UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ENGENHARIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

IGOR GUSTAVO HOELSCHER

DETECÇÃO E CLASSIFICAÇÃO DE SINALIZAÇÃO VERTICAL DE TRÂNSITO EM CENÁRIOS COMPLEXOS

Porto Alegre
2017

IGOR GUSTAVO HOELSCHER

DETECÇÃO E CLASSIFICAÇÃO DE SINALIZAÇÃO VERTICAL DE TRÂNSITO EM CENÁRIOS COMPLEXOS

Dissertação de mestrado apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Rio Grande do Sul como parte dos requisitos para a obtenção do título de Mestre em Engenharia Elétrica.
Área de concentração: Engenharia de Computação

ORIENTADOR: Prof. Dr. Altamiro Amadeu Susin

Porto Alegre
2017

IGOR GUSTAVO HOELSCHER

DETECÇÃO E CLASSIFICAÇÃO DE SINALIZAÇÃO VERTICAL DE TRÂNSITO EM CENÁRIOS COMPLEXOS

Esta dissertação foi julgada adequada para a obtenção do título de Mestre em Engenharia Elétrica e aprovada em sua forma final pelo Orientador e pela Banca Examinadora.

Orientador: _____
Prof. Dr. Altamiro Amadeu Susin, UFRGS – PPGEE
Doutor pela Universidade Federal do Rio Grande do Sul

Banca Examinadora:

Prof. Dr. Edison Pignaton de Freitas, UFRGS – PPGEE
Doutor pela Universidade Federal do Rio Grande do Sul – Porto Alegre, Brasil

Prof. Dr. Claudio Rosito Jung, UFRGS – PPGC
Doutor pela Universidade Federal do Rio Grande do Sul – Porto Alegre, Brasil

Prof. Dr. Tiago Roberto Balen, UFRGS – PG-Micro
Doutor pela Universidade Federal do Rio Grande do Sul –Porto Alegre, Brasil

Coordenador do PPGEE: _____
Prof. Dr. Valner João Brusamarello

Porto Alegre, Março de 2017.

AGRADECIMENTOS

Agradeço, primeiramente, aos meus pais Ivone e Gerson e à minha irmã Ingrid pelo suporte durante essa caminhada. Vocês são, ao mesmo tempo, porto seguro e combustível dos meus dias. Aos demais familiares pelo apoio.

Ao meu orientador e amigo Prof. Dr. Altamiro Amadeu Susin, pela acolhida, confiança e incentivo dados durante esses dois anos de trabalho. Seus ensinamentos foram um marco no meu crescimento pessoal e profissional. Apesar de alcançados nossos objetivos primários, espero poder continuar trabalhando ao seu lado em novos projetos que virão, independente dos caminhos tomados.

À banca, que contribuiu na construção desse documento, para que este transmita da melhor forma possível todo o trabalho desenvolvido durante o período de mestrado.

Aos professores, pesquisadores, doutorandos, mestrandos e graduandos do LaPSI pelas contribuições e debates nos diversos trabalhos que desenvolvemos juntos.

Àqueles que tive o prazer de conhecer durante esse período e que não só foram importantes para o desenvolvimento desse trabalho, mas também foram amigos em tantos outros momentos ímpares que tornaram Porto Alegre singular. Essa etapa da minha vida não teria sido tão especial sem vocês para compartilhar.

À CAPES pela provisão de bolsa de mestrado, permitindo total dedicação ao trabalho. Aos colegas e professores do PPGEE pelos conhecimentos e experiências. Enfim, à todos que de alguma maneira foram importantes para a realização desse trabalho.

RESUMO

A mobilidade é uma marca da nossa civilização. Tanto o transporte de carga quanto o de passageiros compartilham de uma enorme infra-estrutura de conexões operados com o apoio de um sofisticado sistema logístico. Simbiose otimizada de módulos mecânicos e elétricos, os veículos evoluem continuamente com a integração de avanços tecnológicos e são projetados para oferecer o melhor em conforto, segurança, velocidade e economia.

As regulamentações organizam o fluxo de transporte rodoviário e as suas interações, estipulando regras a fim de evitar conflitos. Mas a atividade de condução pode tornar-se estressante em diferentes condições, deixando os condutores humanos propensos a erros de julgamento e criando condições de acidente. Os esforços para reduzir acidentes de trânsito variam desde campanhas de re-educação até novas tecnologias. Esses tópicos têm atraído cada vez mais a atenção de pesquisadores e indústrias para Sistemas de Transporte Inteligentes baseados em imagens.

Este trabalho apresenta um estudo sobre técnicas de detecção e classificação de sinalização vertical de trânsito em imagens de cenários de tráfego complexos. O sistema de reconhecimento visual automático dos sinais destina-se a ser utilizado para o auxílio na atividade de direção de um condutor humano ou como informação para um veículo autônomo. Com base nas normas para sinalização viária, foram testadas duas abordagens para a segmentação de imagens e seleção de regiões de interesse. O primeiro, uma limiarização de cor em conjunto com Descritores de Fourier. Seu desempenho não foi satisfatório. No entanto, utilizando os seus princípios, desenvolveu-se um novo método de filtragem de cores baseado em Lógica Fuzzy que, juntamente com um algoritmo de seleção de regiões estáveis em diferentes tons de cinza (MSER), ganhou robustez à oclusão parcial e a diferentes condições de iluminação.

Para classificação, duas Redes Neurais Convolucionais curtas são apresentadas para reconhecer sinais de trânsito brasileiros e alemães. A proposta é ignorar cálculos complexos ou *features* selecionadas manualmente para filtrar falsos positivos antes do reconhecimento, realizando a confirmação (etapa de detecção) e a classificação simultaneamente. A utilização de métodos do estado da arte para treinamento e otimização melhoraram a eficiência da técnica de aprendizagem da máquina.

Além disso, este trabalho fornece um novo conjunto de imagens com cenários de tráfego em diferentes regiões do Brasil, contendo 2.112 imagens em resolução WSXGA+. As análises qualitativas são mostradas no conjunto de dados brasileiro e uma análise quantitativa com o conjunto de dados alemão apresentou resultados competitivos com outros métodos: 94% de acurácia na extração e 99% de acurácia na classificação.

Palavras-chave: Detecção de Sinalização de Trânsito, Classificação de Sinalização de Trânsito, Segmentação de Imagens, Rede Neural Convolucional.

ABSTRACT

Mobility is an imprint of our civilization. Both freight and passenger transport share a huge infrastructure of connecting links operated with the support of a sophisticated logistic system. As an optimized symbiosis of mechanical and electrical modules, vehicles are evolving continuously with the integration of technological advances and are engineered to offer the best in comfort, safety, speed and economy.

Regulations organize the flow of road transportation machines and help on their interactions, stipulating rules to avoid conflicts. But driving can become stressing on different conditions, leaving human drivers prone to misjudgments and creating accident conditions. Efforts to reduce traffic accidents that may cause injuries and even deaths range from re-education campaigns to new technologies. These topics have increasingly attracted the attention of researchers and industries to Image-based Intelligent Transportation Systems.

This work presents a study on techniques for detecting and classifying traffic signs in images of complex traffic scenarios. The system for automatic visual recognition of signs is intended to be used as an aid for a human driver or as input to an autonomous vehicle. Based on the regulations for road signs, two approaches for image segmentation and selection of regions of interest were tested. The first one, a color thresholding in conjunction with Fourier Descriptors. Its performance was not satisfactory. However, using its principles, a new method of color filtering using Fuzzy Logic was developed which, together with an algorithm that selects stable regions in different shades of gray (MSER), the approach gained robustness to partial occlusion and to different lighting conditions.

For classification, two short Convolutional Neural Networks are presented to recognize both Brazilian and German traffic signs. The proposal is to skip complex calculations or handmade features to filter false positives prior to recognition, making the confirmation (detection step) and the classification simultaneously. State-of-the-art methods for training and optimization improved the machine learning efficiency.

In addition, this work provides a new dataset with traffic scenarios in different regions of Brazil, containing 2,112 images in WSXGA+ resolution. Qualitative analyzes are shown in the Brazilian dataset and a quantitative analysis with the German dataset presented competitive results with other methods: 94% accuracy in extraction and 99% accuracy in the classification.

Keywords: Traffic Sign Detection, Traffic Sign Classification, Image Segmentation, Convolutional Neural Network.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURAS

ADAS	*Advanced Driver Assistance System*
ANN	*Approximate Near Neighbor*
ARTMAP	*Adaptive Resonance Theory—supervised predictive MAPping*
AUC	*Area Under the Curve*
BelgiumTS	*Belgian Traffic Sign Dataset*
BRCNN	*Brazilian Convolutional Neural Network*
BRTSD	*Brazilian Traffic Sign Dataset*
CIE	*Commission Internationale d'Éclairage*
CMYK	*Cyan – Magenta – Yellow – Key(Black)* (Espaço de Cores)
CNN	*Convolutional Neural Network*
CTSD	*Chinese Traffic Sign Dataset*
DARPA	*Defense Advanced Research Projects Agency*
DDT	*Dynamic Driving Task*
DFT	*Discrete Fourier Transform*
DNIT	Departamento Nacional de Infraestrutura de Transportes
FC	Fuzzificação da Cromaticidade
FD	*Fourier Descriptors*
FP	Função de Pertinência
FPGA	*Field-Programmable Gate Array*
GPS	*Global Position System*
GTSCNN	*German Traffic Sign Convolution Neural Network*
GTSDB	*German Traffic Sign Detection Benchmark*
GTSRB	*German Traffic Sign Recognition Benchmark*
HOG	*Histogram of Oriented Gradients*
HSB	*Hue – Saturation – Brightness* (Espaço de Cores)
HSI	*Hue – Saturation – Intensity* (Espaço de Cores)

HSL	*Hue – Saturation – Lightness* (Espaço de Cores)
HSV	*Hue – Saturation – Value* (Espaço de Cores)
IJCNN	*International Joint Conference on Neural Networks*
ILSVRC	*ImageNet Large Scale Visual Recognition Challenge*
IoT	*Internet of Things*
IP	*Intellectual Property*
ITS	*Intelligent Transportation System*
LDA	*Linear Discriminant Analysis*
LIDAR	*Light Detection And Ranging*
LISA	*Laboratory for Intelligent & Safe Automobiles*
LUT	*Look-Up Table*
MIT	*Massachusetts Institute of Technology*
MLP	*Multilayer Perceptron*
MSER	*Maximally Stable Extremal Region*
NHTSA	*National Highway Traffic Safety Administration*
ODD	*Operational Design Domain*
OEDR	*Object and Event Detection and Response*
OMS	Organização Mundial da Saúde
ONU	Organização das Nações Unidas
OpenCV	*Open Source Computer Vision Library*
PRF	Polícia Rodoviária Federal
RGB	*Red – Green – Blue* (Espaço de Cores)
ROI	*Region of Interest*
SAE	*Society of Automotive Engineers*
SGD	*Stochastic Gradient Descent*
SIFT	*Scale-Invariant Feature Transform*
SoC	*System-on-Chip*
SOM	*Self-Organizing Map*
SURF	*Speeded Up Robust Features*
SVM	*Support Vector Machine*
TSDR	*Traffic Sign Detection and Recognition*
UNECE	*United Nations Economic Commission for Europe*
WSXGA+	*Widescreen Super Extended Graphics Array Plus*

YCbCr *Luma – Blue-Difference Chroma – Red-Difference Chroma* (Espaço de Cores)

YUV *Luma – Chroma Component – Chroma Component*

SUMÁRIO

1 INTRODUÇÃO

A denominada Sociedade da Informação e do Conhecimento desponta em direção a conquistas que há 100 anos povoavam, talvez, a ideia de alguns poucos escritores de ficção e "cientistas malucos". Relevantes em diferentes contextos da sociedade, os avanços tecnológicos das últimas décadas trouxeram conforto e qualidade de vida para a sociedade. Motor das revoluções industriais, a tecnologia permitiu a substituição da força humana e animal pela máquina, aumentou a produtividade dos campos, melhorou a saúde com a higiene e os medicamentos, etc. Movendo-se cada vez mais rápido, a capacidade de criar saiu da mecânica para a eletricidade, levando à eletrônica e, por último, à Era Digital.

A flexibilidade dos sistemas eletrônicos digitais permitiu a simbiose das atividades de instrumentação, controle e processamento de informações com praticamente todas as áreas do conhecimento ligadas à tecnologia, trazendo consigo uma explosão de inovações. Evidente, entretanto, que rápidas mudanças e os benefícios que se assomam com isso acompanham encargos, tornando relevante a aquisição de novas capacitações e conhecimentos, o que significa intensificar a capacidade de indivíduos, empresas e países de se adaptar.

Entre os principais setores que estão sendo revolucionados no mundo moderno, o transporte será foco nesse trabalho. Não resta dúvida que o deslocamento por meios rodoviários é um dos eixos que movem a economia na maioria das regiões do mundo. O setor rodoviário brasileiro é especialmente importante pela grande participação que detém no transporte de cargas. Automóveis de transporte individual, veículos de transporte coletivo e veículos de transporte de cargas tem evoluído em aspectos de conforto, acessibilidade, design e segurança. Porém, muitos motoristas são colocados em situações de estresse durante atividades de direção prolongada, gerando falhas de tomada de decisão, desatenção e provocando diversos acidentes.

A Década da Ação pela Segurança no Trânsito[1] exige novos esforços da comunidade para tornar os sistemas de transporte mais seguros e reduzir as estatísticas. Segundo relatório da Organização Mundial da Saúde (2015), 1,25 milhão de pessoas morrem por ano no trânsito, sendo 49% pedestres, ciclistas ou motociclistas, número que se mantém estável desde o ano de 2007. Mesmo considerando essa constância um avanço, a OMS julga necessário maior empenho para que os números alcancem as metas previstas pela Agenda para o Desenvolvimento Sustentável, que prevê uma redução de 50% das mortes e traumatismos causados pelo trânsito até 2020.

Apesar das prospecções e da estabilidade nos números globais, as estatísticas são negativas em países pobres ou emergentes: 85% dos países de baixa renda e 46% dos países

[1] Campanha criada pela Organização das Nações Unidas para estimular medidas de prevenção de acidentes de trânsito (UNITED NATIONS, 2016).

de média renda registraram aumento no número de mortes em acidentes de trânsito no período entre 2010 e 2013. Em conjunto, os países das duas faixas econômicas concentram cerca de 90% das fatalidades, mesmo detendo apenas 54% dos veículos rodando no mundo atualmente. Considerado um dos principais países emergentes da atualidade, o Brasil é o único entre os dez mais populosos do mundo a cumprir o plano de boas práticas legislativas em quatro dos cinco principais fatores de risco no trânsito: uso de cinto de segurança, capacete, limite de velocidade, segurança para crianças e proibição de ingestão de bebida alcoólica antes de dirigir. Mesmo assim, é o país que mais registrou mortes per capita no trânsito na América do Sul (23,4 para cada 100.000 habitantes em 2013 (ORGANIZAÇÃO MUNDIAL DA SAÚDE, 2015)).

Com uma malha viária de 1.580.964 km (CENTRAL INTELLIGENCE AGENCY, 2016), as vias terrestres brasileiras ainda são o principal meio de transporte de pessoas e cargas. Entretanto, em estudo realizado pela Confederação Nacional do Transporte (2016), 58,2% da extensão rodoviária percorrida se encontra em estado classificado como regular, ruim ou péssimo . Quando o quesito é sinalização, a pesquisa mostra que 51,2% das rodovias se encontram em situação regular, ruim ou péssima.

O Departamento Nacional de Infraestrutura de Transportes (DNIT) destaca três abordagens para que o país alcance as metas de redução no número de acidentes. Uma delas é "a engenharia, no sentido de, por um lado, prover o sistema viário de elementos tais que possibilitem a movimentação de veículos e pessoas com fluidez, conforto e segurança, e, por outro, aprimorar a segurança e desempenho dos veículos automotores"(DEPARTAMENTO NACIONAL DE INFRAESTRUTURA DE TRANSPORTES, 2016).

Nessa direção, pesquisadores e líderes do mundo todo se aliam ao rápido avanço da tecnologia para desenvolver e discutir novos métodos e processos que venham aprimorar tanto a infraestrutura das estradas quanto dos veículos. Com aplicações em diversos sistemas – tanto comercializáveis, quanto conceituais – as áreas de processamento de imagens e visão computacional são as mais movimentadas atualmente, liderando o desenvolvimento dos principais Sistemas Inteligentes de Transporte (ITS – do inglês *Intelligent Transport System*).

O avanço da computação e o advento de câmeras cada vez mais eficientes, enquanto seu preço e tamanho diminuem, permite que a lista de aplicações continue crescendo, incluindo desde sistemas de prevenção à colisões, detecção de pista e saída da pista, controle de distância adaptativo, até técnicas de reconhecimento semântico do ambiente no entorno do veículo.

Acompanhando o crescente surgimento desses sistemas, comissões legislativas de vários países do mundo começaram a discutir novas adaptações em suas leis, para a inclusão de tecnologias assistivas em veículos ou, até mesmo, para veículos autônomos circularem nas rodovias federais. Quarenta e seis anos após a Convenção de Viena – que havia definido que os motoristas devem estar em controle do veículo o tempo todo – a Comissão Europeia de Economia das Nações Unidas (UNECE) decidiu preceituar que as tecnologias de condução automatizada de veículos serão permitidas no tráfego, desde que estejam em conformidade com os regulamentos da instituição e possam ser desligadas pelo condutor (ECONOMIC COMISSION FOR EUROPE, 2014).

Nos Estados Unidos da América, a NHTSA (*National Highway Traffic Safety Administration*[2]) passou a adotar o sistema de classificação de veículos autônomos criado pela

[2]Agência do governo americano pertencente ao Departamento dos Transportes que gerencia questões de segurança no trânsito

Sociedade de Engenheiros da Mobilidade (SAE – do inglês *Society of Automotive Engineers*) publicada em 2014 através do padrão J3016 e revisada em setembro de 2016 (SAE INTERNATIONAL, 2016). A classificação é baseada no grau de intervenção e níveis de atenção que ambos sistema e motorista devem empregar nas atividades de condução dinâmica (DDT – do inglês *Dynamic Driving Task*), incluindo subtarefas como a detecção, reconhecimento, classificação e resposta à objetos e eventos (OEDR – do inglês *Object and Event Detection and Response*), além das restrições de domínio de operação (ODD – do inglês *Operational Design Domain*) para o qual os sistemas foram desenvolvidos (clima, tipo de estrada, etc.), como mostra a Tabela 1.

Tabela 1: Níveis de Automação Veicular.

