

**UNIVERSIDADE FEDERAL DE SANTA CATARINA  
DEPARTAMENTO DE COMPUTAÇÃO**

Daniel Bitencourt Pereira

**ANÁLISE E IMPLEMENTAÇÃO DE SISTEMA PARA  
DETECÇÃO DE PLACAS BRASILEIRAS DE  
SINALIZAÇÃO DE TRÂNSITO**

Araranguá

2018



Daniel Bitencourt Pereira

**ANÁLISE E IMPLEMENTAÇÃO DE SISTEMA PARA  
DETECÇÃO DE PLACAS BRASILEIRAS DE  
SINALIZAÇÃO DE TRÂNSITO**

Trabalho de Conclusão de Curso submetido à Universidade Federal de Santa Catarina para a obtenção do Grau de Bacharel em Engenharia de Computação.  
Orientador: Prof. Fabrício de Oliveira Ourique, Dr.

Araranguá

2018

Ficha de identificação da obra elaborada pelo autor,  
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Bitencourt Pereira, Daniel  
Análise e Implementação de Sistema para Detecção  
de Placas Brasileiras de Sinalização de Trânsito /  
Daniel Bitencourt Pereira ; orientador, Fabrício de  
Oliveira Ourique , 2018.  
95 p.

Trabalho de Conclusão de Curso (graduação) -  
Universidade Federal de Santa Catarina, Campus  
Araranguá, Graduação em Engenharia de Computação,  
Araranguá, 2018.

Inclui referências.

1. Engenharia de Computação. 2. Processamento  
Digital de Imagem. 3. Aprendizado de Máquina. 4.  
Descriptor de imagem. I. , Fabrício de Oliveira  
Ourique. II. Universidade Federal de Santa  
Catarina. Graduação em Engenharia de Computação. III.  
Título.

Daniel Bitencourt Pereira

**ANÁLISE E IMPLEMENTAÇÃO DE SISTEMA PARA  
DETECÇÃO DE PLACAS BRASILEIRAS DE  
SINALIZAÇÃO DE TRÂNSITO**

Este Trabalho de Conclusão de Curso foi julgado aprovado para a obtenção do Título de “Bacharel em Engenharia de Computação”, e aprovado em sua forma final pela Universidade Federal de Santa Catarina.

Araranguá, 3 de Julho 2018.



---

Prof. Dr. Eliane Pozzebon  
Coordenadora do Curso

**Banca Examinadora:**



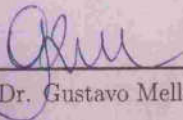
---

Prof. Dr. Fabrício de Oliveira Ourique  
Orientador



---

Prof. Dr. Anderson Luiz Fernandes Perez



---

Prof. Dr. Gustavo Mello Machado



Dedico este trabalho à minha família, em especial ao meu irmão Rafael, que não pode vê-lo concluído.





## AGRADECIMENTOS

Aos meus pais, Dilney Cesar Bitencourt Pereira e Maria das Dores Bitencourt Pereira, pelo total apoio e suporte durante à vida, e incondicional amparo e incentivo aos meus estudos. Aos meus irmãos, Daiani Bitencourt Pereira, Dilney Cesar Bitencourt Pereira Junior e Rafael Bitencourt Pereira, que foram as bases de minha vida e de meu aprendizado. À Universidade Federal de Santa Catarina, que me proporcionou um ensino de qualidade e muitas oportunidades ímpares. Aos meus queridos amigos que à UFSC e os incontáveis churrascos proporcionaram aos longos dos anos. Ao professor Anderson Luiz Fernandes Perez, que além de um ótimo professor, é um excelente amigo e um exemplo de pessoa. Ao Luan Carlos da Silva Casagrande pela imensa ajuda no decorrer da realização deste trabalho. Ao professor Fabrício de Oliveira Ourique, pela orientação e auxílio durante a execução e conclusão deste trabalho. Aos professores Anderson Luiz Fernandes Perez, Gustavo Mello Machado e Tiago Weber por fazerem parte da banca avaliadora. A todos que, de alguma forma, auxiliaram em minha jornada até aqui.



“I do not fear computers. I fear the lack of them.”

(Isaac Asimov)



## RESUMO

Sistemas avançados de assistência ao motorista (ADAS) são sistemas que auxiliam os motoristas na condução de veículos. Entretanto, em razão das falhas humanas, tais sistemas estão em evidência, sobretudo, devido à complexidade envolvida em suas soluções. Sistemas ADAS que envolvem visão computacional e aprendizado de máquina vêm ganhando plena atenção recentemente, principalmente em situações onde o condutor não repara nas informações presentes nas vias. Devido a isso, este Trabalho de Conclusão de Curso compara três métodos descritores de imagem e um identificador de regiões de interesse, todos aplicados para identificação e classificação de placas brasileiras de sinalização de trânsito. Foram conduzidos dois testes. O primeiro utilizou os métodos descritores de imagem HOG, LESH e SFTA, e testou à precisão desses métodos em conjunto com o classificador SVM. O HOG e o LESH apresentaram um excelente desempenho, com acurácia de 96,08% e 91,2% respectivamente. Entretanto, o desempenho do SFTA inviabiliza o uso do mesmo nesta aplicação. O Segundo teste usou o descritor de imagem HOG junto ao MSER. A performance constatada foi baixa, com cerca de 32,85%.

**Palavras-chave:** Processamento de imagem. Aprendizado de máquina. SVM. HOG. MSER. SFTA.



## ABSTRACT

Advanced Driver Assistance System (ADAS) are systems that assist drivers in driving vehicles. However, because of human failures, such systems are in evidence, above all, due to the complexity involved in their solutions. ADAS systems comprising computer vision and machine learning have been obtaining full attention recently, especially in situations where the driver does not notice the information on the roads. Due to this, this work compares three image descriptors methods and one method to identify regions of interest, all applied for the identification and classification of Brazilian traffic signs. Two tests were conducted. The first one used the HOG, LESH and SFTA image descriptor methods, and tested their accuracy in combination with the SVM classifier. HOG and LESH presented excellent performance, with an accuracy of 96.08 % and 91.2 % respectively. However, the performance of the SFTA makes it unfeasible to use it in this application. The second test applied the best performing image descriptor with the MSER. When testing the MSER, a low accuracy was found, at around 32.85 %.

**Keywords:** Image processing. Machine learning. SVM. HOG. MSER. SFTA.





## LISTA DE FIGURAS

Figura 1	Exemplo de ADAS - controle longitudinal e lateral do veículo. ....	32
Figura 2	Exemplo de ADAS - controle de estabilidade (ESC). ....	33
Figura 3	Exemplos de sinalizações verticais de regulamentação. .	35
Figura 4	Exemplos de sinalizações verticais de advertência. ....	36
Figura 5	Exemplos de sinalizações verticais de indicação. ....	36
Figura 6	Exemplo de segmento de uma imagem em escala de cinza e RGB. ....	43
Figura 7	Exemplo de segmento de imagem em escala de cinza utilizando 8-bits para representar a intensidade. Na imagem à esquerda é visível a tonalidade dos <i>pixels</i> . Já a imagem à direita tem seus respectivos valores de intensidade na posição $x$ e $y$ . ....	44
Figura 8	Gradientes de uma imagem. ....	45
Figura 9	Imagem dividida em células de 8x8. ....	46
Figura 10	Construção do histograma - HOG. ....	47
Figura 11	Resultado do HOG na imagem. ....	48
Figura 12	Funcionamento do método SFTA e criação do vetor de características. ....	51
Figura 13	Exemplo de funcionamento do MSER em uma imagem. ....	53
Figura 14	Exemplo de como o <i>threshold</i> influencia nas regiões resultantes do MSER. ....	54
Figura 15	Diagrama de fluxo do Teste I. ....	58
Figura 16	Diagrama de fluxo do Teste II. ....	59
Figura 17	Exemplo de cada categoria de placa brasileira de sinalização de trânsito presente no banco de dados I. ....	63
Figura 18	Arquitetura em árvore dos classificadores SVMs. ....	65



## LISTA DE TABELAS

Tabela 1	Categorias das placas brasileiras de sinalização de trânsito de regulamentação circular. ....	61
Tabela 2	Categorias das placas brasileiras de sinalização de trânsito de advertência losangular. ....	62
Tabela 3	Categorias das placas brasileiras de sinalização de trânsito de regulamentação triangular. ....	62
Tabela 4	Categorias de placas de sinalização de trânsito presentes no banco de dados II. ....	64
Tabela 5	Valores de $C$ e $\gamma$ presentes em todos os classificadores SVM utilizados. ....	66
Tabela 6	Resultado das placas por formato do método HOG. ....	71
Tabela 7	Resultado das placas circulares do método HOG. ....	72
Tabela 8	Resultado das placas triangulares do método HOG. ....	72
Tabela 9	Resultado das placas losangulares do método HOG. ...	73
Tabela 10	Desempenho geral do método HOG. ....	73
Tabela 11	Resultado das placas por formato do método LESH. ...	74
Tabela 12	Resultado das placas circulares do método LESH. ....	74
Tabela 13	Resultado das placas losangulares do método LESH. ...	75
Tabela 14	Resultado das placas triangulares do método LESH. ...	75
Tabela 15	Desempenho geral do método LESH. ....	75
Tabela 16	Resultado das placas por formato do método SFTA. ...	76
Tabela 17	Resultado das placas circulares do método SFTA. ....	77
Tabela 18	Resultado das placas losangulares do método SFTA. ...	77
Tabela 19	Resultado das placas triangulares do método SFTA. ...	78
Tabela 20	Desempenho geral do método SFTA. ....	78
Tabela 21	Resultado das placas de sinalização de trânsito presentes no teste II. ....	79
Tabela 22	Comparativo entre os métodos separados por finalidade do classificador. ....	80



## LISTA DE ABREVIATURAS E SIGLAS

OICA	Organisation Internationale des Constructeurs d'Automobiles.....	25
WHO	World Health Organization .....	25
IPEA	Instituto de Pesquisa Econômica Aplicada.....	25
ADAS	Advanced Driver Assistance Systems.....	26
MSERs	Maximally Stable Extremal Regions.....	28
HOG	Histogram of Oriented Gradient.....	28
SVM	Support Vector Machine .....	28
LESH	Local Energy-based Shape Histogram .....	28
CONTRAN	Conselho Nacional de Trânsito .....	34
RNA	Redes Neurais Artificiais .....	37
RGB	Red-Green-Blue.....	37
MSERs	Maximally Stable Extremal Regions.....	37
HOG	Histogram of Oriented Gradient.....	37
SVM	Support Vector Machine .....	37
HSV	Hue, Saturation and Value.....	38
SFTA	Segmentation-based Fractal Texture Analysis .....	50
TTBD	Two-Threshold Binary Decomposition.....	50



## LISTA DE EQUAÇÕES

3.1	Módulo do gradiente .....	46
3.2	Orientação do gradiente .....	46
3.3	Cálculo da distância euclidiana. ....	47
3.4	Equação do cálculo das frequências locais máximas .....	49
3.5	Equação para cálculo dos histogramas locais .....	49
3.6	Função <i>Gaussian weighting</i> .....	50
3.7	Equação de cálculo do <i>Two-Threshold Binary Decomposition</i> .....	50
3.8	Relação para construção do vetor de características do SFTA .....	51
3.9	Restrição dos dados de treinamento SVM .....	56
3.10	Equações de otimização do problema do SVM .....	56
3.11	Função do Kernel do SVM .....	56





## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b> .....	25
1.1	OBJETIVOS .....	27
1.1.1	Objetivo Geral .....	27
1.1.2	Objetivos Específicos .....	27
1.2	JUSTIFICATIVA .....	27
1.3	METODOLOGIA .....	28
1.4	ORGANIZAÇÃO DO TRABALHO .....	29
<b>2</b>	<b>RESUMO BIBLIOGRÁFICO</b> .....	31
2.1	SISTEMAS AVANÇADOS DE ASSISTÊNCIA AO MOTORISTA - ADAS .....	31
2.2	PLACAS BRASILEIRAS DE SINALIZAÇÃO DE TRÂNSITO .....	34
2.2.1	Sinalização vertical de regulamentação .....	35
2.2.2	Sinalização vertical de advertência .....	35
2.2.3	Sinalizações verticais de indicação .....	36
2.3	UM ESTUDO DO ESTADO DA ARTE EM DETECÇÃO E CLASSIFICAÇÃO DE PLACAS DE SINALIZAÇÃO DE TRÂNSITO .....	37
<b>3</b>	<b>VISÃO COMPUTACIONAL, TÉCNICAS DE PROCESSAMENTO E CLASSIFICAÇÃO DE IMAGENS</b> .....	41
3.1	VISÃO COMPUTACIONAL .....	41
3.1.1	Processamento Digital de Imagens .....	42
3.1.2	<i>Histogram of Oriented Gradients</i> - HOG .....	44
3.1.3	<i>Local Energy-based Shape Histogram</i> - LESH .....	49
3.1.4	<i>Segmentation-based Fractal Texture Analysis</i> - SFTA .....	50
3.1.5	<i>Maximally Stable Extremal Regions</i> - MSER .....	52
3.2	APRENDIZADO DE MÁQUINA .....	55
3.2.1	<i>Support Vector Machine</i> - SVM .....	55
<b>4</b>	<b>METODOLOGIA DOS EXPERIMENTOS</b> .....	57
4.1	DEFINIÇÃO DOS EXPERIMENTOS .....	57
4.2	O BANCO DE DADOS .....	60
4.2.1	Banco de Dados I .....	60
4.2.2	Banco de Dados II .....	64
4.3	CLASSIFICADOR .....	65
4.4	EXTRATORES DE CARACTERÍSTICAS E IDENTIFICADOR DE REGIÕES DE INTERESSE .....	67

4.4.1	<b>HOG</b> .....	67
4.4.2	<b>LESH</b> .....	68
4.4.3	<b>SFTA</b> .....	68
4.4.4	<b>MSER</b> .....	68
4.5	CONFIGURAÇÃO DA MÁQUINA UTILIZADA NOS TESTES .....	69
<b>5</b>	<b>RESULTADOS</b> .....	71
5.1	TESTE 1 - MÉTODO HOG .....	71
5.2	TESTE 1 - MÉTODO LESH .....	73
5.3	TESTE 1 - MÉTODO SFTA .....	76
5.4	TESTE 2.....	78
5.5	COMPARATIVO ENTRE OS MÉTODOS E TESTES...	80
<b>6</b>	<b>CONSIDERAÇÕES FINAIS</b> .....	83
6.1	CONCLUSÃO .....	83
6.2	TRABALHOS FUTUROS.....	84
	<b>REFERÊNCIAS</b> .....	87

## 1 INTRODUÇÃO

No último século, os veículos automotores passaram por inúmeros avanços tecnológicos, e os carros que antigamente eram motivos de riqueza e status transformaram-se em necessidades, tornando-se peças-chaves de diversas economias no século XX. Atualmente, a indústria automobilística é imensa. De acordo com a *Organisation Internationale des Constructeurs d'Automobiles* (OICA), se o setor automobilístico fosse um país, este teria receita bruta equivalente a €2 trilhões por ano (OICA, 2016b). As empresas que fazem parte deste setor investem, anualmente, mais de €84 bilhões em pesquisa e desenvolvimento (P&D) (OICA, 2016b), gerando mais de 8 milhões de empregos em todo o mundo (OICA, 2016a).

Em 2016, foram produzidos no mundo mais de 90 milhões de veículos automotores, sendo o Brasil responsável por grande parte desta produção com mais de 2 milhões de veículos. Apesar da crise econômica enfrentada em 2016, o país ficou em décimo lugar no que diz respeito a produção de automóveis (OICA, 2016c). Segundo estimativas, em 2014 haviam mais de 1,2 bilhão de automóveis em circulação no mundo (PORTAL, 2016) (OICA, 2015). O Brasil apresentou uma frota de mais de 84 milhões de veículos em 2014 (LÍDER, 2015).

Entretanto, todo avanço tecnológico tem suas consequências. De acordo com *World Health Organization* (WHO), morrem mais de 1,25 milhão de pessoas, por ano, em acidentes que envolvem automóveis (WHO, 2015). Mais da metade destas mortes, são de pessoas que não estavam nos automóveis. Além disso, dados mostram que entre 20 e 50 milhões de pessoas se envolvem em acidentes sem vítimas fatais todos os anos (WHO, 2011). De acordo com a WHO, entre os anos de 2011 e 2020, os acidentes custarão mundialmente mais de \$500 bilhões. No Brasil, esses números não são menos alarmantes, segundo os dados retirados da Seguradora Líder, responsável pelos saques do seguro de Danos Pessoais Causados por Veículos Automotores de via Terrestre (DPVAT). No ano de 2015, cerca de 42 mil pessoas perderam suas vidas em acidentes envolvendo automóveis, 500 mil tiveram algum tipo de seqüela permanente e quase 100 mil pessoas tiveram injúrias sem seqüelas permanentes (LÍDER, 2016).

Os números de acidentes não influenciam somente nas vidas afetadas, mas também nos custos gerados. De acordo com a Fundação Instituto de Pesquisa Econômica Aplicada (IPEA), no ano de 2015, os acidentes representaram uma despesa de mais de R\$40 bilhões ao

governo Brasileiro (CARVALHO et al., 2015).

