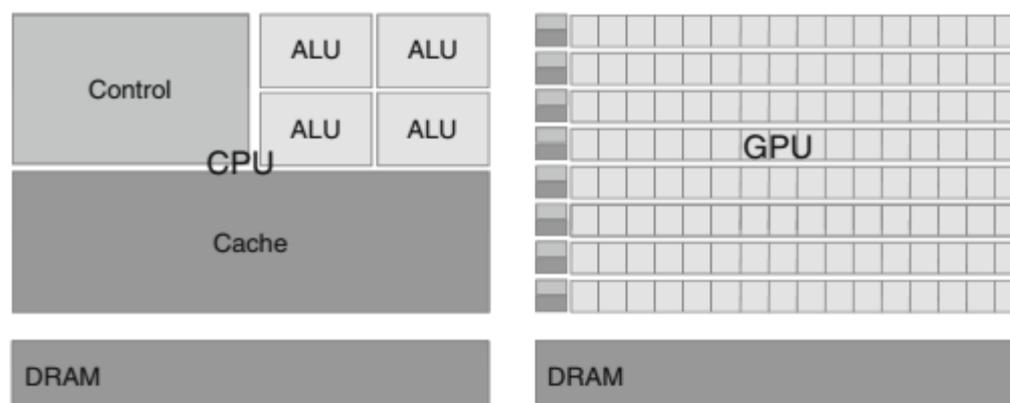


Figura 2.27: Arquitetura das CPUs e GPUs. Extraído de (KIRK; HWU, 2010).

A utilização de GPUs para computação de propósito geral é conhecida como General Purpose Computing GPU (GPGPU). Atualmente existem várias plataformas de programação GPGPU, entre elas se destacam o Compute Unified Device Architecture (CUDA) (NICKOLLS et al. 2008) e Open Computing Language (OpenCL) (STONE; GOHARA; SHI, 2010), ambas possibilitam desenvolver aplicações em uma linguagem de programação de mais alto nível. A principal diferença entre as plataformas é que os programas CUDA são executados somente em placas de vídeo NVIDIA, já os programas OpenCL podem ser executados em diversas plataformas. Uma importante biblioteca focada em processamento de imagens e visão computacional, que é otimizada para CPU Intel e possui recursos para CUDA em GPU, é a Open Source Computer Vision Library (OpenCV). Esta

biblioteca é principalmente utilizada em empresas, grupos de pesquisas e órgãos governamentais.

A biblioteca OpenCV ³ possibilita executar um algoritmo de forma otimizada em CPU Intel e acelerar ainda mais a execução do mesmo utilizando o módulo OpenCV CUDA. Este módulo é composto por um conjunto de classes e funções baseadas na API NVIDIA CUDA, estas utilizam kernels especializados para arquiteturas CUDA modernas de GPUs e fluxo de dados otimizado, possibilitando obter alto desempenho. No entanto, para utilizar os recursos desta biblioteca é preciso obter recursos de hardware e software necessários para o desenvolvimento, possuir conhecimento necessário para configuração bem como para uso adequado das funções e definir quais serão as instruções que serão executadas na CPU e GPU.

2.10 Placa de Identificação de Veículos

A placa veicular brasileira é atualmente regulamentada pelas Resoluções 231⁴ e 241⁵ do Conselho Nacional de Trânsito (CONTRAN) e está em vigor desde 1º de janeiro de 2008, estas resoluções definem especificações técnicas para as placas de identificação de veículos, dentre as especificações a forma e dimensões em milímetros. Segundo estas resoluções as placas de identificação de veículos devem possuir as seguintes dimensões: (a) 130 mm de altura; (b) 400 mm de comprimento, conforme mostra a Figura 2.28.

³ <http://www.opencv.org>

⁴ http://www.denatran.gov.br/download/Resolucoes/RESOLUCAO_231.pdf

⁵ http://www.denatran.gov.br/download/Resolucoes/RESOLUCAO_CONTRAN_241.pdf



Figura 2.28: Dimensões para placas de identificação de veículos. Extraído de (CONTRAN, 2007).

Como pode ser observado na Figura 2.28 a placa veicular possui a forma polígono quadrilátero conforme as dimensões descritas anteriormente. O interesse na placa se deve ao fato dela ser usada como referência para definição da ROI bem como de conter informação na identificação de veículos.

2.11 Logomarcas de Fabricantes de Veículos

Atualmente atuam no Brasil diversos fabricantes de veículos. No segmento de automóveis particulares há uma forte presença de empresas estrangeiras, dentre os fabricantes de veículos mais conhecidos no mundo atuam no mercado brasileiro: Volkswagen e Audi (Alemanha), Fiat (Itália), Chevrolet e Ford (Estados Unidos), Peugeot (França), Hyundai (Coreia do Sul), Toyota, Honda e Nissan (Japão).

Segundo a revista *Pense Carros*⁶ as marcas de veículos particulares mais vendidas do mundo no ano de 2013 foram Toyota, Volkswagen, Ford, Hyundai, Chevrolet e Honda, seguindo essa ordem no ranking de vendas.

⁶ <http://revista.pensecarros.com.br/especial/rs/editorial-veiculos/19,521,4402440,Confira-o-ranking-dos-dez-carros-mais-vendidos-no-mundo.html>

No Brasil as marcas de veículos Volkswagen e Fiat são as mais vendidas. A Tabela 2.1 mostra dez marcas de fabricantes de veículos particulares do mundo que estão presentes no Brasil. Essa tabela é composta também de algumas logomarcas que foram mais utilizadas em abordagens referenciadas neste trabalho.

Tabela 2.1: Marcas de dez fabricantes de veículos particulares do mundo que estão presentes no Brasil.

Toyota		Volkswagen	
Peugeot		Hyundai	
Audi		Honda	
Nissan		Chevrolet	
Fiat		Ford	

Como pode ser observado na Tabela 2.1 cada logomarca possui características que a diferem uma das outras.

2.12 Avaliação de desempenho

Diferentes medidas que permitem a comparação de algoritmos para o reconhecimento de objetos podem ser encontradas na literatura. As mais utilizadas são as taxas de acertos e erros, curvas de precisão e curvas ROC (Receiver Operating Characteristic). Estas medidas são calculadas a partir da tabela de contingência ou matriz de confusão (FAWCETT, 2005). A Tabela 2.2 mostra a tabela de contingência para avaliação de desempenho de algoritmos de reconhecimento de objetos.

Tabela 2.2: Tabela de continência. Extraído e adaptado de (FAWCETT, 2005).

		Uma instância do objeto de referência está presente na imagem analisada	
		Positivo (Sim)	Negativo (Não)
Uma instância do objeto de referência foi detectada na imagem analisada	Positivo (Sim)	Verdadeiro Positivo (VP)	Falso Positivo (FP)
	Negativo (Não)	Falso Negativo (FN)	Verdadeiro Negativo (VN)

Onde:

- VP: ocorre quando o algoritmo detecta uma instância do objeto de referência que está presente na imagem analisada.
- FP: ocorre quando o algoritmo detecta uma instância do objeto de referência que não está presente na imagem analisada.
- FN: ocorre quando o algoritmo não detecta uma instância do objeto de referência que está presente na imagem analisada.
- VN: ocorre quando o algoritmo não detecta uma instância do objeto de referência que não está presente na imagem analisada.

A métrica de avaliação usada no trabalho foi a taxa de acerto, apresentada pela seguinte equação:

$$\text{taxa de acertos} = \frac{VP}{VP + FN} * 100 \quad (2.17)$$

2.13 Considerações Finais

Neste capítulo, foram abordados os conceitos fundamentais relacionados ao trabalho desenvolvido.

Capítulo 3

TRABALHOS CORRELATOS

Este capítulo apresenta algumas abordagens encontradas na literatura que possuem certa relação com o trabalho desenvolvido nesta pesquisa.

3.1 Considerações Iniciais

Na literatura foram encontradas diversas abordagens relacionadas ao problema de reconhecimento de logomarcas de veículos. O tema tem sido abordado pela comunidade acadêmica, o que demonstra sua relevância. A seguir são apresentadas as principais abordagens relacionadas com este trabalho.

3.2 Abordagens

Badura e Foltan (2010) utilizaram o método SURF para fazer o reconhecimento de logomarcas de veículos, destacando a importância do uso do algoritmo SURF no problema de reconhecimento de logomarcas de veículos bem como de métodos que possibilitem melhores resultados, realizando o reconhecimento de 45% a quase 100% das logomarcas nas imagens analisadas para 11 classes de logomarcas. Já Psylos, Anagnostopoulos e Kayafas (2011) combinaram o método *Sliding Concentric Window* (SCW) com o método de Sauvola e Pietikainen (2000), Anagnostopoulos (2006), Kovesi (1999) e uma *Probabilistic*

Neural Network (PNN), destacando a necessidade de métodos rápidos e precisos, realizando o reconhecimento de 85% das logomarcas em 154 ms a 793 ms por imagem analisada para 10 classes de logomarcas. Mais tarde em Lipikornet al. (2014) propuseram um método baseado no SIFT e rede neural de *back-propagation*, realizando o reconhecimento de 55% a 80% das logomarcas nas imagens analisadas para 5 classes de logomarcas.

Sidla, Kottmann e Benesova (2011) ao avaliar varias combinações de detectores (SURF, FERNS) e descritores (Harris, SURF, FAST, LDETECT) de pontos de interesse para detecção de logotipos, constatou que a combinação SURF-SURF, ou seja, do detector SURF com o descritor SURF apresenta o melhor desempenho mesmo em imagens de baixa qualidade ou em ambiente externo, observando que a adoção SURF-SURF seria a melhor opção até o momento, mantendo um melhor compromisso em termos de robustez e velocidade de processamento. Mais tarde Hedayatiet al. (2013) chegou a uma conclusão semelhante ao avaliar o desempenho de diferentes detectores (SIFT, SURF, ORB) e descritores (SIFT, SURF, STAR, ORB, MSER) de pontos de interesse, concluindo que a combinação do detector SURF com o descritor SURF tem uma execução mais rápida não descuidando, contudo, da robustez essencial para o reconhecimento de objetos.

No entanto, devido a importância do algoritmo SURF e de o mesmo não ser rápido o suficiente para melhor atender a exigência de um sistema em tempo real, há também atualmente um grande interesse da comunidade científica em facilitar adoção do mesmo, tais como a abordagem realizada por Huijuam e Qiong (2011) que fizeram a combinação do algoritmo FAST com o algoritmo SURF para acelerar a velocidade de processamento do SURF no reconhecimento de imagens, observando a falta de flexibilidade do algoritmo proposto em reconhecer imagens em diferentes distâncias, devido a falta de escala do algoritmo FAST. Já Xing, Jung-Hae e Byung-Jae (2012) projetaram um algoritmo de substituição e combinaram com o descritor SURF para acelerar o processo de computação dos pontos de interesse. Mais tarde Schweigeret al. (2013) fizeram a linearização do detector SURF para melhorar a velocidade de processamento do algoritmo sem alterar suas características originais. Sendo ainda possível encontrar na literatura outras propostas, tais como, a abordagem realizada por Herakleous e Paullis (2013) que combinaram o SURF com

Fluxo Óptico em um processo otimizado para aplicação em realidade aumentada. Já Cho e Choi (2012) além de combinar o SURF com Fluxo Óptico combinou também com Filtro de Kalman para realizar o reconhecimento e rastreamento de objetos. Mais tarde Chaturvedi e Sethi (2013) utilizou o SURF com Rede Neurais para o problema de reconhecimento de placas de veículos.

Retornando ao problema de reconhecimento de logomarcas de fabricantes de veículos, é possível encontrar na literatura outras propostas, tais como a abordagem de An, Thakoor e Bhanu (2012) que utilizaram um método de super-resolução por Análise de Correlação Canônica (CCA) para realizar o reconhecimento de logomarcas de veículos, realizando o reconhecimento de 80% a 100% e 46.67% a 81.33% das logomarcas nas imagens analisadas para 5 classes de logomarcas. Mais tarde Liu et al. (2014) propuseram um método baseado no *Kernell₂ Regularized Least Square* (RLS) e em um dicionário adaptativo para melhorar o desempenho no reconhecimento de logomarca de veículos, realizando o reconhecimento de logomarcas em 1.20 s por imagem analisada para 21 classes de logomarcas. Nan (2012) combinaram os métodos *Principal Component Analysis* (PCA) e *Kernel Principal Component Analysis* (KPCA) com vários classificadores, realizando o reconhecimento de 95% a 97.5% e 91.7% a 97.9% de logomarcas nas imagens analisadas para 8 classes de logomarcas. Já Zhang, Shen e Chang (2013) apresentaram um método baseado em mapas de saliências e casamento de modelos a fim de realizar uma detecção rápida da logomarca do veículo, realizando o reconhecimento de 1.31% a 94.37% e 1.32% a 75,43% das logomarcas em 210 ms por imagem analisada para 6 classes de logomarcas.