Nível	**Nome**	**Definição**	**DDT**		**DDT reserva**	**ODD**
			Controle Sustentado dos Movimentos Lateral e Longitudinal do Veículo	**OEDR**		
Motorista responsável parcial ou totalmente pela condução do veículo						
0	Sem Automação	Mesmo que com o auxílio de sistemas de segurança, o motorista é responsável por toda a atividade de condução.	Motorista	Motorista	Motorista	n/a
1	Apoio ao Motorista	Controle de sustentação lateral **ou** longitudinal (em ODD específico) realizado pelo sistema, com supervisão do motorista, que realiza o restante das operações de DDT.	Motorista e Sistema	Motorista	Motorista	Limitado
2	Automação Parcial de Direção	Ambos controle de sustentação lateral **e** longitudinal (em ODD específico) realizado pelo sistema, com supervisão do motorista, que realiza o restante as operações de OEDR.	Sistema	Motorista	Motorista	Limitado
Sistema responsável por todas as atividades de condução do veículo (enquanto ativo).						
3	Automação Condicional de Direção	Todas as operações de DDT realizadas pelo sistema em ODD específico, enquanto um humano supervisiona a condução e se mantém atento para assumir o controle quando requisitado.	Sistema	Sistema	Motorista	Limitado
4	Direção Altamente Automatizada	Todas as operações de DDT são realizadas pelo sistema em ODD específico, que se mantém responsável em casos emergenciais, sem esperar a intervenção humana.	Sistema	Sistema	Sistema	Limitado
5	Direção Completamente Automatizada	Operações de DDT realizadas pelo sistema sem restrições, se mantendo responsável em casos emergenciais, sem esperar a intervenção humana.	Sistema	Sistema	Sistema	Ilimitado

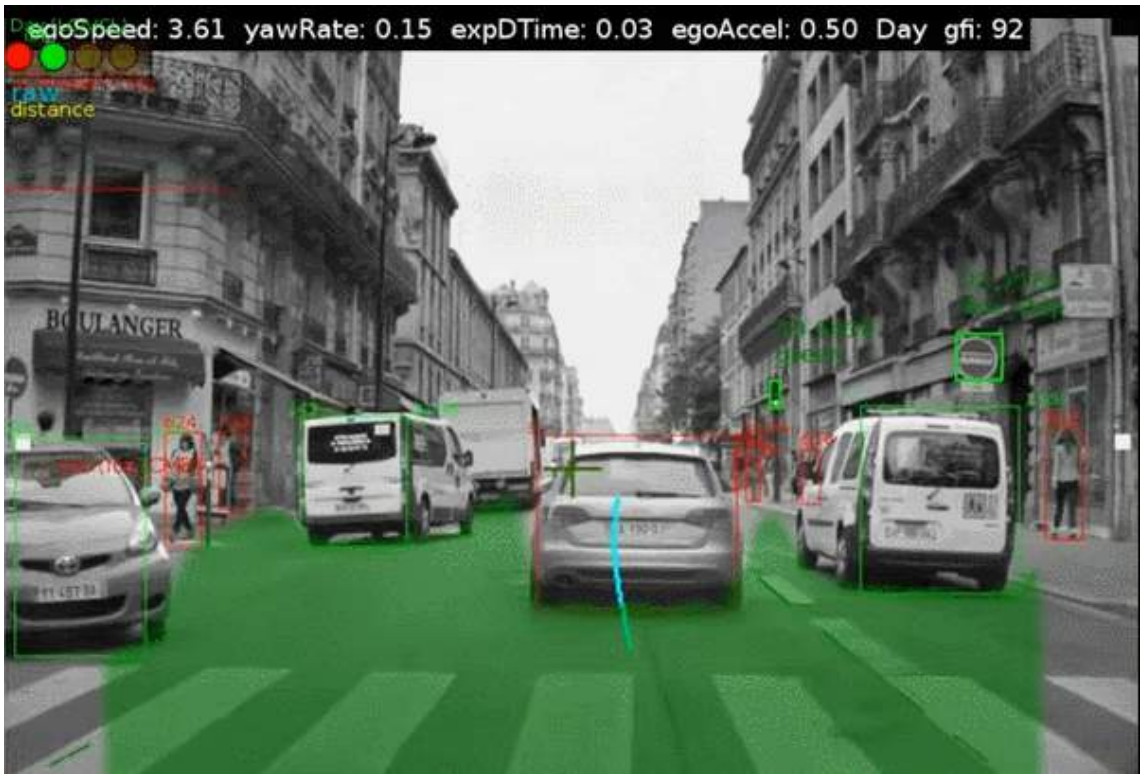
Fonte: SAE INTERNATIONAL (2016)

1.1 Motivação

O comportamento humano ainda é a variável de maior influência nas ocorrências de trânsito. Em relatório anual publicado em 2015, a Polícia Rodoviária Federal apontou as principais causas de acidentes no Brasil: falta de atenção (32% dos casos), velocidade incompatível (20%) e ultrapassagens indevidas (12%). Em relatório, a PRF (2014) indica que "a colisão traseira é o tipo de acidente que mais acontece. É causada principalmente pela falta de atenção, por não se guardar distância de segurança e por se manter uma velocidade incompatível. Entretanto, o tipo de acidente que mais mata é a colisão frontal, causada, especialmente, pelas ultrapassagens forçadas ou em locais sem visibilidade."

Na maioria dos casos, a adição de um sistema capaz de identificar e repassar ao motorista informações relevantes – como sinalização e obstáculos (Figura 1) – no entorno do veículo pode se tornar um fator de prevenção eficiente. Por isso, métodos de reconhecimento de padrões, extração de características, aprendizagem de máquinas, rastreamento de objetos, mapeamento e localização 3D, entre outros, têm sido foco de estudo para o desenvolvimento de soluções para Sistemas Avançados de Apoio ao Motorista (ADAS – do inglês *Advanced Driver Assistance Systems*).

Figura 1: Informações relevantes em um cenário de trânsito.



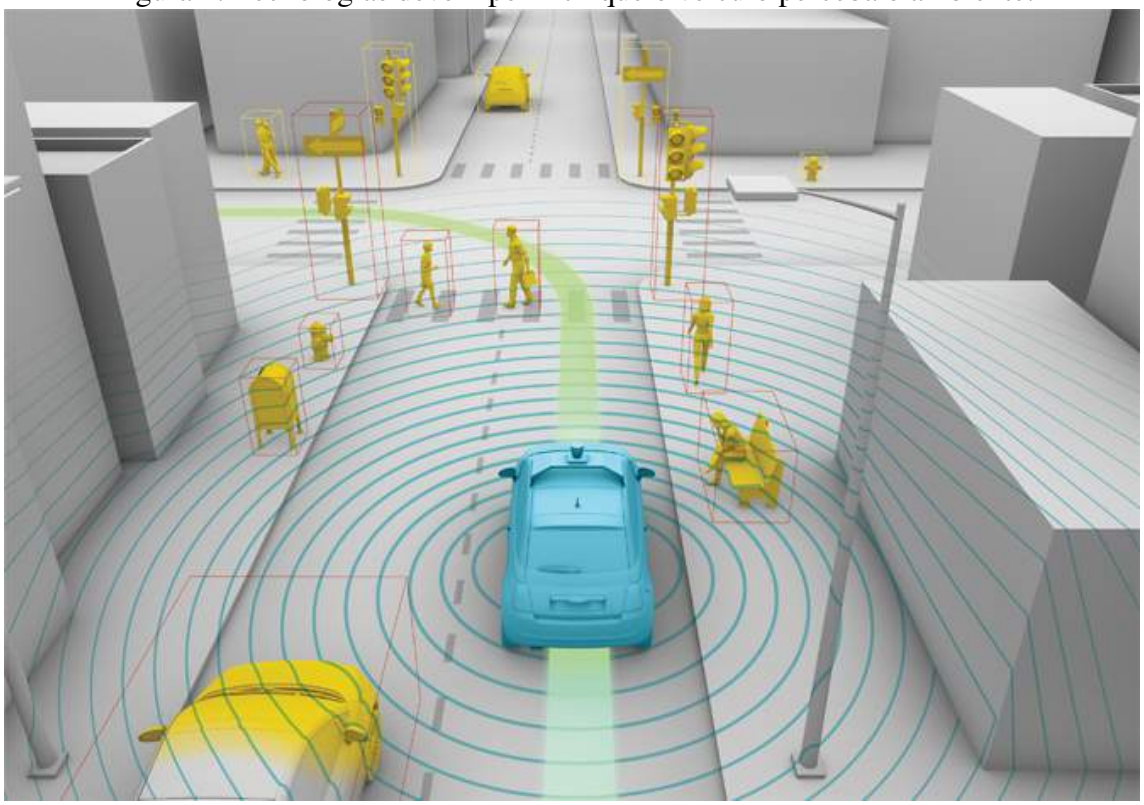
Fonte: MOBILEYE (2015)

Considerado o principal elemento para a segurança do tráfego, a capacidade do motorista de identificar a sinalização de trânsito pode ser afetada se a atividade for realizada por muito tempo ou haja perda de foco. Além disso, os motoristas estão geralmente sobrecarregados com informação e tendem a direcionar sua atenção para elementos de orientação quando estão trafegando em uma região desconhecida, podendo ignorar unidades de regulamentação.

Acredita-se, portanto, que a indústria automobilística passe a dividir suas obsessões

por potência e torque com sensores, câmeras e microprocessadores mais eficientes. Além de um meio de transporte, veículos devem assumir um papel maior na segurança dos seus usuários, tornando-se uma máquina capaz de sentir e ver além do que o motorista seria capaz (Figura 2), comunicar-se com outros veículos e planejar automaticamente um trânsito mais organizado e seguro no seu entorno.

Figura 2: Tecnologias devem permitir que o veículo perceba o ambiente.



Fonte: THE GREATER GOOD (2012).

Por isso, diferentes setores da indústria (Internet, Semicondutores, Software e Comunicações) estão tomando as rédeas de uma nova era veicular. Mais do que uma preocupação com a eficiência do motor, essas empresas procuram o radar perfeito, a câmera com a melhor resolução e profundidade, os protocolos de comunicação mais seguros e os algoritmos mais eficazes. O fato é que instrumentação é o primeiro passo para a evolução automobilística: radares, visão estéreo, LIDAR, câmeras infravermelho, GPS, encoders e uma enxurrada num mundo tomado pelo dilúvio de dados[3].

Big Data tem transformado modelos de negócio, técnicas de processamento de dados, armazenamento de informação e análise de mercado, entre diversos outros tópicos, e alavancou a era da Internet das Coisas (IoT – do inglês *Internet of Things*), permitindo troca de dados e "experiências"entre dispositivos. Caminho este que deve ser seguido também na busca do transporte do futuro, já que, segundo a Intel (2016), um único veículo deve gerar mais de 1 Gb de dados por segundo, provendo informações que podem ser compartilhadas entre os veículos para a construção de um tráfego seguro e organizado.

[3] Abundância de dados científicos (HAAG, 2011)

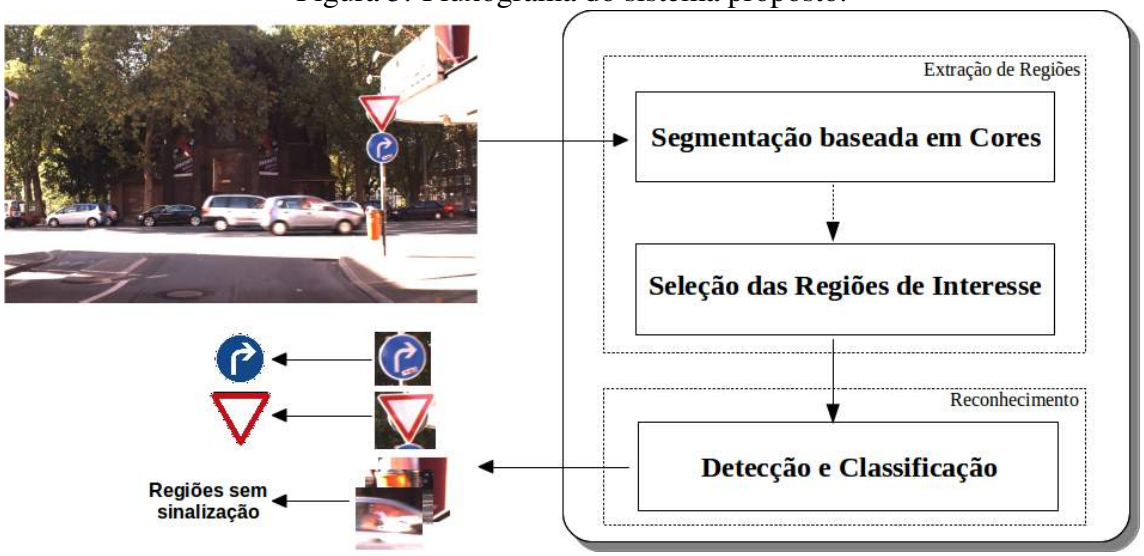
1.2 Objetivos

1.2.1 Objetivos gerais

O objetivo deste trabalho é desenvolver algoritmos de visão computacional para a detecção e o reconhecimento de sinalização vertical de trânsito, para compor um sistema de apoio ao motorista ou o *framework* de um veículo autônomo. Ao se adaptar à normatização federal de cores e formas para sinais de trânsito, o foco do sistema é extrair regiões de interesse a partir de imagens de uma câmera posicionada na frente do veículo e classificar os tipos de sinalização em cenários complexos.

O fluxograma proposto para o sistema é apresentado na Figura 3.

Figura 3: Fluxograma do sistema proposto.



Fonte: Elaborado pelo autor.

1.2.2 Objetivos específicos

1. Identificar os componentes que influenciam a qualidade visual do objeto de interesse em uma imagem real.

2. Apresentar uma técnica de extração de regiões de interesse, baseada na parametrização normativa dos sinais de trânsito.

3. Desenvolver um sistema capaz de classificar os tipos de sinalização de trânsito vertical.

4. Preparar e disponibilizar um conjunto de dados de teste nacional público, com imagens de tráfego em diferentes condições visuais.

2 REVISÃO BIBLIOGRÁFICA

As seções deste capítulo se dedicam a apresentar uma revisão da literatura relacionada ao processo de detecção e reconhecimento de sinalização de trânsito (TSDR – do inglês *Traffic Sign Detection and Recognition*).

O capítulo inicia apresentando um histórico dos problemas e a evolução das diferentes soluções para ADAS e navegação autônoma. Em seguida, são apresentadas as principais abordagens ao problema de TSDR. Ao retratar conceitualmente as principais abordagens encontradas na comunidade científica – destacando vantagens e desvantagens de cada uma, o estudo deve embasar a decisão metodológica de projeto do sistema desenvolvido, por isso essa seção se encerra com uma breve indicação dos trabalhos que inspiraram as implementações desse trabalho.

2.1 Histórico da Navegação Autônoma e ADAS

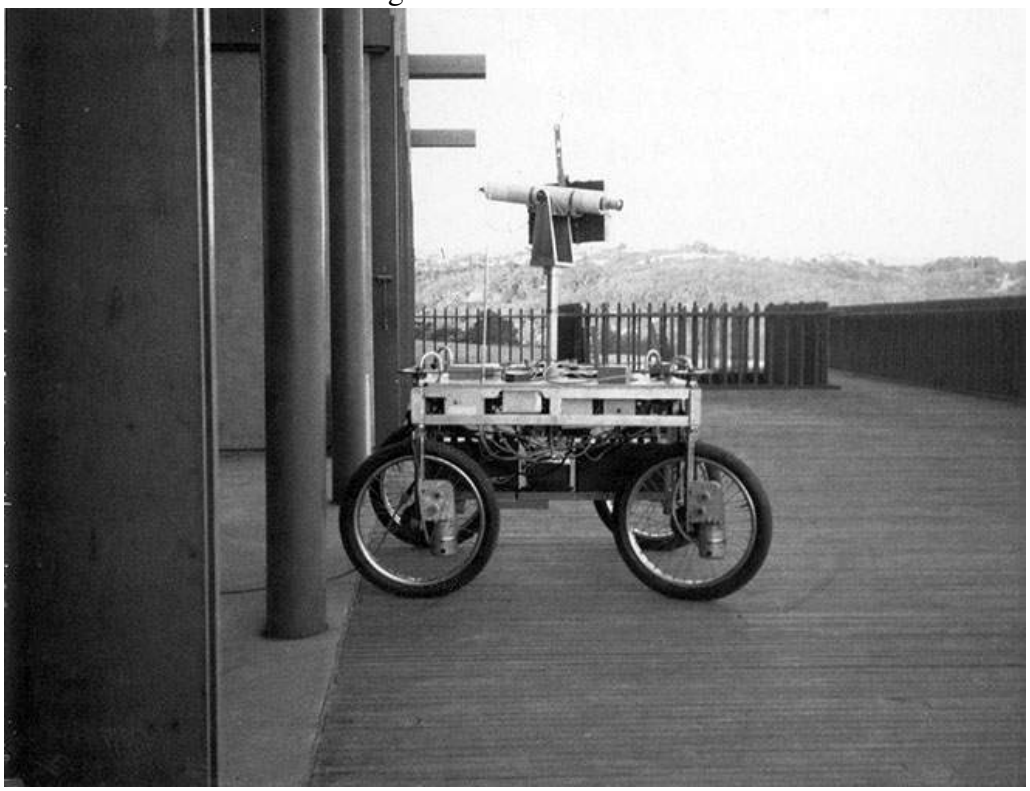
Somos atraídos pela comodidade e eficiência de uma sociedade equipada com veículos autônomos. Máquinas que seriam capazes de organizar definitivamente o tráfego de pessoas e mercadorias, desburocratizar o compartilhamento de transporte e reduzir o tempo que passamos aprisionados na estrada. Imagine ser conduzido até o local de trabalho pelo mesmo veículo que, ao deixá-lo no seu destino, retorna para casa para levar seus filhos para a escola. Ou um sistema de transporte – seja ele público ou corporativo – que otimiza as viagens de uma frota de veículos compartilhados. Sem colisões, milhares de acidentes e mortes no trânsito poderiam ser evitados.

Gradualmente, essa tecnologia cresce, evoluindo do ABS ao controle automático de direção. O que era apenas uma ideia no começo do século 20 chegou ao Stanford Cart (Figura 4) nos anos 1960 que, equipado com uma câmera e controlado por cabo, foi desenhado para ser controlado na Lua por operadores na Terra. Mais tarde, o veículo foi readaptado para trafegar na rua, recebeu diferentes sistemas e em 1979 foi capaz de "atravessar uma sala cheia de cadeiras sem intervenção humana"(VANDERBILT, 2012).

Sistemas de visão foram sendo aprimorados e em 1987 o VaMoRs (Versuchsfahrzeug für autonome Mobilität und Rechnersehen, traduzido do alemão como "Veículo de Teste para a Mobilidade Autônoma e Visão Computacional"), uma van de 5 toneladas e equipada com transputers e PowerPCs foi capaz de rodar sem motorista em rodovias a uma velocidade máxima de 50 km/h.

Sete anos mais tarde, o VaMP, com duas câmeras processando imagens frontais e traseiras de 320x240 pixels, poderia reconhecer marcações de estrada, sua posição relativa na pista e a presença de outros veículos. Em um teste perto de Paris, o carro dirigiu a 130 km/h em um tráfego denso com três pistas, julgando automaticamente se era seguro mudar de faixa (DICKMANNS, 1997). Ambos os veículos juntos acumularam um recorde de

Figura 4: Stanford Cart.



Fonte: VANDERBILT (2012)

cerca de 10.000 km em condução totalmente autônoma em diferentes tipos de estrada.

Em paralelo, o NavLab 5 era preparado para uma viagem de mais de 4.500 km, com sistemas autônomos de manutenção de faixa, aviso de presença de objetos, suporte para mudança de faixa lateral e alerta de curva implementados em um sistema de navegação portátil chamado PANS (JOCHEM et al., 1995). Cerca de 98.2% da viagem foi realizada de maneira autônoma (JOCHEM; POMERLEAU, 1995).

Hoje parte de grandes exposições em renomados museus no mundo todo, esses veículos inspiraram muitos outros, melhor equipados ao longo do tempo. Entre eles está o Stanley (Figura 5), um VW Touareg da Universidade de Stanford e vencedor do DARPA *Grand Challenge* em 2005. A competição criada pela Agência de Projetos de Pesquisa Avançada de Defesa em 2004 reuniu veículos autônomos para a tarefa de percorrer 212 km em terreno off-road sem intervenção humana. O veículo estava equipado com quatro *scanners* à laser, radares de 24 GHz, câmeras e sistemas de visão computacional estéreo e monocular, além de sistema GPS e foi capaz de terminar o circuito em 6 horas e 54 minutos (THRUN et al., 2006).

Novos desafios foram surgindo, incluindo o DARPA *Grand Challenge* de 2007 que se tornou uma competição de navegação autônoma em cenários urbanos e o *Grand Cooperative Driving Challenge* 2011, competição europeia que reuniu 11 grupos de pesquisa para um desafio de navegação autônoma na rodovia A270 nos Países Baixos (LAUER, 2011).

Muitos sistemas e arquiteturas foram padronizados e são utilizados hoje em dia em veículos comerciais. Informações, que nos desafios seriam utilizadas para navegação autônoma, compõem sistemas que não possuem o controle do veículo, mas garantem que o motorista tenha conhecimento delas através de ADAS. Esse tipo de sistema enfrenta

Figura 5: Stanley, carro vencedor do DARPA Grand Challenge 2005.

Fonte: THRUN; MONTEMERLO; DAHLKAMP; STAVENS; ARON (2006)

menos obstáculos para entrar no mercado popular de automóveis, apesar de ainda inflamar discussões sobre as interfaces de comunicação entre a máquina e o ser-humano durante a atividade de direção.

Quem tem a ganhar com isso é a própria indústria de semicondutores. Um relatório publicado pela *McKinsey & Company* sobre as oportunidades de hardware para ADAS até 2025 indica que as receitas globais nesse setor poderiam aumentar de forma constante, atingindo cerca de 4,6 bilhões a 5,3 bilhões de dólares em 2025 (CHOI et al., Feb. 2016). Entre os sistemas que devem gerar maior rendimento estão o de assistência ao estacionamento, frenagem de emergência automática, controle de distância adaptativo e aviso de risco de colisão frontal. Já no campo de componentes, as melhores oportunidades parecem estar nos processadores (gerando uma previsão de 37% da receita total) e nos semicondutores ópticos (28%).