Em 2015, a *National Highway Traffic Safety Administration* (NHTSA) divulgou um estudo que afirma que 94% dos acidentes automobilísticos tem como causa falhas humanas, tais como falta de atenção, embriaguez ao volante, excesso de velocidade, entre outras (SINGH, 2015).

Motivos como estes, servem de incentivo às montadoras na realização de investimentos em P&D em diversas áreas. Atualmente, já existe em desenvolvimento, carros que são denominados autônomos. Capazes de se deslocar sem auxílio de seres humanos. O *Waymo*<sup>®</sup>, veículo autônomo da *Google*<sup>®</sup>, já rodou mais de 3 milhões de milhas em cidades norte americanas (WAYMO, 2016). Por ser novidade, carros autônomos levarão alguns anos para serem acessíveis a toda população mundial.

Concomitantemente ao veículo autônomo, existe uma alternativa simples, que vem sendo pesquisada a mais de três décadas (BENGLER et al., 2014), o Sistemas avançados de assistência ao motorista, conhecido também como *Advanced Driver Assistance Systems* (ADAS). Estes sistemas auxiliam o condutor em situações na qual pode ou não existir a possibilidade de ocorrer um acidente.

Para conduzir um veículo, o motorista deve estar atento e processando excessivas informações constantemente. As informações visuais são, de fato, as mais importantes, pois indicam ao condutor referências sobre outros veículos e as estradas. Muitas vezes, seja por distração ou dificuldade de visualização, algumas informações visuais passam despercebidas. Um tipo de informação comumente ignorada é das placas de sinalização de trânsito. Tais sinalizações verticais servem para indicar aos motoristas sobre legislações, advertências ou atenção em trechos que, se ignorados, podem resultar em acidentes ou multas. Sendo então, de extrema importância para a segurança no trânsito.

O século XXI vem trazendo diversas inovações, onde a tecnologia contribui para o desenvolvimento da segurança. Visto os fatos apresentados, é imprescindível a necessidade de estudar soluções para contribuir com a segurança nas estradas. A partir dessa adversidade, o presente Trabalho de Conclusão de Curso objetiva a abordagem de técnicas de identificação e classificação de placas brasileiras de sinalização de trânsito, e que poderão ser aplicadas a um sistema ADAS passivo, com intuito de auxiliar o condutor nas detecções de placas de sinalização de trânsito.

## 1.1 OBJETIVOS

Esta seção apresenta os objetivos desejados para o desenvolvimento deste trabalho.

### 1.1.1 Objetivo Geral

Estudar, desenvolver e validar técnicas de classificação e reconhecimento das placas brasileiras de sinalização de trânsito, tornando possível o estudo para implantação em automóveis com suporte à ADAS.

### 1.1.2 Objetivos Específicos

O objetivo geral só pode ser alcançado caso os objetivos específicos desta subseção tenham êxito.

1. Realizar o levantamento do estado da arte na área de detecção e classificação de placas de sinalização de trânsito;
2. Estudar possíveis técnicas que poderiam ser empregadas na parte de classificação e identificação;
3. Criar uma de dados com placas brasileiras de sinalização de trânsito para a parte de classificação;
4. Criar uma de dados com placas brasileiras de sinalização de trânsito para a parte de identificação;
5. Validar e testar as técnicas escolhidas com as placas brasileira de sinalização de trânsito.

## 1.2 JUSTIFICATIVA

Embora carros sejam responsáveis por diversos benefícios, os acidentes são um problema crescente. Segundo Singh (2015), 94% dos acidentes no trânsito ocorrem por motivos e falhas que poderiam ser evitadas. Dada a quantidade de mortes, esses dados são ainda mais assustadores, 1,25 milhão por ano (WHO, 2011). Os avanços em sistemas ADAS já salvaram e, ainda salvam, inúmeras vidas. No entanto, devido à quantidade de acidentes, fatais e não fatais, que ainda ocorrem, evidencia-se a falta de pesquisas e aprimoramento na área.

De acordo com Rezaei e Klette (2017), dos diversos ADAS existentes, os que utilizam visão computacional e processamento de imagem serão, um dia, aplicados em veículos autônomos. Sendo estes o ápice da condução segura e confortável em automóveis. Isso tem levado as montadoras e fabricantes à uma busca incessante pela segurança e conforto nos veículos, influenciando-as a fazerem enormes investimentos em P&D em diversas áreas. Inclusive em sistemas que compõem ADAS, e mais especificamente, em sistemas de reconhecimento de placas de sinalização de diversos países.

O reconhecimento de diferentes objetos em imagens é uma tarefa ampla e extremamente complexa. Envolve desde métodos matemáticos, técnicas de extração de informação e uso de diversos classificadores. Neste caso, de reconhecimento de placas de sinalização de trânsito, existe um ambiente muito poluído, com diversas cores e informações. O reconhecimento das placas de sinalização de trânsito nestes ambientes não é tarefa trivial, e apesar de já existirem estudos nesta área, não existe nenhum método com precisão suficiente que seja equivalente ao ser humano. Sendo assim, ainda há de se pesquisar muito neste assunto.

Os dados encontrados durante a pesquisa bibliográfica desse Trabalho de Conclusão de Curso motivam a realização do mesmo, levando em consideração, além do número de acidentes, os custos gerados por eles e os investimentos na área.

### 1.3 METODOLOGIA

Este Trabalho de Conclusão de Curso terá como base o trabalho descrito e implementado por Greenhalgh e Mirmehdi (2012), que utiliza *Maximally Stable Extremal Regions* (MSERs) para detecção, em conjunto com *Histogram of Oriented Gradient* (HOG) e *Support Vector Machine* (SVM) para a classificação das placas de sinalização de trânsito. O trabalho descrito por Zakir, Zafar e Edirisinghe (2011) também será levado em consideração, sendo este utilizador de *Local Energy-based Shape Histogram* (LESH). Além disso, o uso do *Segmentation-based Fractal Texture Analysis* (SFTA), nunca utilizado nesta abordagem, será proposto. Em Greenhalgh e Mirmehdi (2012), é utilizado um banco de dados público de imagens com placas de sinalização de trânsito da Alemanha (STALKAMP et al., 2011). Existem muitos outros bancos de dados públicos de placas de sinalização de trânsito, por exemplo, com placas de sinalização de trânsito da Suécia (LARSSON; FELSBURG;

FORSSEN, 2011), placas de sinalização de trânsito da Bélgica (TIMOFTE; ZIMMERMANN; GOOL, 2014) e placas de sinalização de trânsito da Espanha (MALDONADO-BASCON et al., 2007). Infelizmente, não há nenhum banco de dados com imagens públicas de placas de sinalização de trânsito do Brasil, levando à necessidade de colher diversas imagens para a criação de um banco de dados, e conseqüentemente, para o treinamento e para os testes. Desse modo, um ou mais bancos de dados serão criados para atingir os objetivos propostos do presente trabalho.

O trabalho foi iniciado com o levantamento bibliográfico. O estudo das técnicas usadas por Greenhalgh e Mirmehdi (2012) será feito subseqüentemente, assim como o estudo das técnicas utilizadas por Zakir, Zafar e Edirisinghe (2011) e de outra técnica amplamente utilizada na academia que poderá ser aplicada à esse problema. Por fim, realizou-se a implementação e os testes em *Matrix Laboratory* (MATLAB). Para isso, iniciou com o teste I (sendo este descrito no capítulo 4), que avaliou qual método dentre os escolhidos, em conjunto com o classificador, classificará melhor as placas brasileiras de sinalização de trânsito. Posteriormente, foi implementada e testada a detecção utilizando MSERs (denominado teste II, também descrito no capítulo 4), sendo este, agregado ao método descritor de características que obteve o melhor desempenho. Os testes e resultados revelaram a acurácia do sistema no geral, mas também demonstrou os testes relativos aos métodos descritores de imagem.

## 1.4 ORGANIZAÇÃO DO TRABALHO

A organização deste trabalho se encontra da seguinte maneira:

- O Capítulo 1 serviu para apresentar um problema real a ser abordado por este Trabalho de Conclusão de Curso.
- O Capítulo 2 apresentará uma revisão sobre o estado da arte na classificação e detecção de placas de sinalização de trânsito pela comunidade acadêmica. Também será apresentada uma pequena introdução ao sistema ADAS, assim como uma breve introdução à regulamentação das placas de sinalização de trânsito no Brasil.
- O Capítulo 3 apresenta conceitos mínimos e esclarecimentos para a implementação dos métodos elegidos para este trabalho.
- O Capítulo 4 demonstra como os métodos definidos serão implementados e testados.

- O Capítulo 5 relata os resultados obtidos nos testes com os métodos implementados no Capítulo 4 e de acordo com mesmos descritos pelo Capítulo 3.
- Por fim, o Capítulo 6 aborda comentários sobre os resultados colhidos no Capítulo 5. Também serão feitas observações sobre possíveis trabalhos futuros.



## 2 RESUMO BIBLIOGRÁFICO

Este capítulo visa esclarecer e compreender o ADAS, assim como providenciar uma breve descrição da legislação brasileira de trânsito e de suas sinalizações. Ao fim do capítulo, é realizado o levantamento do estado da arte pertinente ao tema deste trabalho.

### 2.1 SISTEMAS AVANÇADOS DE ASSISTÊNCIA AO MOTORISTA - ADAS

Sistemas avançados de assistência ao motorista (em inglês *Advanced Driver Assistance System* ou ADAS) é um sistema que pode ser inteligente, e que é capaz de auxiliar tanto passivamente quanto ativamente o condutor. Este sistema, segundo Reif (2014), tem como um dos principais objetivos fazer o automóvel reconhecer o ambiente ao redor, e identificar possíveis situações que coloquem a vida do condutor, passageiros ou pedestres em risco, atuando nessas situações a fim de evitar acidentes (REIF, 2014).

Esse sistema tem ganhado plena atenção atualmente devido ao aumento de acidentes causados por humanos, pois estes, não são capazes de responder a situações tão velozmente quanto sistemas automatizados. Em 2014, 94% dos acidentes nos Estados Unidos foram causados por falhas do condutor (SINGH, 2015) e poderiam ser evitados.

De acordo com Winner et al. (2015), o sistema ADAS pode ser separado em cinco níveis:

- **Nível 1:** este nível é caracterizado pela total ausência de automação, onde o condutor é o único responsável pelo controle longitudinal e lateral do veículo;
- **Nível 2:** o condutor ainda é responsável pelo controle longitudinal ou lateral do veículo, sendo um ou outro automatizado parcialmente;
- **Nível 3:** no terceiro nível, existe certo grau de automação pelo controle longitudinal e lateral do veículo. O condutor deve-se manter atento mesmo com este ativo;
- **Nível 4:** neste nível existe elevado grau de automação, no qual o condutor não necessita ficar atento ao tráfego ou na direção do

automóvel. Neste caso, o condutor ainda pode escolher tomar total controle do veículo;

- **Nível 5:** o quinto nível é marcado pela total ausência de controle manual, ou seja, não existe a necessidade de um condutor, pois o veículo é totalmente conduzido autonomamente. Carros autônomos como o Waymo<sup>®</sup> se enquadram neste nível.

A imagem a seguir exemplifica o controle longitudinal e lateral do veículo.

Figura 1: Exemplo de ADAS - controle longitudinal e lateral do veículo.



Fonte: Adaptado de Planing (2014).

Winner et al. (2015) também especifica que existem três **modos de operação** para o ADAS:

- **Modo de operação A:** o ADAS provê informações e funções de aviso sobre situações, sem interferir diretamente no automóvel. Ex.: Reconhecimento de placas de sinalização de trânsito; aviso sonoro ou visual ao sair das faixas da rodovia;

- **Modo de operação B:** o ADAS encarrega-se imediatamente do controle parcial do carro, mas o condutor ainda mantém o controle total. Ex.: Velocidade de cruzeiro; assistente para manter o carro entre as faixas da rodovia.;
- **Modo de operação C:** o ADAS assume total controle do automóvel em situações onde o motorista irá falhar. Ex.: Frenagem de emergência; sistema de anti-colisão.

Em síntese, sistemas que compõem o ADAS estão sendo desenvolvidos a muito tempo. Segundo Shaout, Colella e Awad (2011) e Bengler et al. (2014), tudo começou com o desenvolvimento do controle de cruzeiro no século passado por Nakamura, Ochiai e Tanigawa (1983).

O sistema de estabilidade eletrônico ou *Electronic Stability Control* é um exemplo de ADAS com **modo de operação B** e **nível 3**, pois somente auxilia o condutor em momentos que o sistema prevê que o automóvel irá falhar, caso não auxilie autonomamente o condutor a restabelecer o controle do veículo.

Figura 2: Exemplo de ADAS - controle de estabilidade (ESC).



Fonte: Adaptado de <https://goo.gl/MW8HLe>.

Os principais e mais conhecidos ADAS existentes no mercado de automóveis são:

- Sistema de frenagem anti-travamento ou *Anti-lock Braking System* (ABS);
- Assistente de direção noturna ou *Automotive Night Vision*;
- Sistema de proteção ao pedestre ou *Pedestrian Protection System*;
- Sistema anti-colisão ou *Collision Avoidance System*;

- Assistente de estacionamento ou *Automatic Parking*;
- Assistente para manter o carro entre as faixas da pista ou *Lane Departure Warning System*.

O ADAS pode utilizar diversos sensores (LIDAR, ultrassônico, câmera de vídeo, entre outros), assim como diversos atuadores (freios, aceleradores, buzinas, alertas áudio-visuais, entre outros). Este Trabalho de Conclusão de Curso abordará um sistema que já é presente em inúmeros automóveis ao redor do mundo (TECHNOLOGIES, 2017). O presente sistema visa apresentar uma possível solução para um ADAS passivo que não atuará no controle do veículo, mas irá alertar o condutor sobre placas de sinalização de trânsito presentes na via. O sistema em si, pode ser classificado como **modo de operação A e nível 1**.

## 2.2 PLACAS BRASILEIRAS DE SINALIZAÇÃO DE TRÂNSITO

O Código de Trânsito Brasileiro define sinais de trânsito como:

“Sinais de trânsito são elementos de sinalização viária que se utilizam de placas, marcas viárias, equipamentos de controle luminosos, dispositivos auxiliares, apitos e gestos, destinados exclusivamente a ordenar ou dirigir o trânsito dos veículos e pedestres” (DENATRAN, 2008).

O Conselho Nacional de Trânsito (CONTRAN) regulariza, fiscaliza e define padrões de sinalizações verticais e horizontais, de advertência, regulamentação ou indicação das vias. A sinalização viária deve transmitir mensagens permanentes ou momentâneas que possibilitam aos usuários identificar condições das vias, ordenar e orientar o tráfego das vias (DENATRAN, 2008)(CONTRAN, 2007b) (CONTRAN, 2007a).

De acordo com o CONTRAN, os sinais verticais devem possuir formas padronizadas, associadas as mensagens que se quer transmitir (DENATRAN, 2008).

Tanto os formatos quanto o tamanho são estabelecidos nos manuais do CONTRAN. Os tamanhos variam do tipo de via e do formato, para os principais formatos temos os exemplos abaixo (CONTRAN, 2007b)(CONTRAN, 2007a):

- Círculos;
- Triângulos invertidos;

- Octógonos;
- Losangos;
- Retângulos.

Existe uma série de regulamentações complexas e específicas sobre situações de como colocar, onde colocar, as distâncias corretas e as angulações das sinalizações. Tais regulamentações não fazem parte da abordagem deste trabalho.

### 2.2.1 Sinalização vertical de regulamentação

O manual de sinalização vertical de regulamentação, afirma que as sinalizações deste conjunto transmitem aos usuários condições, proibições e obrigações que governam as vias. As sinalizações de regulamentação devem possuir somente vermelho, branco e preto nas cores (CONTRAN, 2007a).

Figura 3: Exemplos de sinalizações verticais de regulamentação.

- (a) Parada obrigatória. (b) Dê a preferência. (c) Velocidade máxima permitida. (d) Proibido virar à direita.



(a)



(b)



(c)



(d)

Fonte: Manual Brasileiro de Sinalização de Trânsito - VOLUME I - Sinalização Vertical de Regulamentação.

### 2.2.2 Sinalização vertical de advertência

O manual de sinalização vertical de advertência estabelece que este conjunto deve notificar e orientar os usuários de condições, obstáculos ou restrições nas vias. As sinalizações de advertência devem ser nas cores amarelo ou laranja e preto (CONTRAN, 2007b).

Figura 4: Exemplos de sinalizações verticais de advertência.

- (a)Pista sinuosa à esquerda. (b)Obras. (c)Pista escorregadia.  
(d)Passagem sinalizada de pedestres.



(a)



(b)



(c)



(d)

Fonte: Manual Brasileiro de Sinalização de Trânsito - VOLUME II - Sinalização Vertical de Advertência

### 2.2.3 Sinalizações verticais de indicação

As sinalizações verticais de indicação encontram-se fora do escopo do presente trabalho. Escolheu-se retirar estas sinalizações devido a diversidade de sinalizações de indicação existentes.

A fim de caráter informativo e esclarecedor, abaixo há alguns exemplos de sinalização vertical de indicação:

Figura 5: Exemplos de sinalizações verticais de indicação.