Pan e Zhang (2013) utilizaram um algoritmo AdaBoost melhorado com características Haar-like e *Histogram of Oriented Gradients* (HOG) para realizar a detecção do veículo, em seguida uma *Region of Interest* (ROI) que contém o logotipo foi determinada com base na localização da placa do veículo, em por fim a classificação da região do logotipo foi determinada usando *Gentle AdaBoost* e *Support Vector Machine* (SVM), realizando o reconhecimento de 75,90% a 95,18% das logomarcas em 1.42 s por imagem analisada para 11 classes de logomarcas. Já Sam e Tian (2012) apresentaram uma solução para localização e reconhecimento de logotipo de veículos usando um algoritmo de aprendizado de máquina *Modest AdaBoost* para localização do logotipo e *Radial Tchebichef Moments* para realizar o

reconhecimento, realizando o reconhecimento de 92% das logomarcas em 2.2 s por imagem analisada para 10 classes de logomarcas.

Em virtude das propostas citadas acima pode-se destacar algumas observações importantes: as abordagens de Psyllos, Anagnostopoulos e Kayafas (2011) e Zhang, Shen e Chang (2013) tem a vantagem de lidar com uma ROI que contém uma maior quantidade de informação, no entanto a necessidade de processamento aumenta, bem como o nível de complexidade exigido para atingir resultados mais precisos. Os trabalhos de Lipikornet al. (2014), Psyllos, Anagnostopoulos e Kayafas (2011), Pan e Zhang (2013) e Sam e Tian (2012) gastam bastante tempo na fase de treinamento para conseguir uma representatividade forte de amostras. Já Sidla, Kottmann e Benesova (2011) e Hedayatiet al. (2013) destacam o método SURF frente a outras opções da literatura, por outro lado há trabalhos (HUIJUAM; QIONG, 2011; XING; JUNG-HAE; BYUNG-JAE, 2012; SCHWEIGER et al., 2013) visam otimizar o tempo de execução do algoritmo realizando modificações no mesmo, o que acaba na maioria das vezes alterando as suas características, outros propõem novos algoritmos ou combina com outras técnicas, visando uma aplicação específica, apresentando novas opções de contornar o problema. Apesar de novos algoritmos ser importante, para que a proposta de um novo algoritmo seja efetivamente estabelecida é necessário que uma grande parte da comunidade científica adote tal algoritmo. Já a combinação de técnicas (HERAKLEOUS; PAULLIS, 2013; CHO; CHOI, 2012; CHATURVEDI; SETHI, 2013) proporciona novas opções de contornar o problema por meio de aplicações específicas, no entanto é importante observar o domínio do problema devendo ser avaliado em cada caso. No trabalho descrito por Badura e Foltan (2010), o método SURF foi utilizado para o problema de reconhecimento de logomarcas de veículos. Embora o algoritmo SURF possa ser adequado para o problema, o sucesso de aplicação do mesmo depende da integração de outros métodos para diminuir as possibilidades de falsas identificações bem como o tempo de processamento. Outra observação é que os trabalhos acima não descrevem muitos detalhes sobre as imagens usadas nos experimentos.

3.3 Considerações finais

Na Tabela 3.1 é apresentada uma síntese das abordagens que foram apresentadas neste capítulo com os principais métodos utilizados no processo de reconhecimento de logomarcas, bem como as características da abordagem proposta.

Tabela 3.1: Abordagens propostas e métodos empregados

Abordagem	Principais métodos	Imagens de veículos testadas	Classes de logomarcas	Taxa de reconhecimento	Tempo de Treinamento (TT) / Reconhecimento (TR)	Processador
Badura e Foltan (2010)	- ROI - SURF	Imagens fixas frontais de 640x480	11	45% a 100%	TR CPU: não informado	não informado
Psyllos, Anagnostopoulos e Kayafas (2011)	- Sliding Concentric Window (SCW) - (SAUVOLA; PIETIKAINEN, 1999) - (ANAGNOSTOPOULOS, 2006) - (KOVESI, 1999) - Probabilistic Neural Network	110 imagens fixas frontais	10	85%	TT CPU: 576 ms a 1428 ms TR CPU: 154 ms a 793 ms	Pentium IV de 2.0 GHz

	(PNN)					
Lipikornet al. (2014)	- SIFT - Rede neural de back-propagation	220 imagens fixas frontais de 800x600 em boas condições de iluminação.	5	55% a 80%	TT CPU: não informado TR CPU: não informado	não informado
An, Thakoor e Bhanu (2012)	- Canonical Correlation Analysis (CCA)	Imagens fixas frontais	5	80% a 100% e 46.67% a 81.33%	TR CPU: não informado	não informado
Liu et al. (2014)	- Kernel l_2 Regularized Least Square (RLS) - Dicionário adaptativo	Imagens fixas frontais	21	não informado	TR CPU: 1.20 s	não informado
Nan (2012)	- Principal Component Analysis (PCA) - Kernel Principal Component	40 e 48 imagens fixas obtidas por uma câmera fotográfica	8	95% a 97.5% e 91.7% a 97.9%	TR CPU: não informado	não informado

	Analysis(KPCA)	SAMSUNG ST95.				
Zhang, Shen e Chang (2013)	- Mapas de saliências - Casamento de modelos	2413 imagens fixas frontais de 1600 x 1232	6	1.31% a 94.37% e 1.32% a 75.43%	TR CPU: 210 ms	Intel Core2 Duo CPU E4400 2.00 GHz/ 2G RAM
Pan e Zhang (2013)	- Haar-like - Histograms of Oriented Gradients (HOG) - Sobel - Morfologia - ROI - Gentle Adaboost - Support Vector Machine (SVM)	274 imagens fixas frontais de 320x320	11	75.90% a 95.18%	TT CPU: não informado Tempo de detecção CPU: 1.42 s	não informado
Sam e Tian (2012)	- Modest AdaBoost - Radial Tchebichef	200 imagens fixas frontais de 640x480	10	92%	TT CPU: não informado TR CPU: 2.2 s	Intel Core2 Duo 2.66 GHz e 2GB RAM

	moments					
Abordagem proposta	<ul style="list-style-type: none"> - Canny - Conci, Carvalho e Rauber(2009) - ROI - SURF - KNN - RANSAC 	191 imagens fixas - 68 frontais e 123 traseiras de 640x480	10 a 15	45% a 100%	TR CPU: 84.26 ms a 133.73 ms TR GPU-CPU: 73.96 ms a 136.41 ms	CPU Intel Core i5 e GPU NVIDIA GeForce GT 740M de 2GB

Capítulo 4

MATERIAL E MÉTODOS

Este capítulo apresenta os materiais, projeto e as etapas realizadas para o desenvolvimento do método proposto para reconhecimento de logomarcas de veículos usando SURF.

4.1.1 Considerações iniciais

O presente trabalho propõe um método alternativo para o problema de reconhecimento de logomarcas de veículos através de um modelo de utilização eficiente do algoritmo SURF.

4.1.2 Recursos de desenvolvimento

O método foi implementado utilizando diversos recursos: um computador DELL VOSTRO 5470, com 4 GB de memória RAM, sistema operacional Windows 8.1 64 Bits, CPU Intel Core i5, GPU NVIDIA GeForce GT 740M de 2GB, linguagem C++, IDE Visual Studio, CMake para configuração do ambiente de desenvolvimento com auxílio da biblioteca OpenCV e CUDA SDK.

4.1.3 Base de imagens

Os testes foram realizados em 4 bases contendo 191 imagens de veículos estacionados com iluminação diversificada (imagens com dia ensolarado, nublado e com sombra) e com diferentes ângulos de visão. As imagens das Bases I e II foram testadas para 10 classes de logomarcas: Volkswagen, Toyota, Ford, Hyundai, Chevrolet, Honda, Audi, Fiat, Nissan e Peugeot. A Base I contém 10 imagens de veículos nacionais estacionados próximos (uma imagem para cada classe). A Base II contém 30 imagens de veículos nacionais estacionados mais afastados (três imagens para cada classe). As imagens das Bases I e II foram obtidas utilizando a câmera digital CASIO, modelo EX-Z75. As imagens da Base III foram testadas para 15 classes de logomarcas: Volkswagen, Toyota, Ford, Hyundai, Opel, Honda, Audi, Fiat, Nissan, Peugeot, BMW, Citroen, Renault, Seat e Outras. A Base III contém 60 imagens de veículos estrangeiros estacionados próximos e mais afastados obtidas da base Medialab² (quatro imagens para cada classe). As imagens da base IV foram testadas para 13 classes de logomarcas: Volkswagen, Toyota, Ford, Opel, Honda, Audi, Mercedes-Benz, Nissan, Peugeot, BMW, Renault, Seat e Outras. A base IV contém 91 imagens sendo a maioria de traseiras de veículos estrangeiros estacionados próximos e mais afastados obtidas da base Zemris⁷ (7 imagens para cada classe). A Tabela 4.1 descreve as características das bases de imagens.

⁶ <http://www.medialab.ntua.gr/research/LPRdatabase.html>

⁷ <http://www.zemris.fer.hr/projects/LicensePlates/english/results.shtml>

Tabela 4.1: Características das bases de imagens usadas

Imagens de veículos	Base I	Base II	Base III	Base IV
Total	10	30	60	91
Resolução	640x480	640x480	640x480	640x480
Frontal	7	18	39	4
Traseira	3	12	21	87
Dia ensolarado	4	13	23	11
Dia nublado	2	5	13	48
Com sombra	4	12	24	33
Angulação > 30°	3	10	26	22
Distância da Câmera	1,5m a 2,5 m	1,5m a 3,5m	não informado	não informado

4.2 Projeto e desenvolvimento do método proposto

Esta seção explica em detalhes cada etapa do fluxograma do método descrito abaixo.

A solução proposta neste trabalho consiste em um método para realizar o reconhecimento de logomarcas de veículos de forma rápida e robusta, baseada no algoritmo SURF. A Figura 4.1 demonstra as etapas do método proposto.

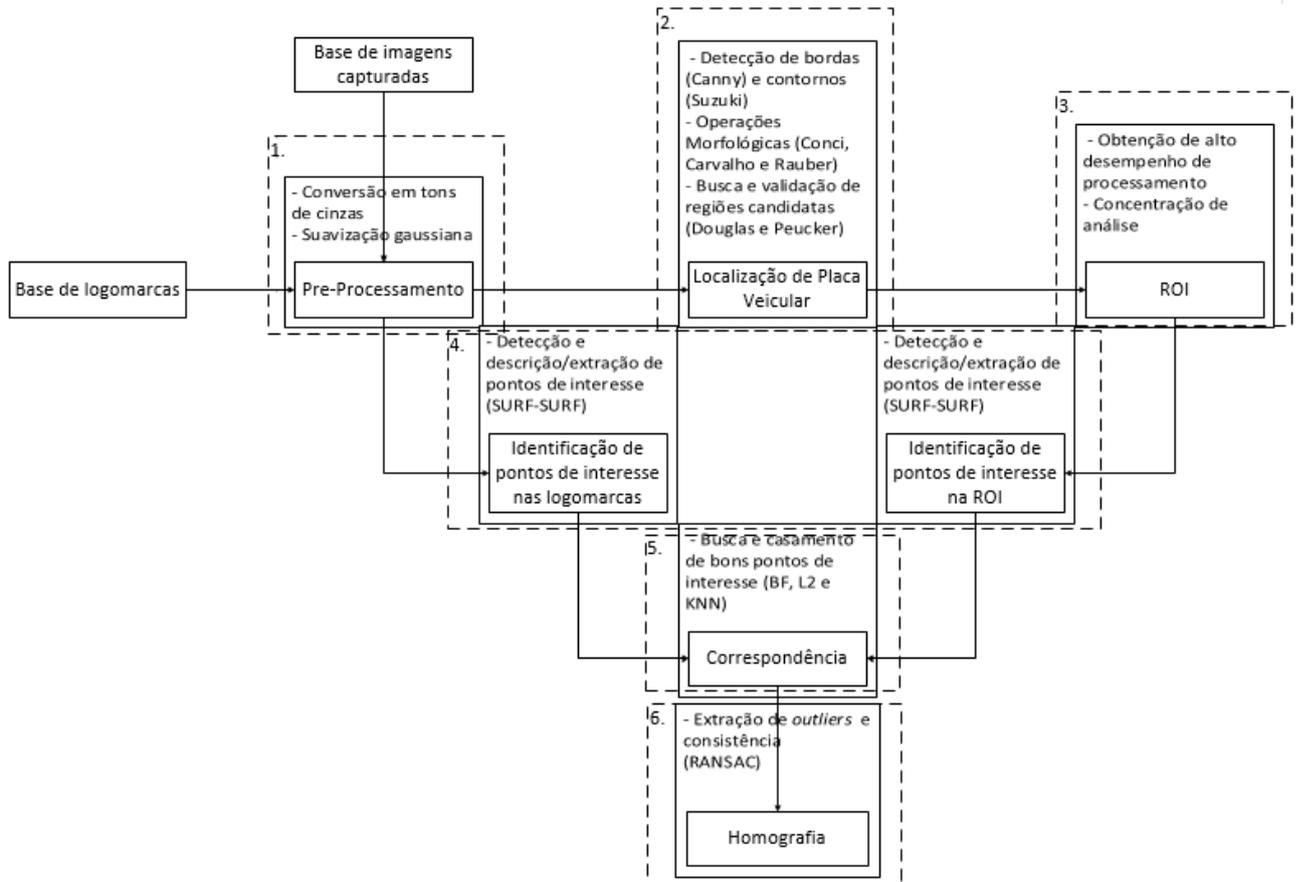


Figura 4.1: Etapas do método proposto

4.2.1 Descrição das etapas

Base de imagens capturadas e base de logomarcas: consiste nas imagens adquiridas e um número de classes de logomarcas. Assim como a Base III e IV, a aquisição das imagens das Bases I e II foi feita em RGB para possibilitar que outros trabalhos sejam desenvolvidos utilizando as mesmas bases e abordando outros problemas. A Figura 4.2 mostra a imagem capturada.