Nissan, Volvo, Audi, Mercedes e muitos outros gigantes da indústria automobilística já anunciaram estar trabalhando em veículos autônomos, acompanhadas por companhias da tecnologia como Google e Apple. Mas o primeiro veículo comercial equipado com sistema de navegação autônoma disponível para o público chegou ao mercado em 2014: Model S da Tesla Motors. O automóvel é equipado com uma câmera estéreo (desenvolvida pela Mobileye) posicionada sobre o para-brisas, radares (desenvolvidos pela Bosch) na grade frontal e sensores ultrasônicos posicionados nos para-choques frontais e traseiros. Os dados fornecidos pelo equipamento são processados pelo sistema AutoPilot, sendo capaz de reconhecer sinalização vertical de trânsito, detectar as marcações horizontais na pista, obstáculos e outros veículos.

Porém, o primeiro sistema acompanha também a primeira fatalidade: após 208 milhões de quilômetros percorridos por veículos Model S, um motorista morreu em uma colisão com outro veículo durante o uso do AutoPilot. A empresa afirmou que "o veículo estava em uma rodovia de múltiplas pistas com o piloto automático ligado quando um trailer fez uma manobra inesperada perpendicular ao *Model S*"(TESLA MOTORS, 2016). Nem o piloto automático e nem o motorista notaram o lado branco do trailer contra a forte iluminação do céu ensolarado, e por isso o freio não foi aplicado.

A Volvo Cars deve testar em 2017 100 SUVs XC90 equipados com a tecnologia de

automação nível 3 *Drive Me* com moradores da cidade de Gotemburgo, na Suécia. O XC90 será equipado com o supercomputador Drive PX 2 da NVIDIA e será conduzido de forma autônoma em certas condições climáticas e em uma estrada que percorre a cidade. Como parte do projeto Drive Me da Volvo, os 100 carros terão uma interface chamada *IntelliSafe Auto Pilot*, permitindo que os motoristas ativem e desativem o modo autônomo através de controles no volante (VOLVO CAR GROUP, 2015).

Fica evidente que repensar maneiras de agilizar o trânsito de pessoas e mercadorias é o principal objetivo dessas tecnologias passos cada vez mais audaciosos são tomados a cada ano, sendo os últimos de conhecimento do público realizados pelas companhias nuTonomy e Otto. A primeira delas (uma startup que nasceu no MIT) introduziu a primeira frota de táxis, com seis veículos Renault Zoe e Mitsubishi i-MiEV, com permissão para realizar transporte de pessoas (inscritos previamente no programa) numa área de aproximadamente 6.5 km^2 em Singapura (VINCENT, 2016). Enquanto que a Otto (uma empresa do grupo Uber) fez a sua primeira entrega com um caminhão Volvo autônomo: 50.000 latas de cerveja por mais de 160 km (HAWKINS, 2016).

Máquinas e pessoas estão cada vez mais conectadas e tecnologias de navegação já traçaram seu caminho no mercado de produtos e serviços. Entretanto, a comunidade não está livre das discussões éticas em torno do assunto e nem todos estão prontos para a iminente revolução no setor.

2.2 Abordagens para TSDR

Detecção e reconhecimento de objetos em cenários externos tem sido alvo de estudos desde o inicio da computação e, aplicações direcionadas à navegação autônoma são encontradas na literatura desde os anos 1980 (ESCALERA et al., 1997).

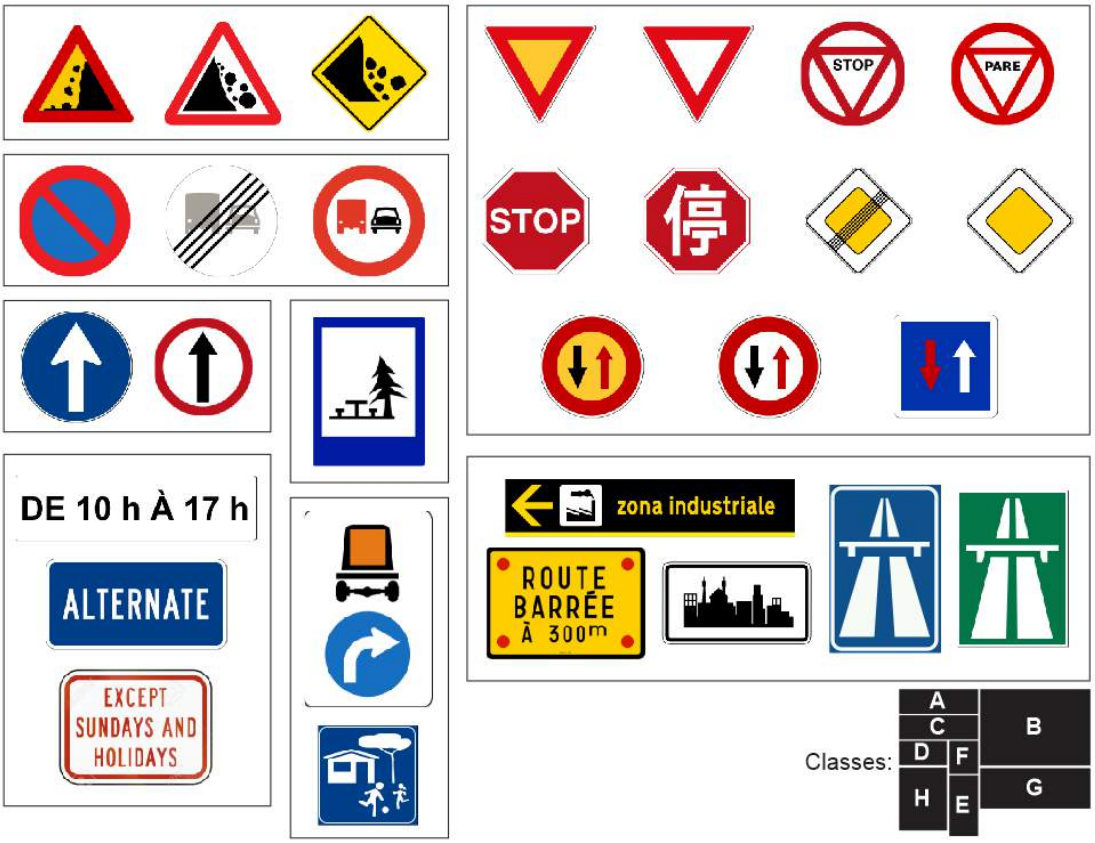
Considerada uma das principais informações para navegação e trânsito de múltiplos veículos na mesma via, os sinais de trânsito foram projetados para serem facilmente reconhecidos e destacados de uma cena pelo humano, e tiveram sua primeira regulamentação internacional determinada durante a Convenção sobre Trânsito Viário de 1968, sendo dispostos em sete categorias (A-H): Sinais de aviso de perigo (A), sinais de prioridade (B), sinais proibitivos ou restritivos (C), sinais obrigatórios (D), informações, instalações ou sinais de serviço (F), sinais de direção, posição ou indicação (G) e, painéis adicionais (H), com exemplifica a Figura 6.

Como um dos 74 países que assinaram a convenção adotada pelo Conselho Social e Econômico da ONU, o Brasil possui um padrão de cores e formas para a sinalização de trânsito regulamentado e apresentado no Manual Brasileiro de Sinalização de Trânsito. São objetos de estudo nesse trabalho as classes nacionais de sinalização vertical de regulamentação (A) (CONSELHO NACIONAL DE TRÂNSITO, 2007a, vol. I) e advertência (B) (CONSELHO NACIONAL DE TRÂNSITO, 2007b, vol. II), como exemplifica a Figura 7.

A sinalização vertical de trânsito apresenta cores, formas e símbolos específicos, ideais para que um humano possa classificá-la. Apesar de, no entanto, não ser a convenção ideal para que uma máquina realize a classificação, a maioria das abordagens em TSDR encontradas na literatura fazem uso dessa padronização para localizar as regiões de interesse na imagem e realizar o reconhecimento dos objetos.

No entanto, várias questões devem ser levadas em consideração num sistema automático de reconhecimento de sinais de trânsito. Por exemplo, a aparência do objeto em uma imagem depende de vários aspectos, como condições de iluminação externa, confi-

Figura 6: Exemplos de placas de sinalização definidas pela Convenção de Viena, organizadas em suas oito classes.



Fonte: Adaptado pelo autor.

Figura 7: Exemplos de placas de regulamentação (A) e advertência (B) do Sistema Nacional de Trânsito.

Fonte: Adaptado pelo autor.

guração da câmera e qualidade do sensor de aquisição. Além disso, a deterioração de um sinal de tráfego devido ao envelhecimento ou vandalismo afeta a sua aparência, enquanto que o tipo de material de revestimento usado para a construção das placas de sinalização também pode causar variações. Obstrução parcial, sombras, perspectiva e inclinação provocadas por deslocamento do suporte ou da própria placa, etc., também são variáveis que afetam a eficiência no reconhecimento desses objetos. Por fim, as imagens de sinal de trânsito tomadas de um veículo em movimento podem sofrer de desfocagem devido ao movimento do veículo e distorções causadas pela trepidação no sistema de captura.

A maioria das dificuldades apresentadas afeta principalmente uma das etapas iniciais de um sistema de visão computacional para TSDR: a segmentação. A segmentação é responsável por subdividir a imagem em componentes (regiões similares entre si) presentes na cena e, portanto, é considerada totalmente dependente da aplicação em que será empregada. Nesse caso, a segmentação deve ser capaz de encontrar regiões que possivelmente compreendem uma placa de sinalização.

Baseado na natureza dos objetos de interesse, existem dois grupos diferentes de segmentação amplamente utilizadas nesses sistemas: a segmentação baseada em cor e a segmentação baseada em forma. A primeira delas geralmente apresenta um custo computacional reduzido, mas pode sofrer com as variações na iluminação em ambientes exteriores. Já a segunda classe de métodos de segmentação é, geralmente, robusta aos principais problemas discutidos na detecção de objetos em cenários externos, entretanto podem levar a erros em casos em que o formato do objeto de interesse tenha sofrido distorções ou esteja sob oclusão parcial. Diversos trabalhos que serão discutidos nessa seção apresentam técnicas de ambas as classes e/ou propõem uma união entre métodos baseados em cor e em forma.

Entretanto, a segmentação, por si só, não garante que o objeto isolado seja realmente o objeto alvo da busca, sendo assim necessário associar a abordagem a outras técnicas de classificação e/ou reconhecimento. Essa seção deve evidenciar a relação entre diferentes abordagens para o problema de TSDR, apresentando uma relação dos principais trabalhos na área.

Em 2001, Cheng *et al.* (2001) e Lucchese e Mitray (2001) já descreviam mais de 150 trabalhos de segmentação baseada em cores. Esse número cresce quando levando em consideração técnicas de segmentação em escala de tons de cinza, passando a ser um tópico que demanda pesquisas exaustivas na área de processamento de imagens em geral e por isso será considerado objeto de interesse paralelo ao foco desse trabalho. A revisão bibliográfica que segue tem o objetivo de apresentar uma evolução natural das diferentes técnicas de segmentação, extração de regiões e classificação, tanto no domínio de cores da imagem quanto pelas suas características de forma, aplicadas em TSDR.

Soluções matemáticas e de visão computacional para o problema de TSDR remetem ao começo dos anos 1990 (LUO; POTLAPALLI, 1994; LUO; POTLAPALLI; HISLOP, 1992a,b; ESTABLE et al., 1994; PICCIOLI et al., 1994). Um dos primeiros trabalhos a descrever com sucesso uma abordagem para o problema em cenários externos foi desenvolvido por De La Escalera *et al.* (1997), ao aplicar uma segmentação no espaço de cores RGB para binarização por limiar. Para evitar que o sistema tenha sua eficiência reduzida com a sensibilidade do espaço de cores RGB às variações de luminosidade, os autores apresentam uma normalização na técnica de limiarização, que busca selecionar apenas os píxeis de cor vermelha. Na extração das regiões segmentadas, o documento apresenta uma técnica de detecção de cantos e características de borda por máscaras, permitindo a seleção de regiões triangulares, retangulares e circulares. Por fim, a classificação é rea-

lizada utilizando uma arquitetura de redes neurais artificiais conhecida como Perceptron de Múltiplas Camadas (do inglês *Multilayer Perceptron* – MLP), para nove classes de sinalização de trânsito vertical.

Utilizando dos mesmos princípios básicos, Vitabile *et al.* (2002) apresentou um sistema que realiza segmentações da imagem com relação às cores vermelha e azul, dessa vez utilizando uma limiarização adaptativa no espaço de cores HSV. A extração é feita pelo cálculo de similaridade entre as regiões segmentadas e um conjunto de amostra que representa o espaço de formas dos sinais de trânsito, com objetivo de obter objetos circulares em azul e vermelho e objetos triangulares de cor vermelha, que no fim são classificados utilizando três MLPs diferentes e descorrelacionadas para um total de 24 classes.

Ambos trabalhos discutidos até o momento não levam em consideração problemas como a oclusão parcial e deformações que podem afetar a apresentação das placas de sinalização no mundo real. Por isso, De La Escalera *et al.* (2003) discute a etapa de extração das regiões como um problema de otimização e aborda a utilização de Algoritmos Genéticos para a seleção da região ótima ao minimizar a sua distância para o conjunto de modelos. Além disso, os autores apresentam uma técnica de segmentação no espaço de cores HSV que combina as informações de matiz e saturação do píxel para gerar uma imagem de tons de cinza. Outra vez, o reconhecimento fica por conta de uma MLP construída para ser robusta em casos de oclusão, rotação e sombreamento.

Comum nesse tipo de sistema, a segmentação da imagem pela informação de cor do píxel reduz o espaço de buscas das regiões de interesse. Entretanto, mesmo que em espaços de cores ditos insensíveis às variações de iluminação presentes em cenários externos, essas técnicas podem acarretar em um número elevado de falsos negativos. Loy e Barnes (2004), apresentaram uma técnica que utiliza a natureza simétrica do triangulo, retângulo e octógono para detectar sinais de trânsito em imagens em escala de cinza. O método elevou a taxa de detecção da época e pode ser implementado em aplicações de tempo real, apesar de ainda ser afetado por diferentes problemas encontrados em cenários complexos.

Para alcançar maior robustez, a utilização de descritores de imagem para o reconhecimento das regiões de interesse começou a ser apresentada a partir do ano de 2007. Utilizando uma segmentação no espaço de cores HSV que implementa uma técnica de crescimento de regiões para agregar objetos próximos e eliminar artefatos (FLEYEH, 2006), Fleyeh *et al.* (2007) realiza a classificação dos sinais de trânsito combinando *Zernike Moments* e *Fuzzy ARTMAP*. Ao realizar uma redução no espaço de características utilizando *Linear Discriminant Analysis* (LDA), o sistema alcançou uma taxa de 100% de classificação dos formatos e 96,0% no reconhecimento dos diferentes sinais de trânsito da classe que limita a velocidade da via.

Keller *et al.* (2008) demonstrou a utilização do algoritmo Viola-Jones (VIOLA; JONES, 2001) – inicialmente construído para reconhecimento de faces – no problema de detecção de sinais de trânsito retangulares, enquanto Zaklouta *et al.* (2011) apresentou a eficácia do descritor HOG (DALAL; TRIGGS, 2005) – originalmente construído para a detecção de humanos – quando combinado com classificadores *Random Forest* no reconhecimento de sinalização de limite de velocidade. Viola-Jones também foi a técnica utilizada na detecção dos sinais por Chen *et al.* (2011), que realizou a classificação ao combinar descritores SURF utilizando um algoritmo clássico de busca – o *Approximate Nearest Neighbor* (ANN). No mesmo ano, Fleyeh *et al.* (2011) utilizou a Transformada de Hough para a detecção de triângulos em imagens segmentadas em HSV por *Self Organizing Maps* (SOM), enquanto que Larsson e Felsberg (2011) implementaram modificações aos Descritores de Fourier para detectar placas de sinalização em imagens em tons de

cinza.

Entretanto, a maioria dos trabalhos apresentados até aquele ano não possuía uma plataforma padronizada de *benchmarking* e a comparação entre os resultados se dava, em sua maioria, de forma qualitativa. Stallkamp *et al.* (2012) introduziu uma base de dados pública com mais de 50.000 imagens de placas de sinalização alemãs, de 43 classes diferentes chamada *German Traffic Signs Recognition Benchmark* (GTSRB). Resultados de classificação também são apresentados no formato da competição organizada pela IJCNN de 2011 e indicaram que o comitê de Redes Neurais Convolucionais (do inglês *Convolutional Neural Network* – CNN) desenvolvido por Ciresan *et al.* (2011) obteve performance superior até mesmo à de humanos.

Merecida menção também à estrutura de CNN desenvolvida por Sermanet e Lecun (2011), comprovando superioridade dessa classe algorítmica no problema de reconhecimento de imagens. Por isso, trabalhos posteriores investiram em diferentes arquiteturas de CNN, alcançando a cada ano melhores resultados tanto na GTSRB, como em bases de dados próprias com classes e estruturas de sinalização de trânsito diferentes. Bases de dados com imagens e vídeos de cenários de trânsito passaram a ser publicadas na comunidade científica, direcionadas tanto ao problema de detecção quanto reconhecimento de placas de sinalização: GTSDB (HOUBEN et al., 2013), LISA Traffic Sign Dataset (MØGELMOSE; TRIVEDI; MOESLUND, 2012), BelgiumTS (MATHIAS et al., 2013), Swedish Traffic Sign Dataset (LARSSON; FELSBERG, 2011) e, CTSD (YANG et al., 2016).

Apesar da surpreendente performance das CNNs, essa classe de classificadores se manteve restrita ao problema de reconhecimento enquanto novas técnicas de detecção continuaram a surgir. Yang *et al.* (2016) apresentou uma segmentação de imagens baseada em cores que cria um mapa de probabilidades em tons de cinza para as cores de interesse, permitindo a utilização da técnica MSER (do inglês *Maximally Stable Extremal Regions*) para seleção de regiões conexas e Máquinas de Vetores de Suporte (do inglês *Support Vector Machine* – SVM) para eliminar regiões que não compreendem uma placa de sinalização, baseada nas suas características HOG. Considerada como estado da arte na detecção em tempo real, esse sistema foi testado nas bases de dados GTSDB e CTSD obtendo altas taxas de recuperação de regiões com o menor tempo de processamento da literatura. O reconhecimento ficou por conta de uma CNN com apenas duas camadas de convolução e duas camadas MLP, obtendo alta performance enquanto alcança uma taxa de classificação de 97,75% no GTSRB.

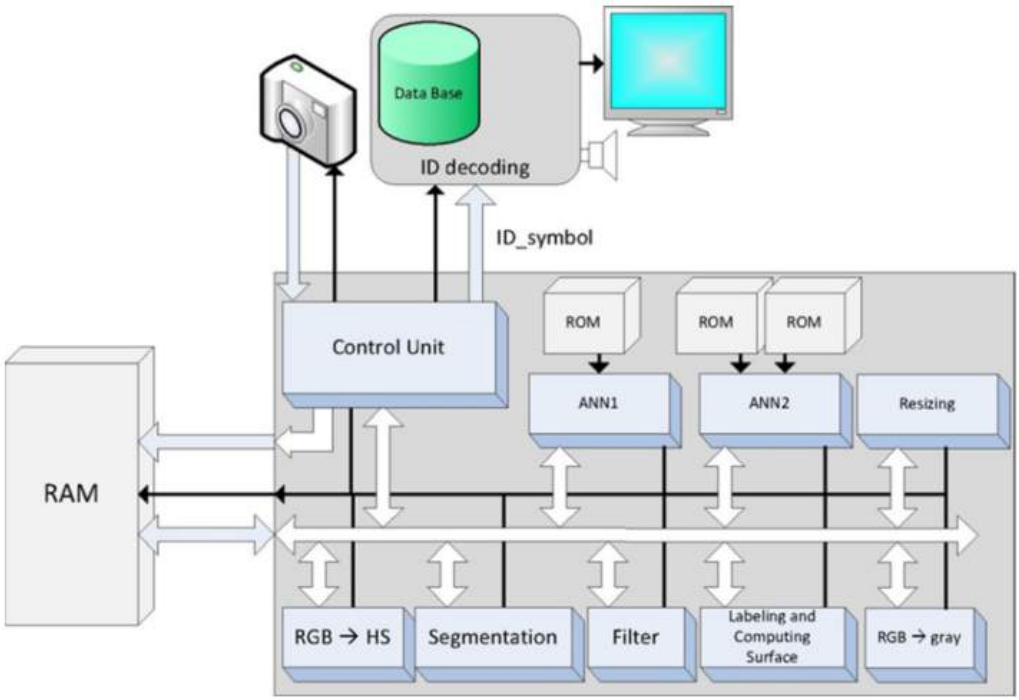
O cenário em que se enquadram as CNNs também depende do crescimento no volume de dados disponível. A maioria dos trabalhos que implementam esse tipo de arquitetura em TSDR aplica técnicas de aumento de dados artificial mesmo em grandes bases como a GTSRB. Para levar à uma nova revolução no campo de soluções para TSDR, Zhu *et al.* (2016) propuseram a realização de ambas tarefas, detecção e classificação, simultaneamente utilizando uma única estrutura de CNN puramente convolucional. Para isso, os autores criaram uma base de dados de cenários de trânsito com 100.000 imagens – 111 vezes maior que a GTSDB – de alta resolução, obtidas com a partir da Tencent Maps que realiza um serviço equivalente ao Google Street View na China. A Rede Neural Convolucional apresentada pelos autores possui oito níveis de convolução e para uma entrada contendo uma imagem 680x480x3 de um cenário de trânsito ela exibe na saída as regiões detectadas com probabilidade de conter um sinal de trânsito (84% de acurácia) e a sua classe (88% de acurácia).