- (a)Indicação de serviços auxiliares. (b)Indicação de distância de atrativos turísticos. (c)Indicação de rodovias e estradas - Rodovia Federal.



(a)



(b)



(c)

Fonte: Manual Brasileiro de Sinalização de Trânsito - VOLUME III - Sinalização Vertical de Indicação.

### 2.3 UM ESTUDO DO ESTADO DA ARTE EM DETECÇÃO E CLASSIFICAÇÃO DE PLACAS DE SINALIZAÇÃO DE TRÂNSITO

Algoritmos para detecção e classificação de placas de sinalização de trânsito vem sendo pesquisados há alguns anos. No trabalho de Ritter (1992), pioneiro no assunto, o autor optou por usar segmentação de cor e rede neural artificial (RNA). Muito tem sido feito nos últimos anos, para melhorar a acurácia do sistema, assim como o custo computacional e velocidade total de processamento. O estado da arte levantado encontrou soluções diferentes para a resolução do mesmo problema.

O trabalho desenvolvido por Sebanja e Megherbi (2010) propõe um sistema que baseia-se em segmentação no espaço de cor *Red-Green-Blue* (RGB), separando todas as partes da imagem que possuam cores definidas entre um intervalo de *threshold*. Na parte de identificação dos formatos e classificação foi decidido o uso de *principal component analysis*, que retira características da parte segmentada da imagem usando autovetores. A imagem, após a segmentação, passa por uma etapa de pré-processamento que rotaciona, escalona e retira informações desnecessárias da mesma. Com uma acurácia de cerca de 99,2%, o sistema levou 2,5 segundos para processar cada imagem, considerando-se impossível sua utilização em tempo real.

O uso das técnicas de MSER, HOG e SVM foi proposto por Greenhalgh e Mirmehdi (2012). O MSERs identifica possíveis regiões mais estáveis que se enquadram em um *threshold*. A imagem primeiro é transformada em escala de cinza normalizada, para intensificar os valores de *Red* e *Blue*, que são as cores das placas de sinalização de trânsito do banco de dados do treinamento. Após extrair todas as regiões com cores que parecem placas de sinalização de trânsito, usa-se HOG para extrair as características que serão usadas no classificador SVM. É usado vários SVMs, onde cada um é treinado para identificar uma característica somente, por exemplo, um SVM para identificar as cores, outro para os formatos e outro para a classe das placas de sinalização de trânsito. Segundo os autores, o método proposto teve uma precisão de 89%.

Qin, Fang e Zhao (2010) abordaram um sistema usando o espaço de cor em RGB para calcular a distância das cores em relação à outro ponto referencial. Neste caso, é necessário usar um *threshold* pois as placas sofrem pequenas variações de cor causadas por diversas situações, como por exemplo iluminação ou serração. Para à escolha de um *threshold* adequado, foi realizado um estudo estatístico em diversas

amostras. A imagem é binarizada usando o valor de *threshold*, e então é usado um classificador multi-classe SVM, que faz a classificação de forma, cor e características retiradas da imagem. Os resultados foram positivos para aplicações em tempo real, mas o próprio autor indica à necessidade de mais estudos para a escolha do *threshold* dinamicamente.

Gonzalez et al. (2011) usou sinal infravermelho para auxiliar na detecção das placas de sinalização de trânsito. O sinal infravermelho é refletido pelas placas de sinalização de trânsito, que sempre possuem cores reflexivas, então as partes com reflexo são extraídas. A transformada de *Hough* é usada para identificar os formatos. O autor decidiu usar *Kalman Filter* para auxiliar no rastreamento das placas de sinalização de trânsito. Várias características são extraídas para classificar placas de sinalização de trânsito, mas não é descrito o método usado. Os resultados são positivos e com uma acurácia relatada de 98% segundo o autor.

Em outro trabalho, foi usado MSER para identificar possíveis regiões com placas de sinalização de trânsito. Essas regiões têm suas características extraídas com *Color Global* e *Local Oriented Edge Magnitude Pattern*. Essas características serão usadas em um classificador SVM. O sistema foi proposto por Yuan et al. (2014) e obteve um resultado muito positivo, com precisão de 97% no banco de dados de imagens criado e testado pelo autor.

Em 2015 foi divulgado um estudo onde se propôs um sistema para reconhecimento de caracteres em placas de sinalização de trânsito. O trabalho produzido por Greenhalgh e Mirmehdi (2012), se baseia em MSERs, transformada de *Hough* e *Hue, Saturation and Value* (HSV). O MSERs provê informações de regiões que podem ou não ter ocorrência de placas, a transformada de *Hough* encontra as regiões que se assemelham às linhas do contorno de placas de sinalização de trânsito. Então é utilizado *Optical Character Recognition* para identificar os caracteres. Para melhorar o tempo computacional, definiu-se regiões de interesse na imagem. O sistema funcionou em 89% dos testes.

Recentemente, Yang et al. (2016) utilizou uma técnica de probabilidade de cor que transforma a imagem em um mapa de probabilidade de placas de sinalização de trânsito. O mapa de probabilidade de cor é estimado com valores manualmente retirados de amostras de imagens com placas de sinalização de trânsito. É usado o espaço de cor *ohta*, pois é invariante a iluminação. O MSERs é extraído desse mapa de probabilidade de cor, onde é possível encontrar regiões estáveis que podem ser placas de sinalização de trânsito. As características HOG são



extraídas no mapa de probabilidade e na imagem em escala de cinza. Esses dois conjuntos de características são concatenadas e usados no classificador multi-classe SVM que classifica em super classes. Para classificar em subclasses, foi escolhido usar um *Convolutional Neural Network*. O sistema teve uma precisão total de 98% nos testes.

Em Zakir, Zafar e Edirisinghe (2011), os autores propõem um sistema completo de reconhecimento e classificação, onde para o reconhecimento de regiões de interesse, é utilizado segmentação por cor usando HSV. O *Contourlet Transform* é utilizado em conjunto com SVM para reconhecer o formato. Enquanto para o classificador, eles usaram LESH para extrair as características junto com o SVM para classificar. O HSV extrai da imagem, as regiões que possuem as cores presentes em placas de sinalização de trânsito. No caso, os autores utilizaram banco de dados próprio com cerca de 1200 imagens de placas de sinalização de trânsito utilizadas na Inglaterra. Ao identificar as regiões de interesse, eles enviam essas regiões para o *Contourlet Transform* que extrai características daquela região e as envia para um SVM, sendo este o responsável por classificar o formato. Se o formato for algum desejado, aquela região é enviada para o descritor de imagem LESH. Por último, este extrai as características e as envia para um classificador SVM. Importante expor que as imagens enviadas para o LESH estão segmentadas acromaticamente. Segundo os autores, o método utilizado obteve de 96% até 98% de acurácia nos testes realizados.



### 3 VISÃO COMPUTACIONAL, TÉCNICAS DE PROCESSAMENTO E CLASSIFICAÇÃO DE IMAGENS

Este Capítulo aborda os aspectos de visão computacional e processamento digital de imagem utilizados neste trabalho. Uma descrição e definição de processamento digital de imagens será exemplificada, e subsequentemente, serão definidas as técnicas de processamento de imagens utilizadas neste trabalho. Ao fim do Capítulo é apresentado a técnica de aprendizado de máquina escolhida para este trabalho.

#### 3.1 VISÃO COMPUTACIONAL

O ser humano tem uma capacidade única de interpretação de dados em tempo real. Os olhos humanos conseguem captar as imagens do mundo real, e o cérebro fica responsável pela extração de informações dessas imagens e pela tomada de decisão de acordo com as imagens obtidas pelos olhos. Podemos exemplificar da seguinte maneira: imagina-se uma pessoa que esteja dirigindo um automóvel em uma rodovia qualquer, os olhos desta pessoa já estão recebendo alta carga de informações como a rota da pista, velocidade, lombadas, outros carros, sinalizações, entre outros. Em determinado momento, um cachorro atravessa a rodovia correndo, este mesmo condutor deve:

- Captar a imagem com os olhos, ou seja, ele deve estar prestando atenção na rodovia;
- O cérebro deve processar essa imagem captada pelos olhos e encontrar o possível problema na imagem, neste caso o cachorro;
- O condutor deve agir de acordo com os dados processados pelo cérebro, a fim de evitar o acidente com o cachorro. O condutor deve, em milésimos de segundo, decidir qual manobra será feita.

Para os humanos, o exemplo acima, apesar de ser uma situação de risco, é uma tarefa fácil. Os seres humanos estão acostumados a tomar decisões em pouco tempo, o cérebro consegue ser rápido o suficiente para compreender as imagens recebidas pelos olhos e exigir uma decisão do corpo. Este fenômeno, que é simples para os humanos, em um computador não é tão trivial. Visão computacional é a área de estudo que procura deixar o computador capaz de, a partir de imagens

digitais, compreender e abstrair dados que possam ser representados de alguma maneira, ou que se possa tirar informações importantes ou tomar decisões. Deixa-se o computador capaz de “enxergar” uma imagem como os humanos fazem, conseguindo inferir em situações simples como reconhecer objetos ou cores.

De acordo com Pedrini e Schwartz (2008), a visão computacional existe para auxiliar na resolução de problemas com demasiada complexidade, utilizando como base e exemplo as funções cognitivas humanas, que tomam decisões baseadas em imagens captadas pelos olhos. Kaehler e Bradski (2017), definem visão computacional como a transformação de dados adquiridos por imagens digitais em outra forma de representação ou variável de decisão.

Visão computacional é uma área extremamente ampla. Geralmente, é associada à algum método de inteligência artificial. Alguns exemplos de estudos dentro da área de visão computacional são:

- Processamento de sinais;
- Processamento de imagem;
- Análise de imagem;
- Reconhecimento de padrões com uso de inteligência artificial;
- Visão de máquina;
- Modelagem tridimensional.

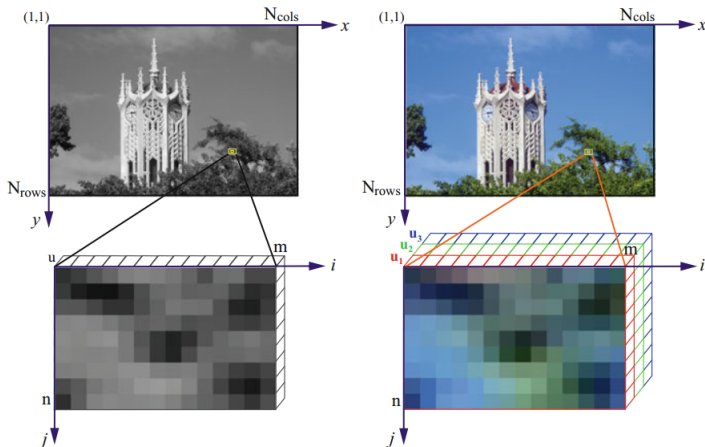
Neste trabalho, temos como objetivo, a partir de uma imagem adquirida por uma câmera, retirar informações para que possamos indicar se nesta imagem existe ou não uma placa de sinalização de trânsito. Assim sendo, estaremos abordando métodos de processamento de imagem que auxiliarão na aquisição de dados que poderão ser usados no classificador, que será descrito posteriormente neste capítulo.

### 3.1.1 Processamento Digital de Imagens

O autor Shih (2010) interpreta imagem de uma forma mais teórica, como sendo um função de sinais de duas dimensões. De forma simplificada, Pedrini e Schwartz (2008) e Gonzalez (2009) determinam imagem como uma função de intensidade no espaço de  $f(x, y)$ , onde  $x$  e  $y$  determinam a intensidade naquela posição. Gonzalez (2009) afirma que caso  $x$  e  $y$  sejam finitos e discretos, as imagens são chamadas de imagens

digitais. Função de sinais de duas dimensões são a demonstração de imagem na forma mais básica, ou seja em escala de cinza. As funções em três dimensões que descrevem as imagens possuem bandas de cores. Essas bandas de cores também são conhecidos por canais de cores. Normalmente utiliza-se *Red-Green-Blue*, ou popularmente conhecidas como RGB, pois são fáceis de serem representadas. Existem muitos outros espaços de cores, mas os mesmos não fazem parte deste trabalho. A concatenação de diversas imagens ao longo do tempo, sejam elas em escala de cinza, RGB ou qualquer outro espaço de cor, é como formamos um vídeo.

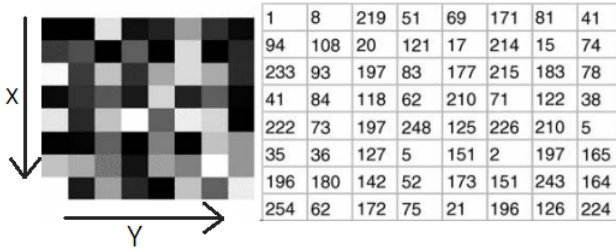
Figura 6: Exemplo de segmento de uma imagem em escala de cinza e RGB.



Fonte: Adaptado de Rezaei e Klette (2017).

Pedriani e Schwartz (2008) definem que cada ponto de  $f(x, y)$  é chamado de *pixel*. A junção de diversos *pixels* constituem um quadro, que formam juntos uma imagem. A união de dois ou mais quadros de *pixels* organizam o espaço de cor, e a junção de diversos quadros de espaços de cores formam os vídeos. Cada *frame* de um vídeo é uma imagem (GONZALEZ, 2009) (SHIH, 2010).

Figura 7: Exemplo de segmento de imagem em escala de cinza utilizando 8-bits para representar a intensidade. Na imagem à esquerda é visível a tonalidade dos *pixels*. Já a imagem à direita tem seus respectivos valores de intensidade na posição  $x$  e  $y$ .



Fonte: Adaptado de Shih (2010).

Shih (2010), afirma que análise digital de imagem é o processo que torna a imagem digitalizada em algum dado ou variável de decisão. O processo de digitalização de imagem é feito pelo sensor que capta luz analógica do ambiente e a discretiza em função de intensidade e espaço em  $f(x, y)$ . Concluímos então, que o processamento digital de imagens é o uso de algoritmos digitais para extrair ou modificar características de imagens digitais com intuito de extrair informações desejadas.

### 3.1.2 *Histogram of Oriented Gradients* - HOG

*Histogram of Oriented Gradients* ou HOG, ou em tradução literal Histograma da orientação dos Gradientes, é um descritor de imagens (extrator de características) imensamente utilizado atualmente para detecção de objetos ou pedestres. Foi descrito pela primeira vez na patente proposta por McConnell (1982). Seu uso na comunidade científica foi iniciado por Dalal e Triggs (2005), que propuseram o uso do HOG como parte de um sistema para identificação de pedestres.

Diferente de outros descritores de imagens, o amplo uso de HOG vem de sua capacidade de representar o objeto globalmente por um simples vetor, facilitando o uso em técnicas de aprendizado de máquina. O tamanho deste vetor depende de algumas configurações, mas podem alcançar tamanhos na ordem de 100 mil dimensões, causando um custo computacional elevado.

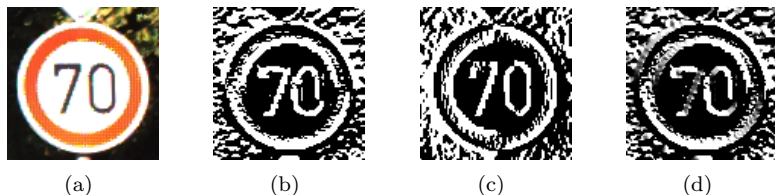
De forma simples, HOG funciona colhendo informações importantes de direção de gradiente de uma imagem dividida em células.

Posteriormente, o algoritmo calcula o módulo e a direção de seus gradientes, divide a imagem em células, calcula o histograma dos gradientes e os concatena em um único vetor (BRADSKI, 2000) (HSIAO; CHAN; WANG, 2016) (MALLICK, 2016) (MCCORMICK, 2013).

Na primeira etapa do algoritmo, calcula-se os gradientes horizontais e verticais da imagem inteira, como na Figura 8.

Figura 8: Gradientes de uma imagem.

(a) Imagem original (b) Gradientes horizontais (c) Gradientes verticais  
(d) Magnitude dos gradientes.



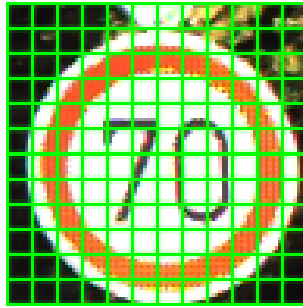
Fonte: Autor.

De acordo com Dalal e Triggs (2005), caso a imagem seja colorida, cada banda de cor terá seu valor de gradiente calculado. Somente a banda de cor com o maior valor de gradiente irá para a próxima etapa.

Observando as imagens da Figura 8, vemos que o cálculo dos gradientes retiram inúmeras informações desnecessárias na imagem. Também percebe-se que apenas informações abruptas (alta frequência) são mantidas, isto é, regiões suaves (baixa frequência) na imagem são retiradas.

No segundo passo, o algoritmo divide a imagem em células. O tamanho de célula utilizado por Dalal e Triggs (2005) é de  $8 \times 8$  pixels, mas esse tamanho pode ser modificado. Supondo que o tamanho seja maior, haverá muita informação descartada no próximo passo. Caso o tamanho de célula seja menor, haverá demasiada informação, fazendo com que o vetor final fique muito grande. Cada caso deverá ser analisado isoladamente, para que se possa definir um tamanho de célula.