Figura 4.2 : Imagem capturada

1. **Pré-processamento:** tem como objetivo melhorar a imagem, de forma a auxiliar os processos seguintes. Nesta etapa é realizada a conversão da imagem RGB capturada em tons de cinzas bem como a suavização gaussiana da mesma para diminuição de ruído. A Figura 4.3 mostra a imagem em tons de cinzas e a Figura 4.4 mostra a imagem suavizada.



Figura 4.3: Imagem em tons de cinzas



Figura 4.4 : Imagem suavizada

- 2. Localização de placa veicular:** a localização da placa do veículo é usada como referência para definir uma região de interesse na imagem analisada. Nesta etapa é aplicado o operador de Canny (1986) para detectar as bordas da imagem. As bordas caracterizam os contornos dos objetos, sendo muito úteis no processo de segmentação e localização de regiões candidatas. A Figura 4.5 mostra a imagem após a aplicação do detector de bordas.

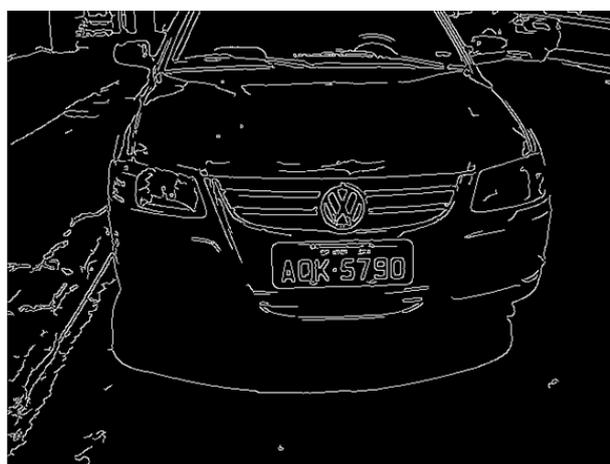


Figura 4.5 : Imagem com as bordas destacadas

Os contornos dos objetos são detectados usando o método de Suzuki (1985) e representação poligonal. Os contornos são então organizados em uma árvore e manipulados de forma a obter informações de área e aproximação poligonal. O método de Douglas e Peucker (1973) é utilizado

para realizar a simplificação no processo de aproximação. Informações de ângulo e razão também são utilizadas no processo de busca e validação de regiões candidatas a placa veicular. Caso uma determinada região candidata a placa veicular seja validada, é desenhado um retângulo verde indicando sua localização. A Figura 4.6 mostra a placa do veículo localizada na imagem.



Figura 4.6 : Placa do veículo localizada na imagem

Caso contrário é aplicado o método de Conci, Carvalho e Rauber (2009) adaptado com o método anterior. A adaptação é realizada após a utilização do detector de bordas de Canny. A Figura 4.7 mostra uma outra imagem após aplicação do detector de bordas.



Figura 4.7 : Imagem com a aplicação do operador de Canny

O método de Conci, Carvalho e Rauber (2009) é utilizado para realizar uma segmentação por operações morfológica binária. Aplica-se o fechamento com um elemento estruturante horizontal linear, com a largura igual à maior separação entre os caracteres possíveis para converter os caracteres da placa em uma região branca. A Figura 4.8 mostra a imagem com o fechamento.

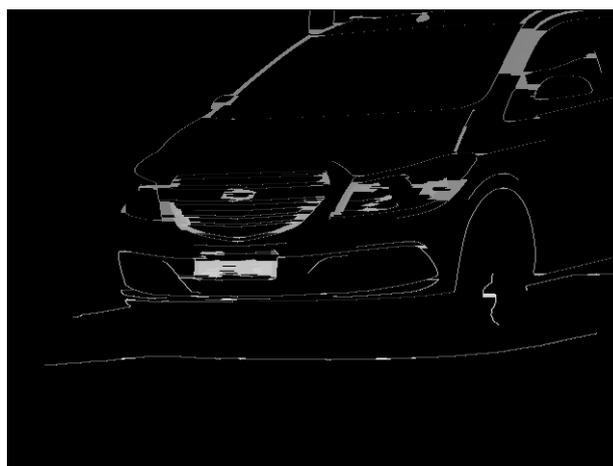


Figura 4.8 : Imagem com fechamento

Em seguida é realizado uma abertura com um elemento estruturante vertical linear para eliminar os objetos de altura inferior à altura mínima dos caracteres. A Figura 4.9 mostra a imagem com abertura mínima.

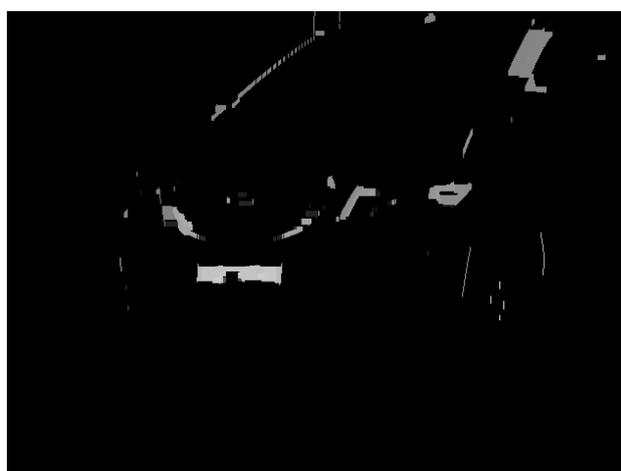


Figura 4.9 : Imagem com abertura mínima

Depois é realizada outra abertura para eliminar os objetos de altura inferior a altura máxima dos caracteres. Eliminando desta forma também a região da placa. A Figura 4.10 mostra a imagem com abertura máxima.

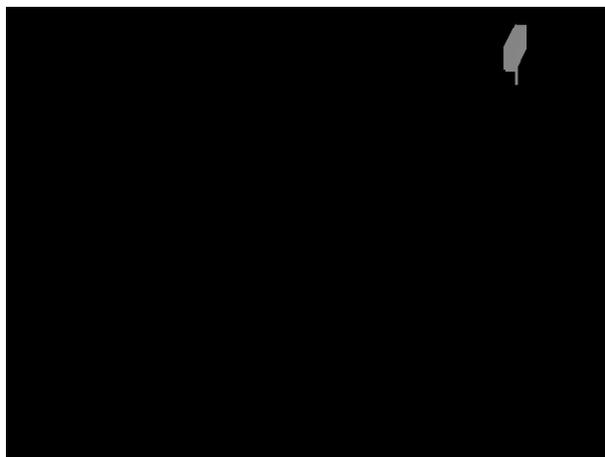


Figura 4.10 : Imagem com abertura máxima

A imagem do passo anterior Figura 4.9 é subtraída com a imagem do passo atual Figura 4.10 obtendo uma imagem resultante contendo apenas a região da placa e um pouco de ruído residual. A Figura 4.11 mostra a imagem subtraída.

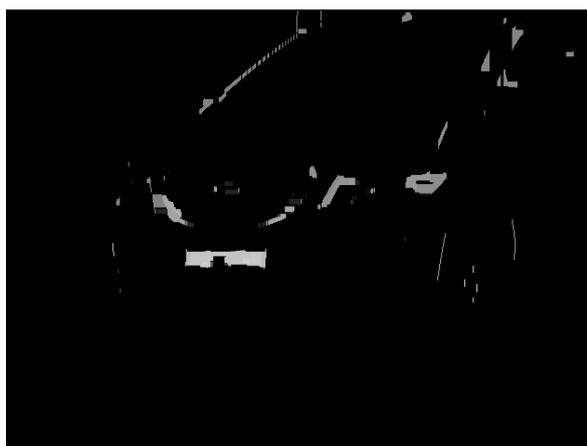


Figura 4.11 : Imagem subtraída

Na sequência é realizado uma abertura com um elemento estruturante horizontal linear de largura menor do que a largura mínima da placa para

eliminar o ruído residual. A Figura 4.12 mostra a imagem com eliminação da largura mínima.

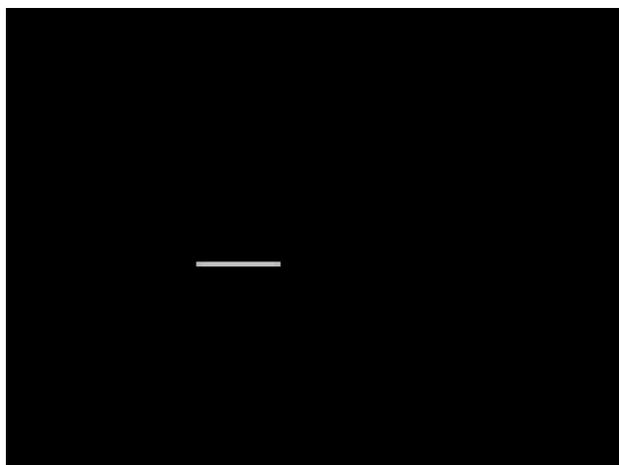


Figura 4.12 : Imagem com eliminação de largura mínima

Por fim, aplica-se uma dilatação para destacar a região da placa. Obtendo dessa forma a posição da placa na imagem capturada. A Figura 4.13 mostra a imagem dilatada.

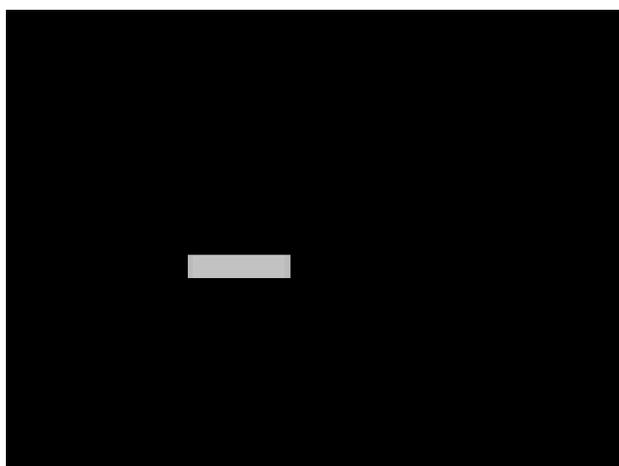


Figura 4.13 : Imagem dilatada

- 3. ROI:** a região de interesse é definida com base na localização da placa e é utilizada para se obter um alto desempenho em termos de processamento bem como concentrar a análise em uma determinada região na imagem analisada. A Figura 4.14 mostra a placa do veículo localizada e uma

região de interesse definida acima da placa. A Figura 4.15 mostra somente a ROI.



Figura 4.14 : ROI definida acima da placa



Figura 4.15 : ROI

- 4. Identificação de pontos de interesse:** nesta etapa aplica-se o método SURF (BAY et al., 2008) para detectar e descrever determinados pontos das logomarcas bem como da região de interesse na imagem analisada para obtenção de um conjunto de elementos marcados. A Figura 4.16 mostra os pontos de interesse identificados na ROI e a Figura 4.17 nas logomarcas.



Figura 4.16 : Pontos de interesse identificados na ROI



Figura 4.17 : Pontos de interesse identificados nas logomarcas

5. **Correspondência:** após a identificação dos pontos de interesse a correspondência entre descritores das logomarcas e de uma ROI da imagem capturada é analisada de modo a se afirmar se há ou não casamento entre as imagens. Para isso, aplica-se o método de busca exaustiva e o método do vizinho mais próximo por distância euclidiana realizando a busca e casamento de bons pontos de interesse. A Figura 18 mostra os pontos interesse correspondidos.

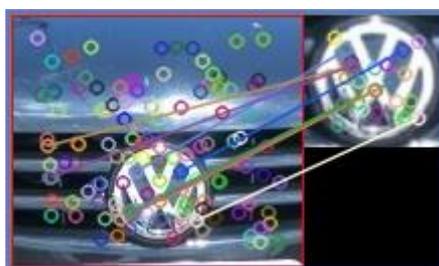


Figura 4.18 : Correspondência de pontos de interesse

6. **Homografia:** ao final do processo utiliza-se o método RANSAC (FISCHIER; BOLLES, 1981) para realizar a consistência da logomarca reconhecida na imagem analisada, excluindo todos os pontos de interesse

que não fazem parte da logomarca. A Figura 4.19 mostra a logomarca reconhecida na imagem analisada.