2.2.1 Tempo Real e Implementações em Hardware

Arquiteturas de hardware, plataformas de processamento, quantidade de dados, tamanho das imagens e dificuldade da busca são algumas das variáveis que influenciam significativamente a taxa de detecção e classificação de uma abordagem para TSDR. A maior parte da literatura não discute implementações de hardware das suas soluções e realiza comparações de tempo de processamento com implementações em software em computadores pessoais. No entanto, fica claro que uma solução em TSDR é apenas um dos vários módulos que deverão compor um sistema embarcado de navegação autônoma e/ou assistência ao motorista e, por isso, estarão competindo entre si numa rede de sensores e atuadores complexa com computação distribuída de baixo consumo de energia.

O problema pode assumir novos níveis de complexidade e tornam-se foco de outros trabalhos que buscam acelerar soluções criadas em software. Souani, Faiedh e Besbes (2014) apresentam uma das principais arquiteturas de TSDR em hardware encontradas na literatura, ainda que utilizando algoritmos simples de detecção e reconhecimento. Módulos de conversão de espaço de cores, segmentação utilizando binarização por limiar, filtros de imagem, análise e classificação de formas e reconhecimento de interiores utilizando redes neurais artificiais foram implementadas em uma FPGA Virtex 4 (como mostra a Figura 8). Rodando à 88 MHz, o sistema é capaz de realizar detecções a cada 64 metros em um veículo à 120 km/h, com uma taxa de acerto no reconhecimento de 82%.

Figura 8: Diagrama de blocos do sistema TSDR.



Fonte: SOUANI; FAIEDH; BESBES (2014).

Oruklu, em conjunto com Waite (2013) e Han (2014), descreve dois sistemas de co-design entre hardware e software utilizando processadores de software e IPs em hardware para TSDR, o primeiro deles implementado em uma arquitetura com Microblaze (XILINX, 2016) (Virtex 5) e o segundo em uma placa de desenvolvimento ZedBoard (AVNET, 2016) (processador ARM Cortex™-A9 integrados com Kintex®-7 de 28nm),

obtendo um ganho de performance de oito vezes entre as duas implementações. O trabalho abre espaços para aplicações em *single-board computers* – como a Raspberry Pi – e novos *Systems-on-Chip* (SoCs) com processadores de uso genérico associados à co-processadores especializados, incluindo a Parallella (ADAPTEVA, 2014) e os módulos Jetson da NVIDIA (NVIDIA, 2016).

Outra possível solução para acelerar o processo de detecção e classificação é limitar a área de busca dos objetos de interesse. De Paula (2015, p. 79) utiliza as informações de uma câmera calibrada para, de acordo com o Manual Brasileiro de Sinalização de Trânsito (CONSELHO NACIONAL DE TRÂNSITO, 2007a), reduzir o espaço de busca na imagem. Assim, utilizando descritores HOG e um classificador binário SVM, a abordagem foi capaz de reduzir o tempo de busca e classificação de objetos 2,02s para 25,49ms e aumentar a acurácia média de 90,44% para 99,38%.

A evolução no poder de processamento permite que aplicações com maior complexidade computacional sejam adaptados para plataformas embarcadas, cada vez menores e mais potentes. Além disso, o advento de algoritmos com paralelismo intrínseco (como, por exemplo,as CNNs) facilitam a sua implementação em sistemas distribuídos. Por isso, novas abordagens que buscam otimizar os módulos de um sistema TSDR sem perder qualidade apresentam ainda mais potencial para explorar novas arquiteturas de hardware.

3 BENCHMARKS E BASES DE IMAGENS

Até a introdução de bases de dados públicas, contendo imagens de cenários de trânsito ou regiões contendo placas de sinalização, a maioria dos métodos publicados apresentavam resultados excelentes nas próprias imagens. Tipicamente, a maioria alcançava taxas de reconhecimento acima de 95%, com poucos falsos positivos, porém nenhum deles providenciava qualquer meio de comparar a eficiência com outras técnicas. Publicados quase que simultaneamente, as bases de dados sueca (LARSSON; FELSBERG, 2011) e alemã (STALLKAMP et al., 2012) iniciaram uma era de padronização de *benchmarks* para os problemas de detecção e reconhecimento de sinalização de trânsito.

O conjunto de imagens sueco foi criado ao longo de 350 km de rodovias e estradas urbanas da Suécia. Uma câmera colorida de 1,3 megapixels foi posicionada dentro de um veículo na altura do painel e direcionada para o para-brisa do carro, ligeiramente para a direita, para cobrir as regiões onde usualmente está presente a maioria dos sinais de trânsito. A câmera possuía uma distância focal de 6,5mm, com aproximadamente 41° de ângulo de visão. A base contém mais de 20.000 quadros, com anotações para cerca de 20%, contendo toda a visão frontal de um veículo do cenário de trânsito.

O GTSRB, conjunto de referência com imagens de sinalização de trânsito para reconhecimento, possui 51.840 imagens de placas de sinalização alemãs, classificadas em 43 classes diferentes e foi criado a partir de uma filmagem de aproximadamente 10 horas durante a atividade de direção em diferentes estradas da Alemanha, durante o dia. Diferentes horas do dia e diferentes estações do ano permitem a variabilidade nas características de iluminação das imagens, que foram gravadas nos meses de março, outubro e novembro de 2010. A câmera utilizada, uma Prosilica GC 1380CH com controle de exposição automático, fez a filmagem da visão frontal do veículo a uma taxa de 25 quadros por segundo, resolução de 1360×1024 píxeis e armazenamento no padrão Bayer (BAYER, 1976).

Este último é, desde a sua apresentação, um dos mais utilizados pela literatura para validar técnicas de reconhecimento de placas de sinalização e veio acompanhado de uma comparação entre diversos métodos. Suas mais de 50 mil imagens possuem tamanho que varia entre 15×15 e 250×250 píxeis, não necessariamente quadradas e centralizadas e, está dividido em conjunto de treinamento e de teste (com 39.209 e 12.630 imagens, respectivamente).

Entretanto, esse conjunto não possui regiões falsas de sinalização, nem cenários de trânsito para o problema de detecção. Por isso, o grupo de Visão Computacional em Tempo Real do Instituto de Computação Neural da universidade alemã *Ruhr-Universität Bochum* disponibilizou em 2013 o *benchmark* GTSDB, contendo 900 imagens de cenários de trânsito da Alemanha (e dividias em conjunto de treinamento, com 600 imagens, e teste, com 300 imagens). O sinais de trânsito presentes nesses cenários são divididos em três superclasses – proibitório, perigo e mandatório – de acordo com a forma e cores, para

permitir que diferentes abordagens possam ser validadas (HOUBEN et al., 2013).

Uma das mais diversas na literatura, a base de dados de sinais de trânsito LISA (MØGELMOSE; TRIVEDI; MOESLUND, 2012) possui 6.610 quadros, com 7.855 anotações de 47 classes de sinalização dos Estados Unidos da América. Obtidas com câmeras diferentes, algumas imagens se apresentam coloridas e outras em tons de cinza, com resolução variando entre 640×480 e 1024×522 e contendo objetos com tamanho entre 6×6 e 167×168.

O BelgiumTS, conjunto de imagens de sinalização de trânsito belga do Instituto Federal de Tecnologia de Zurique, também é dividido entre detecção (TIMOFTE; ZIMMERMANN; Van Gool, 2014) e classificação (TIMOFTE; GOOL, 2011) e contem 13.444 anotações para 4.565 placas de sinalização fisicamente diferentes em 9.006 imagens. Cada anotação marcada um objeto visível que está a, no máximo, 50 metros de distância da câmera.

Na Ásia, Yang *et al.* (2016) disponibilizou o conjunto CTSD com 1.100 imagens de cenários de trânsito chinês, dividido em conjunto de treinamento e teste (com 700 e 400 imagens, respectivamente). As imagens possuem diferentes resoluções – variando entre 1024×768 e 1280×720 – e contém 1.574 anotações de placas de sinalização, dividias nas mesmas superclasses que o GTSDB.

Recentemente, Zhu *et al.* (2016) apresentaram uma das maiores bases de dados de imagens para TSDR completamente anotadas. O Tsinghua-Tencent 100K contem 100 mil panoramas obtidos a partir do Tencent Street View – serviço chinês similar ao Google Street View – de cinco cidades da China, com cenários de trânsito tanto em grandes centros urbanos como em periferias. O conjunto possui mais de 30 mil placas de sinalização diferentes e cobre diferentes variações de iluminação, condições climáticas, posicionamento e oclusão dos objetos.

Para uma análise qualitativa dos métodos e discussão do problema de TSDR no Brasil, esse texto se propõe a apresentar a base de cenários de trânsito brasileira: BRTSD[1]. Essa base foi construída no Laboratório de Processamento de Sinais e Imagens do Departamento de Engenharia Elétrica da UFRGS e contém 2.112 cenários de trânsito em diferentes climas e regiões do país, cujas imagens estão em resolução WXSGA+ (1680×1050). Ainda sem anotações oficiais, uma contagem preliminar indicou a presença de 3.597 placas de sinalização de regulamentação (cor predominante vermelha) e 544 placas de sinalização de advertência (cor predominante amarela).

Para validação quantitativa das técnicas propostas e comparação de resultados, esse trabalho irá utilizar as bases GTSRB e GTSDB. O GTSDB permite a avaliação da precisão do módulo de extração de regiões de interesse, enquanto que o GTSRB permite o treinamento e validação da abordagem para classificação, em conjunto com a eficiência do sistema como um todo nas imagens do GTSDB.

[1] *http://lapsi.eletro.ufrgs.br/Download/BRTSD/*

4 MÓDULO DE EXTRAÇÃO DE REGIÕES

As cores e formas de uma placa de sinalização de trânsito são informações chave para que o motorista consiga identificá-las durante a atividade de direção em qualquer cenário. São projetadas para se destacar do fundo de uma imagem, com cores brilhantes e formatos geométricos de fácil identificação. Geralmente contem pictogramas de alto contraste e que representam mensagens curtas e mundialmente estabelecidas.

Entretanto, o reconhecimento pode ser prejudicado quando esses objetos sofrem com deteriorações devido à qualidade do material de fabricação, ao envelhecimento ou ao vandalismo. Além disso, o usuário pode encontrar dificuldades de detectar sinais de trânsito durante uma atividade de direção cansativa e estressante, em regiões com maior concentração de objetos visuais, oclusão parcial, iluminação desfavorável ou baixa visibilidade.

Além disso, a maioria das abordagens robustas nesse tipo de situação são geralmente complexas matematicamente e requerem mais recursos computacionais, ou se utilizam de uma combinação de sensores (câmeras de visão estéreo, LIDAR ou radares).

O foco desse trabalho é desenvolver técnicas e recomendar algoritmos com potencial para integrar sistemas embarcados de tempo real e que utilizem apenas a informação de uma câmera monocular para detectar e reconhecer sinalização de trânsito. Portanto, essa seção deve apresentar toda a construção do pipeline de extração de Regiões de Interesse (ROIs, do inglês *Regions of Interest*), projetado para conter os módulos de pré-processamento, segmentação baseada em cores e seleção de objetos (com um fluxo como apresentado pela Figura 9.

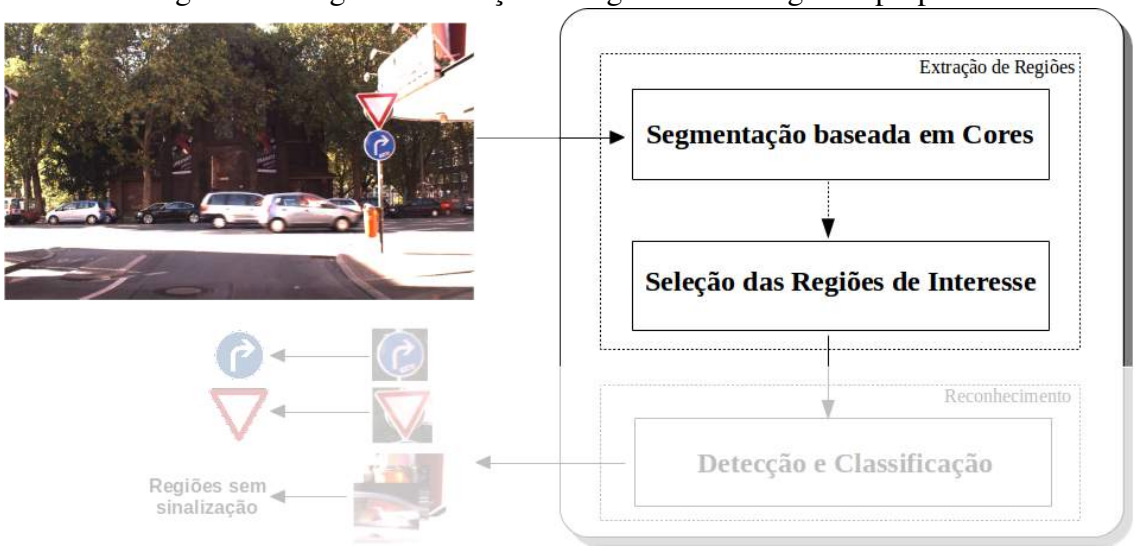
Os três estágios que realizam a extração de ROIs são geralmente dependentes da aplicação e as técnicas que se utilizam em cada uma delas estão fortemente conectadas entre si. Isso significa que o pré-processamento deve facilitar que as regiões de interesse sejam destacadas da imagem pela segmentação e que esse método permita a aplicação do algoritmo seletor em seu máximo. Por isso, dois *frameworks* de extração foram estudados e serão apresentados, ambos discutem maneiras de selecionar as regiões de interesse utilizando as informações de cor e forma dos objetos.

Após a introdução das técnicas estudadas e a sua implementação, a seção irá discutir os prós e contras de cada um dos métodos e apresentar resultados de extração nas bases de dados GTSDB e BRTSD, com uma comparação com o estado-da-arte.

4.1 Teoria de Cores

Cor é uma das principais informações na área de detecção e reconhecimento de objetos, permitindo segmentar imagens utilizando as propriedades de coloração dos materiais que compõem os objetos de interesse. Como mencionado anteriormente, a sinalização de trânsito definiu, por meio da Convenção de Viena, a utilização de diferentes colora-

Figura 9: Estágios de extração de regiões no fluxograma proposto.



Fonte: Elaborado pelo autor.

ções para exibir informações chave no reconhecimento desses objetos em um cenário de trânsito.

Espaços de cores são representações matemáticas para organizar e descrever conjuntos de cores como combinações de níveis de luz monocromática em diferentes comprimentos de onda (DUBOIS, 2009). Esses modelos permitem representar uma cor como sendo uma tupla de números – diferentes propriedades em cada modelo (por exemplo, três no espaço de cores RGB ou quatro no CMYK).

Em 1931, a Comissão Internacional de Iluminação (CIE – *Commission Internationale d'Éclairage*) definiu uma padrão de cores, fornecendo funções de correspondência de cores para dois conjuntos primários: um conjunto de primários monocromáticos vermelhos, verdes e azuis (RGB), e uma base transformada referida como XYZ (conhecida como triestímulos).

Mais tarde, outros espaços de cores foram criados – enquanto os já existentes foram modificados – e otimizados para diferentes aplicações. O número de espaço de cores hoje é gigantesco, mas a maioria deles possui suas bases nos principais, apresentados abaixo:

- RGB: modelo utilizado na captura e apresentação de imagens digitais;
- CMYK: modelo utilizado para impressão de alta qualidade;
- HSI, HSV(HSB) e HLS: modelos criados para simular o modo como o ser humano percebe as cores;
- YIO, YUV e YCbCr: geralmente utilizados em sistemas de video (televisão, por exemplo), e;
- XYZ: modelo que descreve as cores físicas primárias.

Segundo o Vocabulário de Iluminação CIE de 1987, Cor é o "atributo da percepção visual que consiste em qualquer combinação de conteúdo cromático e acromático. Este atributo pode ser descrito por nomes de cores cromáticas, como amarelo, laranja, marrom, vermelho, rosa, verde, azul, roxo, etc., ou por nomes de cores acromáticas, como

branco, cinza, preto, etc., e qualificado como brilhante, turvo, claro, escuro, etc., ou por combinações de tais nomes"(FAIRCHILD, 2013).

Porém essa definição é acompanhada por uma nota, indicando que a cor percebida depende de diferentes variáveis que não são controladas pela cor real do objeto. Em cenários externos, por exemplo, onde principalmente a iluminação não é uma variável de controle, a percepção da cor de um objeto por parte do observador pode variar de acordo com:

- a distribuição espectral da cor do estímulo produzido pelo iluminante;
- o tamanho, forma, estrutura e envolvente da área de estímulo;
- o estado de adaptação do sistema visual do observador, e;
- a experiência do observador, com relação à observações similares.

Ou seja, a mesma placa de sinalização pode ter coloração aparente totalmente diferente quando estamos vendo o objeto em diferentes distâncias, diferentes angulações, diferentes horários do dia ou em imagens capturadas com câmeras diferentes.

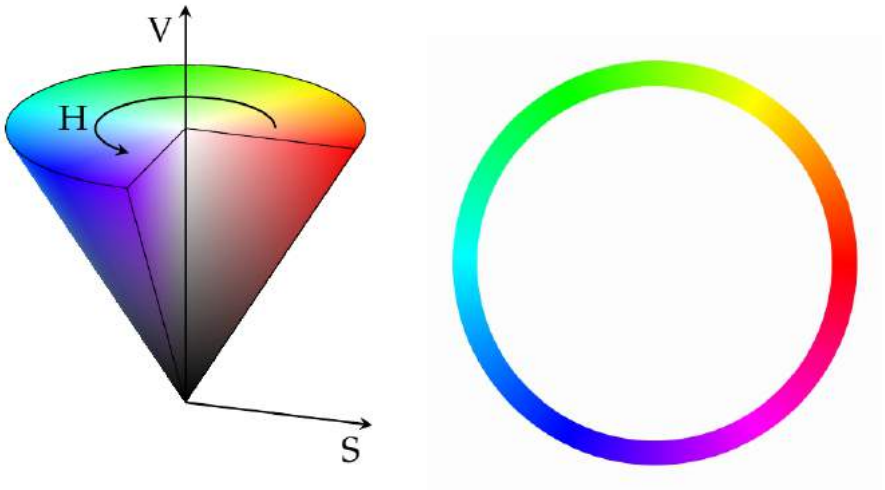
Para reduzir o efeito dessa variabilidade em técnicas de segmentação baseada em cores, diversos trabalhos utilizam espaços de cores capazes de representar níveis de cromaticidade e iluminação em canais diferentes, facilitando a seleção de regiões cromáticas da cor de interesse. Entre um dos mais utilizados está o canal HSV, ou HSB, cujos canais são Matiz (*Hue*), Saturação (*Saturation*) e Valor/Brilho (*Value*), inicialmente descrito por Smith (1978) como "Modelo Hexcone"para utilização em computação gráfica (Figura 10 (a)).

Segundo Fairchild (2013), Matiz é o valor atribuído à percepção do observador sobre a cor do objeto e em como essa cor se assemelha a uma das cores – vermelho, amarelo, verde e azul – ou a uma combinação de pares adjacentes destas cores consideradas em um anel fechado (Figura 10 (b)). O valor de matiz é obtido a partir dos atributos primários e possui a mesma formulação em qualquer espaço de cores, sendo descrito como ângulos que variam de 0 a 360°. Além disso, sua relação com RGB possui propriedades importantes na detecção de objetos:

- Invariância à escala/multiplicação: $hue(R, G, B) = hue(\alpha R, \alpha G, \alpha B)$, para todo α, respeitando $(\alpha R, \alpha G, \alpha B) \in ([0, 255], [0, 255], [0, 255])$;
- Invariância ao deslocamento/soma: $hue(R, G, B) = hue(R + \beta, G + \beta, B + \beta)$, para todo β, respeitando $(R + \beta, G + \beta, B + \beta) \in ([0, 255], [0, 255], [0, 255])$, e;
- Invariância à mudança de saturação, ou seja, mesmo que a pureza da cor sofra variações, ainda é possível obter o valor de matiz.