Figura 9: Imagem dividida em células de 8x8.



Fonte: Autor.

Após a divisão da imagem em células, a terceira etapa é o algoritmo para calcular o histograma do gradiente de cada célula. Essa etapa se divide em duas partes. Inicialmente, é calculado o gradiente, e para isso, usa-se cada *pixel* que está contido nessa célula. O algoritmo irá calcular a diferença entre o *pixel* da esquerda ao *pixel* da direita, assim como a diferença dos *pixels* da vertical. Com isso, é encontrado dois valores, um que corresponde a mudança no eixo  $x$  e outra no eixo  $y$  do *pixel* em questão. Com esses valores e com as Equações 3.1 e 3.2, encontra-se o módulo e a orientação do gradiente daquele *pixel* respectivamente. Onde  $g_x$  e  $g_y$  são os gradientes referentes ao *pixel* na posição  $x$  e  $y$ .

$$m(x, y) = \sqrt{g_x(x, y)^2 + g_y(x, y)^2} \quad (3.1)$$

$$\Theta(x, y) = \tan^{-1} \frac{g_y(x, y)}{g_x(x, y)} \quad (3.2)$$

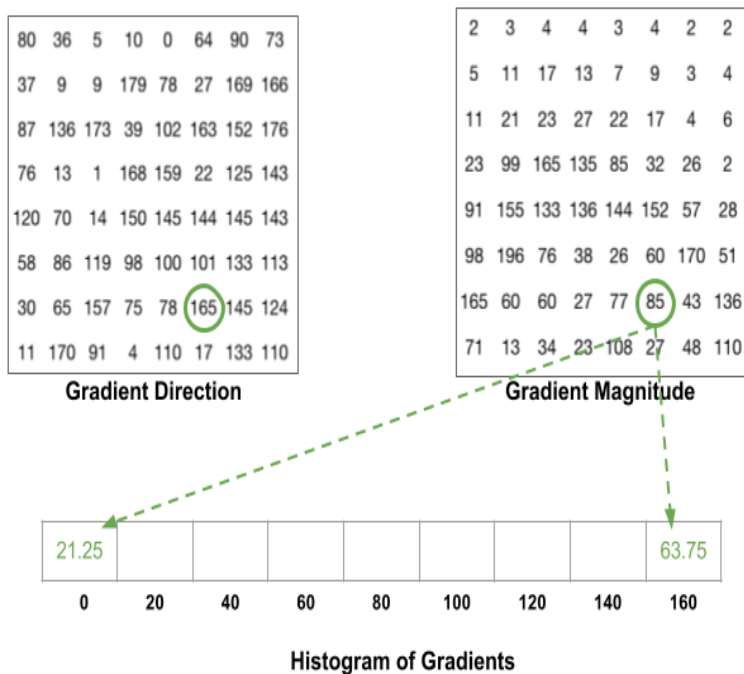
Após calculado o gradiente de todos os *pixels*, é construído o histograma. Em Dalal e Triggs (2005), eles usam *9-bin* (9 divisões, sendo cada uma com 20 valores unitários) no histograma, que é a quantidade de grupos no histograma, valor que é customizável e irá influenciar no custo computacional também. Como os valores da direção do gradiente são representados por graus, ou seja, eles possuem valores que vão de  $0^\circ$  até  $180^\circ$  (pois são *unsigned* e valores opostos são representados pelos mesmos números), usando *9-bin* teremos nove grupos, ou seja, cada grupo terá  $20^\circ$ .

Na construção do histograma, o algoritmo leva em consideração o valor da orientação, e o distribui equivalentemente entre os valores de



orientação possível. Na Figura 10, vemos que o valor de magnitude é de 85 e o valor da orientação é de 165°. Como não existe uma divisão com 165°, o algoritmo acaba por mensurar a contribuição desse valor para as divisões de 160° e 0°. Isto serve para deixar mais fidedigno os valores no histograma.

Figura 10: Construção do histograma - HOG.



Fonte: Mallick (2016).

Na quarta etapa do algoritmo, é feita uma normalização em todas as células, para que não haja discrepância entre células irmãs, visto que o cálculo desses gradientes é influenciado por condições de iluminação.

Para entender o poder da normalização, vamos supor um vetor com os valores [128 64 32] e calcular distância euclidiana (ou  $L_2 - norm$ ), presente na equação 3.3. Com isso teremos:

$$L^2 - norm = \sqrt{128^2 + 64^2 + 32^2} = 146,64 \quad (3.3)$$

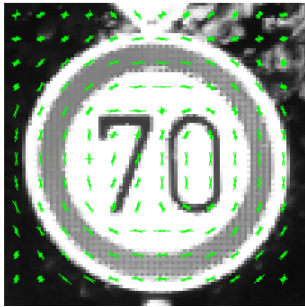
Usando o valor de 146.64 para dividir cada elemento do vetor,

teremos um vetor com os valores  $[0,87 \ 0,43 \ 0,22]$ . Agora, vamos supor um outro vetor com valores três vezes maiores e calcularmos a distância euclidiana.

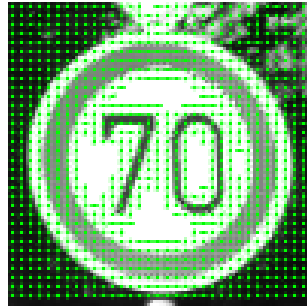
Nota-se que, caso dividirmos o vetor  $[384 \ 192 \ 96]$  por 439,92, teremos o vetor  $[0,87 \ 0,43 \ 0,22]$ . Percebe-se que a normalização retira a escala.

Figura 11: Resultado do HOG na imagem.

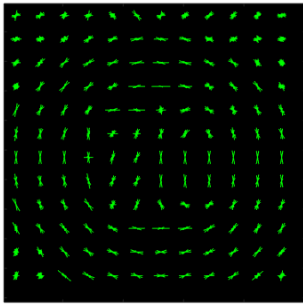
(a) Resultado do HOG na imagem com célula de 8x8 (b) Resultado do HOG na imagem com célula de 3x3 (c) Resultado somente HOG com célula de 8x8 (d) Resultado somente HOG com célula de 3x3.



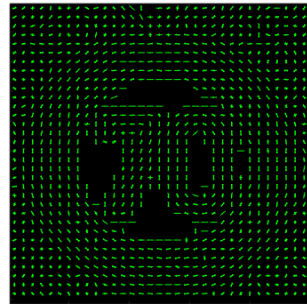
(a)



(b)



(c)



(d)

Fonte: Autor.

De acordo com Dalal e Triggs (2005), ao invés de normalizar cada célula individualmente, o algoritmo normaliza um grupo de células. Dalal e Triggs (2005) usaram grupos de células com tamanho de  $2 \times 2$  pixels, e com sobreposição de 50% entre elas. Após a normalização, o vetor HOG está pronto para ser utilizado.

Na imagem 11, foram testados dois tamanhos de célula, 8x8 e 3x3. O resultado usando célula de 8x8 está nas imagens (a) e (c), e possui muito menos informações. O vetor HOG da célula 8x8 tinha 4356 valores. Já as Figuras (b) e (d) demonstram o resultado HOG da célula com 3x3. Este vetor possui 36864 valores. Vemos que quanto menor a célula, maior o tamanho do vetor de resultado.

### 3.1.3 Local Energy-based Shape Histogram - LESH

O método LESH foi proposto por Sarfraz e Hellwich (2008) e se baseia no método apresentado por Morrone e Owens (1987). Morrone e Owens (1987) descreveram uma maneira de calcular a energia local chamada de *Local Energy Model*. Sarfraz e Hellwich (2008) propuseram criar um histograma a partir do cálculo da energia local.

Para calcular a energia local encontra-se na imagem pontos onde os componentes de frequência locais são máximos. A Equação 3.4 descreve como essas frequências locais máximas são calculadas.

$$E(x) = \max_{-\phi(x) \in [0, 2\pi]} \frac{\sum_n A_n \cos(\phi_n(x) - \bar{\phi}_n(x))}{\sum_n A_n} \quad (3.4)$$

Onde  $A_n$  e  $\phi_n$  são a magnitude e a fase do  $n$ -ésimo componente de *fourier*. Para preservar informações subjacentes de fase, são empregados filtros lineares com pares simétricos e anti-simétricos. Para isso, optou-se por aplicar *Gabor wavelets kernels* especificamente configurados com cinco frequências espaciais e oito orientações. São usados então cada ponto da imagem com frequências locais máximas para cada valor de escala e orientação, produzindo assim, um valor complexo composta com os pares simétricos e anti-simétricos dos filtros.

Com o *Local Energy Model* em mente, e sabendo que as energias locais calculadas realçam as informações de contornos e bordas, Sarfraz e Hellwich (2008) decidiram gerar um histograma local que acumula a energia local em cada orientação do filtro e em diferentes sub-regiões da imagem. Os histogramas locais são calculados baseados em diferentes sub-regiões da imagem, sendo então, concatenados. O histograma local é calculado de acordo com a Equação 3.5.

$$h_{r,b} = \sum W_r \times E \times \delta(L - b) \quad (3.5)$$

Onde  $b$  representa a divisão atual do histograma,  $L$  é o mapa de

orientação,  $E$  é a energia local calculada a partir da equação 3.4.  $W$  é a função *Gaussian weighting* no centro da região  $r$ . O  $W$  é calculado pela Equação 3.6.

$$W_r = \frac{1}{\sqrt{2\pi\sigma}} e^{-(x-r_{x_0})^2 + (y-r_{y_0})^2 / \sigma^2} \quad (3.6)$$

A Equação 3.6 fornece margens mais suaves entre as divisões dos histogramas. No caso do autor, foram extraídos oito orientações no filtro em 16 partições de imagem, criando um vetor com 128 dimensões. No caso deste trabalho, o valor utilizado foi 256 partições, sendo assim, o vetor final tem 2048 dimensões.

### 3.1.4 Segmentation-based Fractal Texture Analysis - SFTA

Segmentation-based Fractal Texture Analysis (SFTA) é um método proposto por Costa, Humpire-Mamani e Traina (2012) visando a análise e extração de características de textura. Este método pode ser dividido em duas partes principais. A primeira é a conversão da imagem para escala de cinza. A segunda é a utilização da imagem em escala de cinza em um método chamado *Two-Threshold Binary Decomposition* (TTBD), onde a imagem em escala de cinza é decomposta em um conjunto de imagens binarizadas em um espaço fractal, e então, são calculadas as regiões de bordas (COSTA; HUMPIRE-MAMANI; TRAINA, 2012) (SANTHI; ACHARJYA; EZHILARASAN, 2015).

O algoritmo de TTBD recebe a imagem em escala de cinza e retorna um conjunto de imagens binárias. TTBD começa computando dois valores de *threshold*, e selecionando na imagem em escala de cinza, os componentes que estão entre essa faixa de *threshold*. Segundo Costa, Humpire-Mamani e Traina (2012), o uso do algoritmo de *Otsu* encontra o valor de *threshold* que minimiza a variância intra-classe.

$$I_b(x, y) = \begin{cases} 1 & \text{se } t_L < I(x, y) < t_U \\ 0 & \text{Caso Contrário} \end{cases} \quad (3.7)$$

Na Equação 3.7, temos o resultado de uma imagem binária, onde  $t_L$  é o valor de *threshold* inferior e  $t_U$  é o valor de *threshold* superior. Esse processo é realizado diversas vezes, tendo como resultado um conjunto de imagens binárias.

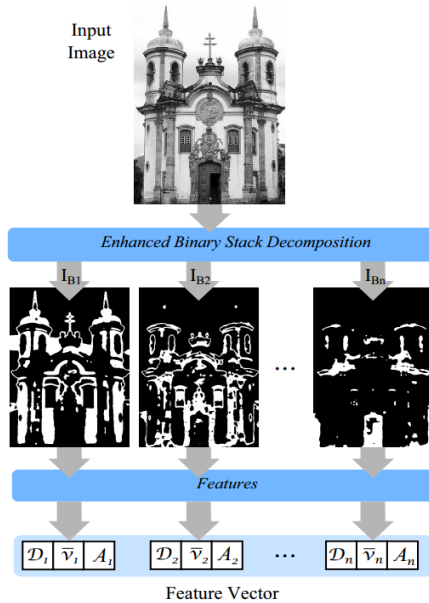
Depois de calculado o conjunto de imagens binárias, é construído

o vetor de características SFTA usando a relação da Equação 3.8.

$$\Delta(x, y) = \begin{cases} 1 & \text{se } \exists (x', y') \in N_8[(x, y)] : \\ & I_b(x', y') = 0 \wedge \\ & I_b(x, y) = 1 \\ 0, & \text{Caso Contrário} \end{cases} \quad (3.8)$$

Onde  $N_8[(x, y)]$  é o conjunto de *pixels* que possui 8 vizinhos em  $(x, y)$ .  $\Delta(x, y)$  recebe 1 se o valor correspondente da posição  $(x, y)$  for 1 em  $I_b = (x, y)$  e está posição tiver pelo menos 1 *pixel* vizinho com valor 0, caso contrário o valor de  $\Delta(x, y)$  é 0.

Figura 12: Funcionamento do método SFTA e criação do vetor de características.



Fonte: Costa, Humpire-Mamani e Traina (2012).

O resultado destas extrações é composto por três características de cada imagem binária, sendo elas: as dimensões fractais; as médias de cinza; a área de cinza ou área dos *pixels*. A imagem 12 demonstra a entrada e a saída do SFTA. O custo computacional do SFTA é linear, sendo então uma boa opção para detecção de bordas e texturas. De

acordo com Costa, Humpire-Mamani e Traina (2012), SFTA teve performance e acurácia muito superior a outros métodos de extração de características como *Gabor* e *Haralick*.

### 3.1.5 *Maximally Stable Extremal Regions* - MSER

*Maximally Stable Extremal Regions* ou MSER, foi proposto por Matas et al. (2002) inicialmente para achar regiões de interesse que possuíssem correspondentes confiáveis em duas imagens, ou seja, em imagens ou câmeras estereoscópicas para reconstrução de imagens em três dimensões ou encontrar seus correspondentes.

Esse método teve grande apelo em outras áreas e hoje também é utilizado em diversas aplicações diferentes do designado inicialmente. MSER já foi utilizado em rastreamento de objetos (DONOSER; BISCHOF, 2006), detecção de placas de sinalização de trânsito (GREENHALGH; MIRMEHDI, 2012), detecção e reconhecimento de caracteres com *Optical Character Recognition* (OCR) (ISLAM et al., 2016), inspeção de estradas de ferro (DUBEY; JAFFERY, 2016) e reconhecimento de placas veiculares (GU et al., 2015).

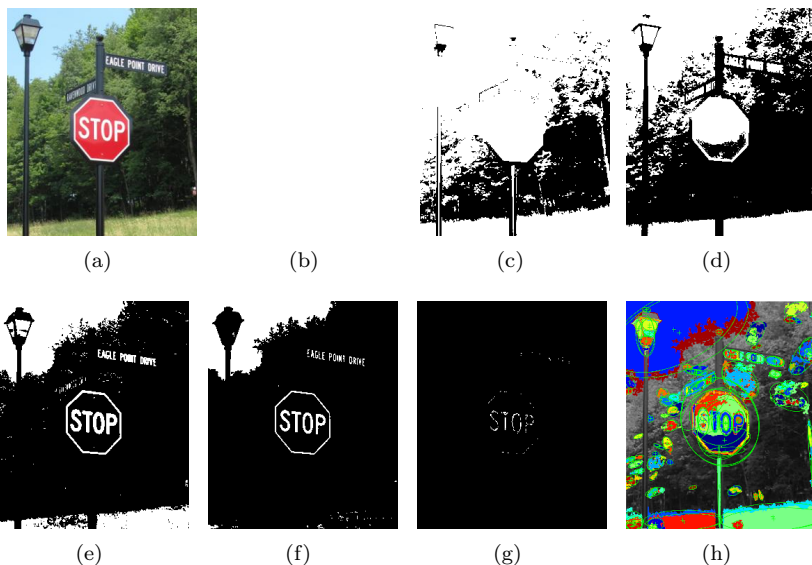
Em Matas et al. (2002) é definido o nome *extremal region* como uma região de componentes conectados, ou *pixels* em uma imagem, que possuem mais alta ou mais baixa intensidade que os outros componentes conectados ou *pixels* nas extremidades. *Maximally Stable Extremal Regions* é definido na teoria como sendo a seqüência de *extremal regions* aninhadas, ou seja, uma sequencia de *extremal regions* maximamente estáveis. Simplificadamente, o MSER encontra as regiões de uma imagem incrementando o valor de *threshold* em todos os valores de intensidade possíveis.