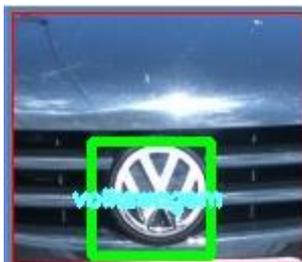


Figura 4.19 : Extração de outliers e consistência

4.3 Considerações finais

Neste capítulo foi apresentado o projeto e as etapas realizadas para o desenvolvimento do método proposto para reconhecimento de logomarcas de veículos usando SURF.

Capítulo 5

RESULTADOS E DISCUSSÕES

Este capítulo serão apresentados os resultados e as discussões sobre os experimentos realizados para a avaliação do método desenvolvido.

5.1 Considerações iniciais

Os experimentos descritos neste capítulo serviram de base para uma série de análises quantitativas, buscando-se assim avaliar a efetividade do método proposto.

Os dados coletados foram avaliados com base na tabela de continência, utilizando-se o software Excel para a geração de medidas estatísticas e gráficos para melhor visualização.

5.2 Validação do método

Foram realizados diversos experimentos em uma imagem de veículo a fim de verificar a viabilidade e definir a melhor combinação de métodos para se resolver o problema. A Tabela 5.1 apresenta os resultados dos testes (experimentos de 1 a 17) que foram realizados usando a imagem da Figura 4.2 de 640x480, os quais apresentaram 100% de taxa de acerto, destacando o Tempo de Processamento (TP) total de cada experimento realizado bem como o TP de cada técnica empregada no

método proposto (constituído por parte 1 e parte 2) usando CPU e GPU. A Figura 5.1 demonstram esses resultados.

Tabela 5.1 : Experimentos de validação

Experimento	Conversão em tons de cinzas	Suavização gaussiana	Canny	Conci, Carvalho e Rauber (2009)	Detecção de contorno	ROI	SURF	SIFT	BF L2 K=2	FLANN	RANSAC	Tempo Total (ms)
1	CPU						CPU		CPU			2070.985
2	CPU						CPU			CPU		2230.414
3	CPU							CPU	CPU			2360.580
4	CPU							CPU		CPU		2571.284
5	CPU						CPU		CPU		CPU	2068.946
6	CPU	CPU	CPU		CPU	CPU		CPU	CPU		CPU	198.060
7	CPU	CPU	CPU		CPU	CPU		CPU		CPU	CPU	288.771
8 – Método proposto parte 1 em CPU	CPU	CPU	CPU		CPU	CPU	CPU		CPU		CPU	109.501
TP (ms)	1.954	3.191	10.220		4.323	0.040	85.388		1.411		1.845	
9	CPU	CPU	CPU		CPU	CPU	CPU			CPU	CPU	197.552
10 – Método proposto parte 2 em CPU	CPU	CPU	CPU	CPU	CPU	CPU	CPU		CPU		CPU	129.787
TP em ms	1.766	3.007	9.339	65.944	1.537	0.028	39.050		0.554		1.300	
11	CPU	CPU	CPU	CPU	CPU	CPU	CPU			CPU	CPU	152.128

12 - Método proposto parte 1 em GPU	GPU	GPU	GPU		CPU	CPU	GPU		GPU		CPU	38.535
TP (ms)	1.866	1.548	6.309				4.500		1.458			
13	GPU	GPU	GPU		CPU	CPU	GPU			CPU	CPU	93.755
14 - Método proposto parte 2 em GPU	GPU	GPU	GPU	GPU	CPU	CPU	GPU		GPU		CPU	51.349
TP (ms)	1.705	1.271	5.579	27.353			3.737		1.591			
15	GPU	GPU	GPU	GPU	CPU	CPU	GPU			CPU	CPU	76.104
16 - Método proposto parte 2 em GPU	GPU	GPU	GPU	GPU-CPU	CPU	CPU	GPU		GPU		CPU	44.938
TP (ms)				21.307								
17	GPU	GPU	GPU	GPU-CPU	CPU	CPU	GPU			CPU	CPU	71.425

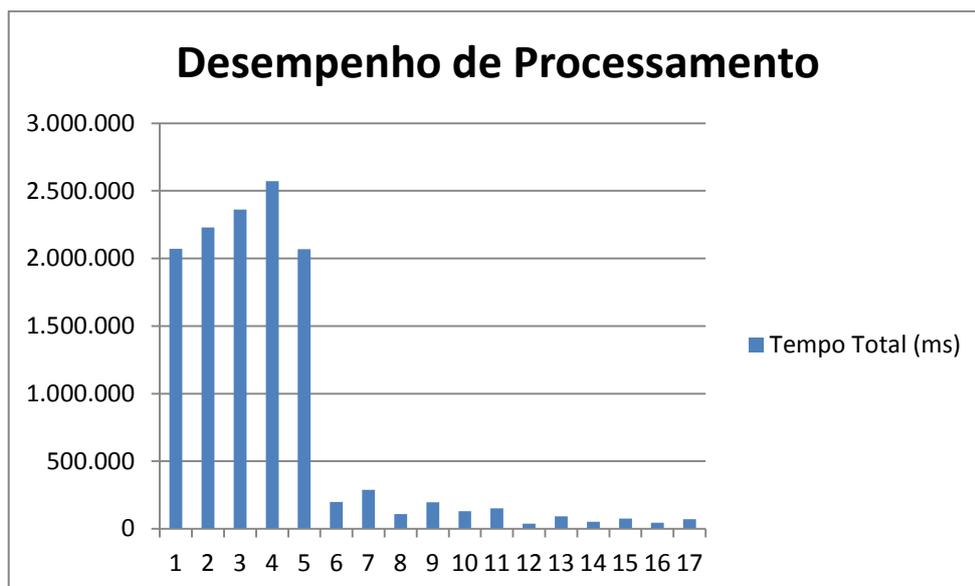


Figura 5.1 : Desempenho de processamento em uma imagem

Como pode ser observado na Tabela 5.1 e na figura acima o experimento 1 e 5 possibilitou reconhecer a logomarca do veículo gastando menos tempo de execução que o experimento 2 devido o uso do método SURF por busca exaustiva e o método do vizinho mais próximo por distância euclidiana, similarmente ocorreu com o experimento 3 em relação ao experimento 4. No entanto o experimento 1 e 5 mostrou resultados mais promissores para melhor atender as necessidades de um sistema de reconhecimento de objetos em termos de velocidade de processamento. O experimento 5 possibilitou realizar a consistência da logomarca reconhecida na imagem analisada excluindo todos os pontos de interesse que não fazem parte da logomarca.

Já a utilização de uma ROI com os experimentos de 6 a 11 agregou dois valores muito importantes para se resolver o problema de reconhecimento de logomarcas de veículos:

- Possibilitou obter um alto desempenho em termos de processamento, uma vez que toda etapa de reconhecimento ficou somente em uma pequena porção da imagem, não sendo necessário processar a imagem inteira.
- Concentrar a análise, conseqüentemente melhorando a precisão do sistema ao reconhecer a logomarca do veículo usando SURF, uma vez que somente os pontos de interesse dentro da ROI foram considerados no processo de análise.

Os experimentos 8 e 9 mostraram resultados mais promissores. No entanto semelhantemente ao experimento 1, o método do experimento 8 mostrou-se ser o mais viável para melhor atender as necessidades de um sistema de reconhecimento de objetos em termos de velocidade de processamento. No entanto o método do experimento 8 não seria tão robusto em imagens de veículos de cores claras devido a cor da placa ser mais próxima da cor do veículo, não havendo tanta diferenciação nos níveis de cinzas para definição de bordas. Sendo assim optou-se por realizar uma adaptação do método do experimento 8 com o método de Conci, Carvalho e Rauber (2009) no experimento 10 e 11 para realizar uma segmentação por operações morfológica binária. Melhorando a robustez e agregando velocidade em termos de busca e validação de regiões candidatas a placa veicular definindo o método proposto para reconhecimento de logomarcas de veículos.

O tempo de processamento foi ainda mais otimizado em GPU para as principais técnicas empregadas no método proposto com os experimentos de 12 a 17. A Figura 5.2 demonstra os resultados comparativos dos experimentos 8 e 12 e a Figura 5.3 dos experimentos 10, 14 e 16.