Entretanto, a expressividade da matiz é controlado pelas informações de Brilho e Saturação. O Brilho é o atributo dado à percepção de acordo com a quantidade de luz que uma área é capaz de emitir ou refletir e, no espaço de cores HSV, é representado como uma porcentagem (0 - 100%). Já a Saturação é a pureza ou quantidade de cor daquela área em relação ao seu brilho, representado em HSV também por uma porcentagem (0-100%). Quando o brilho ou a saturação é muito baixo, o valor de matiz não tem significado. Ainda, quando o valor de saturação se encontra está abaixo de um certo limiar, o valor de matiz é instável e sofre variações bruscas.

Figura 10: Representação do Espaço de Cores HSV e o anel de atributos de matiz.



(a) Modelo Hexcone. (b) Anel de matiz.

Fonte: Adaptado pelo autor.

O objetivo da segmentação baseada em cores nesse trabalho é reduzir o espaço de busca por Regiões de Interesse (ROIs) na imagem, permitindo maior eficiência na seleção de candidatos para classificação e reduzindo o tempo de processamento. As abordagens escolhidas para essa etapa (em ambos os frameworks) será a de seleção de píxeis de interesse, sem pesquisa na vizinhança, levando em consideração a sua posição em um certo espaço de cores.

4.2 Limiarização e Descrição de Bordas

Essa seção irá apresentar uma introdução sobre o espaço de cores HSV, utilizado tanto na segmentação por limiar como na segmentação por fuzzificação. Utilizando limiarização, o sistema gera uma imagem binarizada, contendo objetos da cor de intersse. Para filtrar esse objetos, a técnica apresentada nessa seção será a descriação de bordas utilizando a transformada de Fourier unidimensional.

4.2.1 Segmentação Binária Baseada em Cores

Como já discutido, o espaço de cores HSV é considerado um dos mais robustos às variações de luminosidade em imagens de cenários externos. Esse conjunto reune as características de matiz de um píxel em um único canal, que é controlado pela saturação e brilho.

Vitabile *et al.* (2002) definiu os limites para a expressividade da matiz, introduzindo três regiões diferentes de cromaticidade no espaço de cores HSV, levando em consideração as propriedades relacionais dos três canais e a influência dos valores de saturação e brilho na matiz de um píxel:

- Área acromática: $s \leq 0.25$ ou $v \leq 0.20$ e $v \geq 0.9$;
- Área cromática instável: $0.25 < s < 0.5$ e $0.2 < v < 0.9$;
- Área cromática: $s \geq 0.5$ e $0.2 < v < 0.9$;

onde s é o valor de saturação e v o valor de brilho.

Esses intervalos são levados em consideração no método de segmentação criado por Fleyeh (2006). O algoritmo cria uma máscara marcando os píxeis de acordo com (1), para detecção de cor vermelha. Para eliminar pequenas regiões ruidosas na segmentação, o método aplica crescimento de regiões utilizando "sementes".

$$pixel = \begin{cases} 255, & \text{if} \begin{cases} H < 10 \text{ or } H > 240 \\ S \geq 40 \\ 30 \leq V \leq 230 \end{cases} \\ 0, & \text{otherwise} \end{cases} \qquad (H, S, V) \in [0, 255]. \tag{1}$$

Entretanto, o método requer a conversão da imagem na escala RGB para HSV, cuja formulação (Equação (2)) possui equações não-lineares, aumentando a complexidade computacional do método. Além disso, a técnica de crescimento de regiões utilizando sementes é aplicada apenas para remoção de objetos muito pequenos na imagem, podendo ser substituída apenas por uma regra de seleção no método de extração. Esse método, com pequenas variações, passou a ser usado de forma recorrente em sistemas de TSDR mais recentes (NGUYEN; RYONG; KYU, 2014; CHEN et al., 2011; SOUANI; FAIEDH; BESBES, 2014; FLEYEH; ROCH, 2013), incluindo implementações em hardware (HAN; ORUKLU, 2014).

$$\begin{aligned} R' &= R/255.0 \\ G' &= G/255.0 \\ B' &= B/255.0 \\ C_{max} &= max(R', G', B') \\ C_{min} &= min(R', G', B') \\ \Delta &= C_{max} - C_{min} \\ H &= \begin{cases} 0^\circ & , \Delta = 0 \\ 60^\circ \times (\frac{G'-B'}{\Delta} mod 6) & , C_{max} = R' \\ 60^\circ \times (\frac{B'-R'}{\Delta} + 2) & , C_{max} = G' \\ 60^\circ \times (\frac{R'-G'}{\Delta} + 4) & , C_{max} = B' \end{cases} \\ S &= \begin{cases} 0 & , C_{max} = 0 \\ \frac{\Delta}{C_{max}} & , C_{max} \neq 0 \end{cases} \\ V &= C_{max} \end{aligned} \tag{2}$$

A relação direta e a dependência do espaço de cores RGB utilizado faz com que cada tom físico das cores vermelha, verde e azul e suas combinações possua uma única combinação em HSV. Além disso, essa relação depende totalmente o sistema RGB utilizado pelo dispositivo, fazendo com que imagens capturadas com ferramentas diferentes possam apresentar valores HSV diferentes para a mesma cor. A unicidade nessa relação permite que as regras aplicadas no espaço de cores HSV sejam utilizadas diretamente no espaço de cores RGB.

Por isso, nossa implementação da segmentação apresentada por Fleyeh utiliza uma tabela de consulta (LUT – do inglês *Look-up Table*) tridimensional de tamanho $255 \times 255 \times 255$. Como os limiares de segmentação são fixos e cada combinação de RGB possui uma correspondente HSV, calcula-se previamente todos os valores de segmentação para

toda combinação possível de RGB, armazenando esse valores na LUT. Assim, o valor de segmentação para um píxel em RGB com valores (r, g, b) pode ser obtido acessando a posição $[r, g, b]$ da tabela.

O método também foi expandido para segmentar imagens com relação às cores azul e amarela, movendo os limiares de H para a região que corresponde a cada uma das cores no anel de matiz. Assim, cada uma das cores predominantes em placas de sinalização vertical de trânsito possui uma LUT de segmentação e gera uma imagem binarizada que destaca regiões com píxeis de cor aparente similar às cores de interesse. Essa técnica é capaz de gerar três imagens segmentadas a partir de uma imagem RGB de tamanho 1680×1050 em apenas 32ms, utilizando um computador pessoal, com sistema operacional Linuxe processador i5-6600k (4GHz).

A Figura 11 apresenta exemplos da Segmentação por Limiarização aplicada para as cores vermelha e amarela em cenários de trânsito do Brasil, enquanto a Figura 12 mostra o algoritmo sendo aplicado para as cores vermelha e azul em cenários de trânsito da Alemanhã.

Figura 11: Exemplos de Segmentação por Limiarização em cenários de trânsito do Brasil. Para cada cenário é apresentada sua imagem segmentada (binarizada) e o resultado da operação AND com a imagem original, a fim de ilustrar a cor de interesse.

Fonte: Elaborado pelo autor.

4.2.2 Descritores de Fourier

Introduzido por Granlund (1972) para o reconhecimento de dígitos manuscritos, os Descritores de Fourier (FD, do inglês *Fourier Descriptors*) são uma classe de descritores que se utiliza da Transformada Discreta de Fourier (DFT, do inglês *Discrete Fourier Transform*) unidimensional para obter padrões de comportamento de contornos de formas bidimensionais. O método se aproveita de propriedades da DFT para encontrar atributos robustos à transformações afim: rotação, escala e translação.

O método assume que um contorno fechado c, cujos pontos possuem coordenadas x, y, pode ser representado como uma função complexa contínua e periódica

$$c(l) = c(l + L) = x(l) + iy(l) \tag{3}$$

Figura 12: Exemplos de Segmentação por Limiarização em cenários de trânsito da Alemanha. Para cada cenário é apresentada sua imagem segmentada (binarizada) e o resultado da operação AND com a imagem original, a fim de ilustrar a cor de interesse.

Fonte: Elaborado pelo autor.

onde L é o comprimento do contorno. Dessa forma, é possível obter os coeficientes de Fourier utilizando a Transformada de Fourier 1D de $c(l)$

$$C(n) = \frac{1}{L} \int_{l=0}^{L} c(l) exp(-\frac{i2\pi nl}{L}) dl \tag{4}$$

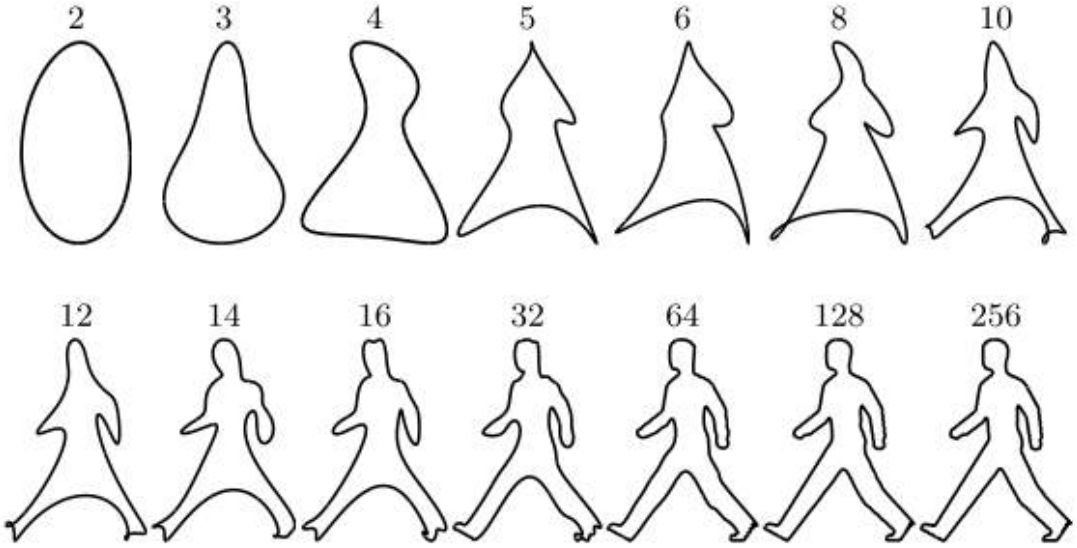
para $n = 0, ..., N$, onde $N \leq T$ é o número de descritores. A discretização da transformada ocorre porque o contorno, em uma imagem bidimensional, é descrito como sendo uma função discreta $c[l] = x[l] + iy[l]$.

O comportamento físico dos descritores está diretamente relacionado à interpretação da Transformada de Fourier para sinais. A utilização de componentes de baixa frequência permite a reconstrução de uma versão suavizada do contorno, enquanto que um número maior de descritores permite uma aproximação maior do comportamento do contorno e de suas variações de alta frequência no espaço da imagem. Larsson, Felsberg e Forssen (2011) apresentaram o exemplo da utilização de um número variado de descritores para representar o contorno do sinal de pedestre presente em uma das placas de sinalização suecas, como mostra a Figura 13.

Para tornar esse tipo de descritor robusto às transformações afins supracitadas, é necessário analisar como operações de escala, rotação e translação afetam os coeficientes de Fourier. Uma operação de translação sobre um contorno $c(l)$ em uma imagem I é basicamente a soma de uma coordenada x, y constante a todos os pontos da curva, ou seja, $c(l) \rightarrow c(l) + (x_0, y_0)$. O único coeficiente de Fourier afetado por essa operação é a componente DC $C(0)$, que, ao ser ignorada, torna os $N - 1$ coeficientes restantes invariantes à translação. A operação de escala de um contorno ($c(l) \rightarrow \alpha c(l)$) afeta apenas a magnitude dos coeficientes, podendo ser sobrepujada pela normalização dos $C(1...N)$ coeficientes em relação à energia do sinal, fazendo com que $||C||^2 = 1$.

Por fim, a operação de rotação do contorno $c(l)$ em ϕ radianos em sentido anti-horário corresponde à multiplicação de $c(l)$ por $exp(i\phi)$, que modifica a fase dos coeficientes de

Figura 13: Reconstrução do contorno interno de um sinal de cruzamento pedestre sueco usando um número crescente de coeficientes de Fourier.



Fonte: LARSSON; FELSBERG; FORSSEN (2011)

Fourier pela simples adição de um offset constante (Equation (5)).

$$c(l) \rightarrow exp(i\phi)c(l) \Rightarrow C(n) \rightarrow exp(i\phi)C(n) \tag{5}$$

Apesar de muitos trabalhos também ignorarem o sinal de fase dos coeficientes de Fourier, considerando os Descritores de Fourier como sendo $|C(n)|$ para $n = 1, ..., N$ e normalizados pela energia, Larsson *et al.* mostraram que dois contornos podem ser muito diferentes mesmo que a magnitude da DFT seja a mesma. Por isso, os autores criaram um método de correspondência entre dois contornos c_1 e c_2 dado por:

$$e = 2 - 2max_l Re\{r_{12}(l)\} \tag{6}$$

para $r_{12}(l) = \mathcal{F}^{-1}\{c_1 \cdot c_2\}(l)$

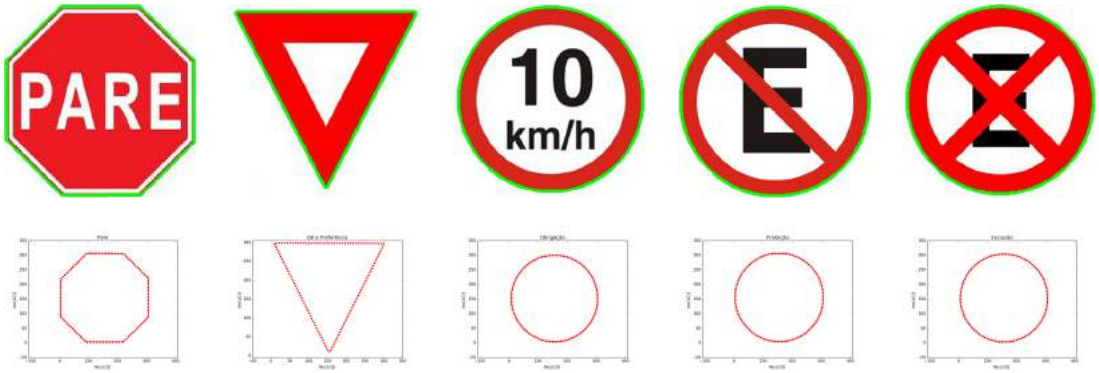
No mesmo ano, Larsson e Felsberg mostraram a aplicação desses atributos no reconhecimento de placas de sinalização suecas, obtendo o FD de todos os contornos em uma imagem em tons de cinza e cruzando com o banco de dados de contornos padrões (LARSSON; FELSBERG, 2011).

Entretanto, este trablaho utiliza o FD como método de confirmação de regiões, realizando uma busca por contornos externos, na imagem segmentada, similares aos contornos de placas de sinalização (circulares, triangulares ou octogonais). Isso significa que todos os objetos na imagem terão (para o seu contorno externo) um FD que será cruzado com a base de contornos externos padrão de cada sinalização de trânsito.

A Figura 14 apresenta alguns dos formatos padrões para as placas de sinalização do sistema brasileiro de trânsito, juntamente com o sinal gerado a partir do seu contorno externo. A Figura 15 cruza os Descritores de Fourier para cada um dos contornos, demonstrando a diferença entre as componentes de frequência para os sinais únicos ("Pare"e "Dê a Preferência") para com os sinais de regulamentação padrão ("Obrigação", "Proibição"e "Exclusão", de contorno circular).

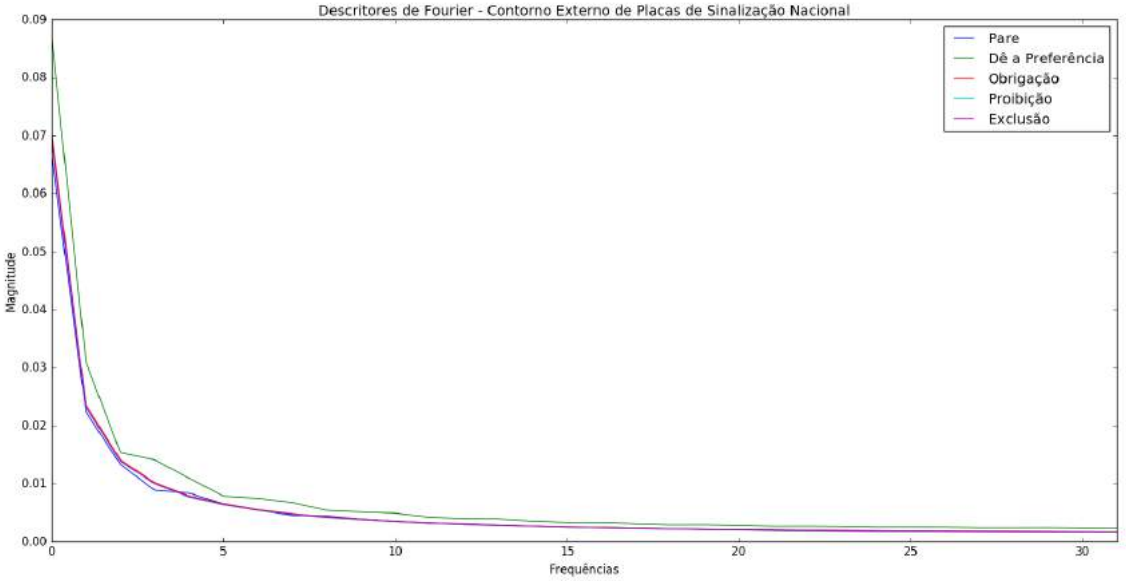
Ao analisar a Figura 15 é possível notar uma diferença no comportamento dos descritores para as curvas de formato octogonal e triangular, em relação aos contornos circulares. Essa diferença (e também a similaridade entre os três contornos circulares) pode

Figura 14: Exemplos de placas de regulamentação, seus contornos (em verde) e funções complexas associadas.



Fonte: Elaborado pelo autor.

Figura 15: 32 Descritores de Fourier para cada um dos contornos ilustrados na Figura 14.



Fonte: Elaborado pelo autor.

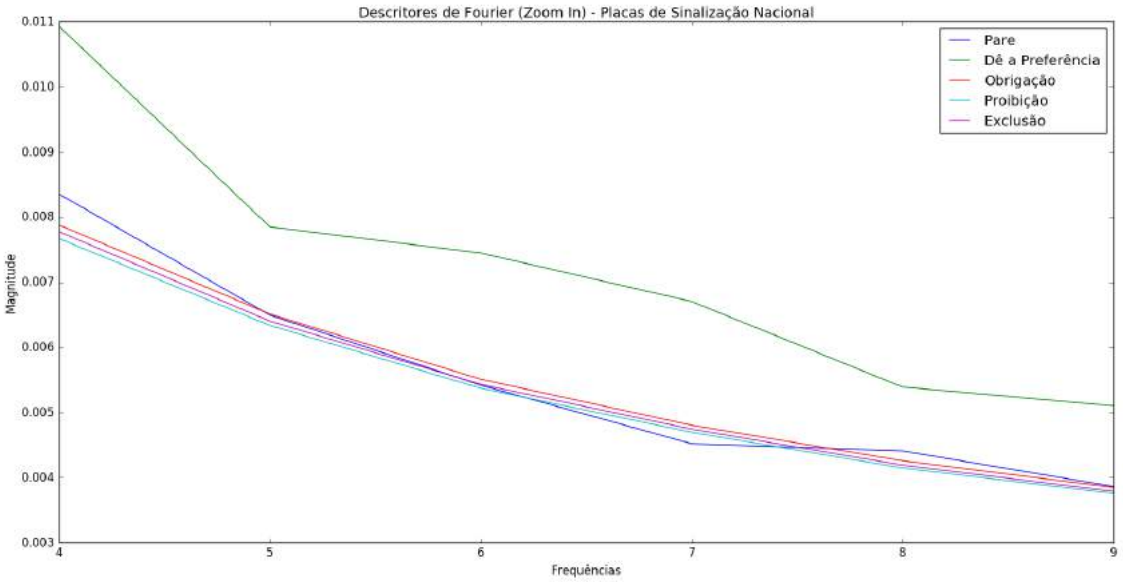
ser melhor vista quando analisamos apenas o intervalo de FDs entre $[4, 9]$, como mostra a Figura 16.