Para exemplificar o conceito desse algoritmo de processamento de imagens, supõe-se uma imagem em escala de cinzas com valores de intensidade entre 0 e 255. Varia-se o valor de um *threshold* entre 0 e 255, e então pega-se todos os valores dos *pixels* dessa imagem abaixo de um valor de *threshold* e os determina como sendo *pixels* pretos. Os *pixels* que tem valor acima deste valor de *threshold* serão brancos. A cada variação de *threshold* encontram-se regiões (componentes conectados). Essas regiões são as *extremal regions* (regiões extremas) da imagem, naquele valor de *threshold*. A cada etapa desse processo aumenta-se o valor de *threshold*, conseqüentemente algumas regiões geradas quando o *threshold* estava baixo vão pertencer e então unir-se a regiões geradas com *threshold* mais alto (MATAS et al., 2002) (HOLLOWAY; LONDON; DA-

VIES, 2012) (TUYTELAARS; MIKOLAJCZYK, 2008). O algoritmo MSER está representado na Figura 13, onde a Figura 13-(h) é o resultado de todas as regiões encontradas na Figura 13-(a).

Figura 13: Exemplo de funcionamento do MSER em uma imagem.

- (a) Imagem original (b)  $Threshold = 0$  (c)  $Threshold = 50$   
 (d)  $Threshold = 100$  (e)  $Threshold = 150$  (f)  $Threshold = 200$   
 (g)  $Threshold = 250$  (h) Resultado final de todas as regiões e suas intensidades.

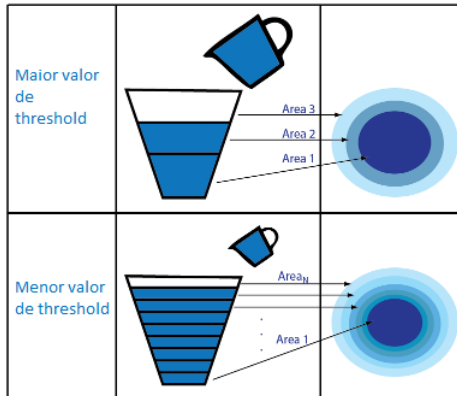


Fonte: Autor.

Outro exemplo é demonstrado na Figura 14, onde é possível ver como as regiões com componentes conectados são de fato extraídas através do valor de *threshold*.

O MSER inicialmente proposto é bem eficiente, com um custo computacional quase linear, de  $O(n \log(\log(n)))$ , onde  $n$  é o número total de *pixels* na imagem. Nistér e Stewénius (2008) propuseram uma implementação do algoritmo MSER que no pior caso tem o custo de  $O(n)$ .

Figura 14: Exemplo de como o *threshold* influencia nas regiões resultantes do MSER.



Fonte: Adaptado de MathWorks (2012).

Algumas outras características foram observadas por Matas et al. (2002), e além das mencionadas acima, o método MSER tem as seguintes propriedades:

- Invariante para *affine transformations* ou transformações afins, isto é, pequenas distorções não influenciam na detecção das regiões.
- A extração das regiões do MSER é muito estável, visto que a região é selecionada somente se estiver em outra região quando ambas estão incluídas no valor de *threshold*.
- O método é invariante à translação, rotação e escalonamento quando sutis. Esta característica é funcional e fundamental em reconstrução de imagens estereoscópica.
- Detecção multi-escalar, ou seja, regiões grandes e pequenas podem ser detectadas igualmente.
- Tem um custo computacional baixo.

MSER não funciona bem quando a imagem é borrada, pois não há distinção entre os extremos das regiões.



## 3.2 APRENDIZADO DE MÁQUINA

Inteligência artificial é o ramo da computação que estuda técnicas que proporcionam, de alguma forma, capacidade de decisão as máquinas ou dispositivos. Uma de suas vertentes é o aprendizado de máquina. Esta, estuda métodos e técnicas para que os algoritmos consigam aprender a partir de exemplos.

O aprendizado de máquina pode ser supervisionado ou não-supervisionado. O primeiro indica que o algoritmo, durante o treinamento, já saberá o resultado final, ou seja, há um supervisor que dirá se o resultado condiz com a realidade desejada. Geralmente é utilizada para classificação ou regressão, já que treinamos o algoritmo com exemplos. A segunda, o treinamento se dá sem a existência de exemplos, deixando-o explorar os resultados possíveis. Normalmente se utiliza esta última técnica em problemas de associação ou agrupamento (LORENA et al., 2007).

Uma técnica amplamente utilizada de aprendizado de máquina supervisionado é o *support vector machine*, ou em tradução máquina de vetores de suporte. Uma abordagem mais aprofundada desta técnica se dará na próxima subseção.

### 3.2.1 *Support Vector Machine* - SVM

Como observado anteriormente, máquina de vetores de suporte é uma técnica de aprendizado de máquina supervisionado. Os autores em Cortes e Vapnik (1995) descreveram o uso desta técnica quando estudavam a teoria do aprendizado estatístico. A aplicação desta técnica se popularizou devido à sua grande capacidade de generalização e sua ótima eficiência (AKANDE et al., 2014) (CASAGRANDE et al., 2017).

Segundo Hsu, Chang e Lin (2003), a classificação envolve separar os dados em dados de treinamento e dados de testes. Cada dado de entrada no treinamento tem um rótulo, ou seja, tal dado é pertencente a uma classe, sendo que essa classe pode possuir diversos atributos. O objetivo principal do SVM é criar um modelo que consiga prever a classe dos dados de teste, apenas com os atributos desse dado de teste (HSU; CHANG; LIN, 2003).

A partir de um conjunto de dados de treinamento, rotulados de acordo com a Equação em 3.9, o algoritmo do SVM procura uma

solução para o problema de otimização descrito na Equação em 3.10.

$$\begin{aligned} &(x_i, y_i), i = 1, \dots, l \\ &\text{onde } x_i \in R^n \\ &\text{e } y_i \in 1, -1^l \end{aligned} \tag{3.9}$$

$$\begin{aligned} &\min_{\mathbf{w}, b, \xi} \frac{1}{2} \mathbf{w}^T \mathbf{w} + C \sum_{i=1}^l \xi_i \\ &\text{sujeito à } y_i(\mathbf{w}^T \phi(x_i) + b) \geq 1 - \xi_i \\ &\text{e } \xi_i \geq 0 \end{aligned} \tag{3.10}$$

Então, os vetores de treinamento  $x_i$  são mapeados pela função  $\phi$  em um espaço dimensional superior. O SVM procura um hiperplano que separe linearmente com a maior margem possível dentro deste espaço dimensional superior.  $C > 0$  é o parâmetro de penalidade (HSU; CHANG; LIN, 2003)(CORTES; VAPNIK, 1995). Como descrito em Casagrande et al. (2017), os valores da penalidade possuem um papel importante na acurácia do sistema. Um valor elevado de  $C$  pode distorcer a acurácia, tornando-a alta na fase de treinamento e baixa na fase de testes. Um valor muito pequeno de  $C$  pode deixar o sistema com uma precisão muito baixa (HSU; CHANG; LIN, 2003) (CASAGRANDE et al., 2017). Assim como o valor de gama ( $\gamma$ ) pode alterar o resultado final, valores altos de  $\gamma$  podem causar *overfitting*, e valores muito baixos podem causar *underfitting* (HSU; CHANG; LIN, 2003) (CASAGRANDE et al., 2017).

Entretanto, muitos autores utilizam uma função de *kernel* que está descrita na Equação 3.11 (HSU; CHANG; LIN, 2003).

$$K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2), \gamma > 0 \tag{3.11}$$

As funções definidas para o *kernel* são descritas para mapear os dados no espaço dimensional superior. A função em 3.11 é chamada de *radial basis function* e mapeia os dados de forma não linear. Diversas funções de *kernel* são propostas e resultam em diferentes soluções e performance no SVM. No presente trabalho optou-se na implementação descrita por Hsu, Chang e Lin (2003), onde o *kernel* utilizado é o *radial basis function*, pois o mesmo demonstra uma ótima capacidade de generalização.

## 4 METODOLOGIA DOS EXPERIMENTOS

O presente capítulo visa esclarecer a metodologia utilizada para a implementação e realização dos testes. Na Seção 4.1, é discutido os dois testes que foram aplicadas neste trabalho. Já a Seção 4.2 apresenta as duas bases de dados de placas brasileiras de sinalização de trânsito criadas para este trabalho, visto que não foi encontrada nenhuma base de dados pública. Na Seção 4.3 é apresentada a arquitetura do classificador SVM, assim como as variações de sua configuração entre os diferentes métodos descritores de imagens utilizados. Por fim, a Seção 4.4 aborda todas as configurações que foram empregadas nos métodos usados. A Seção 4.5 se refere a configuração do computador no qual os testes foram realizados.

### 4.1 DEFINIÇÃO DOS EXPERIMENTOS

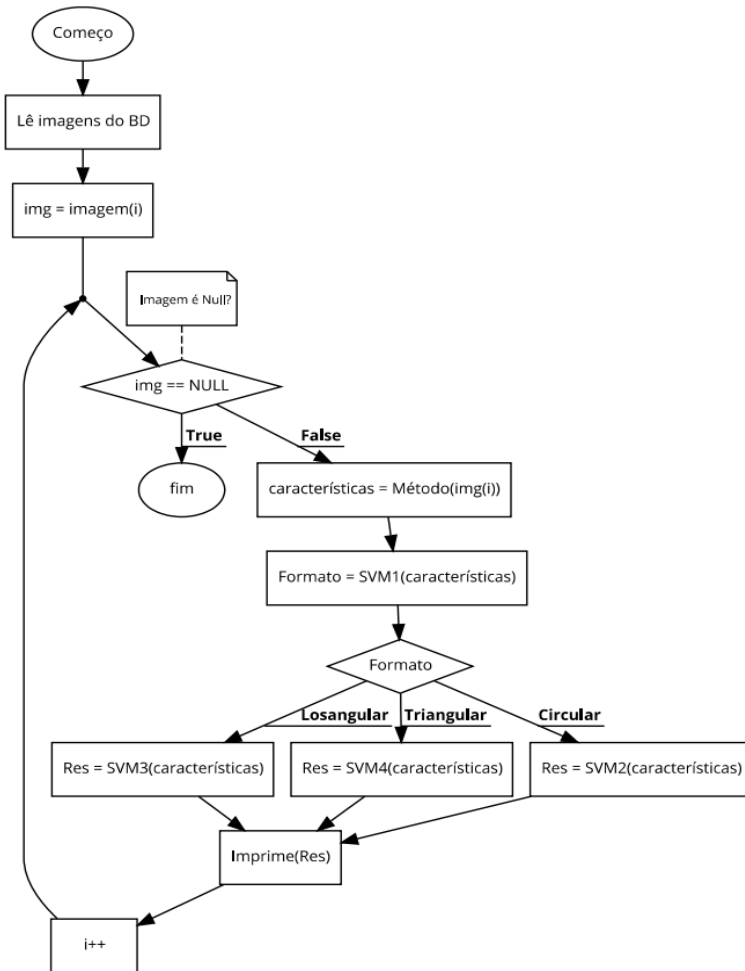
O objetivo deste trabalho é avaliar e validar os algoritmos propostos para identificar e classificar placas brasileiras de sinalização de trânsito. Com isso em mente, definiu-se a realização de dois experimentos.

O primeiro irá validar os métodos de extração de características e o classificador. Serão testados três métodos distintos, mas todos possuirão somente um tipo de classificador que, no caso deste trabalho, é o SVM. Cada imagem terá suas características extraídas por cada um dos três métodos, e com isso, será utilizado o vetor de resultados para treinar o classificador. Após o classificador treinado, as imagens de teste terão suas características extraídas, e o resultado disso será utilizado no classificador, tornando possível à predição de determinada placa à qual classe ela pertence. O diagrama de fluxo está demonstrado na Figura 15.

O segundo teste irá validar o método de identificação de regiões que são mais estáveis na imagem, o que as torna possíveis placas de sinalização de trânsito. Neste caso, será utilizado o MSER e o método que melhor for avaliado no primeiro teste para extrair as características. O algoritmo deste teste irá ler cada imagem e extrair as regiões estáveis com o MSER. Após, cada região irá para o método escolhido para extrair características. Com os resultados da extração de característica de cada região, o classificador irá identificar se a região é uma placa ou não e a qual classe ela pertence. A Figura 16 demonstra o diagrama de

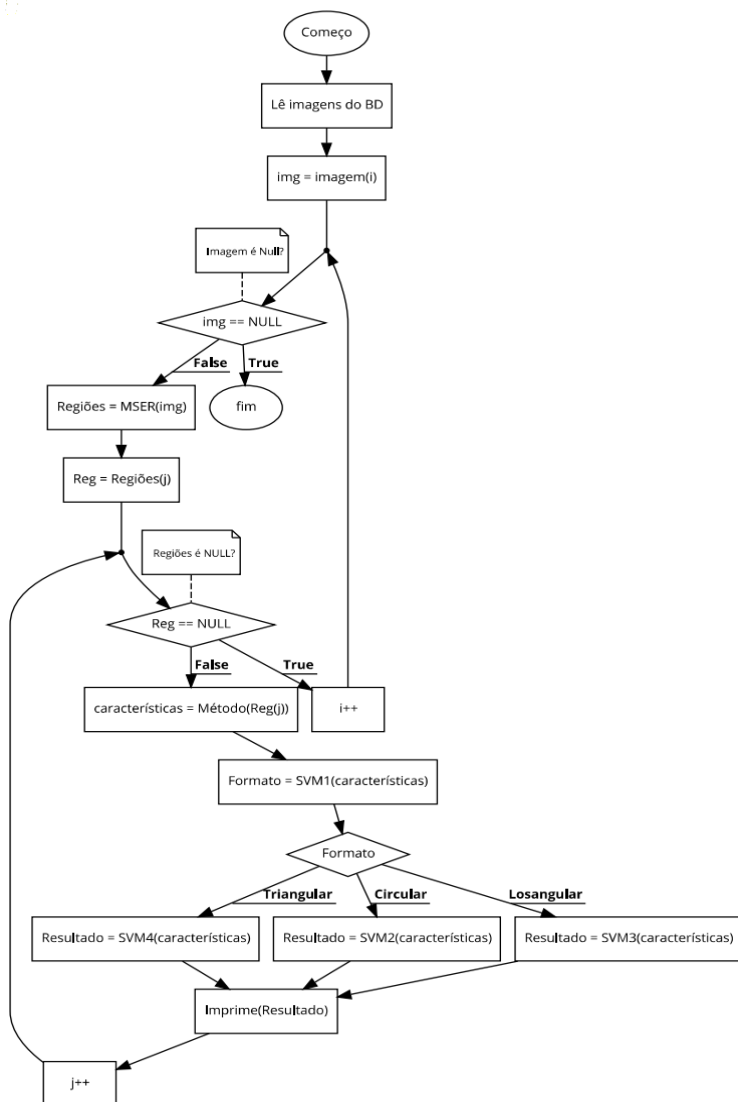
fluxo do segundo teste.

Figura 15: Diagrama de fluxo do Teste I.



Fonte: Autor.

Figura 16: Diagrama de fluxo do Teste II.



Fonte: Autor.

Com esses dois testes, conseguiremos definir a melhor abordagem, e se os métodos escolhidos para a identificação e classificação de

placas brasileiras de sinalização de trânsito são viáveis.

## 4.2 O BANCO DE DADOS

Serão criados dois bancos de dados para que se possa realizar os testes. O banco de dados I servirá para o primeiro teste, e o banco de dados 2 será utilizado no segundo teste.

### 4.2.1 Banco de Dados I

Devido a inexistência de um banco de dados público de placas de sinalização de trânsito Brasileira, foi necessária, para a realização deste trabalho, a criação de um banco de dados para os testes. Ao montar um banco de dados, é fundamental colher imagens que con dizem com a realidade, ou seja, deve-se coletar imagens de tamanhos diferentes, pequenas variações na angulação ou rotação, distintos tipos de luminosidade e imagens com qualidades diversas.

No caso deste trabalho, a coleta das imagens é algo extremamente custoso, pois requer um carro coletando as imagens em diferentes situações rodando em diferentes estradas. Para contornar o tempo necessário circulando por vias e agilizar o processo de montagem do banco de dados, escolheu-se utilizar imagens retiradas do *Google Street View*. Esta abordagem mostrou-se positiva nos testes, visto que é possível salvar imagens em alta resolução (4800x2888 *pixels*), tornando possível utiliza-lás.

O bando de dados criado<sup>1</sup> possui 475 imagens, sendo inicialmente divididas pelos seguintes formatos:

- 266 placas circulares;
- 184 placas losangulares;
- 25 placas triangulares.

Reitera-se que as placas circulares e triangulares são de regulamentação, e as placas losangulares são de advertência.

Para validar os testes, o banco de dados foi separado entre 80% das imagens para teste e validação na etapa de treinamento, e 20%

---

<sup>1</sup>Banco de dados I poderá ser consultado no seguinte link: <https://goo.gl/K7XqMB>

para os testes finais. A separação é feita aleatoriamente, para não haver tendências no classificador. O objetivo desta separação é avaliar os métodos propostos com imagens desconhecidas da etapa de treinamento, resultando em uma resposta mais fidedigna. Neste banco de dados temos:

- 373 placas para teste e validação na etapa de treinamento;
- 102 placas para os testes finais.

O tamanho da imagem foi padronizado para  $50 \times 50$  *pixels*. Esta etapa se fez necessária devido a alguns dos métodos utilizados, os quais possuem o tamanho do vetor de resultados variando pelo tamanho das imagens. Uma imagem com tamanho maior poderia melhorar o resultado final, mas sacrificaria o custo computacional envolvido. Neste caso, optou-se por um tamanho menor pois o custo computacional para o treinamento, principalmente para realizar a otimização que será discutida na próxima seção, seria extremamente elevado.