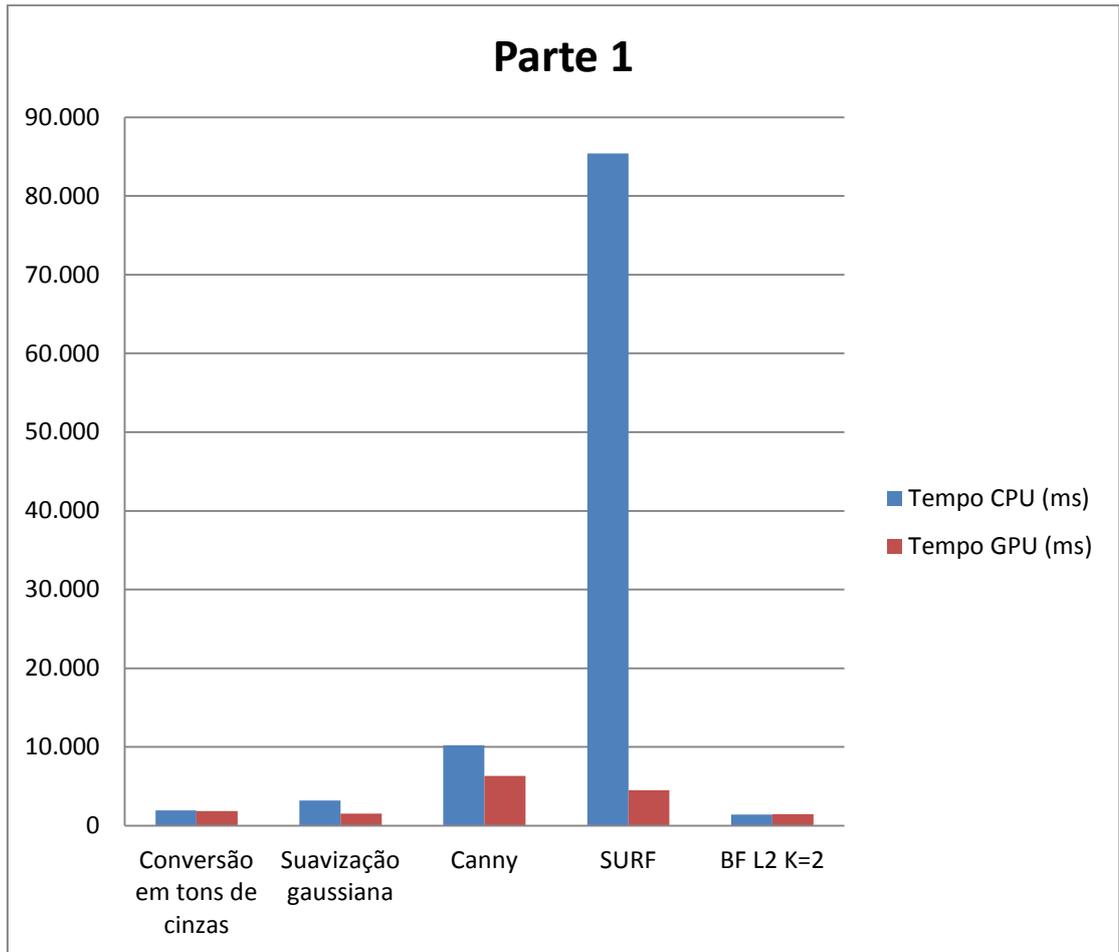


Figura 5.2 : Método proposto parte 1

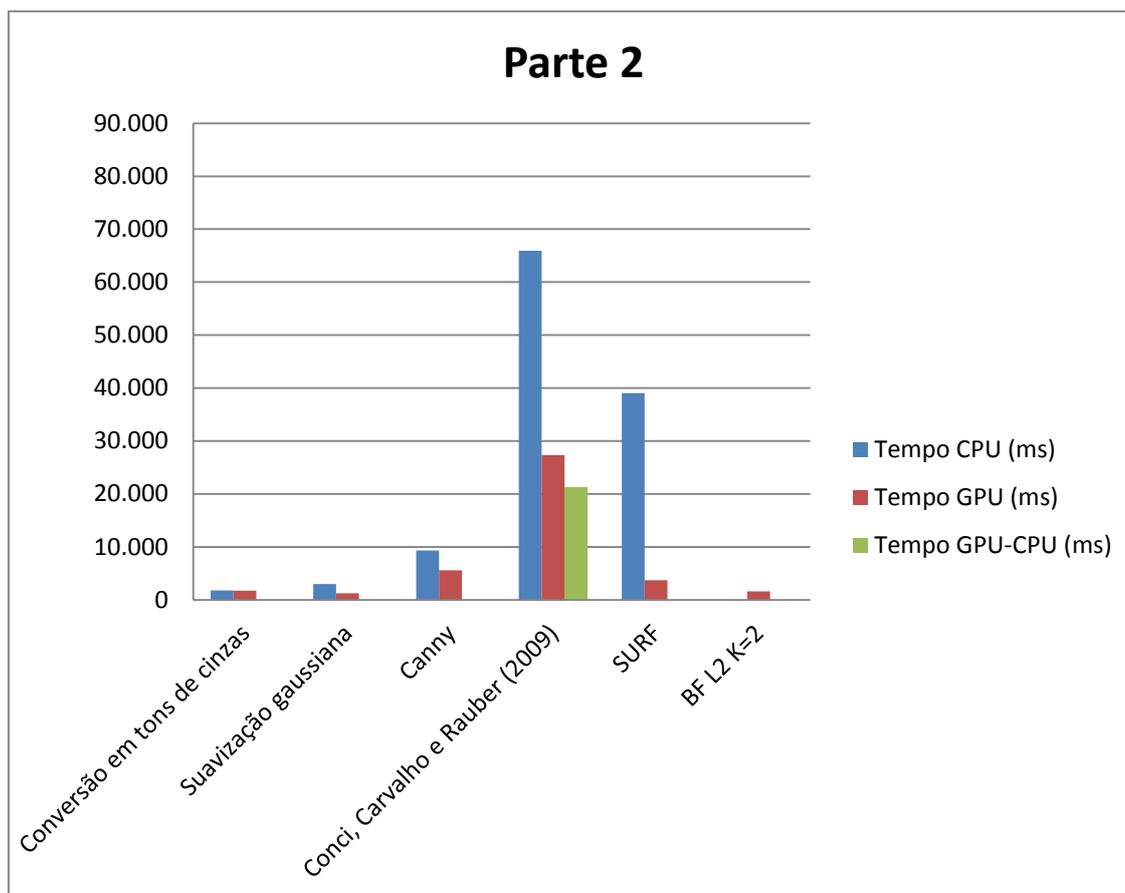


Figura 5.3 : Método proposto parte 2

A Tabela 5.2 e a Figura 5.4 mostra o resultado da parte adaptada do método de Conci, Carvalho e Rauber (2009) na CPU com experimento 10, na GPU com experimento 14 e somente a operação de dilatação na CPU com o experimento 16.

Tabela 5.2 : Método de Conci, Carvalho e Rauber (2009)

Experimento	Fechamento linear horizontal (ms)	Abertura linear vertical elim alt min (ms)	Abertura linear vertical elim alt max (ms)	Subtração (ms)	Abertura linear horizontal elim larg min (ms)	Dilatação (ms)
10 - CPU	9.152	3.932	9.948	0.469	33.503	8.940
14 - GPU	1.543	1.474	3.108	0.223	5.544	15.461
16 - GPU-CPU	1.561	1.514	3.120	0.278	5.652	9.182

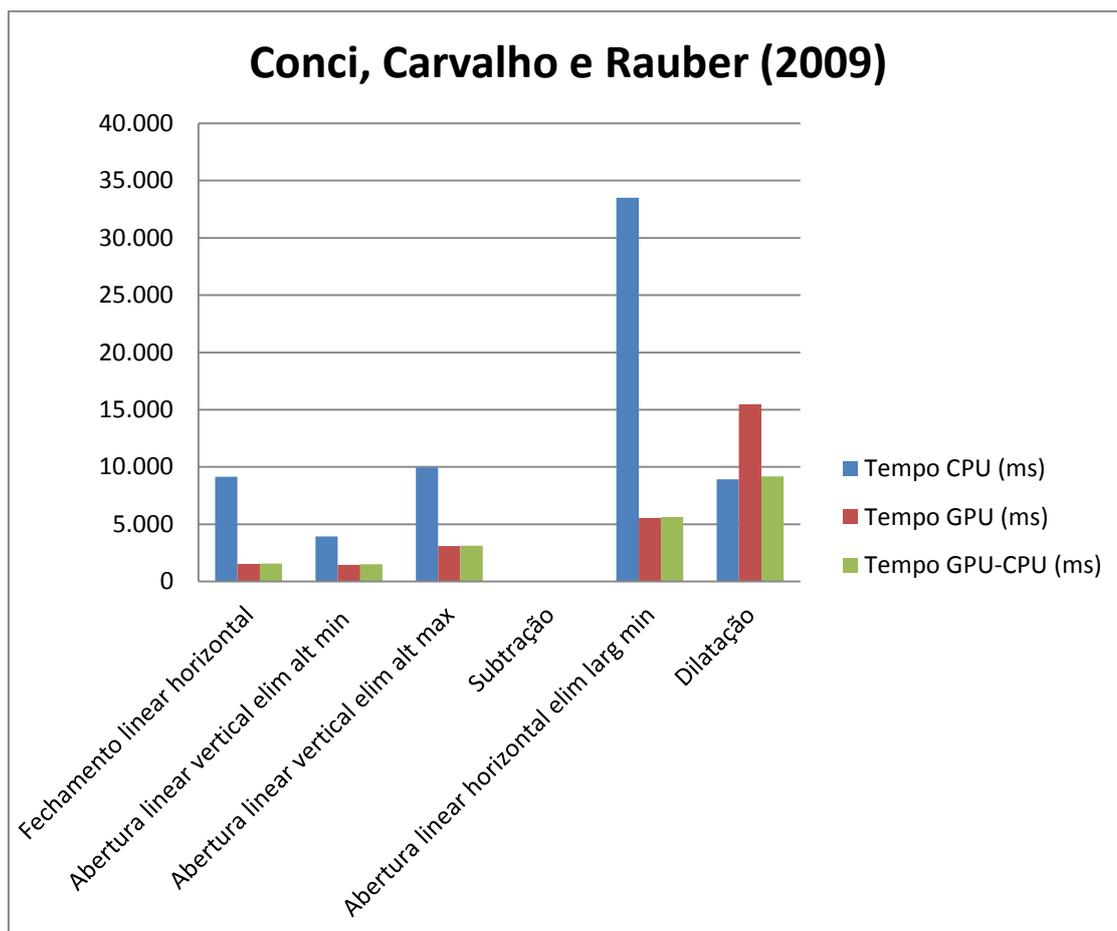


Figura 5.4 : Método de Conci, Carvalho e Rauber (2009)

5.3 Aplicação do método

As tabelas e os gráficos abaixo mostram os resultados obtidos da aplicação do método proposto por busca exaustiva e por FLANN em CPU e GPU com as bases de imagens da Tabela 4.1.

Os resultados dos experimentos 3 e 4 (método proposto) para todas as bases mostram que a aplicação por busca exaustiva é mais viável em termos de processamento do que usando FLANN (experimentos 1 e 2).

Tabela 5.3 : Experimentos de aplicação base I

Experimento	Conversão em tons de cinzas	Suavização gaussiana	Canny	Conci, Carvalho e Rauber (2009)	Detecção de contorno	ROI	SURF	BF L2 K=2	FLANN	RANSAC	Tempo Médio (ms)	Desvio Padrão (ms)
1 - Método usando FLANN CPU	CPU	CPU	CPU	CPU	CPU	CPU	CPU		CPU	CPU	298,84	237,85
2 - Método usando FLANN GPU	GPU	GPU	GPU	GPU-CPU	CPU	CPU	GPU		CPU	CPU	211,65	116,21
3 - Método proposto usando BF CPU	CPU	CPU	CPU	CPU	CPU	CPU	CPU	CPU		CPU	84,26	43,00
4 - Método proposto usando BF GPU	GPU	GPU	GPU	GPU-CPU	CPU	CPU	GPU	GPU		CPU	73,96	17,81