Entretanto, esse método encontra algumas dificuldades ao ser aplicado em imagens de mundo real. A obtenção de contornos fechados requer que o objeto esteja completo, fazendo com que a técnica seja sensível à oclusões parciais nos objetos de interesse. A solução implementada nesse caso leva em consideração que todos os formatos externos das placas de sinalização de interesse são convexos. Dessa forma, é possível aproximar o contorno externo por um contorno geométrico convexo e fechado.

A Figura 17 mostra alguns exemplos de placas de sinalização do BRTSD onde o contorno é alterado porque parte do segmento de cor vermelha do objeto está sob oclusão ou danificado. A aproximação convexa e fechada desse contorno pode ser uma solução, no entanto, a similaridade desse contorno com o formato ideal depende estritamente do tamanho da área de oclusão ou deformação.

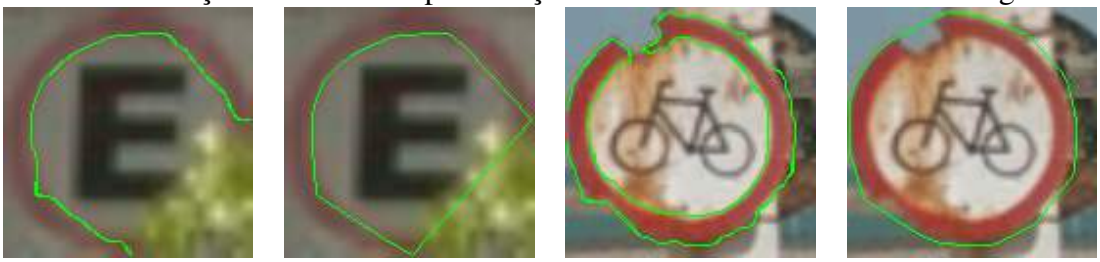
Outro problema para se obter as FDs está no fato de que a segmentação pode incluir

Figura 16: Destaque para variações no comportamento dos Descritores de Fourier no intervalo entre $[4, 9]$.



Fonte: Elaborado pelo autor.

Figura 17: Objetos sob oclusão parcial ou danificados tornam a obtenção dos FDs mais difícil. Uma solução é obter uma aproximação convexa fechada do contorno original.



Fonte: Elaborado pelo autor.

regiões de fundo da imagem quando estas possuem uma cor similar à cor de interesse (como acontece no exemplo de segmentação para cor amarela da Figura 11). Isso faz com que toda aquela região seja considerada como um único objeto, deformando o contorno externo e inviabilizando o uso dos Descritores de Fourier.

Apesar dos testes indicarem, portanto, que esse método não possui a robustez que é objetivo desse trabalho, os problemas apresentados inspiraram o estudo de uma nova abordagem para o problema de extração de regiões de interesse da imagem, apresentada na próxima seção. A ideia é permitir que os píxeis recebam graus de certeza quanto à sua cor, utilizando as áreas de cromaticidade introduzidas por Vitabile *et al.* em 2002. Com isso, o objetivo é permitir a extração de objetos mesmo que eles estejam sob oclusão parcial, danificados ou em uma região de baixo contraste.

4.3 Segmentação Fuzzy e Regiões de Estabilidade

Como já discutido, limiarização é a abordagem mais comum em sistemas TSDR que realizam segmentação baseada em cores e já foi aplicada em diferentes espaços de cores.

Apesar de ser o mais intuitivo, o espaço de cores RGB é também ou mais sensível aos distúrbios causados pela variação na luminosidade em ambientes externos. Por isso, como mostrou a seção 4.2, diferentes pesquisadas desenvolvem algoritmos de segmentação em outros espaços de cores (como HSV, CIELab, etc.).

Entretanto, quando sob a aplicação de um limiar – como o do pipeline anterior, geralmente as imagens coloridas são mapeadas para uma imagem binarizada (onde os píxeis com cor acima ou dentro do limiar são mapeados com intensidade máxima e o seu complemento é mapeado com intensidade mínima). Esse tipo de segmentação pode causar quebra de regiões se os limites de limiarização forem muito restritivos, fazendo com que o seletor não encontre prováveis candidatos ou elimine estes se a sua forma não estiver completa. Ainda, o abrandamento dos limites pode causar a seleção de muitos falsos candidatos ou agrupar regiões aos objetos de interesse.

Para solucionar alguns desses problemas, esse trabalho propõe realizar uma segmentação por filtros de cores, utilizando lógica Fuzzy e teoria de cromaticidade. Píxeis que estão dentro do limiar continuarão recebendo um valor máximo de segmentação. Porém, píxeis com coloração sobre a região de limiarização deverão receber valores de intensidade entre 0-255 de acordo com a sua proximidade a região cromática.

Esse método, chamado de Fuzzificação da Cromaticidade (FC), gera uma imagem em tons de cinza que destaca as regiões que possuem a cor de interesse com graus de certeza entre 0-255. As características dessa imagem permitem a utilização do algoritmo detecção chamado *Maximally Stable Extremal Regions* (MSER, em tradução livre Regiões Extremas Maximamente Estáveis), que fará a seleção de regiões conectadas com graus de fuzzificação parecidos. Detalhes das técnicas serão discutidos nas próximas subseções, seguidas de resultados e discussões.

4.3.1 Fuzzificação da Cromaticidade

Essa abordagem foi construída, novamente, no espaço de cores HSV. Entretanto, o framework não irá aplicar uma binarização na imagem por limiarização, mas sim um filtro de cores baseado em teoria Fuzzy. A técnica foi inspirada no método apresentado por De La Escalera *et al.* (DE LA ESCALERA; ARMINGOL; MATA, 2003) e tem a intenção de incluir regiões de cromaticidade instável e acromaticidade, descritas no estudo feito por Vitabile *et al.* (2002), à imagem segmentada, com diferentes tons de cinza.

Lógica Fuzzy, ou Lógica Difusa – termo encontrado em alguns livros traduzidos para o português, é uma forma de lógica multivalorada empregado geralmente para lidar com o conceito de verdade parcial, onde o valor de verdade pode variar entre completamente verdadeiro e completamente falso. Esses graus podem ser gerenciados por funções específicas (de Funções de Pertinência – FPs), que atribui graus de pertinência, tipicamente um número real no intervalo de $[0, 1]$, para os elementos de um universo (CINTULA; FERMULLER; NOGUERA, 2016).

O método (chamado Fuzificação da Cromaticidade – FC) propõe a utilização de Teoria Fuzzy para realizar a transformação de uma imagem RGB para um mapa em escala de cinza que destaca regiões com coloração de interesse, baseado na proximidade entre a matiz de um píxel e a matiz de busca. Essa proximidade é penalizada pela distância entre o píxel e a região cromática.

Por isso, são criados modelos de fuzzificação para cada uma das cores, levando em consideração o seu intervalo no canal de matiz e os limites de região cromática, cromática instável e acromática. Para cada píxel, os valores de matiz, saturação e brilho recebem graus de pertinência de acordo com as FPs especificadas no modelo. A saída da segmen-

tação é, como descrito na Lógica Fuzzy, níveis de verdade no intervalo entre $[0, 1]$ que representam a certeza de que o píxel é da cor de intersse.

Uma Função de Pertinência é uma função característica que define como cada um dos pontos no espaço de entradas é mapeado para o espaço de graus de pertinência. Nesse trabalho, serão utilizadas as funções de formato Π, formato S e formato Z. Uma FP de formato Π é construída de acordo com a Equação (7) onde os parâmetros a e d são considerados "pés"da curva, enquanto os parâmetros b e c são os "ombros". O parâmetro x é a entrada que será mapeada para o universo de graus de pertinência.

$$f(x; a, b, c, d) = \begin{cases} 0, & x \leq a \\ 2(\frac{x-a}{b-a})^2, & a < x \leq \frac{a+b}{2} \\ 1 - 2(\frac{x-b}{b-a})^2, & \frac{a+b}{2} \leq x \leq b \\ 1, & b \leq x \leq c \\ 1 - 2(\frac{x-c}{d-c})^2, & c \leq x \leq \frac{c+d}{2} \\ 2(\frac{x-d}{d-c})^2, & \frac{c+d}{2} < x \leq d \\ 0, & x \geq d \end{cases} \tag{7}$$

Mandal *et al.* (2012) descreve a utilização de diferentes FPs para predição de dados de uma série temporal, indicando que a função de formato Π obteve o menor erro associado. Essa curva têm como principal característica um comportamento de afastamento lento a partir do nível difuso alto (valor verdade 1) até o centro da curva (valor verdade 0.5), onde começa a se aproximar mais rapidamente do nível difuso baixo (valor verdade 0). Esse comportamento é também o principal motivo para a escolha desse tipo de FP, já que permite que valores mais próximos da região de certeza de uma cor de interesse recebam graus de pertinência mais altos (próximos de 1), enquanto que o afastamento é punido com uma rápida aproximação do grau de pertinência 0.

Note que uma Função de Pertinência de formato Π pode ser representada como a multiplicação de uma FP de formato S (Equação (8)) e outra FP de formato Z (Equação (9)). Além disso, o complemento da função Π pode ser escrito como uma soma de funções de S e Z,.

$$f(x; a, b) = \begin{cases} 0, & x \leq a \\ 2(\frac{x-a}{b-a})^2, & a < x \leq \frac{a+b}{2} \\ 1 - 2(\frac{x-b}{b-a})^2, & \frac{a+b}{2} \leq x \leq b \\ 1, & x \geq b \end{cases} \tag{8}$$

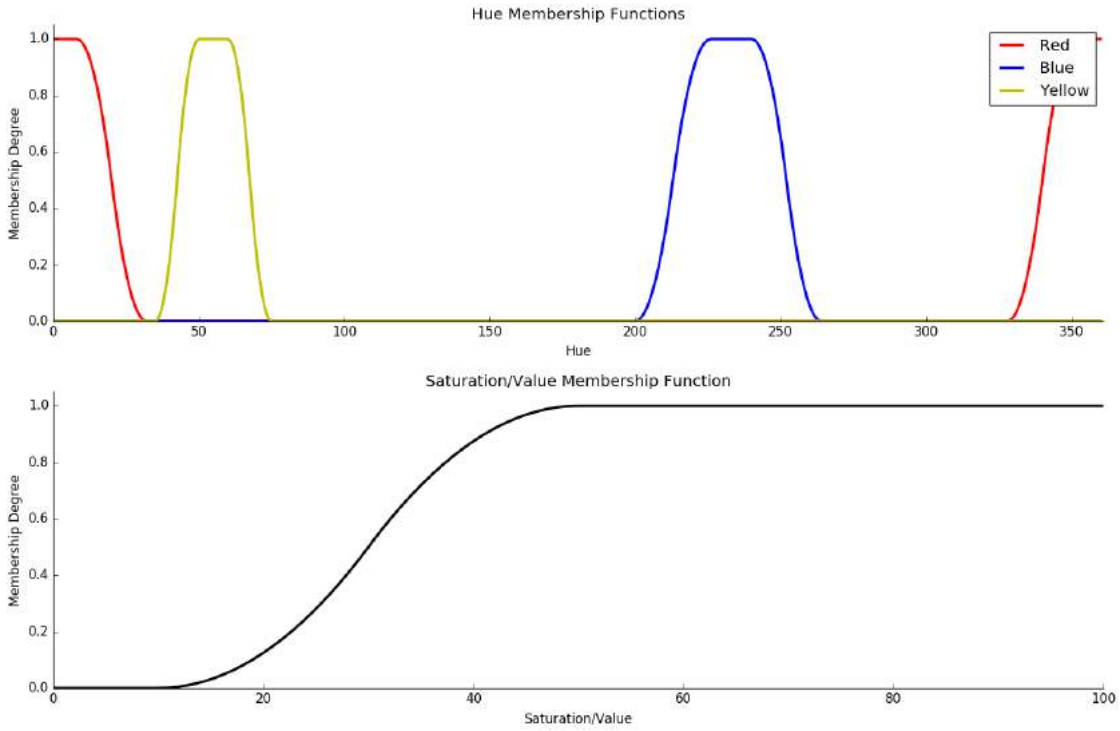
$$f(x; a, b) = \begin{cases} 1, & x \leq a \\ 1 - 2(\frac{x-a}{b-a})^2, & a \leq x \leq \frac{a+b}{2} \\ 2(\frac{x-b}{b-a})^2, & \frac{a+b}{2} < x \leq b \\ 0, & x \geq b \end{cases} \tag{9}$$

Essa definição é necessária definir os limites para fuzificação da matiz, saturação e brilho de um píxel. As cores predominantes nos sinais de trânsito – vermelho e amarelo no Brasil e, vermelho e azul na Alemanha – são representadas por uma combinação de FPs em cada um dos canais do espaço de cores HSV.

Uma cor de interesse possui uma matiz correspondente, informação que é utilizada para centralizar a Função de Pertinência de formato Π. Como o intervalo de H da cor

vermelha está presente no início e no fim do anel de matiz, essa cor é representada, portanto, pela soma de uma curva S e uma curva Z. Já os canais de saturação e brilho foram modelados com funções de formato S. A Figura 18 apresenta as funções de pertinência para cada uma das cores de interesse (vermelho, amarelo e azul) e suas respectivas matizes no universo H do espaço de cores HSV, além da curva para fuzzificação das informações de S e V.

Figura 18: Funções de pertinência para fuzificação das informações de Matiz, Saturação e Brilho de um píxel.



Fonte: Elaborado pelo autor.

O algoritmo permite uma flexibilidade nos limites para cada uma das funções de pertinência, tornando possível realizar ajustes para aumentar ou diminuir a abrangência de cores e tons que são segmentadas pelo modelo e seus graus de pertinência.

Após a fuzzificação, o modelo emprega a regra de defuzzificação (Equação (13), definida empiricamente). O relacionamento entre os canais é feito ao obter o grau de pertinência h (Equação (10) – a partir da Equação (7) ou (8)+(9), para segmentação da cor vermelha) e os graus de pertinência s (Equação (11)) e v (Equação (12)) – utilizando a Equação (8) – para as informações de matiz, saturação e brilho de cada um dos píxeis.

$$h = HueMF(x[i, j, hue]; h_{limits}) \tag{10}$$

$$s = SaturationMF(x[i, j, sat]; s_{limits}) \tag{11}$$

$$v = ValueMF(x[i, j, val]; v_{limits}) \tag{12}$$

$$out[i, j] = h \times s \times v \tag{13}$$

O valor de $out[i, j]$, contido no intervalo $[0, 1]$, pode ser considerado como a probabilidade (ou grau de certeza) de um dado píxel ser da cor de interesse, o que permite que

alguns dos píxeis que flutuam nos limites das áreas cromática instável e acromática sejam representados com intensidades na saída. Além disso, o valor da saída pode ser representado com oito bits num intervalo entre 0 e 255, afim de facilitar a visualização da imagem em tons de cinza.

A Figura 19 apresenta exemplos de segmentação utilizando o método FC para as cores vermelha e amarela em cenários de trânsito do Brasil, enquanto que a Figura 20 mostra o algoritmo FC sendo aplicado para as cores vermelha e azul em cenários de trânsito da Alemanha.

Figura 19: Exemplos de segmentação por FC em cenários de trânsito do Brasil. Para cada cenário é apresentada sua imagem segmentada com valores de intensidade pintados da cor de interesse, para fins de ilustração.

(a) Segmentação da cor vermelha.

(b) Segmentação da cor amarela.

Fonte: Elaborado pelo autor.

A principal diferença entre o algoritmo FC e segmentação por binarização é a inclusão de regiões de cor similar à cor de interesse com graus de certeza. Para ilustrar o ganho que essa nova abordagem pode acrescentar, a Figura 21 apresenta um exemplo de segmentação aplicada à uma região da imagem que possui a borda de uma placa triangular alemã. Nesse caso é possível notar que parte da placa está sob incidência de luz, alterando a cor perceptível daquele segmento. Na Figura 22(b) é possível notar que essa região foi ignorada pela segmentação por Limiarização, enquanto que a Figura 22(c) mostra que, na segmentação por FC, essa região foi incluída à imagem de saída, mesmo que com tons de cinza de menor intensidade.

Apesar de ser modelado no espaço de cores e melhor visualizado no espaço de cores HSV, os píxeis de uma imagem RGB não precisam ser convertidos para passarem pelo

Figura 20: Exemplos de segmentação por FC em cenários de trânsito da Alemanha. Para cada cenário é apresentada sua imagem segmentada com valores de intensidade pintados da cor de interesse, para fins de ilustração.

(a) Segmentação da cor vermelha.

(b) Segmentação da cor azul.

Fonte: Elaborado pelo autor.

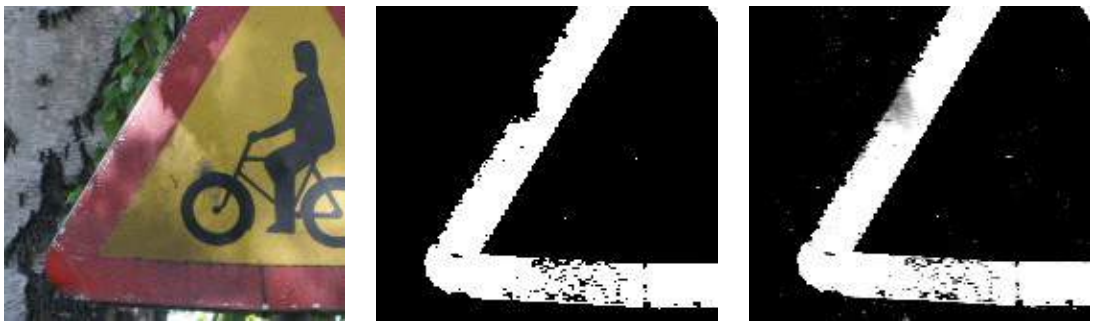
processo de fuzzificação e defuzzificação, já que as regras poderiam ser convertidas para o espaço de cores RGB.

Além disso, realizar todo o processo de inferência é relativamente complexo e se torna ineficiente efetuar todos os mapeamentos para cada píxel da imagem, a cada quadro processado. Esse requisito pode ser eliminado, já que após definidos os limites para as funções de pertinência, o processo Fuzzy não possui nenhum outro parâmetro adaptável ou dependente da entrada.

Ao manter as propriedades de inferência e os limites das curvas de fuzzificação, cada combinação de cores RGB terá apenas um grau de pertinência constante associado. Ou seja, se cada combinação RGB é mapeada para um único valor HSV e cada combinação de matiz, saturação e brilho possui uma probabilidade de ser da cor de interesse, então cada combinação RGB possui apenas um valor de saída. Dessa forma, todo o processo de fuzzificação, agregamento e deffuzificação pode ser simplificado utilizando tabelas de consulta (LUTs, do inglês *Look-Up Tables*).

Portanto calcula-se os valores de segmentação para todas as 256^3 combinações de RGB e armazena-se numa LUT de aproximadamente 16MBytes, para cada uma das cores. Assim, cada píxel pode receber um valor de segmentação ao acessar o índice da LUT correspondente ao RGB (por exemplo, píxel com $R = 200$, $G = 10$, $B = 60$ tem um valor de segmentação salvo na LUT[200][10][60]). Ao fazer isso, o tempo necessário para seg-

Figura 21: Comparação entre as segmentações por Limiarização e FC.



(a) Recorte parcial de uma placa de sinalização alemã.

(b) Segmentação por Limiarização.

(c) Segmentação por FC.

Fonte: Elaborado pelo autor.

mentação de uma imagem de 1680x1050 para as três cores predominantes foi reduzido para aproximadamente 32ms usando um computador pessoal, com sistema operacional Linux, processador i5-3570k (3.4GHz) e 16GB de memória RAM, com implementação em C++ sem a utilização de multiprocessamento. Além disso, a complexidade de espaço pode ser reduzida ao usar representações de cor com 7 bits (2MBytes), ou 6 bits (0.26MBytes), sem afetar drasticamente a eficiência do processo de inferência.

A técnica pode ser utilizada em qualquer outra aplicação que requer uma segmentação de imagens utilizando informações de cores dos objetos, já que é possível adaptar funções de pertinência em qualquer intervalo de matiz. Além disso, em sistemas com requisitos críticos de tempo, o acesso à LUT pode ser feito de maneira concorrente, permitindo aplicações de tempo real e o projeto de hardware específico.