Tabela 1: Categorias das placas brasileiras de sinalização de trânsito de regulamentação circular.

<b>Placas circulares</b>	<b>Treino e Validação (80%)</b>	<b>Teste Final (20%)</b>
40 km/h	23	6
60 km/h	22	6
80 km/h	56	14
Pare	29	8
Proibido estacionar	19	5
Proibido parar e estacionar	24	7
Proibido retornar pela esquerda	12	4
Proibido virar à direita	2	1
Proibido virar à esquerda	4	1
Siga em frente	18	5
<b>Total</b>	<b>209</b>	<b>57</b>

Tabela 2: Categorias das placas brasileiras de sinalização de trânsito de advertência losangular.

<b>Placas losangulares</b>	<b>Treino e Validação (80%)</b>	<b>Teste Final (20%)</b>
Curva acentuada à direita	14	6
Curva acentuada à esquerda	14	6
Lombada	56	14
Obras à frente	5	8
Faixa de pedestres	43	5
Semáforo à frente	12	7
<b>Total</b>	<b>144</b>	<b>40</b>

Tabela 3: Categorias das placas brasileiras de sinalização de trânsito de regulamentação triangular.

<b>Placas triangulares</b>	<b>Treino e Validação (80%)</b>	<b>Teste Final (20%)</b>
Dê a preferência	20	5
Outras	16	0
<b>Total</b>	<b>36</b>	<b>5</b>

No total, teremos dez categorias de placas circulares de regulamentação descritas na Tabela 1. Já na categoria de placas de advertência losangulares temos seis categorias, que estão totalizadas na Tabela 2. Para a placa de regulamentação triangular temos apenas duas categorias, ambas descritas na Tabela 3. Ao adicionar uma categoria de outras, conseguiremos distinguir se uma placa desta categoria é realmente uma placa triangular de “dê a preferência”.

Pelo fato do *Google Street View* mostrar fotos somente retiradas durante o dia, o presente banco de dados carece de imagens com placas de sinalização em ambientes escuros ou noturnos.

A Figura 17 dispõe um exemplo de placa de sinalização de trânsito de cada categoria presente no banco de dados criado. As imagens foram escolhidas aleatoriamente da base de dados.



Figura 17: Exemplo de cada categoria de placa brasileira de sinalização de trânsito presente no banco de dados I.

(a) 40 km/h (b) 60 km/h (c) 80 km/h (d) Pare (e) Proibido estacionar  
 (f) Proibido parar e estacionar (g) Proibido retornar pela esquerda  
 (h) Proibido virar à direita (i) Proibido virar à esquerda (j) Siga em frente  
 (k) Curva acentuada à direita (l) Curva acentuada à esquerda  
 (m) Lombada (n) Obras à frente (o) Faixa de pedestres (p) Semáforo à frente  
 (q) Dê a preferência



Fonte: Autor.

### 4.2.2 Banco de Dados II

Este banco de dados tem como objetivo apenas validar o segundo teste. Para isso, foram coletadas imagens distintas das placas em ambientes poluídos, ou seja, as placas estão na imagem se misturando em ambientes aleatórios, como florestas, prédios e carros. O método MSER deverá encontrar nas imagens regiões estáveis que poderão ser, ou não, placas de sinalização de trânsito.

Seguindo a lógica da criação do banco de dados I, neste banco de dados as imagens também foram coletadas utilizando o *Google Street View*.

Tabela 4: Categorias de placas de sinalização de trânsito presentes no banco de dados II.

Classes de Placas	Quantidade
40 km/h	5
60 km/h	2
80 km/h	5
Pare	6
Proibido estacionar	1
Proibido parar e estacionar	1
Proibido retornar pela esquerda	3
Proibido virar à direita	6
Proibido virar à esquerda	5
Siga em frente	4
Curva acentuada à direita	7
Curva acentuada à esquerda	7
Lombada	3
Faixa de pedestres	6
Semáforo à frente	2
Dê a Preferência	7
<b>TOTAL</b>	<b>70</b>

Este banco de dados<sup>2</sup> possui 59 imagens, sendo que nestas imagens temos um total de 70 placas brasileiras de sinalização de trânsito, pertencentes a 16 classes distintas. A Tabela 4 mostra como as placas de sinalização de trânsito estão categorizadas.

---

<sup>2</sup>Banco de dados II poderá ser consultado no seguinte link: <https://goo.gl/ahmvXD>

Diferente do banco de dados I, este banco de dados não será separado com uma parte de treino e validação e outra de testes. Isso se deve pelo fato de que no segundo teste não haverá a necessidade de treinar um classificador, então as imagens serão utilizadas diretamente para testes.

### 4.3 CLASSIFICADOR

O classificador desempenhará um importante papel nos testes, sendo assim, sua configuração e definição de arquitetura irão influenciar diretamente nos resultados finais.

A fim de padronização, à arquitetura definida do classificador será a mesma em todos os testes, alterando apenas os parâmetros da função do *kernel*. A função de *kernel* utilizada neste trabalho é a *radial basis function*. Os parâmetros que serão alterados em cada SVM serão o  $C$  e  $\gamma$ , ambas definições estão descritas na Seção 3.2.1.

Uma árvore de classificadores será utilizada, a qual podemos separar em dois níveis. O primeiro nível classificará o formato da placa desejada. O segundo nível irá definir qual classe de placa de sinalização está sendo classificada. A Figura 18 demonstra como a arquitetura em árvore do classificador está disposta.

Figura 18: Arquitetura em árvore dos classificadores SVMs.

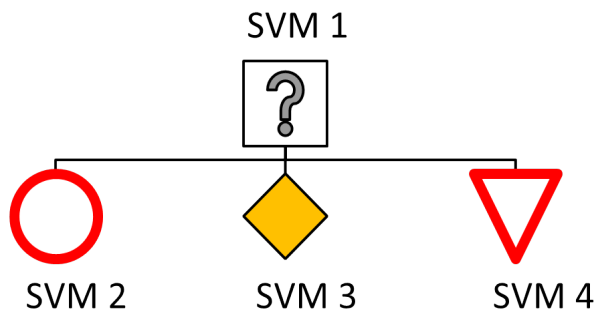


Tabela 5: Valores de  $C$  e  $\gamma$  presentes em todos os classificadores SVM utilizados.

Método	Classificador	$C$	$\gamma$
<b>HOG</b>	<i>SVM 1</i>	46,416	46,416
	<i>SVM 2</i>	1000	46,416
	<i>SVM 3</i>	215,44	46,416
	<i>SVM 4</i>	46,416	46,416
<b>LESH</b>	<i>SVM 1</i>	0,46416	10
	<i>SVM 2</i>	1000	215,44
	<i>SVM 3</i>	46,416	46,416
	<i>SVM 4</i>	2,1544	10
<b>SFTA</b>	<i>SVM 1</i>	1000	1000
	<i>SVM 2</i>	215,44	1000
	<i>SVM 3</i>	1000	1000
	<i>SVM 4</i>	0,46416	1000

Onde o classificador SVM 1 é um SVM multiclasse com três classes, onde a classificação será apenas pelo formato da placa como relatado na Seção 4.2.

O classificador SVM 2 é um SVM multiclasse com dez classes, que classificará apenas as placas de regulamentação circulares descritas na Tabela 1.

Já o SVM 3 é um classificador multiclasse com seis classes, e que tem como objetivo apenas classificar as placas de advertência losangulares, como as apresentadas na Tabela 2.

O SVM 4 é o classificador responsável por classificar a placa de regulamentação triangular. Este é o único classificador SVM binário utilizado nesta arquitetura. Sua definição foi necessária pois temos somente uma classe de placa que se enquadra nesta categoria, sendo assim, é necessário ter exatidão de que o resultado final deste classificador será correto, resultando positivo caso for uma placa de Dê a Preferência negativo caso contrário.

Para o treinamento do SVM 4, foram utilizadas as 20 placas de “Dê a Preferência” e 16 placas categorizadas como outras. Essas 16 placas pertencem cada um a uma categoria do banco de dados I, com exceção da placa de “Dê a Preferência”. Desta forma, o classificador será treinado para classificar com exatidão somente placas de “Dê a Preferência”.

Todos os classificadores tiveram seus valores de  $C$  e  $\gamma$  customi-

zados usando o *grid search algorithm* presente no MATLAB. A Tabela 5 mostra todos os valores de  $C$  e  $\gamma$  de todos os classificadores criados e utilizados neste trabalho.

#### 4.4 EXTRATORES DE CARACTERÍSTICAS E IDENTIFICADOR DE REGIÕES DE INTERESSE

O presente trabalho realiza a comparação entre três métodos difundidos pela literatura acadêmica. São eles o HOG, LESH e SFTA. Posteriormente é avaliado o desempenho de um identificador de regiões estáveis na imagem, que no nosso caso é o MSER.

O fluxograma dos testes será igual para os três métodos, como demonstrado na Figura 15. As configurações utilizadas em cada um dos métodos serão discutidas nas próximas subseções.

##### 4.4.1 HOG

Na configuração do HOG, o algoritmo permite algumas customizações que podem influenciar na performance computacional e desempenho nos testes. Após extensivos ensaios, definiu-se uma configuração ideal. Os valores serão descritos abaixo:

- CellSize: definido como 3x3 (já que o tamanho das imagens é de 50x50 *pixels*, um valor pequeno de célula tornou o vetor de resultados com 8100 dimensões);
- BlockSize: definido como 2x2 (valor para normalização, como esclarecido em 3.1.2)
- BlockOverlap: Definido como BlockSize/2 (valor para normalização, como esclarecido em 3.1.2);
- NumBins: definido como 9 (divisões do histograma);
- UseSignedOrientation: Definido como false.

Como resultado, cada vetor de resultados do HOG tem 8100 dimensões.

#### 4.4.2 LESH

O método LESH possui algumas configurações relacionadas ao *Gabor Filter*, tamanho das partições e similaridade. Após alguns testes, chegou-se a uma configuração desejável.

- Scale: definido como 5 (configuração do *gabor filter*);
- Orient: definido como 8 (configuração do *gabor filter*);
- Partition: definido como 16 (sendo assim, o vetor de saída tem 2048 dimensões);
- Similaridade: definido como 0 (ou seja, *jeffrey divergence*).

Com estes ajustes, o vetor de resultados do LESH tem 2048 dimensões.

#### 4.4.3 SFTA

No SFTA, a única configuração era o valor de “nt”, que define o número de *thresholds*. Neste caso, após variados testes o valor ótimo encontrado é 20. Segundo Costa, Humpire-Mamani e Traina (2012), o vetor de resultados terá o tamanho de  $V_{dim} = 6 \times nt - 3$ , ou seja, teremos 117 dimensões no vetor de resultados do SFTA.

#### 4.4.4 MSER

O MSER foi o identificador de regiões escolhido para o segundo teste. Entretanto, sua configuração altera, em demasia, o resultado final. Como visto em 3.1.5, o MSER pode ser computacionalmente inviável caso retorne muitas regiões na imagem. Levando isso em conta, optou-se por uma configuração mais conservadora, fazendo com que somente regiões realmente estáveis na imagem fossem detectadas. Contudo, tal configuração que priorizou o custo computacional, acabou por condenar o desempenho do MSER. As configurações escolhidas foram:

- ThresholdDelta: definido como 10 (número de incrementos que definem a estabilidade da região);
- RegionAreaRange: definido de 100 até 12000 (área das regiões testadas em *pixels*);

- MaxAreaVariation: definido como 0.25 (máxima variação de área para cada *threshold*);
- ROI: definido com o tamanho da imagem (denota qual parte da imagem é a região de interesse).

Com essas configurações, o MSER foi utilizado para descobrir regiões estáveis na imagem. Os resultados serão apresentados no Capítulo 5.

#### 4.5 CONFIGURAÇÃO DA MÁQUINA UTILIZADA NOS TESTES

Todos os testes foram realizados em uma máquina com Windows 10, Processador Intel i5-3340M, 8GB de memória RAM e SSD de 256GB. O MATLAB versão R2017a foi utilizado para executar os testes.





## 5 RESULTADOS

Nesta Capítulo serão apresentados os resultados dos testes descritos na Seção 4.1. No primeiro teste, inicialmente foi utilizado o método HOG. Em seguida, o método LESH foi utilizado. Por fim, o terceiro método aplicado foi o SFTA. Em cada teste inicialmente, serão apresentados os resultados do primeiro nível da árvore de classificadores SVM, que reconhece o formato das placas de sinalização de trânsito. Posteriormente, os resultados do segundo nível da árvore de classificadores SVM serão apresentados, sendo este, o nível que reconhece a categoria da placa de sinalização de trânsito em questão.

Todas as seguintes tabelas relacionadas ao primeiro teste terão cinco colunas. A primeira coluna identifica o formato ou a categoria da placa. A segunda coluna apresenta a quantidade de placas de sinalização de trânsito que o classificador acertou. A terceira coluna mostra a quantidade de placas de sinalização de trânsito que o classificador errou. A quarta coluna exibe o total de placas de sinalização de trânsito presentes em determinada categoria. A quinta coluna representa a taxa de acerto, em porcentagem, do classificador em relação as classificações corretas.

### 5.1 TESTE 1 - MÉTODO HOG

Inicialmente, o método HOG teve um ótimo desempenho para reconhecer os formatos. A Tabela 6 expõe os resultados.

Tabela 6: Resultado das placas por formato do método HOG.

<b>Formato</b>	<b>Acerto</b>	<b>Erro</b>	<b>Total</b>	<b>Taxa de acerto (%)</b>
Circular	57	0	57	100%
Losango	40	0	40	100%
Triangular	5	0	5	100%
<b>TOTAL</b>	<b>102</b>	<b>0</b>	<b>102</b>	<b>100%</b>

Ao observar a Tabela 6, vemos que este método teve 100% de acurácia para os formatos das placas de sinalização de trânsito.

Para os testes no segundo nível da árvore de classificadores, as Tabelas 7, 8 e 9 apresentam os resultados dos classificadores de placas

circulares, triangulares e losangulares respectivamente.

Seguindo a tendência do primeiro nível, o método HOG foi excelente em todos os classificadores do segundo nível. Apenas quatro placas foram classificadas erradas. Uma placa de 40 km/h foi classificada como sendo de 80 km/h. Duas placas de 60 km/h foram classificadas como sendo de 80 km/h. Uma placa de 80 km/h foi classificada como sendo de 60 km/h.

Tabela 7: Resultado das placas circulares do método HOG.

Placas Circulares	Acerto	Erro	Total	Taxa de acerto (%)
40 km/h	5	1	6	83%
60 km/h	4	2	6	67%
80 km/h	13	1	14	93%
Pare	8	0	8	100%
Proibido estacionar	5	0	5	100%
Proibido parar e estacionar	7	0	7	100%
Proibido retornar pela esquerda	4	0	4	100%
Proibido virar à direita	1	0	1	100%
Proibido virar à esquerda	1	0	1	100%
Siga em frente	5	0	5	100%
<b>TOTAL</b>	<b>53</b>	<b>4</b>	<b>57</b>	<b>93%</b>

Tabela 8: Resultado das placas triangulares do método HOG.

Placa Triangular	Acerto	Erro	Total	Taxa de acerto (%)
Dê preferência	5	0	5	100%
<b>TOTAL</b>	<b>5</b>	<b>0</b>	<b>5</b>	<b>100%</b>

Tabela 9: Resultado das placas losangulares do método HOG.

<b>Placas Losangulares</b>	<b>Acerto</b>	<b>Erro</b>	<b>Total</b>	<b>Taxa de acerto (%)</b>
Curva acentuada à direita	4	0	4	100%
Curva acentuada à esquerda	4	0	4	100%
Lombada	15	0	15	100%
Obras à frente	2	0	2	100%
Faixa de pedestres	11	0	11	100%
Semáforo à frente	4	0	4	100%
<b>TOTAL</b>	<b>40</b>	<b>0</b>	<b>40</b>	<b>100%</b>

Tabela 10: Desempenho geral do método HOG.

<b>Desempenho geral</b>	<b>Acerto</b>	<b>Erro</b>	<b>Total</b>	<b>Taxa de acerto (%)</b>
<b>TOTAL</b>	<b>98</b>	<b>4</b>	<b>102</b>	<b>96,08%</b>

Analisando a Tabela 8, vemos que o método HOG foi excelente para separar as placas de Dê a Preferência das demais. Examinando a Tabela 10, vemos o comportamento da árvore de classificadores. Neste caso, o desempenho geral do método HOG foi de 96.08%, que indica que o HOG é um excelente método para ser aplicado em sistemas que classificam placas brasileiras de sinalização de trânsito.

## 5.2 TESTE 1 - MÉTODO LESH

O método LESH apresentou uma boa performance. Para o primeiro nível da árvore de classificação, o método LESH errou apenas uma placa losangular, que a classificou como sendo circular. A Tabela 11 exibe os resultados deste primeiro nível.

Tabela 11: Resultado das placas por formato do método LESH.