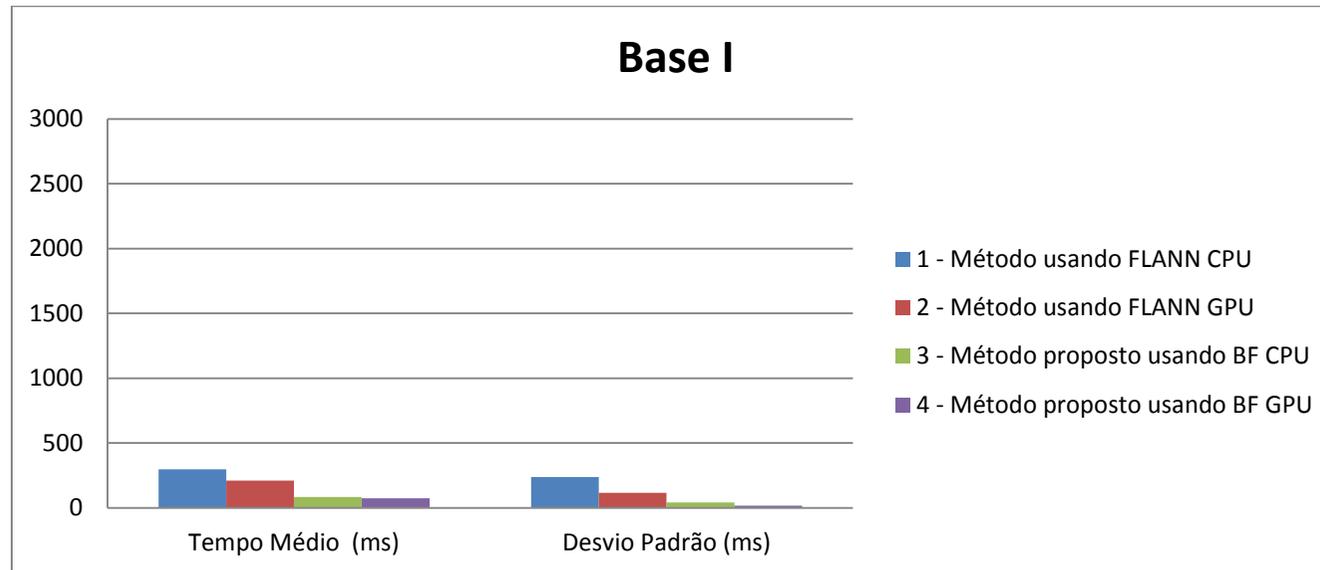


Figura 5.5 : Desempenho de processamento base I

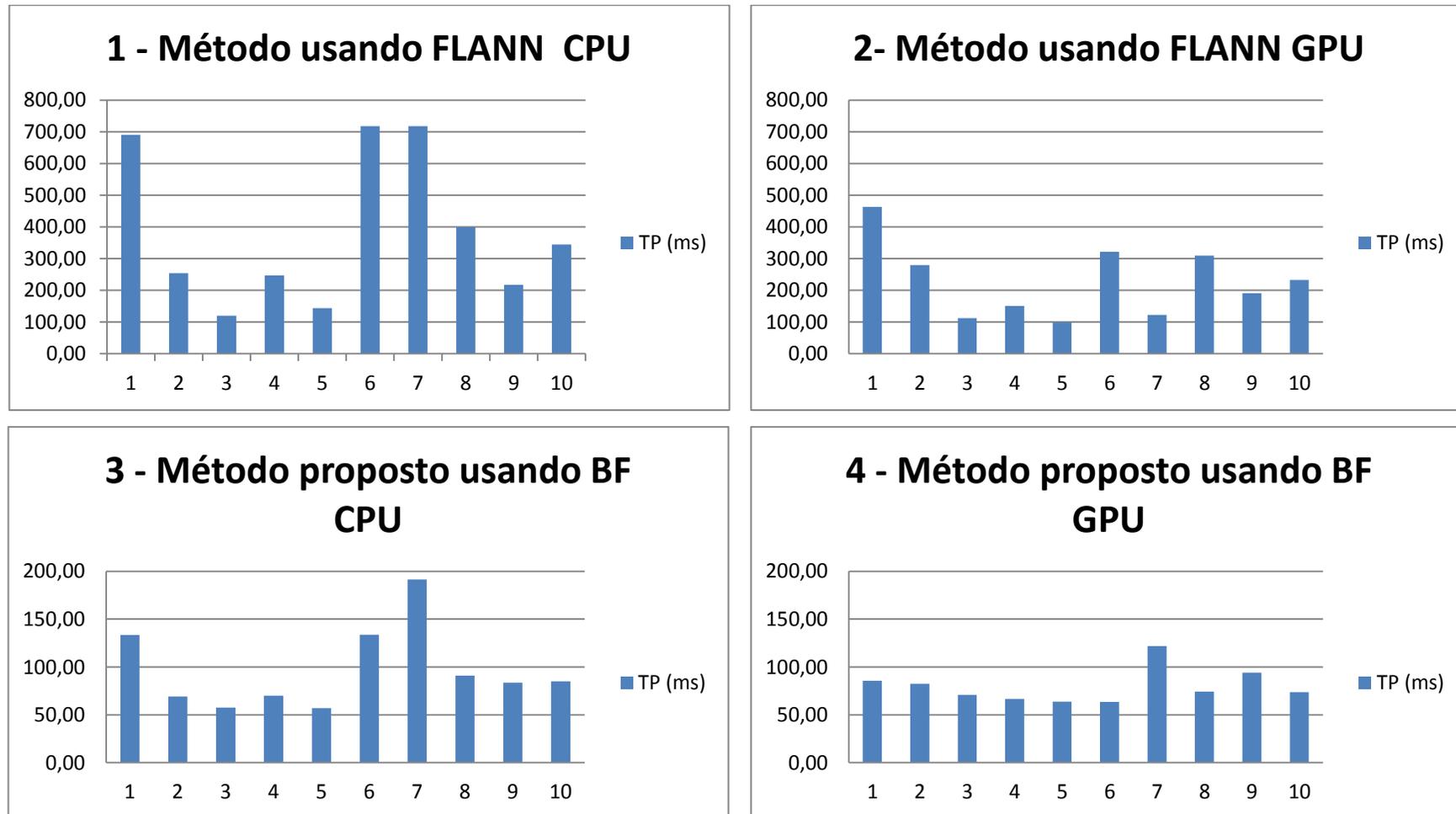


Figura 5.6 : Desempenho de processamento por imagem base I

Tabela 5.4 : Experimentos de aplicação base II

Experimento	Conversão em tons de cinzas	Suavização gaussiana	Canny	Conci, Carvalho e Rauber (2009)	Deteccção de contorno	ROI	SURF	BF L2 K=2	FLANN	RANSAC	Tempo Médio (ms)	Desvio Padrão (ms)
1 - Método usando FLANN CPU	CPU	CPU	CPU	CPU	CPU	CPU	CPU		CPU	CPU	973,11	1063,40
2 - Método usando FLANN GPU	GPU	GPU	GPU	GPU-CPU	CPU	CPU	GPU		CPU	CPU	602,03	442,49
3 - Método proposto usando BF CPU	CPU	CPU	CPU	CPU	CPU	CPU	CPU	CPU		CPU	109,51	63,49
4 - Método proposto usando BF GPU	GPU	GPU	GPU	GPU-CPU	CPU	CPU	GPU	GPU		CPU	106,67	38,73

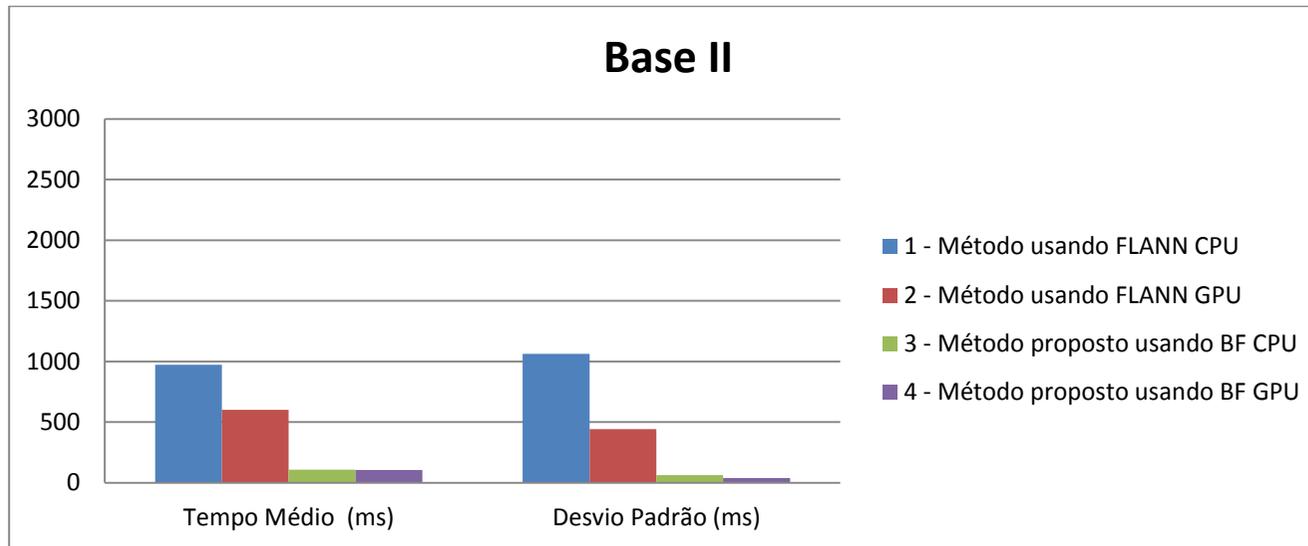


Figura 5.7 : Desempenho de processamento base II

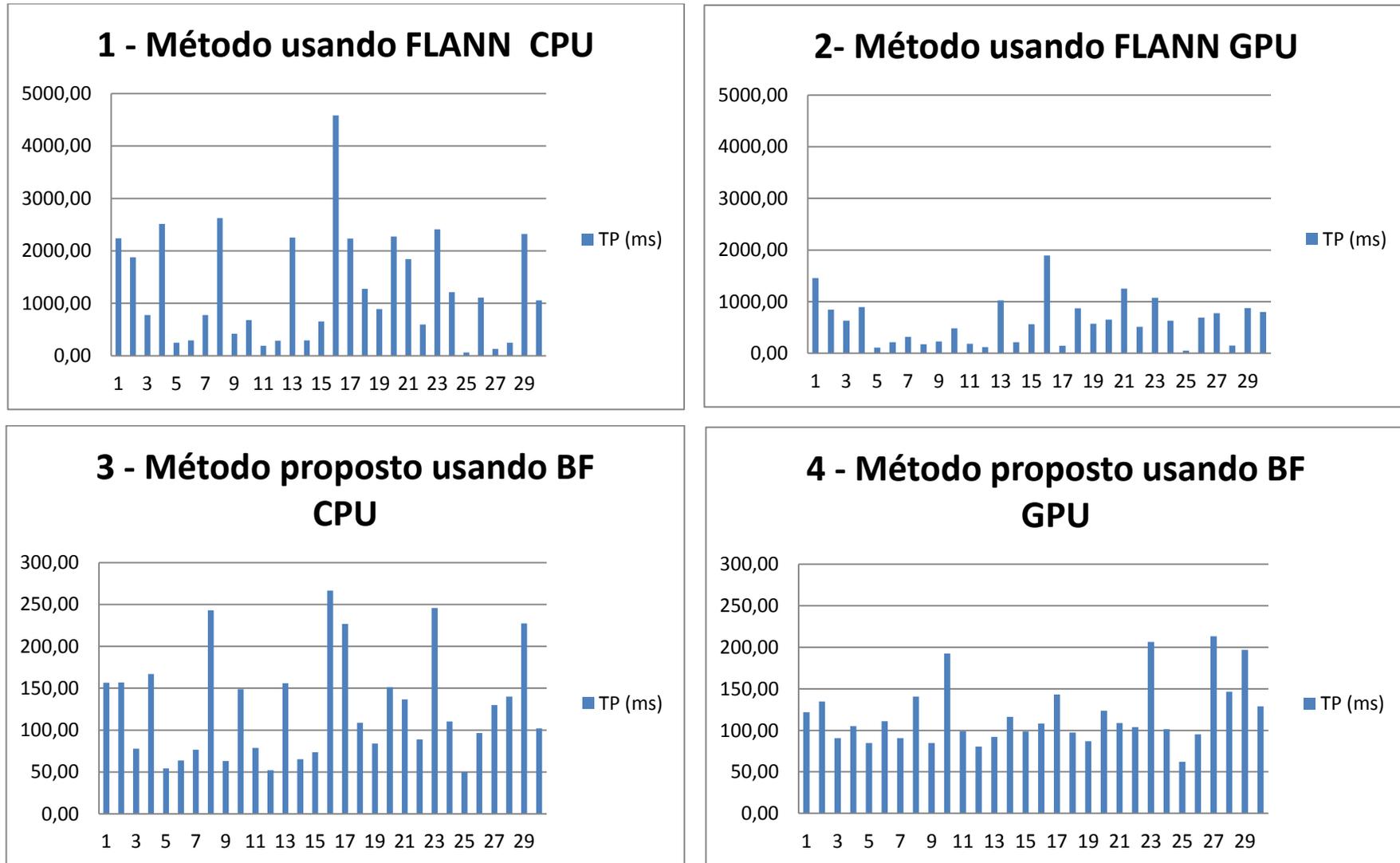


Figura 5.8 : Desempenho de processamento por imagem base II

Tabela 5.5 : Experimentos de aplicação base III

Experimento	Conversão em tons de cinzas	Suavização gaussiana	Canny	Conci, Carvalho e Rauber (2009)	Deteccção de contorno	ROI	SURF	BF L2 K=2	FLANN	RANSAC	Tempo Médio (ms)	Desvio Padrão (ms)
1 - Método usando FLANN CPU	CPU	CPU	CPU	CPU	CPU	CPU	CPU		CPU	CPU	892,48	2386,30
2 - Método usando FLANN GPU	GPU	GPU	GPU	GPU-CPU	CPU	CPU	GPU		CPU	CPU	167,41	744,17
3 - Método proposto usando BF CPU	CPU	CPU	CPU	CPU	CPU	CPU	CPU	CPU		CPU	133,73	129,75
4 - Método proposto usando BF GPU	GPU	GPU	GPU	GPU-CPU	CPU	CPU	GPU	GPU		CPU	136,41	55,92