4.3.2 Seleção de Regiões de Interesse

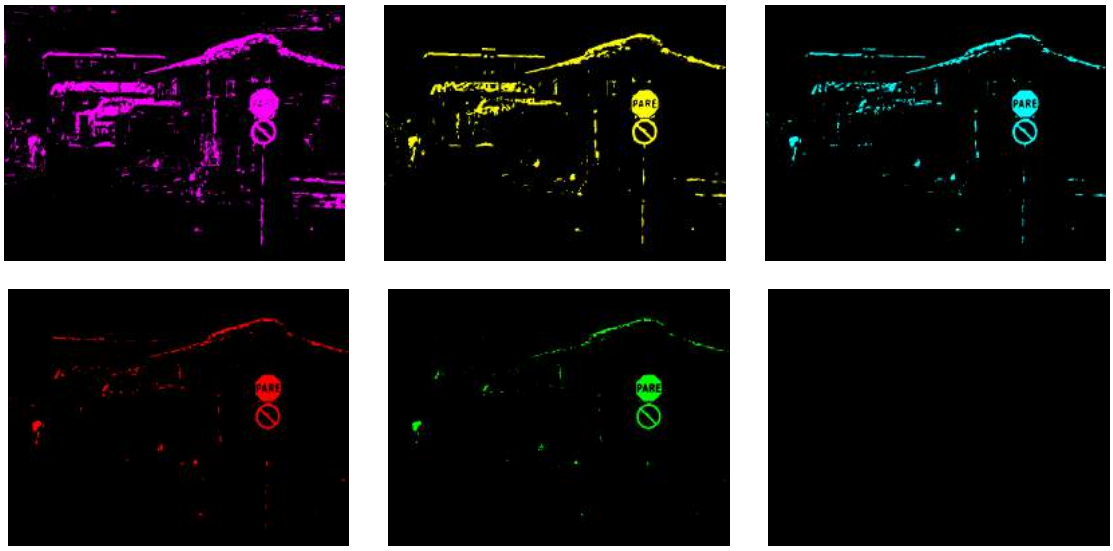
A imagem segmentada pela técnica de Fuzzificação da Cromaticidade permite destacar, com graus de certeza, as regiões que possuem propriedades de coloração parecidas com as cores predominantes na sinalização vertical de trânsito. Essa propriedade, no entanto, não garante que apenas píxeis que compõem placas de sinalização receberam valores de segmentação altos. Além disso, outros objetos podem estar conectados à uma Região de Interesse e uma mesma ROI pode ter píxeis de diferentes intensidades (variação nos graus de certeza de cada píxel).

Por isso, é necessário realizar a seleção de regiões conectadas analizando o perfil da vizinhança e o formato final do objeto extraído. Para realizar essa etapa, foi selecionado o algoritmo *Maximally Stable Extrema Regions* (MSER, em tradução livre Regiões Extremas Maximamente Estáveis), aplicado pela primeira vez na detecção de sinalização de trânsito em imagens por Larsson e Felsberg (2011) em conjunto com os Descritores de Fourier.

O MSER foi proposto por Matas *et al.* (2002) para encontrar correspondências entre elementos de duas imagens obtidas de diferentes pontos de vista de um mesmo objeto. Basicamente, o MSER é usado, em uma imagem em tons de cinza, para selecionar regiões conectadas que mantém seu formato estável durante um intervalo de níveis de intensidade definido por Δ. As Figuras 22 e 23 mostram os resultados de segmentação Fuzzy (apresentadas na Figura 19) binarizadas em diferentes limiares no intervalo entre $[10, 250]$, ilustrando o processo de modificação nas propriedades dos objetos em cada um dos graus

de pertinência de saída. Para cada um dos limiares, os objetos conectados podem ser considerados MSERs diferentes.

Figura 22: Imagem de segmentação em tons de cinza da Figura 20(a) binarizada em níveis linearmente espaçados entre no intervalo de $[10, 250]$, da esquerda superior para a direita inferior.



Fonte: Elaborado pelo autor.

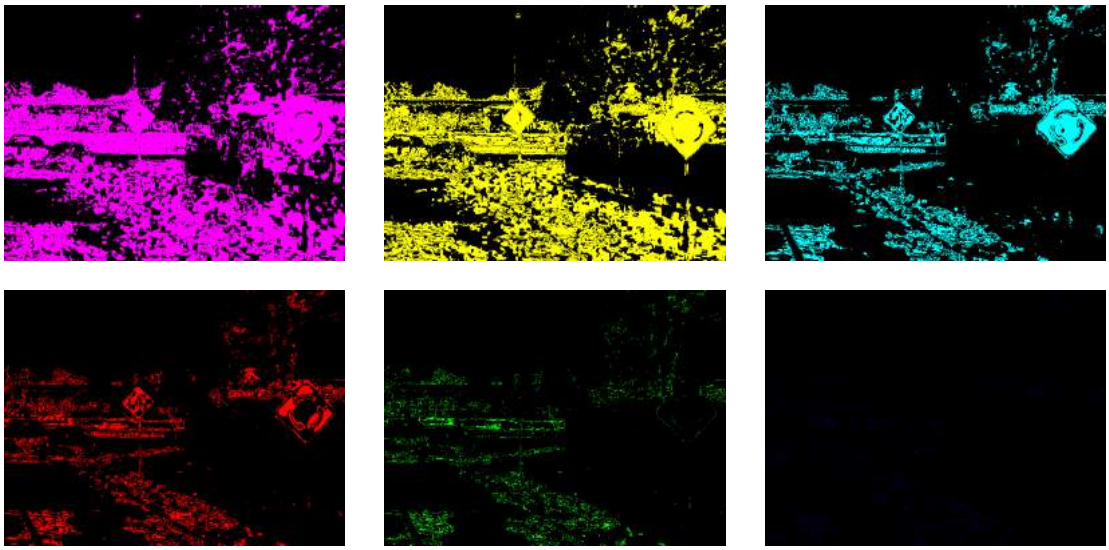
Sua compatibilidade nesse estágio do sistema se deve ao fato de que diversas regiões da imagem segmentada possuem estabilidade por diferentes intervalos de limiar no processo de binarização. Esse tipo de objeto possui as mesmas propriedades descritas como áreas de interesse do algoritmo:

- Invariância à transformações afins no domínio da intensidade;
- Covariância à preservação de adjacência;
- Estabilidade no domínio da intensidade, e;
- Detecção em múltiplas escalas.

No algoritmo original, o conjunto de MSERs pode ser listado utilizando uma implementação com complexidade computacional de ordem $\mathcal{O}(n\alpha(n))$, onde n é o número de píxeis na imagem e $\alpha(n)$ é o inverso da função de Ackermann ($\alpha(n) \leq 4$ em qualquer n prático) (MATAS et al., 2002). Inicialmente, a lista de píxeis é ordenada pela intensidade. Em seguida, os píxeis são adicionados à imagem seguindo a ordem, mas sem restrições de direção (crescente ou decrescente) e uma lista de componentes conectados e suas áreas é mantida usando um algoritmo de busca-e-união.

A quasi-linearidade do algoritmo garante eficiência na prática. Essa implementação (com variações nos algoritmos de ordenação e busca-e-união) é utilizada por diversas bibliotecas de processamento de imagem e visão computacional, como o OpenCV (www.opencv.org) ou VLFeat (www.vlfeat.org). A implementação da biblioteca OpenCV leva cerca de 260ms para extrair MSERs de uma imagem de 1680x1050 (1.764.000 píxeis) em um computador pessoal, com sistema operacional Linuxe processador i5-6600k (4GHz).

Figura 23: Imagem de segmentação em tons de cinza da Figura 20(b) binarizada em níveis linearmente espaçados entre no intervalo de $[10, 250]$, da esquerda superior para a direita inferior.



Fonte: Elaborado pelo autor.

A técnica possui diversas similaridades com o algoritmo de segmentação Watershed por imersão apresentado por Vincent e Soille (1991) – também conhecido como Método das Linhas Divisoras de Água (LDA). A comparação entre os dois trabalhos é utilizada por Nistér e Stewénius (2008) para apresentar alterações no método original que reduziram o custo computacional com uma implementação de ordem $\mathcal{O}(n)$, mantendo diversas propriedades do algoritmo original.

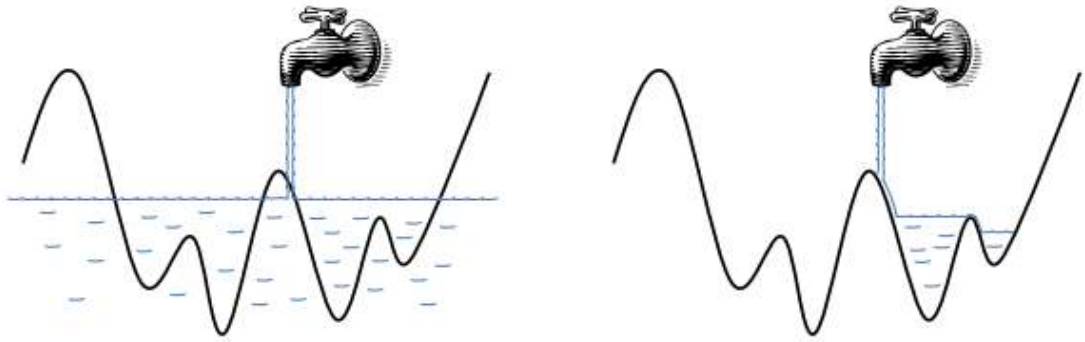
Análogo ao método de imersão, o perfil de tons de cinza da imagem pode ser comparado ao mapa de altitudes de uma paisagem de campo. Ao abrir vazão, o nível de água cresce gradualmente até que toda a paisagem esteja coberta. No algoritmo original, o nível sobe igualmente em toda a imagem, como se a água estivesse entrando por baixo.

Porém, a técnica apresentada em 2008 possui um perfil de inundação. Ao invés de imergir a imagem em um recipiente com o mesmo nível de água em todos os lugares, a inundação ocorre como se a paisagem fosse opaca e a água fosse despejada em algum ponto arbitrariamente selecionado (píxel). A água enche primeiramente a bacia onde a água é despejada sobre. Se a água continua a correr, a região então transborda sobre a outras partes da paisagem. A Figura 24 ilustra o exemplo utilizado.

A implementação de Nistér e Stewénieus mantém o controle da "corrente descendente"de água, registrando tanto os componentes conectados quanto os níveis de cinza na imagem. As regiões que forem preenchidas pela enchente são consideradas MSERs. Ao considerar que a diferença de complexidade recai basicamente ao valor de $\alpha(n)$, esse algoritmo pode realizar até 4 vezes menos operações e, utilizando a mesma configuração testada anteriormente, foi capaz de extrair regiões em apenas 72ms.

Greenhalgh e Mirmehdi (2012) utilizaram o MSER sobre a imagem em tons de cinza do cenário de trânsito, para extrair representações de sinalização de trânsito que continham fundo branco, e sobre as imagens normalizadas nas cores vermelha e azul para placas com fundo vermelho e azul, respectivamente. Entretanto, essa abordagem gera um número elevado de regiões selecionadas, por não utilizar nenhum tipo de passo prévio para restringir o universo de busca.

Figura 24: Analogia de imersão e inundação para exemplificar as duas implementações do MSER.



Fonte: NISTÉR; STEWÉNIUS (2008)

Em 2015, Yang *et al.* descreveram a utilização do MSER sobre a imagem de saída do processo de segmentação baseado em cores. O trabalho introduziu a criação de um mapa de probabilidades para cada cor de interesse no qual são selecionadas as MSERs, já que que o método aumenta o contraste entre os objetos segmentados e o plano de fundo, melhorando a precisão e facilitando a extração das ROIs, ao mesmo tempo em que diminui o número de objetos selecionados (YANG et al., 2016).

Da mesma forma, o MSER é aplicado na imagem segmentada pelo nosso algoritmo. As regiões selecionadas são então pré-filtradas quanto à proporção de tela (que deve ser menor que 2:1).

4.4 Resultados

Essa seção irá apresentar alguns parâmetros de implementação dos algoritmos utilizados, bem como sua validação utilizando duas das bases de dados já apresentadas: BRTSD e GTSDB.

O Dataset de imagens de trânsito brasileiro, criado no Laboratório de Processamento de Sinais e Imagens do Departamento de Engenharia Elétrica da UFRGS e batizado com o acrônimo de BRTSD (*Brazilian Traffic Sign Dataset*), contem 2,112 imagens em resolução WSXGA+ (1680×1050) obtidas manualmente utilizando a ferramenta Street View do Google. As imagens contem cenários de trânsito urbano e rural de diferentes cidades do país, apresentando diferentes ambientes de iluminação natural e poluição visual, com objetos deformados tanto pela ação do homem, quanto pelo envelhecimento e qualidade do material utilizado na fabricação. Em contagem manual realizada por humanos, foram identificados 3,597 sinais de trânsito da classe nacional de regulamentação (placas vermelhas) e 544 sinais da classe nacional de advertência (placas amarelas), com tamanho mínimo de 12×12 píxeis.

Por ainda não possuir uma demarcação oficial completa, a base BRTSD será utilizada para validar de maneira qualitativa as opções metodológicas de extração de regiões de interesse. A primeira delas, utilizando uma segmentação discreta de cores (Segmentação por Limiarização) associada à obtenção de Descritores de Fourier não teve seus resultados computados, por apresentar ainda na concepção problemas que estão presentes na maioria dos casos encontrados nas imagens de trânsito brasileiras.

Os resultados obtidos também por Larsson e Felsberg, propondo uma combinação

mais robusta de Descritores de Fourier e Modelos Espaciais para todas os contornos de uma imagem de trânsito em tons de cinza, não competem com o estado-da-arte atual, chegando apresentar uma precisão de apenas 59.25% e taxa de recall 47.76% para a classe de placas de "Dê a Preferência"utilizando o conjunto de imagens de verão da base sueca (LARSSON; FELSBERG, 2011).

Essa abordagem inicial, no entanto, permitiu a criação da técnica de segmentação utilizando Lógica Difusa, chamada Fuzzificação da Cromaticidade, associada diretamente à seleção de objetos utilizando o algoritmo MSER. Os limites utilizados para as Funções de Pertinência que modelam cada uma das cores de interesse foram geradas após uma análise das regiões de interesse no Anel de Matiz e regiões de cromaticidade definidas por Vitabile *et al.*.

Pra realizar operações básicas com imagens, como a conversão entre espaços de cores, as implementações utilizaram funções da biblioteca de código aberto OpenCV. O espaço de cores HSV é representado no OpenCV em intervalos de $H \in [0, 180)$, $S \in [0, 256)$ e $V \in [0, 256)$. A cor vermelha está próxima aos ângulos iniciais/finais do Anel de Matiz, e por isso a função de formato Π que a descreve deve compreender essa região de certeza, além de assumir valores de incerteza próximos aos tons de roxo e laranja, por isso os parâmetros de Π para essa cor são: $(164, 174, 6, 14)$. A mesma regra foi utilizada para as cores amarelo e azul, que possuem como parâmetros para as suas funções Π os valores $(12, 18, 24, 32)$ e $(102, 116, 124, 138)$, respectivamente. Os parâmetros selecionados das FPs de formato S utilizadas para modelar os canais S e V do método se baseiam nas regiões de cromaticidade e, no intervalo utilizado pelo OpenCV, são $(50, 128)$ e $(0, 50)$.

O algoritmo MSER utiliza uma implementação de complexidade linear (NISTÉR; STEWÉNIUS, 2008)[1] como detector de regiões conexas e foi configurado com os seguintes parâmetros: $(1, 8e-5, 3e-2, 0.35, 0.7, False)$, que correspondem ao valor de Δ, área mínima em relação ao tamanho da imagem, área máxima, variação máxima, diversidade mínima e ordem da vizinha conectável. Ao utilizar um $\Delta = 1$, o algoritmo permite a extração de objetos em todos os níveis de cinza, com uma variabilidade máxima nos tons de cinza vizinhos de 35%. A diversidade determina a quantidade de área que determina se duas regiões próximas são o mesmo objeto, ou seja, quanto menor o valor de diversidade mínima, mais objetos conectados que ocupam a mesma área da imagem serão extraídos como MSERs diferentes. A ordem de vizinhança determina apenas se será utilizada uma Vizinhança-4 ($False$) ou Vizinhança-8 ($True$) para determinar se um píxel pertence ou não ao objeto.

Para uma analise quantitativa dos resultados do método proposto, a técnica foi validada na base imagens alemã para detecção. O GTSDB é um um conjunto de imagens de referência para o problema de detecção de sinalização de trânsito, sendo amplamente utilizado na literatura. A base contém 900 imagens de tamanho 1360×800 e anotação para 1.213 sinais de trânsito com tamanho variando entre 15×15 e 128×128 píxeis. O banco divide as placas em 43 classes diferentes, das quais 38 são coloridas (computando um número total de 1.083 objetos de coloração vermelha ou azul), sendo estas o foco desse trabalho (HOUBEN et al., 2013).

A tabela 2 apresenta uma comparação quantitativa para os resultados de extração utilizando o método FC+MSER no GTSDB, em relação ao método proposto por Yang *et al.* (2016), que possui uma abordagem bastante similar. A tabela apresenta o número médio de regiões extraídas para cada imagem (# Regiões), taxa de recall, número de falsos negativos (# FN), número de cenários testados da base (# Cenários) e performance.

[1] Disponível em https://github.com/idiap/mser

Tabela 2: Resultados para extração de regiões de interesse no GTSDB.

	# Regiões	Recall	# FN	# Cenário	Tempo(s)
(YANG et al., 2016)	325	99.5%	01	300	0.067
FC+MSER	54.42	94.0%	65	900	0.072

Fonte: Adaptado pelo autor.

Apesar de apresentar uma taxa de recall abaixo do resultado esperado, o teste foi realizado com todas as 900 imagens do GTSDB e apresentou uma taxa de regiões selecionadas por imagem aproximadamente 6 vezes menor que o número de seleções feitas pelo método apresentado por Yang *et al.*.

Reduzir o número de regiões selecionadas para cada imagem se torna importante nessa classe de métodos já que cada proposição precisa ser analisada a fim de determinar se aquela região é ou não uma placa de sinalização. No método proposto por Yang *et al.*, por exemplo, o sistema calcula o Histograma dos Gradientes (utilizando uma alternativa que leva em consideração os canais de cores do objeto) para cada uma das proposições e utiliza Máquinas de Vetores de Suporte para determinar se a região se enquadra em uma das três *superclasses* determinadas pelo GTSDB: proibição, perigo e obrigação. Depois disso, a técnica classifica as regiões nas classes coloridas do GTSRB utilizando Redes Neurais Convolucionais.

Esse trabalho propõe substituir a etapa de detecção (utilizando HOG+SVM) por uma arquitetura de Rede Neural Convolucional capaz de classificar diretamente as regiões selecionadas nas subclasses do GTSRB, no passo em que elimina os falsos positivos do estágio de extração utilizando uma classe negativa. Teoria, metodologia de implementação e resultados para o módulo de classificação serão apresentados no próximo capítulo.

A performance apresentada pela técnica de seleção automática de regiões de interesse compete diretamente com o estado-da-arte e a sua concepção permite implementações em hardware específico. O tempo médio de 72ms foi obtido em um computador pessoal, com sistema operacional Linux, processador i5-3570k (3.4GHz) e 16GB de memória RAM, com implementação em C++.

Para aumentar a precisão do modulo de extração, os limites para as Funções de Pertinência Fuzzy e os parametros do MSER podem ser ajustados para cobrir uma maior área no universo de cores e objetos. Entretanto, o abrandamento desses atributos podem aumentar o número de proposições, o que deve aumentar o tempo de processamento para cada imagem. Para concluir esse capítulo e ilustrar o funcionamento desse estágio, a Figura 25 apresenta extrações feitas corretamente nas cenas usadas como exemplo nas Figuras 19 e 20.

Figura 25: Regiões de interesse extraídas corretamente dos cenários apresentados nas Figuras 19 e 20.

Fonte: Elaborado pelo autor.

5 MÓDULO DE CLASSIFICAÇÃO

Até o momento, o sistema proposto ainda não é capaz de determinar se as regiões extraídas são, realmente, placas de sinalização de trânsito. Na verdade, a maioria dos candidatos pode ser considerada apenas regiões de fundo da imagem ou objetos que possuem cor e tamanho similares aos de interesse desse trabalho.

Como já discutido no Capítulo 2, para a maior parte da literatura, no entanto, o foco do estágio de classificação é identificar à qual classe pertence cada uma das placas de sinalização extraídas de uma imagem, que já foram confirmadas por um estágio prévio de detecção, responsável por eliminar os falsos positivos. O próprio *benchmark* para TSDR criado pelo grupo de Visão Computacional em Tempo Real do Instituto de Computação Neural da universidade alemã Ruhr-Universität Bochum, que suporta as bases de imagens GTSDB (HOUBEN et al., 2013) e GTSRB (STALLKAMP et al., 2012) apresentadas no Capítulo 3, separa o problema em detecção e reconhecimento, sendo que essa última etapa não leva em consideração a existência de regiões que não são placas de sinalização.