<b>Formato</b>	<b>Acerto</b>	<b>Erro</b>	<b>Total</b>	<b>Taxa de acerto (%)</b>
Circular	57	0	57	100%
Losango	39	1	40	97,5%
Triangular	5	0	5	100%
<b>TOTAL</b>	<b>101</b>	<b>1</b>	<b>102</b>	<b>99%</b>

As Tabelas 12, 13 e 14 apresentam os resultados colhidos no segundo nível da árvore para as categorias circulares, losangulares e triangulares, respectivamente.

Na Tabela 12, duas placas de 40 km/h foram classificadas como 80 km/h. Duas placas de 60 km/h também foram classificadas como 80 km/h. Nos dois erros na categoria de 80 km/h, uma placa foi classificada como sendo de 60 km/h e outra como sendo placa de Proibido Parar e Estacionar. A placa de Proibido Virar à Direita foi categorizada como placa de Proibido Parar e Estacionar.

Tabela 12: Resultado das placas circulares do método LESH.

<b>Placas Circulares</b>	<b>Acerto</b>	<b>Erro</b>	<b>Total</b>	<b>Taxa de acerto (%)</b>
40 km/h	4	2	6	66,7%
60 km/h	4	2	6	66,7%
80 km/h	12	2	14	85,7%
Pare	8	0	8	100%
Proibido estacionar	5	0	5	100%
Proibido parar e estacionar	7	0	7	100%
Proibido retornar pela esquerda	3	1	4	75%
Proibido virar à direita	0	1	1	0%
Proibido virar à esquerda	1	0	1	100%
Siga em frente	5	0	5	100%
<b>TOTAL</b>	<b>49</b>	<b>8</b>	<b>57</b>	<b>85,9%</b>

Tabela 13: Resultado das placas losangulares do método LESH.

<b>Placas Losangulares</b>	<b>Acerto</b>	<b>Erro</b>	<b>Total</b>	<b>Taxa de acerto (%)</b>
Curva acentuada à direita	4	0	4	100%
Curva acentuada à esquerda	4	0	4	100%
Lombada	14	1	15	93,3%
Obras à frente	2	0	2	100%
Faixa de pedestres	11	0	11	100%
Semáforo à frente	4	0	4	100%
<b>TOTAL</b>	<b>39</b>	<b>1</b>	<b>40</b>	<b>97,5%</b>

Já o desempenho do SVM que classifica as placas losangulares foi muito superior do que o SVM que classifica as placas circulares, visto que esses dois SVMs são os que mais tem classes. O único erro de classificação foi uma placa de Lombada que foi categorizada como placa de Pare.

Tabela 14: Resultado das placas triangulares do método LESH.

<b>Placa Triangular</b>	<b>Acerto</b>	<b>Erro</b>	<b>Total</b>	<b>Taxa de acerto (%)</b>
Dê preferência	5	0	5	100%
<b>TOTAL</b>	<b>5</b>	<b>0</b>	<b>5</b>	<b>100%</b>

No SVM que classifica a placa de Dê a Preferência, vê-se que o método LESH teve uma ótima acurácia.

Tabela 15: Desempenho geral do método LESH.

<b>Desempenho geral</b>	<b>Acerto</b>	<b>Erro</b>	<b>Total</b>	<b>Taxa de acerto (%)</b>
<b>TOTAL</b>	<b>93</b>	<b>9</b>	<b>102</b>	<b>91,2%</b>

O desempenho geral do método LESH está na Tabela 15. Apesar de classificar nove placas erroneamente, o método teve uma performance aceitável, com taxa de acerto de 91.2%.

### 5.3 TESTE 1 - MÉTODO SFTA

O método SFTA não teve uma boa performance praticamente em nenhum dos testes. A Tabela 16 começa demonstrando os resultados para o primeiro nível da árvore de classificadores, que categoriza pelo formato das placas.

Tabela 16: Resultado das placas por formato do método SFTA.

<b>Formato</b>	<b>Acerto</b>	<b>Erro</b>	<b>Total</b>	<b>Taxa de acerto (%)</b>
Circular	47	10	57	82.4%
Losango	34	6	40	85%
Triangular	5	0	5	100%
<b>TOTAL</b>	<b>86</b>	<b>16</b>	<b>102</b>	<b>84.3%</b>

Ao analisar a Tabela 16, vemos que este método ocasionou o erro de classificação de dez placas que eram para serem categorizadas como circulares, mas foram categorizadas como losangulares. Também observa-se que seis placas foram classificadas como losangulares mas eram circulares.

Seguindo os testes, os resultados do segundo nível da árvore de classificadores estão presentes nas Tabelas 17, 18 e 19, que correspondem aos resultados dos classificadores das placas circulares, losangulares e triangulares respectivamente.

De acordo com a Tabela 17, o método SFTA teve uma performance para as placas circulares muito abaixo do esperado. Das seis placas de 40 km/h, quatro foram categorizadas erroneamente, sendo uma de 60 km/h, duas como placa de Proibido Estacionar e uma como placa de Lombada. Duas placas de 60 km/h foram classificadas como sendo placa de 40 km/h, e uma de 60 km/h foi classificada como sendo placa de Lombada. Oito placas de 80 km/h foram classificadas erradas da seguinte maneira: uma placa de 40 km/h, três placas de 60 km/h, uma placa de Pare, uma placa de Proibido Parar e Estacionar, uma placa de Lombada, e uma placa de Curva Acentuada à Direita. Uma placa de Pare foi categorizada como placa de Faixa de Pedestres. Duas placas de Proibido Estacionar foram classificadas como placa de Faixa de Pedestres. Uma placa de Proibido Parar e Estacionar foi classificada como placa de Lombada. Uma placa de Proibido Retornar pela Esquerda foi categorizada como Proibido Parar e Estacionar. A placa de Proibido Virar à Direita foi classificada como Curva Acentuada à

Direita. Já a placa de Proibido Virar à Esquerda foi categorizada como Proibido Estacionar. Uma placa de Siga em Frente foi categorizada como placa de Lombada. Como resultado final, das 57 placas circulares, apenas 34 foram corretamente classificadas.

Tabela 17: Resultado das placas circulares do método SFTA.

Placas Circulares	Acerto	Erro	Total	Taxa de acerto (%)
40 km/h	2	4	6	33,3%
60 km/h	3	3	6	50%
80 km/h	6	8	14	42,9%
Pare	7	1	8	87,5%
Proibido estacionar	3	2	5	60%
Proibido parar e estacionar	6	1	7	85,7%
Proibido retornar pela esquerda	3	1	4	75%
Proibido virar à direita	0	1	1	0%
Proibido virar à esquerda	0	1	1	0%
Siga em frente	4	1	5	80%
<b>TOTAL</b>	<b>34</b>	<b>23</b>	<b>57</b>	<b>59,6%</b>

Tabela 18: Resultado das placas losangulares do método SFTA.

Placas Losangulares	Acerto	Erro	Total	Taxa de acerto (%)
Curva acentuada à direita	2	2	4	50%
Curva acentuada à esquerda	0	4	4	0%
Lombada	13	2	15	86,7%
Obras à frente	1	1	2	50%
Faixa de pedestres	6	5	11	54,5%
Semáforo à frente	2	2	4	50%
<b>TOTAL</b>	<b>24</b>	<b>16</b>	<b>40</b>	<b>60%</b>

Apesar de ligeiramente melhor do que as placas circulares, as placas Losangulares também tiveram uma baixa acurácia. Na Tabela 13, duas placas de Curva Acentuada à Direita foram classificadas como sendo uma placa de 60 km/h e outra como Curva Acentuada à Esquerda. Três placas de Curvas Acentuadas à Esquerda foram classificadas como Curva Acentuada à Direita e uma como sendo a placa de 60 km/h. Duas placas de lombadas foram categorizadas erradas, uma como placa de Semáforo e outra como placa de Siga em Frente. Uma placa de Obras à Frente foi classificada como placa de Lombada. Quatro placas de Faixa de Pedestres foram classificadas como placa de lombada, e uma como placa de 80 km/h. Uma placa de Semáforo foi categorizada como placa de Siga em Frente e outra como sendo placa de Proibido Parar e Estacionar.

Tabela 19: Resultado das placas triangulares do método SFTA.

Placa Triangular	Acerto	Erro	Total	Taxa de acerto (%)
Dê preferência	5	0	5	100%
<b>TOTAL</b>	<b>5</b>	<b>0</b>	<b>5</b>	<b>100%</b>

Já para as placas triangulares, o método conseguiu distingui-las bem, como demonstrado na Tabela 14.

O método SFTA teve um desempenho geral muito pobre, demonstrado na Tabela 15. Das 102 placas do banco de dados de testes, apenas 63 foram corretamente classificadas. A taxa de acerto deste método foi de 61.8%.

Tabela 20: Desempenho geral do método SFTA.

Desempenho geral	Acerto	Erro	Total	Taxa de acerto (%)
<b>TOTAL</b>	<b>63</b>	<b>39</b>	<b>102</b>	<b>61,8%</b>

## 5.4 TESTE 2

Neste segundo teste, usou-se o método HOG como extrator de características. Isso deveu-se pelo fato de que o HOG teve uma performance muito superior aos outros dois métodos testados.

Seguindo o diagrama de fluxo presente na Figura 16, o método HOG extraiu as características de cada região de cada placa presente



no banco de dados II, enviando cada resultado para o classificador. Com essa abordagem, obteve-se os resultados presentes na Tabela 21. Ao analisar a referida tabela, identificamos uma baixíssima acurácia do MSER.

Apesar da baixa acurácia, o classificador identificou todas as placas que existiam nas regiões providas pelo MSER, ou seja, o MSER provavelmente não é o método mais aconselhável para identificar possíveis regiões com placas de sinalização de trânsito. Esta baixa precisão também pode ter sido causada pelas imagens, que provavelmente não tinham qualidade suficiente para o MSER.

Tabela 21: Resultado das placas de sinalização de trânsito presentes no teste II.

Classes de Placas	Acertos	Erros	Total	Taxa de acerto (%)
40 km/h	1	4	5	20%
60 km/h	0	2	2	0%
80 km/h	2	3	5	40%
Pare	5	1	6	83,3%
Proibido estacionar	0	1	1	0%
Proibido parar e estacionar	1	0	1	100%
Proibido retornar pela esquerda	1	2	3	33,3%
Proibido virar à direita	1	5	6	16,7%
Proibido virar à esquerda	2	3	5	40%
Siga em frente	1	3	4	25%
Curva acentuada à direita	0	7	7	0%
Curva acentuada à esquerda	0	7	7	0%
Lombada	1	2	3	33,3%
Faixa de pedestres	1	5	6	16,7
Semáforo à frente	0	2	2	0%
Dê a Preferência	7	7	0	100%
<b>TOTAL</b>	<b>23</b>	<b>47</b>	<b>70</b>	<b>32,85%</b>

## 5.5 COMPARATIVO ENTRE OS MÉTODOS E TESTES

Em relação ao teste I, observamos que o HOG teve uma acurácia muito superior aos outros dois métodos testados. Entretanto, o método LESH também obteve uma performance muito boa. Porém, quando comparado aos dois primeiros métodos, os testes relacionados ao SFTA invalidam o uso do mesmo.

Cada resultado, de acordo com o classificador definido na arquitetura, será demonstrado a seguir. A Tabela 22 relaciona o comparativo de todos os resultados para os três métodos.

Visivelmente, para o primeiro classificador, o HOG e o LESH estão excelentes. No entanto, já é constatado o baixo desempenho do SFTA nesta aplicação.

Tabela 22: Comparativo entre os métodos separados por finalidade do classificador.

Classificador	Método	Acerto	Erro	Total	Taxa
					de Acerto (%)
SVM 1	<i>HOG</i>	102	0	102	100%
	<i>LESH</i>	101	1	102	99%
	<i>SFTA</i>	86	16	102	84,3%
SVM 2	<i>HOG</i>	53	4	57	93%
	<i>LESH</i>	49	8	57	85,9%
	<i>SFTA</i>	34	23	57	56,6%
SVM 3	<i>HOG</i>	40	0	40	100%
	<i>LESH</i>	39	1	40	97,5%
	<i>SFTA</i>	24	16	40	60%
SVM 4	<i>HOG</i>	5	0	5	100%
	<i>LESH</i>	5	0	5	100%
	<i>SFTA</i>	5	0	5	100%
Performance Geral	<i>HOG</i>	98	4	102	96,08%
	<i>LESH</i>	93	9	102	91,2%
	<i>SFTA</i>	63	39	102	61,8%

Nos testes com o SVM 2, todos os métodos resultaram em alguns erros. No entanto, o HOG obteve a melhor eficiência. Novamente, o SFTA possuiu a pior performance entre os métodos testados.

De acordo com a Tabela 22, constata-se ainda à superioridade

do HOG perante os outros métodos apresentados para o classificador SVM 3. Outra vez, o método LESH não ficou muito pior que o HOG. No entanto, o SFTA obteve resultados ínfimos.

Ao testar os métodos no classificador SVM 4, que identifica as placas de regulamentação triangulares, todos os métodos foram bem sucedidos e com 100% de acurácia. Este fenômeno pode ser explicado quando observamos o banco de dados desta categoria. A placa triangular é a única presente neste grupo, distinguindo-se muito das outras placas. Contudo, as placas circulares possuem muitas semelhanças entre si, até mesmo para o método HOG, sendo estas as únicas classes classificadas erradas pelo mesmo.

Como performance geral, é significativo que o desempenho do HOG foi distinto dos outros métodos. Com uma performance de 96,08%, o HOG obteve o primeiro lugar dentre os métodos escolhidos. De fato, o LESH não obteve um resultado ruim, pois 91,2% é um resultado razoável. Mas, quando olhamos para o SFTA percebemos uma performance com 61,8% de precisão. Na prática, podemos descartar o uso do SFTA nesta aplicação, quando esta é utilizada em conjunto com o SVM.

Para o teste II, ressalta-se que foi empregado o uso do HOG como método que extraiu as características. Ao reparar a Tabela 21, percebemos uma acurácia de 32,85% no teste final. Um resultado péssimo, principalmente quando comparamos com o trabalho de Greenhalgh e Mirmehdi (2012). Apesar dos bancos de dados distintos utilizados entre o trabalho de Greenhalgh e Mirmehdi (2012) e o presente trabalho, podemos apontar a baixa performance resultante do banco de dados. Visto que as imagens para a segunda base de dados são do *Google Street View*, e estas podem não estar com qualidade suficiente para este sistema. Apesar da boa resolução, elas provavelmente possuíam alguns ruídos. Isto explicaria a baixa performance em detectar regiões estáveis, que poderiam conter prováveis placas de sinalização de trânsito. Como comentado na seção 3.1.5, quando as regiões de alta frequência não são bem distintas do restante, o MSER não consegue identificar regiões estáveis. Importante ressaltar que em todas as regiões que possivelmente possuíam placas de sinalização de trânsito, e que foram enviadas à árvore de classificadores HOG, foram classificadas corretamente. Sendo assim, o HOG obteve 100% de acurácia nas placas que foram identificadas.



## 6 CONSIDERAÇÕES FINAIS

Neste Capítulo, será dada as conclusões sobre os resultados coletados no Capítulo 5. Em seguida, é abordada as propostas de trabalhos futuros.

### 6.1 CONCLUSÃO

O presente trabalho comparou métodos descritores de imagens e um identificador de regiões de interesse. Dois tipos de experimentos foram realizados. O primeiro comparou o desempenho dos métodos descritores de imagens. O Segundo experimento mensurou a acurácia do sistema quando o descritor de imagens é utilizado em conjunto com o método que identifica regiões de interesse.

Inicialmente, foi observado o predomínio do método HOG, perante os outros dois métodos. Isto deveu-se pelo fato de que o HOG consegue extrair informações de orientação dos gradientes locais, ou seja, cada parte da imagem contribui de forma igual. Isto faz com que pequenas parcelas em uma imagem sejam comparadas com as mesmas pequenas parcelas em outra imagem.

Os resultados apresentados pelo método LESH foram positivos. Como comentado na Seção 3.1.3, o fato do método em questão calcular a energia local e ela realçar os contornos e bordas auxiliou nos resultados. Outra razão dos resultados positivos pode ser o fato de que o LESH é calculado a partir de frequências presentes na imagem. Isto indica que placas brasileiras de sinalização de trânsito semelhantes possuem frequências próximas. Isto é bem observado em placas que possuíam muitas similaridades, como as de velocidade máxima permitida.

Ao analisar o desempenho obtido com o uso do SFTA, observamos uma baixa acurácia. O motivo disto pode estar relacionado ao fato de as imagens utilizadas na base de dados terem um tamanho de  $50 \times 50$  *pixels*, que pode ser pequeno demais. A relação na afirmação anterior vem ao observar o principal funcionamento do SFTA, descrito pela Seção 3.1.4. O fato é que ao definirmos o número de *thresholds* em uma imagem pequena, isso pode acarretar que as sub-imagens que saem do algoritmo TTBD não tenham representantes em um certo intervalo de *thresholds*, tornando algumas sub-imagens totalmente pretas e sem qualquer informação.