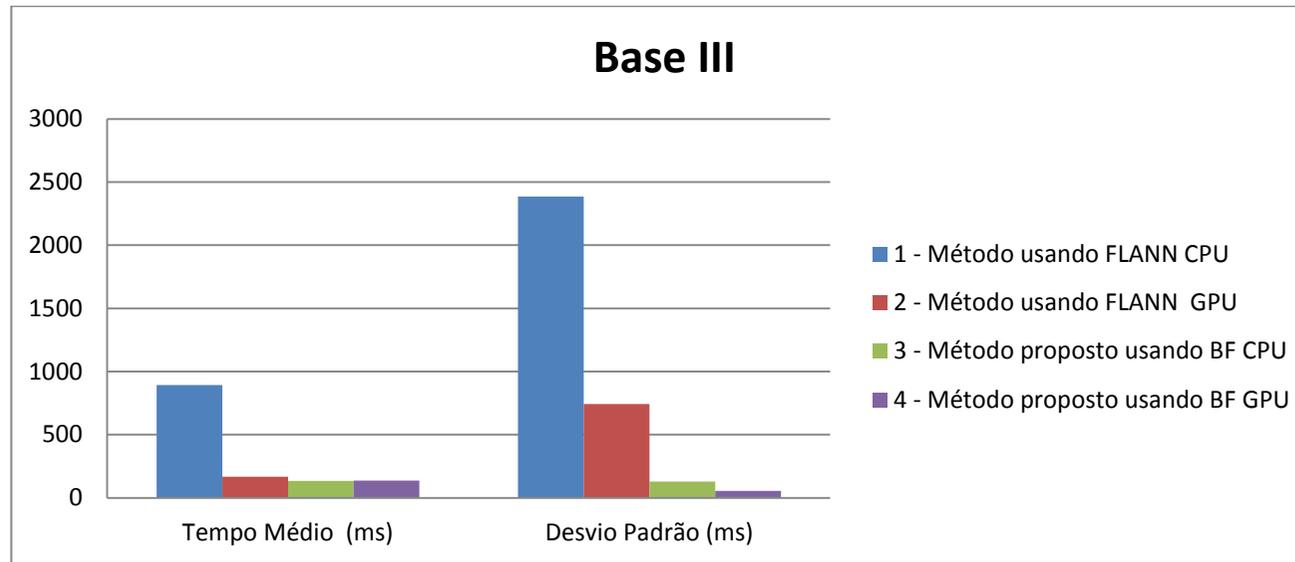


Figura 5.9 : Desempenho de processamento base III

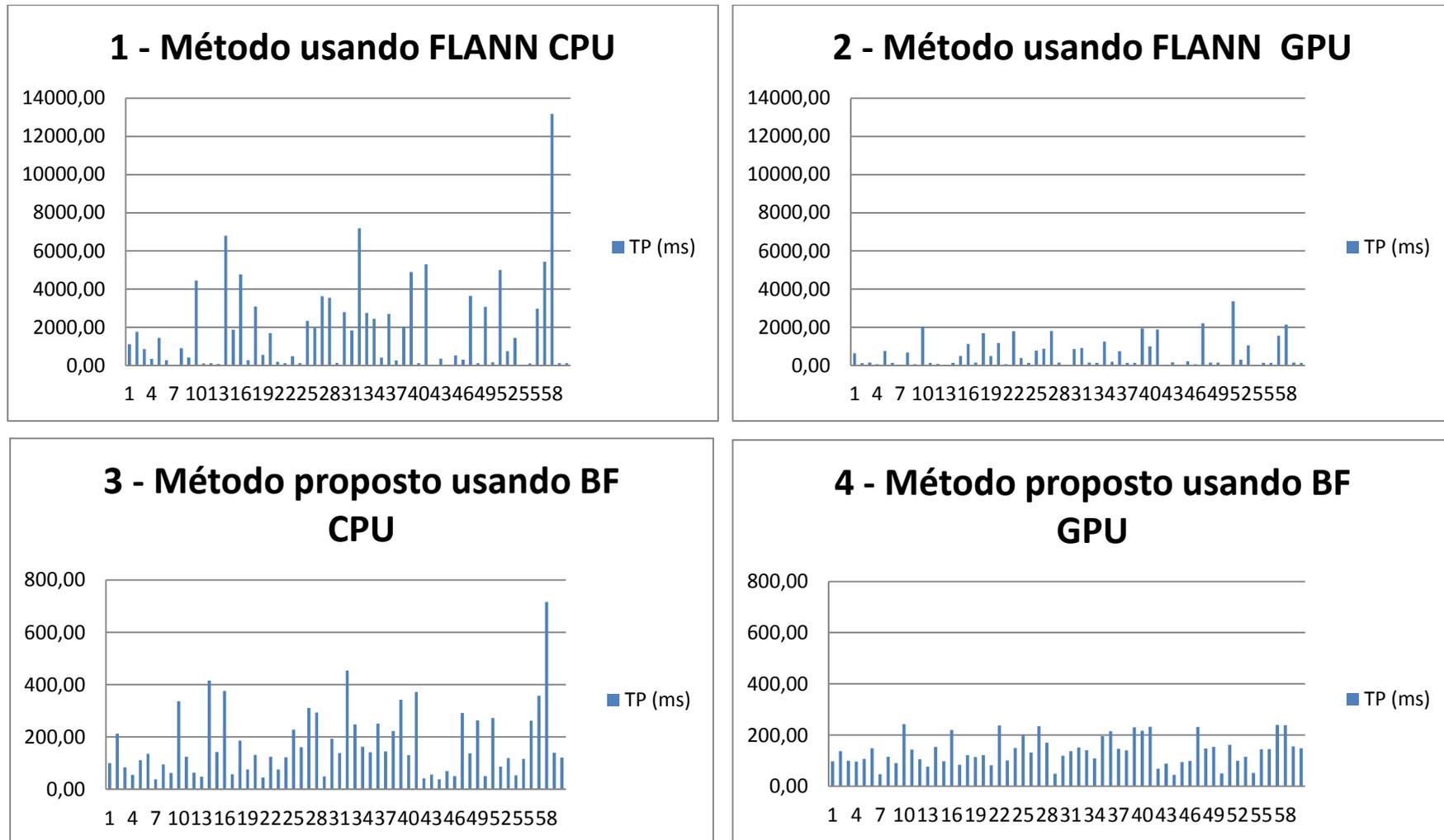


Figura 5.10 : Desempenho de processamento por imagem base III

Tabela 5.6: Experimentos de aplicação base IV

Base IV												
Experimento	Conversão em tons de cinzas	Suavização gaussiana	Canny	Conci, Carvalho e Rauber (2009)	Deteção de contorno	ROI	SURF	BF L2 K=2	FLANN	RANSAC	Tempo Médio (ms)	Desvio Padrão (ms)
1 - Método usando FLANN CPU	CPU	CPU	CPU	CPU	CPU	CPU	CPU		CPU	CPU	587,41	2051,02
2 - Método usando FLANN GPU	GPU	GPU	GPU	GPU-CPU	CPU	CPU	GPU		CPU	CPU	243,35	817,77
3 - Método proposto usando BF CPU	CPU	CPU	CPU	CPU	CPU	CPU	CPU	CPU		CPU	96,49	111,58
4 - Método proposto usando BF GPU	GPU	GPU	GPU	GPU-CPU	CPU	CPU	GPU	GPU		CPU	100,06	69,01

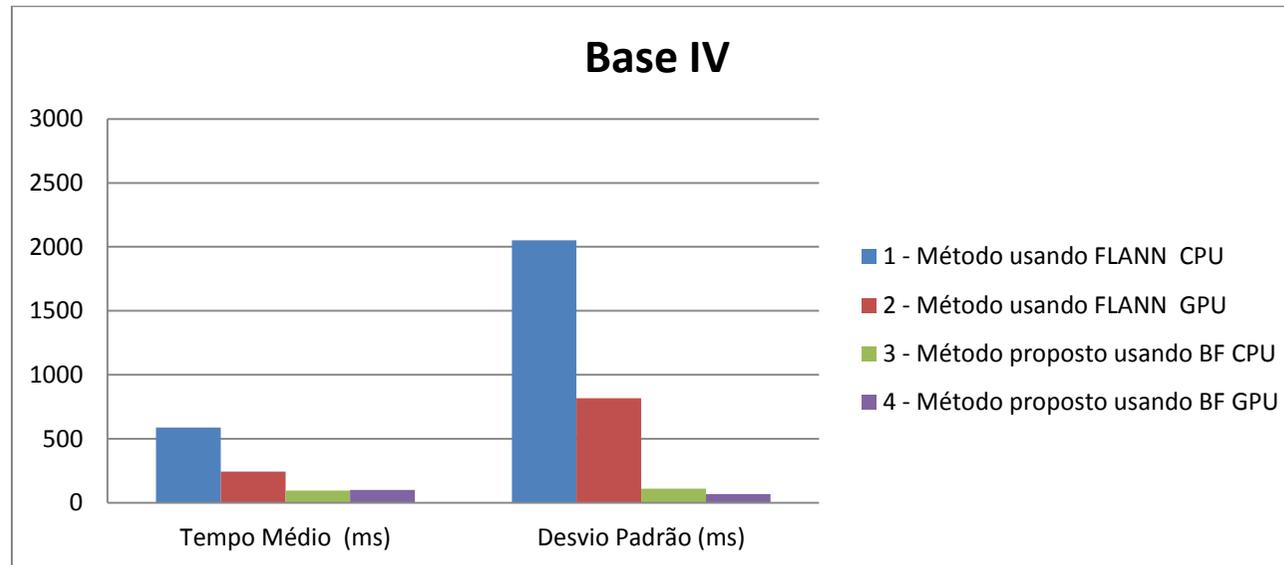


Figura 5.11 Desempenho de processamento base IV

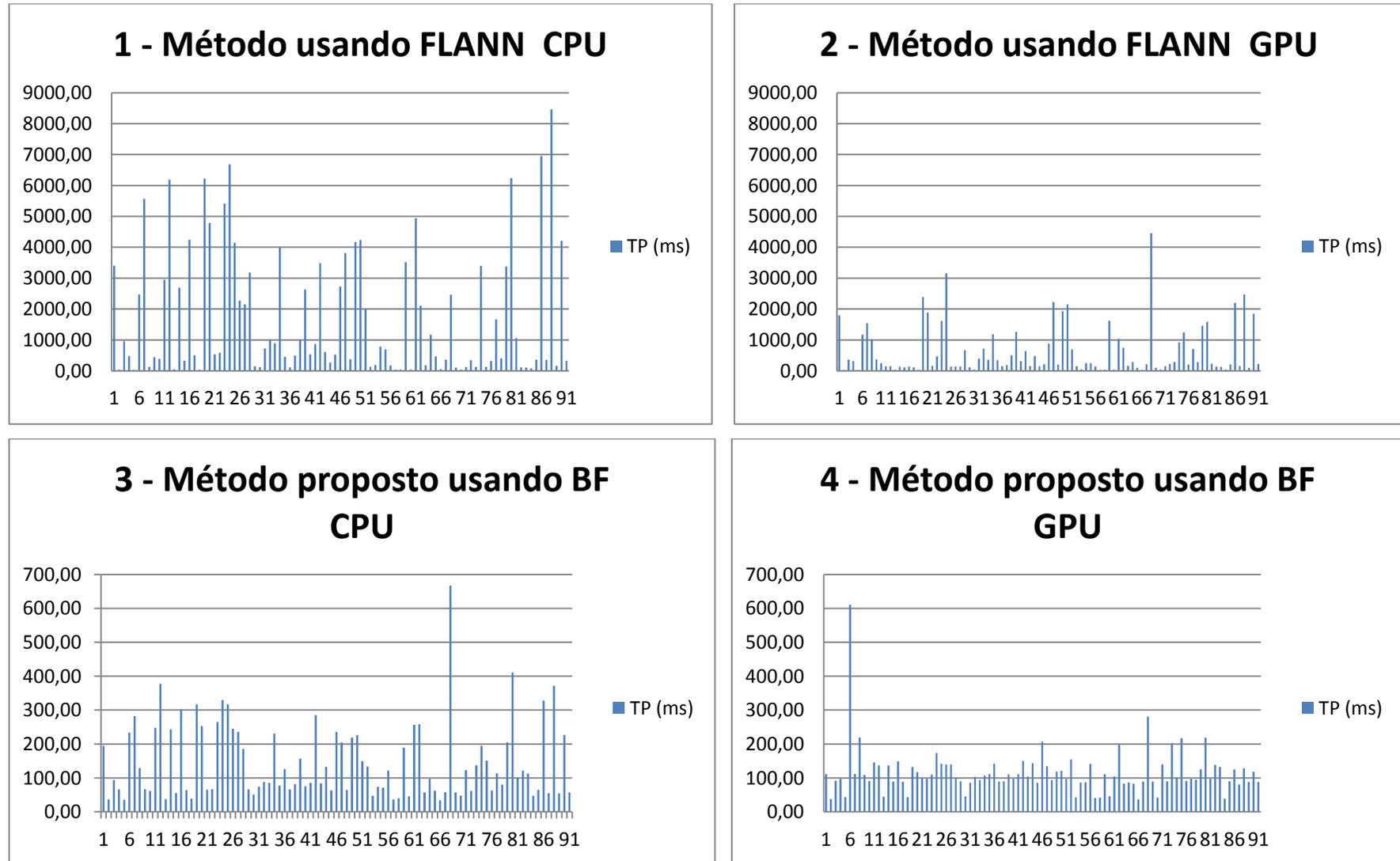


Figura 5.12: Desempenho de processamento por imagem base IV

As tabelas e as figuras acima mostram também que o desempenho em termos de processamento foi ainda mais otimizado com a execução da maioria das etapas do método proposto em uma GPU, conseqüentemente um melhor resultado quando comparado aos trabalhos anteriores apresentados no Capítulo 3.

O método proposto realizou o reconhecimento de logomarcas de veículos com 100% de taxa de acerto para base I, 64 % para a base II, 45% para a base III e IV.

O desempenho de reconhecimento bem como o tempo de processamento por imagem pode variar uma vez que depende da qualidade da imagem analisada, do sucesso da etapa de definição da ROI e do número de bons pontos de interesse encontrados para correspondência.

5.4 Considerações finais

Os experimentos de validação possibilitaram realizar comparações para se chegar no método apresentado na Figura 4.1.

Os resultados dos experimentos para avaliação do método desenvolvido mostraram que o método proposto pode ser uma opção alternativa no problema de reconhecimento de logomarcas de veículos.

Capítulo 6

CONCLUSÕES

Neste trabalho foi proposto um método alternativo para reconhecimento de logomarcas de veículos. O método mostrou ser flexível em reconhecer logomarcas de veículos em diferentes pontos de vista. Possibilitou localizar a placa de veículos de forma rápida por meio de ajustes de parâmetros com métodos de detecção de bordas e contorno e com auxílio de operações morfológica binária, sendo ainda mais rápido em alguns casos em que uma região de interesse foi definida de forma mais delimitada. Outro fator importante foi que o desempenho em termos de processamento foi ainda mais otimizado com a execução da maioria das etapas do método proposto em uma GPU.

O método proposto realizou o reconhecimento de logomarcas de veículos com uma taxa de sucesso entre 45% e 100 % das imagens analisadas, com ganhos em velocidade de processamento em relação a outras abordagens. Tanto a taxa de acerto como o tempo de processamento por imagem apresentaram variações significativas entre as imagens utilizadas. Isso se deve à qualidade e características das imagens da Base III e IV, que são mais heterogêneas do que as imagens da Base I e II. Como esperado, esses fatores influenciaram no sucesso das etapas de definição da ROI e seleção do número de bons pontos de interesse.

Os resultados sugerem que o método proposto, baseado na integração de diversas técnicas, possui potencial para ser refinado, e disponibilizado como alternativa eficiente na solução do problema de reconhecimento de logomarcas de veículos.

6.1 Trabalhos futuros

Em trabalhos futuros, seria interessante, a investigação de:

- Um modelo que melhore ainda mais a velocidade em termos de processamento do método proposto neste trabalho bem como sua robustez para melhor atender as necessidades do problema de reconhecimento de logomarcas de veículos.
- Um modelo que facilite a adoção do método SURF em outra área de aplicação.
- Um modelo que utilize o método ORB para reconhecimento de objetos em sistemas embarcados (HEDAYATI et al., 2013; PHAN et al., 2013; PHAN; NA; KIM, 2014; RUBLEE et al., 2011; LEE; BYUN, 2013; KIM et al., 2014).

PUBLICAÇÕES

A partir deste trabalho, foi publicado o seguinte artigo em evento científico:

MACEDO, C.; FERNANDES, M. M. Recognition of Vehicles Logos using SURF. WVC 2015. USP - São Carlos, Brasil: USP, 2015.

REFERÊNCIAS

AN L.; THAKOOR N.; BHANU B. Vehicle Logo Super-resolution by Canonical Correlation Analysis. **Image Processing (ICIP), 2012 19th IEEE International Conference on.**, p.2229-2232, 2012.

ANAGNOSTOPOULOS, C.N.E. et al. A License Plate-Recognition Algorithm for Intelligent Transportation System Applications. **Intelligent Transportation Systems, IEEE Transactions on.**, v.7, n.3, p.377-392, 2006.

AGRAWAL, M.; KONOLIGE, K.; BLAS, M. R. CenSurE: Center Surround Extremas for Realtime Feature Detection and Matching, **Computer Vision – ECCV 2008.**, v. 5305, p. 102–115, 2008.

BADURA, S.; FOLTAN, S. Advanced scale-space, invariant, low detailed feature recognition from images - car brand recognition. **Computer Science and Information Technology (IMCSIT), Proceedings of the 2010 International Multiconference on.**, p.19-23, 2010.

BAY, H. et al. Speeded-Up Robust Features(SURF). **Computer Vision and Image Understanding.**, v.110, n. 3, p. 346-359, 2008.

BAY, H.; TUYTELAARS, T.; GOOL, L. V. SURF: Speeded Up Robust Features. **Computer Vision – ECCV 2006.**, v.3951, p. 404-417, 2006.

CONCI A.; AZEVEDO E.; LETA F. R. Computação Gráfica.v. 2, Rio de Janeiro: Elsevier, 2008.

CALONDER M.; LEPETIT V.; FUA P. BRIEF: Binary Robust Independent Elementary Features. **Computer Vision – ECCV 2010.**, v. 6314, p. 778–792, 2010.

CONCI, A.; CARVALHO, J.E.R.; RAUBER, T. W.A complete system for vehicle plate localization, segmentation and recognition in real life scene.**Latin America Transactions, IEEE (Revista IEEE America Latina)**., v.7, n.5, p.497-506, 2009.

COVER, T.; HART, P., Nearest neighbor pattern classification. **Information Theory, IEEE Transactions on**., v.13, n.1, p.21-27, 1967.

CHATURVEDI, A.; SETHI, N. Automatic License Plate Recognition System using SURF Features and RBF Neural Network. **International Journal of Computer Applications**.,v. 70, n. 27, p. 37-41,2013.

CHOI J.; CHO Y.A Study on Algorithm Fusion for Recognition and Tracking of Moving Robot.**World Academy of Science, Engineering and Technology, International Science Index**., v. 6, n. 12, 1562-1567, 2012.

CANNY, J. A. Computational Approach to Edge Detection. **Pattern Analysis and Machine Intelligence, IEEE Transactions on**., v. 8, n. 6, p.679-698, 1986.

CANNY, J. A. Finding edges and lines in images.**M.I.T. Artificial Intell.Lab., Cambridge**., MA, Rep. AI-TR-720, 1983.

COLLET, A. et al. Object Recognition and Full Pose Registration from a Single Image for Robotic Manipulation. **Robotics and Automation, 2009.ICRA '09. IEEE International Conference on**., p.48-55, 2009.

CONCI, A; DE CARVALHO, J. E. R.; RAUBER, T.W. A complete system for vehicle plate localization, segmentation and recognition in real life scene. **Latin America Transactions, IEEE (Revista IEEE America Latina)** , v. 7, n. 5, p.497-506, 2009.

DOUGLAS, D.; PEUCKER, T. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. **The Canadian Cartographer**., v. 10, n. 2, p. 112-122, 1973.

EMANI H.; FATHI M.; RAAHEMIFAR K. Real Time Vehicle Make and Model Recognition Based on Hierarchical Classification. **International Journal of Machine Learning and Computing.**, v. 4, n. 2, p. 142-145, 2014.

FISCHLER M. A.; BOLLES R. C. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. **Commun. ACM.**, v. 24, n. 6, p. 381-395, 1981.

FAWCETT T. An introduction to ROC analysis, Pattern Recognition Letters. **Pattern Recognition Letters.**,v. 27, n. 8, p. 861-874, 2005.

GONZALEZ, R. C.; WOODS, R. E. Processamento de Imagens Digitais. São Paulo: EdgardBlücher, 2000.

GONZALEZ, R. C.; WOODS, R. E. Processamento de Imagens Digitais. São Paulo: Pearson Prentice Hall, 2010.

HARRIS C.; STEPHENS M.A combined corner and edge detector. **In Proc. of Fourth Alvey Vision Conference**, p. 147-151, 1988.

HEDAYATI, M. et al. Performances of Invariant Feature Detectors in Real-Time Video Applications. **Springer International Publishing.**, v. 8237, p. 193-205, 2013.

HASSANZADEH, S.; POURGHASSEM, H.A Fast Logo Recognition Algorithm in Noisy Document Images. **Intelligent Computation and Bio-Medical Instrumentation (ICBMI), 2011 International Conference on.**, p. 64-67, 2011.

HERAKLEOUS, K.; POULLIS, C. Improving Augmented Reality Applications with Optical Flow. **Image Processing (ICIP), 2013 20th IEEE International Conference on.**, p. 3403-3406, 2013.

HARTLEY, R.; ZISSERMAN, A. Multiple View Geometry in Computer Vision.2.ed. Cambridge University Press, 2003.

HUIJUAN Z; QIONG H. Fast image matching based-on improved SURF algorithm. **Electronics, Communications and Control (ICECC), 2011 International Conference on.**, p.1460-1463, 2011.

LEE, K.; BYUN K. A hardware design of optimized ORB algorithm with reduced hardware cost. **Advanced Science and Technology Letters.**, v.43, p. 58-62, 2013.

LIPIKORN, R. et al. Vehicle logo recognition based on interior structure using SIFT descriptor and neural network. **Information Science, Electronics and Electrical Engineering (ISEEE), 2014 International Conference on.**, v. 3, p.1595-1599, 2014.

LIU W. et al. A kernel-based l2 norm regularized least square algorithm for vehicle logo recognition. **Digital Signal Processing (DSP), 2014 19th International Conference on.**, p.631-635, 2014.

LOWE, D. G. Object recognition from local scale-invariant features. **Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on.**, v. 2, p. 1150-1157, 1999.

LOWE, D. G. "Distinctive Image Features from Scale-invariant Keypoints". **International Journal of Computer Vision.**, v. 60, n. 2, p. 91-110, 2004.

MUJA, M.; LOWE, D.G.Fast approximate nearest neighbors with automatic algorithm configuration. **(2009) VISAPP 2009 - Proceedings of the 4th International Conference on Computer Vision Theory and Applications.**, v. 1, p. 331-340, 2009.

MATAS, J. et al. Robust wide-baseline stereo from maximally stable extremal regions. **British Machine Vision Computing 2002.**, v. 22, n. 10, p. 761-767, 2004.

MOESLUND, T. B. Introduction to Video and Image Processing, Building Real Systems and Applications. New York: Springer, 2012.

MARANA, A. N.; BREGA J. R. F. Técnicas e Ferramentas de Processamento de Imagens Digitais e Aplicações em Realidade Virtual e Misturada. Bauru: Unesp/FC, Editora Canal 6, 2008.

MARQUES, J. S. Reconhecimento de Padrões: Métodos Estatísticos e Neurais. 2. ed. Lisboa: IST Press, 2005.

MAO, S. et al. Rapid vehicle logo region detection based on information theory. **Computers & Electrical Engineering**, v. 39, n. 3, p. 863-872, 2013.

NICKOLLS, J. et al. Scalable Parallel Programming with CUDA. **Queue**, v.6, n. 2, p. 40-53, 2008.

NAN, Z. Vehicle-logo recognition based on classifier combinations. **System Science, Engineering Design and Manufacturing Informatization (ICSEM), 2012 3rd International Conference on**, v.2, p. 278-281, 2012.

OKABE, T.; SATO, Y. Object recognition based on photometric alignment using RANSAC. **Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on**, v.1, p. 221-228, 2003.

TAN P.; STEINBACH, M.; KUMAR, V. Introduction to Data Mining, Adison-Wesley, 2006.

PSYLLOS A.; ANAGNOSTOPOULOS C. N.; KAYAFAS E. Vehicle model recognition from frontal view image measurements. **Computer Standards & Interfaces**, v. 33, n. 2, p. 142-151, 2011.

PEDRINI, H.; SCHWARTZ, W. R. Análise de Imagens Digitais: Princípios, Algoritmos e Aplicações. São Paulo: Thomson Learning, 2008.

PAN, H.; ZHANG, B. An Integrative Approach to Accurate Vehicle Logo Detection. **Journal of Electrical and Computer Engineering**, p. 1-12, 2013.

PHAN, D. et al. Object Recognition by Combining Binary Local Invariant Features and Color Histogram. **Pattern Recognition (ACPR), 2013 2nd IAPR Asian Conference on.**, p.466-470, 2013.

PHAN, D.; NA, I.; KIM, S. Local features and histogram based planar object recognition. **In Proceedings of the 8th International Conference on Ubiquitous Information Management and Communication.**, p. 1-4, 2014.

RUBLEE, E et al. ORB: An efficient alternative to SIFT or SURF. **Computer Vision (ICCV), 2011 IEEE International Conference on.**, p.2564-2571, 2011.

ROSTEN, E.; DRUMMOND, T. Machine learning for high-speed corner detection. **In European Conference on Computer Vision.**, v. 1, p. 430–443. 2006.

ROSTEN, E.; DRUMMOND, T. Fusing points and lines for high performance tracking. **Computer Vision, 2005.ICCV 2005. Tenth IEEE International Conference on.**, v. 2, p. 1508-1515, 2005.

SAM, K.; TIAN X. Vehicle Logo Recognition Using Modest AdaBoost and Radial Tchebichef Moments. **Proceedings of 2012 4th International Conference on Machine Learning and Computing.**, v. 25, p. 91-95, 2012.

SIDLA O.; KOTTMANN M.; BENESOVA W. Real-time pose invariant logo and pattern detection. **Proc. SPIE, Intelligent Robots and Computer Vision XXVIII: Algorithms and Techniques.**, v. 7878, 2011.

SHI, J.; TOMASI, C. Good features to track, **Computer Vision and Pattern Recognition, 1994. Proceedings CVPR '94., 1994 IEEE Computer Society Conference on.**, p. 593-600, 1994.

SUZUKI S.; ABE K. Topological structural analysis of digitized binary images by border following. **Computer Vision, Graphics, and Image Processing.**, v. 30, n.1, p. 32-46, 1985.

STONE, J. E.; GOHARA, D.; SHI G. OpenCL: A Parallel Programming Standard for Heterogeneous Computing Systems. **Computing in Science Engineering.**, v. 12, n. 3, p. 66-73, 2010.

SCHWEIGER, F. et al. Speeded-up SURF: Design of an efficient multiscale feature detector. **Image Processing (ICIP), 2013 20th IEEE International Conference on.**, p. 3475-3478, 2013.

SAUVOLA J.; PIETIKÄINEN M. Adaptive document image binarization. **Pattern Recognition.**, v. 33, n. 2, p. 225-236, 2000.

KIRK D. B.; HWU W. W. Programming Massively Parallel Processors: A Hands-on Approach, A Hands-on Approach. Elsevier, 2010.

XING.X.; JUNG-HAE C.; BYUNG-JAE C. A Replacement Algorithm of Fast Computing Interest Point's Orientation and Descriptor in SURF. **International Conference on Advances in Electrical and Electronics Engineering.**, p. 336-338, 2012.

WANG Y.; LI N.; WU Y. Application of Edge Testing Operator in Vehicle Logo Recognition. **Intelligent Systems and Applications (ISA), 2010 2nd International Workshop on.**, p.1-3, 2010.

YANG, H. et al. An Efficient Method for Vehicle Model Identification via Logo Recognition. **Computational and Information Sciences (ICCIS), 2013 Fifth International Conference on.**, p.1080-1083, 2013.

ZHOU, K. et al. Hybridization of appearance and symmetry for vehicle-logo localization. **Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on.**, p.1396-1401, 2012.

KOVESI P. "Image Features From Phase Congruency", **Videre: A Journal of Computer Vision Research, MIT Press.**, v. 1, n. 3, p. 1-27, 1999.

KIM et al. Performance Analysis of ORB Image Matching Based on Android. **International Journal of Software Engineering and Its Applications.**, v. 8, n. 3, p. 11-20, 2014.

ZHANG F.; SHEN Y.; CHANG H.A Visual Saliency Based Method for Vehicle Logo Detection.Proc. SPIE, **Fifth International Conference on Digital Image Processing (ICDIP 2013).**, v. 8878, 2013.