Ao associar o método HOG com o MSER, é observado um péssimo

desempenho. Como descrito ao fim da seção 5.5, o motivo pode estar relacionado ao banco de dados, que pode não ter qualidade o suficiente para que as regiões estáveis das imagens sejam extraídas corretamente. Outro motivo pode estar relacionado ao tamanho das imagens que, por serem muito grandes e possuírem informações demais, acabam por atrapalhar com pequenos detalhes entre as regiões estáveis e não estáveis. Entretanto, tal afirmação é impossível senão contestada com a criação de outro banco de dados de placas brasileiras de sinalização de trânsito, e a realização dos mesmos testes com as mesmas configurações.

Concluindo-se no presente trabalho, foi reconhecido positivamente o uso do método HOG, como descritor de imagem em conjunto com o SVM, para resolução do problema abordado inicialmente por este trabalho. Todavia, é possível o aperfeiçoamento do método LESH com diferentes configurações ou classificadores. No entanto, o uso do SFTA é possivelmente descartável para a resolução do problema proposto.

O uso do MSER ainda deve ser estudado com maior cautela, apesar do baixo desempenho, o trabalho descrito por Greenhalgh e Mirmehdi (2012) indica um desempenho muito superior do conjunto HOG, SVM e MSER. Tal indicação não foi observada neste trabalho.

## 6.2 TRABALHOS FUTUROS

Para futuros trabalhos poderá ser considerado aumentar as duas bases de dados. Isso vai aumentar a acurácia do sistema, visto que a maior quantidade de exemplos indica que o sistema estaria melhor preparado e generalizado para situações reais.

A comparação dos métodos aqui apresentados, testados em outros classificadores, é uma outra importante sugestão. O problema de reconhecimento em imagens é complexo, sendo então, impossível afirmar que um classificador é superior aos outros sem antes testá-los. Existem muitos classificadores que seriam bons opositores para o SVM, como por exemplo o PNN.

Apesar do HOG ter a melhor performance dentre os três métodos testado, o método LESH também teve uma precisão interessante. Portanto, é um método que merece mais atenção, por isso, é digno de mais testes com configurações e usando classificadores diferentes.

Apesar da baixa performance do MSER, ele ainda é um método amplamente utilizado na comunidade acadêmica. Seu uso nesta aplicação, segundo os testes, não é o mais indicado. Devido a isso, o estudo de alternativas ao MSER aplicados a este problema seria uma ótima

opção. Uma alternativa básica, mas que é amplamente utilizada é a segmentação da imagem por cor.

Por fim, também é uma alternativa testar os métodos utilizando a biblioteca de visão computacional *Open Source Computer Vision Library* (OpenCV). Este teste poderia indicar a viabilidade do sistema para aplicações em tempo real.





## REFERÊNCIAS

- AKANDE, A. et al. Performance comparison of svm and ann in predicting compressive strength of concrete. v. 16, p. 88–94, 09 2014.
- BENGLER, K. et al. Three decades of driver assistance systems: Review and future perspectives. **IEEE Intelligent Transportation Systems Magazine**, v. 6, n. 4, p. 6–22, 2014. ISSN 15249050.
- BRADSKI, G. The OpenCV Library. **Dr. Dobb's Journal of Software Tools**, 2000.
- CARVALHO, C. H. R. de et al. **Estimativa dos Custos dos Acidentes de Trânsito no Brasil com Base na Atualização Simplificada das Pesquisas Anteriores do Ipea**. 2015. [Online; Acessado em 15 de Maio de 2017]. Disponível em: <<https://goo.gl/pnQU5H>>.
- CASAGRANDE, L. et al. Probabilistic neural network and wavelet transform for mapping of phragmites australis using low altitude remote sensing. In: **2017 30th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)**. [S.l.: s.n.], 2017. p. 269–276.
- CONTRAN. **Manual Brasileiro de Sinalização de Trânsito - VOLUME I - Sinalização Vertical de Regulamentação**. 2007. [Online; Acessado em 1 de Junho de 2017]. Disponível em: <<https://goo.gl/4P6rNy>>.
- CONTRAN. **Manual Brasileiro de Sinalização de Trânsito - VOLUME II -Sinalização Vertical de Advertência**. 2007. [Online; Acessado em 1 de Junho de 2017]. Disponível em: <<https://goo.gl/iQEG7d>>.
- CORTES, C.; VAPNIK, V. Support-vector networks. **Mach. Learn.**, Kluwer Academic Publishers, v. 20, n. 3, set. 1995. ISSN 0885-6125. Disponível em: <<https://doi.org/10.1023/A:1022627411411>>.
- COSTA, A. F.; HUMPIRE-MAMANI, G.; TRAINA, A. J. M. An efficient algorithm for fractal analysis of textures. In: **2012 25th SIBGRAPI Conference on Graphics, Patterns and Images**. [S.l.: s.n.], 2012. p. 39–46. ISSN 1530-1834.

DALAL, N.; TRIGGS, B. Histograms of oriented gradients for human detection. In: **2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)**. [S.l.: s.n.], 2005. v. 1, p. 886–893 vol. 1. ISSN 1063-6919.

DENATRAN. **Conselho Nacional De Trânsito - Código de Trânsito Brasileiro**. 2008. [Online; Acessado em 1 de Junho de 2017]. Disponível em: <<https://goo.gl/M2KkMC>>.

DONOSER, M.; BISCHOF, H. Efficient maximally stable extremal region (mscr) tracking. In: **2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)**. [S.l.: s.n.], 2006. v. 1, p. 553–560. ISSN 1063-6919.

DUBEY, A. K.; JAFFERY, Z. A. Maximally stable extremal region marking-based railway track surface defect sensing. **IEEE Sensors Journal**, v. 16, n. 24, p. 9047–9052, Dec 2016. ISSN 1530-437X.

GONZALEZ, . et al. Automatic traffic signs and panels inspection system using computer vision. **IEEE Transactions on Intelligent Transportation Systems**, v. 12, n. 2, p. 485–499, June 2011. ISSN 1524-9050.

GONZALEZ, R. C. **Processamento Digital De Imagens 3a Edição**. Pearson, 2009. ISBN 9788576054016. Disponível em: <<https://goo.gl/wp9W48>>.

GREENHALGH, J.; MIRMEHDI, M. Real-time detection and recognition of road traffic signs. **IEEE Transactions on Intelligent Transportation Systems**, v. 13, n. 4, p. 1498–1506, 2012. ISSN 1524-9050.

GU, Q. et al. Multi-scaled license plate detection based on the label-moveable maximal mscr clique. **Optical Review**, v. 22, n. 4, p. 669–678, Aug 2015. ISSN 1349-9432. Disponível em: <<https://doi.org/10.1007/s10043-015-0103-8>>.

HOLLOWAY, E. R. R.; LONDON, U. of; DAVIES, U. **Computer and Machine Vision, Fourth Edition: Theory, Algorithms, Practicalities**. 4th. ed. Elsevier Science Publishing Co Inc, 2012. ISBN 0123869080. Disponível em: <<https://goo.gl/bLWejM>>.

HSIAO, S. F.; CHAN, J. M.; WANG, C. H. Hardware design of histograms of oriented gradients based on local binary pattern and

binarization. In: **2016 IEEE Asia Pacific Conference on Circuits and Systems (APCCAS)**. [S.l.: s.n.], 2016. p. 433–435.

HSU, C.-W.; CHANG, C.-C.; LIN, C.-J. A practical guide to support vector classification. v. 101, p. 1396–1400, 01 2003.

ISLAM, M. R. et al. Text detection and recognition using enhanced mser detection and a novel ocr technique. In: **2016 5th International Conference on Informatics, Electronics and Vision (ICIEV)**. [S.l.: s.n.], 2016. p. 15–20.

KAEHLER, A.; BRADSKI, G. **Learning OpenCV 3 Computer Vision in C++ with the OpenCV Library**. O'Reilly UK Ltd., 2017. ISBN 1491937998. Disponível em: <<https://goo.gl/AT1my9>>.

LARSSON, F.; FELSBURG, M.; FORSSÉN, P.-E. Correlating fourier descriptors of local patches for road sign recognition. **IET Computer Vision**, v. 5, n. 4, p. 244–254, July 2011. ISSN 1751-9632.

LÍDER, S. **Anuário estatístico 2014 - Seguradora Líder - Administradora do seguro DPVAT**. 2015. [Online; Acessado em 15 de Maio de 2017]. Disponível em: <<https://goo.gl/rBLcEe>>.

LÍDER, S. **Números de saques ao DPVAT no ano de 2015 - Seguradora Líder - Administradora do seguro DPVAT**. 2016. [Online; Acessado em 15 de Maio de 2017]. Disponível em: <<https://goo.gl/kW9FgF>>.

LORENA, A. et al. Uma introdução ao suporte vector machines. v. 14, 01 2007.

MALDONADO-BASCON, S. et al. Road-sign detection and recognition based on support vector machines. **IEEE Transactions on Intelligent Transportation Systems**, v. 8, n. 2, p. 264–278, June 2007. ISSN 1524-9050.

MALLICK, S. **Histogram of Oriented Gradients**. 2016. [Online; Acessado em 21 de Março de 2018]. Disponível em: <<http://www.learnopencv.com/histogram-of-oriented-gradients/>>.

MATAS, J. et al. Robust wide-baseline stereo from maximally stable extremal regions. In: CITESEER. **In British Machine Vision Conference**. [S.l.], 2002.

MATHWORKS, T. **Detect MSER Features**. 2012. [Online; Acessado em 01 de Maio de 2018]. Disponível em: <<https://www.mathworks.com/help/vision/ref/detectmserfeatures.html>>.

Robert K. McConnell. **Method of and apparatus for pattern recognition**. 1982. US4567610A.

MCCORMICK, C. **HOG Person Detector Tutorial**. 2013. [Online; Acessado em 26 de Março de 2018]. Disponível em: <<http://mccormickml.com/2013/05/09/hog-person-detector-tutorial/>>.

MORRONE, M. C.; OWENS, R. A. Feature detection from local energy. **Pattern Recogn. Lett.**, Elsevier Science Inc., New York, NY, USA, v. 6, n. 5, p. 303–313, dez. 1987. ISSN 0167-8655. Disponível em: <[http://dx.doi.org/10.1016/0167-8655\(87\)90013-4](http://dx.doi.org/10.1016/0167-8655(87)90013-4)>.

NAKAMURA, K.; OCHIAI, T.; TANIGAWA, K. Application of microprocessor to cruise-control system. **IEEE Transactions on Industrial Electronics**, IE-30, n. 2, p. 108–113, 1983. ISSN 15579948.

NISTÉR, D.; STEWÉNIUS, H. Linear time maximally stable extremal regions. In: \_\_\_\_\_. **Computer Vision – ECCV 2008: 10th European Conference on Computer Vision, Marseille, France, October 12-18, 2008, Proceedings, Part II**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008. p. 183–196. ISBN 978-3-540-88688-4. Disponível em: <[https://doi.org/10.1007/978-3-540-88688-4\\_14](https://doi.org/10.1007/978-3-540-88688-4_14)>.

OICA. **Motorization Rate 2014 Worldwide**. 2015. [Online; Acessado em 15 de Maio de 2017]. Disponível em: <<https://goo.gl/gky8C1>>.

OICA. **Employment**. 2016. [Online; Acessado em 15 de Maio de 2017]. Disponível em: <<https://goo.gl/4mA2qU>>.

OICA. **A Growth Industry**. 2016. [Online; Acessado em 15 de Maio de 2017]. Disponível em: <<https://goo.gl/vi7tY1>>.

OICA. **Production Statistics**. 2016. [Online; Acessado em 15 de Maio de 2017]. Disponível em: <<https://goo.gl/3rE7m6>>.

PEDRINI, H.; SCHWARTZ, W. **Análise de imagens digitais: princípios, algoritmos e aplicações**. CENGAGE LEARNING NACIONAL, 2008. ISBN 9788522105953. Disponível em: <<https://goo.gl/jcr4Wg>>.

PLANING, P. **Innovation Acceptance**. Springer Fachmedien Wiesbaden, 2014. 1–326 p. ISBN 9783658050054. Disponível em: <[http://www.ebook.de/de/product/23077729-/patrick\\_planing\\_innovation\\_acceptance.html](http://www.ebook.de/de/product/23077729-/patrick_planing_innovation_acceptance.html)>.

PORTAL, S. **Number of passenger cars and commercial vehicles in use worldwide from 2006 to 2014 in (1,000 units)**. 2016. [Online; Acessado em 15 de Maio de 2017]. Disponível em: <<https://goo.gl/JRvNZH>>.

QIN, F.; FANG, B.; ZHAO, H. Traffic sign segmentation and recognition in scene images. **2010 Chinese Conference on Pattern Recognition (CCPR)**, p. 1–5, 2010. Disponível em: <<https://goo.gl/JKBjrC>>.

REIF, K. **Brakes, Brake Control and Driver Assistance Systems**. Vieweg+Teubner Verlag, 2014. ISBN 3658039779. Disponível em: <<http://link.springer.com/10.1007/978-3-658-03978-3>>.

REZAEI, M.; KLETTE, R. **Computer Vision for Driver Assistance**. Springer-Verlag GmbH, 2017. ISBN 3319505491. Disponível em: <<http://link.springer.com/10.1007/978-3-319-50551-0>>.

RITTER, W. Traffic sign recognition in color image sequences. **Proceedings of the Intelligent Vehicles '92 Symposium**, p. 12–17, 1992. Disponível em: <<https://goo.gl/rhD66J>>.

SANTHI, V.; ACHARJYA, D. P.; EZHILARASAN, M. **Emerging Technologies in Intelligent Applications for Image and Video Processing**. [S.l.]: IGI Global, 2015. ISBN 1466696850.

SARFRAZ, M.; HELLWICH, O. Head pose estimation in face recognition across pose scenarios. v. 1, p. 235–242, 01 2008.

SEBANJA, I.; MEGHERBI, D. B. Automatic detection and recognition of traffic road signs for intelligent autonomous unmanned vehicles for urban surveillance and rescue. **2010 IEEE International Conference on Technologies for Homeland Security, HST 2010**, p. 132–138, 2010.

SHAOUT, A.; COLELLA, D.; AWAD, S. Advanced driver assistance systems - past, present and future. **Computer Engineering**

**Conference (ICENCO), 2011 Seventh International**, p. 72–82, 2011. Disponível em: <<https://goo.gl/ViBfn5>>.

SHIH, F. Y. **Image Processing and Pattern Recognition: Fundamentals and Techniques**. IEEE COMPUTER SOC PR, 2010. ISBN 0470404612. Disponível em: <<https://goo.gl/CvU8HH>>.

SINGH, S. Critical reasons for crashes investigated in the national motor vehicle crash causation survey. **National Highway Traffic Safety Administration**, n. February, p. 1–2, 2015. Disponível em: <<https://goo.gl/j2iKVi>>.

STALLKAMP, J. et al. The german traffic sign recognition benchmark: A multi-class classification competition. In: **The 2011 International Joint Conference on Neural Networks**. [S.l.: s.n.], 2011. p. 1453–1460. ISSN 2161-4393.

TECHNOLOGIES, D. **Delphi Integrated Radar and Camera System**. 2017.

TIMOFTE, R.; ZIMMERMANN, K.; GOOL, L. V. Multi-view traffic sign detection, recognition, and 3d localisation. **Machine Vision and Applications**, v. 25, n. 3, p. 633–647, 2014. ISSN 1432-1769. Disponível em: <<http://dx.doi.org/10.1007/s00138-011-0391-3>>.

TUYTELAARS, T.; MIKOLAJCZYK, K. **Local Invariant Feature Detectors: A Survey**. Hanover, MA, USA: Now Publishers Inc., 2008. ISBN 9781601981387. Disponível em: <<https://goo.gl/g5N9T5>>.

WAYMO. **Google Autonomous Vehicle**. 2016. [Online; Acessado em 15 de Maio de 2017]. Disponível em: <<https://waymo.com/ontheroad/>>.

WHO. Global status report on road safety 2015. **WHO Library Cataloguing-in-Publication Data Global**, p. 340, 2015. ISSN 1475-5785. Disponível em: <<https://goo.gl/ubs3zC>>.

WHO, U. N. R. S. C. Global plan for the decade of action for road safety 2011 - 2020. **Geneva: WHO**, p. 25, 2011. ISSN 1564-0604. Disponível em: <<https://goo.gl/AakFMC>>.

WINNER, H. et al. **Handbook of Driver Assistance Systems**. Springer-Verlag GmbH, 2015. 1602 p. <http://link.springer.com/10.1007/978-3-319-12352-3>. ISBN

3319123513. Disponível em: <[http://www.ebook.de/de/product-/22813222/handbook\\_of\\_driver\\_assistance\\_systems.html](http://www.ebook.de/de/product-/22813222/handbook_of_driver_assistance_systems.html)>.

YANG, Y. et al. Towards real-time traffic sign detection and classification. **IEEE Transactions on Intelligent Transportation Systems**, v. 17, n. 7, p. 2022–2031, 2016. ISSN 15249050.

YUAN, X. et al. Robust traffic sign recognition based on color global and local oriented edge magnitude patterns. **IEEE Transactions on Intelligent Transportation Systems**, v. 15, n. 4, p. 1466–1474, 2014. ISSN 15249050.

ZAKIR, U.; ZAFAR, I.; EDIRISINGHE, E. Road sign detection and recognition by using local energy based shape histogram (lesh). v. 4, p. 17, 01 2011.