Para filtrar as regiões selecionadas em métodos de segmentação baseada em cores, alguns trabalhos recomendam a utilização de detectores de formato, utilizando as propriedades geométricas dos objetos de interesse para introduzir métodos de detecção de *corners* (ESCALERA et al., 1997), corelação entre templates (VITABILE; GENTILE; SORBELLO, 2002; HUYNH-THE; NGUYEN THANH; TRAN CONG, 2014), atributos de forma (SOENDORO; SUPRIANA, 2011; KHAN; BHUIYAN; ADHAMI, 2011; BUI-MINH et al., 2012; ABUKHAIT et al., 2012) ou combinações de features de dimensão reduzida como os Descritores Fourier e Modelos Espaciais (LARSSON; FELSBERG; FORSSEN, 2011).

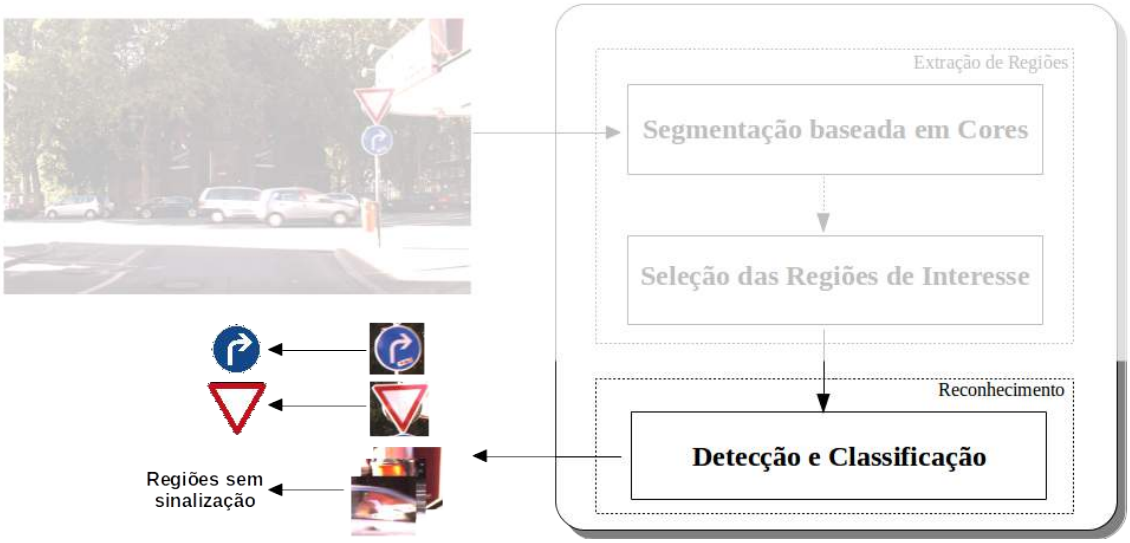
Entretanto, descritores de formato possuem uma alta taxa de erros em cenários complexos, podendo levar o sistema à classificações errôneas quando não estão previstos situações em que a sinalização de trânsito está sob oclusão parcial, danificada ou possui alguma similaridade com outros objetos do cenário. Por isso, trabalhos mais recentes têm apresentado a utilização de features de alta dimensionalidade como Viola-Jones (KELLER et al., 2008; CHEN et al., 2011), SURF (do inglês *Speed-Up Robust Features* (CHEN et al., 2011), descritores do tipo Haar (baseados na Transformada Wavelet) (KELLER et al., 2008) ou o Histograma dos Gradientes Orientados (HOG) (ZAKLOUTA; STANCIULESCU, 2011; YANG et al., 2016) como entrada para métodos de aprendizagem de máquina com a finalidade de confirmar se a região extraída é, ou não, uma placa de sinalização.

Apesar de terem sido bastante utilizados também na etapa de reconhecimento, o cálculo das desses descritores para todas as regiões antes da tomada de decisão pode ser uma tarefa de complexidade computacional elevada e faz com que a abordagem tenha uma performance reduzida, impedindo a sua utilização em sistemas de tempo real.

Yang *et al.* (YANG et al., 2016), entretanto, apresentou uma abordagem utilizando HOG+SVM para eliminar falsos positivos antes da etapa de classificação, que faz uso de uma Rede Neural Convolucional (CNN), representando um ganho de performance elevado em relação aos outros métodos. Porém, mesmo que utilizando um método de filtragem de falsos positivos, os autores descrevem a necessidade de incluir uma classe na CNN a fim de marcar todas as regiões que não contém placas de sinalização de interesse mas passaram despercebidas pelo passo intermediário.

Por isso, e considerando o avanço das pesquisas em relação aos métodos de *Deep Learning* – tradução livre Aprendizagem Profunda – e CNN já discutidos no Capítulo 2, o módulo de classificação desse trabalho propõe a eliminação de estágios de filtragem de falsos positivos ao realizar simultaneamente, para todas as MSERs selecionadas, a detecção e o reconhecimento utilizando uma CNN de tamanho reduzido (que será referenciada nas próximas seções como *short-CNN*) como mostra a região em destaque do fluxograma da Figura 26.

Figura 26: Estágios de reconhecimento no fluxograma proposto.



Fonte: Elaborado pelo autor.

5.1 Teoria de Deep Learning e Redes Neurais Convolucionais

Deep learning é uma das áreas mais recentes do campo da Aprendizagem de Máquina e Inteligência Artificial, onde modelos matemáticos são criados para que a máquina seja capaz de extrair padrões a partir de dados brutos para apresentar soluções em problemas específicos. Entretanto, para problemas complexos, um sistema de aprendizagem de máquina precisa reunir modelos menores para gerar um conceito de mais alto nível. Se esse processo de inferência puder ser descrito por um grafo profundo (com múltiplas camadas), então a abordagem é considerada *Deep Learning* (GOODFELLOW; BENGIO; COURVILLE, 2016).

Os exemplos mais clássicos de um modelo de *Deep Learning* são as Redes Profundas Feedfoward ou Perceptron de Múltiplas Camadas (MLP, do inglês *MultiLayer Perceptron*). Um MLP nada mais é do que uma função matemática capaz de mapear o conjunto de entrada para o conjunto de saída. No caso de Redes Profundas, essa função é formada pela composição de muitas funções mais simples, ou seja, cada função mais simples é

responsável por mapear o conjunto de entrada para uma nova representação, que é utilizada pela função seguinte, até que na função final a entrada é representada no conjunto de saíde desejado.

Redes Neural Convolucional é uma arquitetura especial de redes neurais descrita inicialmente por LeCun (1989) para o processamento de dados que possuem uma estrutura do tipo grade bem definida, como, por exemplo, imagens – considerada uma grade 2D de píxeis. Entretanto, na prática, CNNs só começaram a assumir o topo de performance nessa década, com o aumento no número de dados disponíveis e por consequência no volume de datasets. Exemplo disso é o ImageNet Large Scale Visual Recognition Challenge 2010 (ILSVRC2010), uma base de dados de imagens com mais de 1,2 milhão de imagens de alta resolução, separadas em 1.000 classes diferentes (RUSSAKOVSKY et al., 2015).

O método foi rapidamente adaptado a partir do problema de classificação de imagens (KRIZHEVSKY; SUTSKEVER; HINTON, 2012) para o reconhecimento de objetos, especialmente placas de sinalização de trânsito (CIRESAN et al., 2011; SERMANET; LECUN, 2011; CIRESAN et al., 2012; JIN; FU; ZHANG, 2014). Atualmente, CNN é o modelo mais popular de *Deep Learning* no processamento de imagens, já que a sua principal vantagem é que a entrada do modelo são os píxeis da imagem ao invés de utilizar *features* pré-calculadas e selecionadas manualmente.

A dependência de um número elevado de imagens para o treinamento tem sido a principal barreira para a obtenção de resultados mais precisos ou a sua utilização em diferentes frentes do problema de TSDR (como seleção de regiões). Zhu *et al.* (2016) discute uma abordagem *end-to-end* para realizar simultaneamente a detecção e o reconhecimento de sinais de trânsito, sem qualquer pré-processamento ou segmentação, utilizando uma CNN profunda. Para que o modelo pudesse obter resultados competitivos no reconhecimento, porém, os autores introduziram o Tsinghua-Tencent 100K Benchmark, uma base de dados 111 vezes maior que o GTSDB, com imagens de alta resolução de cenários de trânsito chines.

A principal diferença entre CNNs e MLPs está no uso de uma operação linear conhecida por Convolução no lugar da clássica multiplicação de matrizes. Em processamento de imagens, a operação de convolução ($*$) é utilizada principalmente em filtros lineares no domínio espacial. Gonzales e Woods (2008, p. 149) define uma convolução espacial (ou 2D) como sendo:

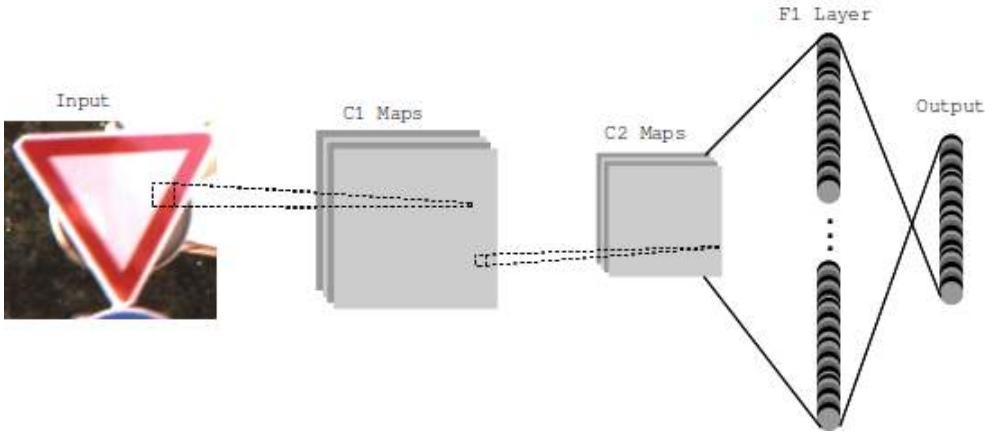
$$w(x,y) * f(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x-s, y-t) \tag{14}$$

e está sempre acompanhada por termos como máscara ou *kernel* de convolução, que denota a função $w(x,y)$ ou filtro de convolução. Os modelos de CNN que serão apresentados nas próximas seções para solucionar o problema de classificação são formados por uma sucessão de camadas de convolução e subamostragem conectadas à uma MLP, como ilustra a Figura 27 (com variações nos hiperparametros – tamanho das máscaras de convolução, número de camadas, número de neurônios, etc.). Essa arquitetura recebe como entrada uma imagem em RGB e fará o mapeamento para o conjunto de saída, que representa as probabilidades de a imagem pertencer à cada uma das classes.

5.1.1 Camadas de Convolução

Tipicamente, uma camada de convolução é composta de três estágios de processamento (Figura 28). No primeiro passo, a camada realiza diversas operações lineares de

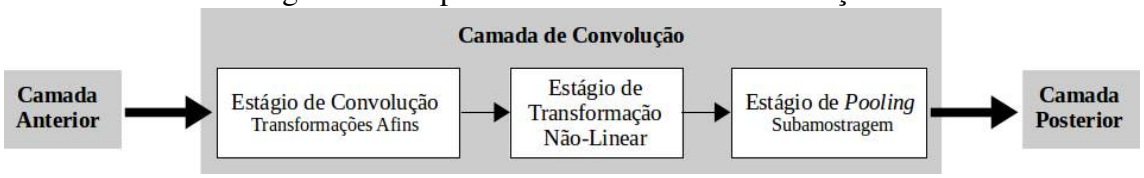
Figura 27: Arquitetura genérica de CNN utilizada no problema de classificação de sinalização vertical de trânsito.



Fonte: Elaborado pelo autor.

convolução sobre a entrada. Para produzir relações não-lineares, o resultado dessa operação é utilizado como parâmetro de uma função de ativação não-linear. O último estágio geralmente contem uma operação de *pooling* (ou mineração) utilizado para modificar a estrutura da saída – realizando, por exemplo, uma subamostragem.

Figura 28: Etapas de uma camada de convolução.



Fonte: GOODFELLOW; BENGIO; COURVILLE (2016).

5.1.1.1 Estágio de Convolução

Goodfellow *et al.* (2016) discutem três propriedades importantes presentes na operação de convolução que aumentam o interesse da comunidade científica pelas Redes Convolucionais: interações esparsas, compartilhamento de parâmetros e representações equivariantes.

Interações Esparsas: Sejam as camadas C_1 e C_2 subsequentes em uma rede neural artificial. Geralmente, redes tradicionais descrevem a relação entre suas camadas como uma multiplicação de matrizes: $C_1W \rightarrow C_2$, onde W é a matriz de pesos que conecta os neurônios de C_1 com os neurônios de C_2. Dessa forma, cada unidade presente na camada C_2 é influenciada por todas as unidades de C_1.

Entretanto, CNNs tipicamente possuem conexões esparsas ao se utilizar uma máscara de convolução menor que a entrada. Por exemplo, seja a imagem I (15) e a máscara W (16). O resultado da operação $W * I$ é uma matriz I' dada pelas Equações (17) e (18). Ou seja, cada unidade de I' (I'_{xy}) depende apenas da sua vizinhança de tamanho 3×3. A Figura 29 ilustra a primeira dessas propriedades.

$$I = \begin{bmatrix} i_{00} & i_{01} & i_{02} & i_{03} & i_{04} \\ i_{10} & i_{11} & i_{12} & i_{13} & i_{14} \\ i_{20} & i_{21} & i_{22} & i_{23} & i_{24} \\ i_{30} & i_{31} & i_{32} & i_{33} & i_{34} \end{bmatrix} \tag{15}$$

$$W = \begin{bmatrix} w_{00} & w_{01} & w_{02} \\ w_{10} & w_{11} & w_{12} \\ w_{20} & w_{21} & w_{22} \end{bmatrix} \tag{16}$$

$$C = W * I = \begin{bmatrix} c_{00} & c_{01} & c_{02} & c_{03} & c_{04} \\ c_{10} & c_{11} & c_{12} & c_{13} & c_{14} \\ c_{20} & c_{21} & c_{22} & c_{23} & c_{24} \\ c_{30} & c_{31} & c_{32} & c_{33} & c_{34} \end{bmatrix} \tag{17}$$

$$\begin{cases} c_{00} = w_{11}i_{00} + w_{10}i_{01} + w_{01}i_{10} + w_{00}i_{11} \\ \vdots \\ c_{22} = w_{22}i_{11} + w_{21}i_{12} + w_{20}i_{13} + w_{12}i_{21} + w_{11}i_{22} + w_{10}i_{23} \\ \vdots \\ c_{34} = w_{22}i_{23} + w_{21}i_{24} + w_{12}i_{33} + w_{11}i_{34} \end{cases} \tag{18}$$

Compartilhamento de Parâmetros: Sejam as mesmas camadas C_1 e C_2 e a sua conexão representada por $C_1W \rightarrow C_2$. Em redes neurais tradicionais, cada peso da matriz W está relacionado diretamente ao mapeamento de uma única unidade de C_1 para uma única unidade C_2, ou seja: $C_1(x)W(x,y) \rightarrow C_2(y)$.

Como mostram as Equações (17) e (18), os valores do resultado da convolução entre uma imagem I e uma máscara W compartilham entre si todos os pesos de W. Por isso, ao invés de aprender um conjunto de pesos para cada unidade da camada posterior, o mesmo conjunto (máscara) é utilizado para mapear todos os valores da entrada para a saída. Ao fazer isso, a complexidade computacional para o cálculo da camada C_2 a partir de C_1 não muda, entretanto a complexidade de espaço é reduzida, porque só é necessário armazenar a máscara W. A Figura 30 ilustra essa propriedade, utilizando uma flecha em destaque para representar o mesmo peso em ambos os casos.

Representações Equivariantes: Equivariância à translação é uma propriedade que ocorre por consequência do formato especial de compartilhamento de parâmetros da operação de convolução. Uma função $f(x)$ é dita equivariante à função g se $f(g(x)) = g(f(x))$.

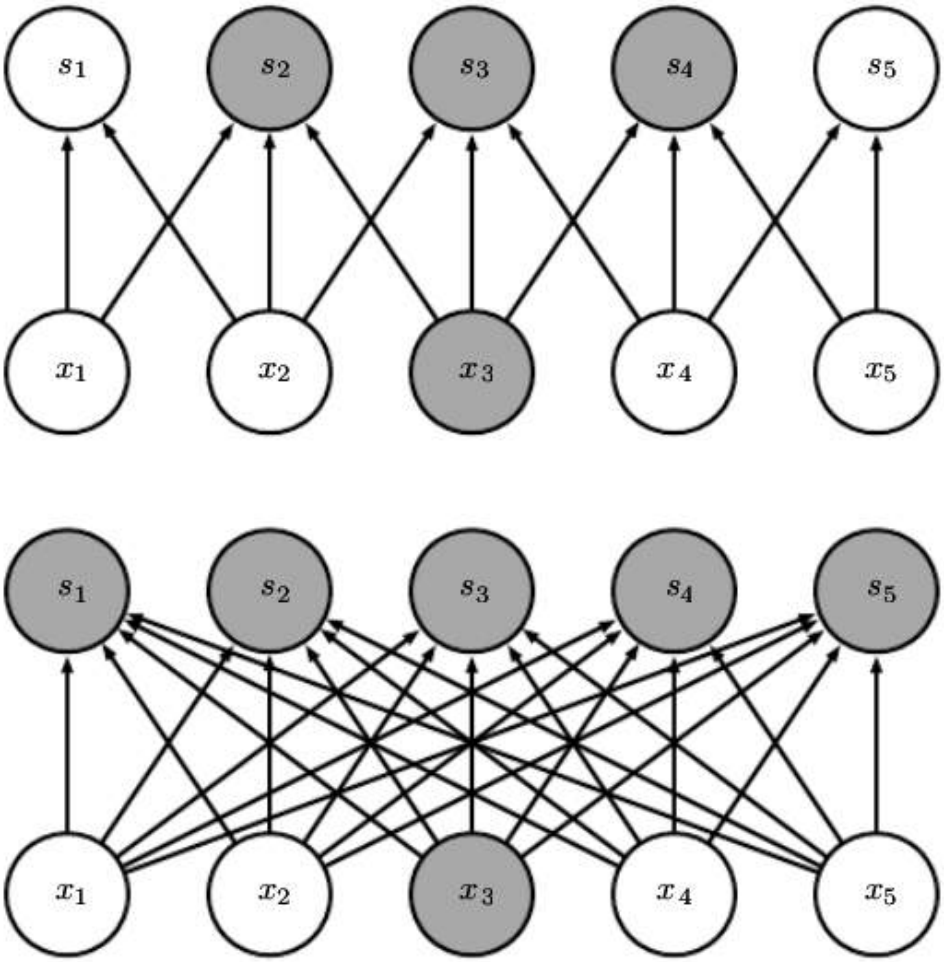
Seja g a função que mapeia $I \rightarrow I'$ pela operação de translação à direita $I'(x,y) = I(x-1,y)$. Convolução é equivariante à g, já que se aplicada essa transformação a I e depois a operação de convolução, o resultado será igual ao processo de convoluir I e só então aplicar a transformação g na saída (Equação (19)).

$$g(I(x,y)) * W = I'(x,y) * W = g(I(x,y) * W) \tag{19}$$

5.1.1.2 Estágio de Transformação Não-Linear

Convolução é uma operação linear sobre o operando, ou seja, esse estágio só é capaz de criar relações lineares entre a entrada e a saída. Caso uma rede seja construído utilizando apenas operações de convoluções (ou multiplicação de matrizes), o modelo estará

Figura 29: Exemplos de iteração esparsa (acima) e densa (abaixo). Em camadas conectadas totalmente, a unidade x_3 influencia todas as unidades posteriores, o que não acontece em camadas conectadas por convolução com uma máscara de tamanho inferior à entrada.



Fonte: GOODFELLOW; BENGIO; COURVILLE (2016, p. 336).

gerando combinações lineares de funções lineares de mapeamento da entrada para a saída. Problemas complexos, no entanto, requerem um modelo não-linear da solução. Essa propriedade é adicionada ao modelo utilizando o estágio de Transformação Não-Linear na Camada de Convolução (Figura 28).

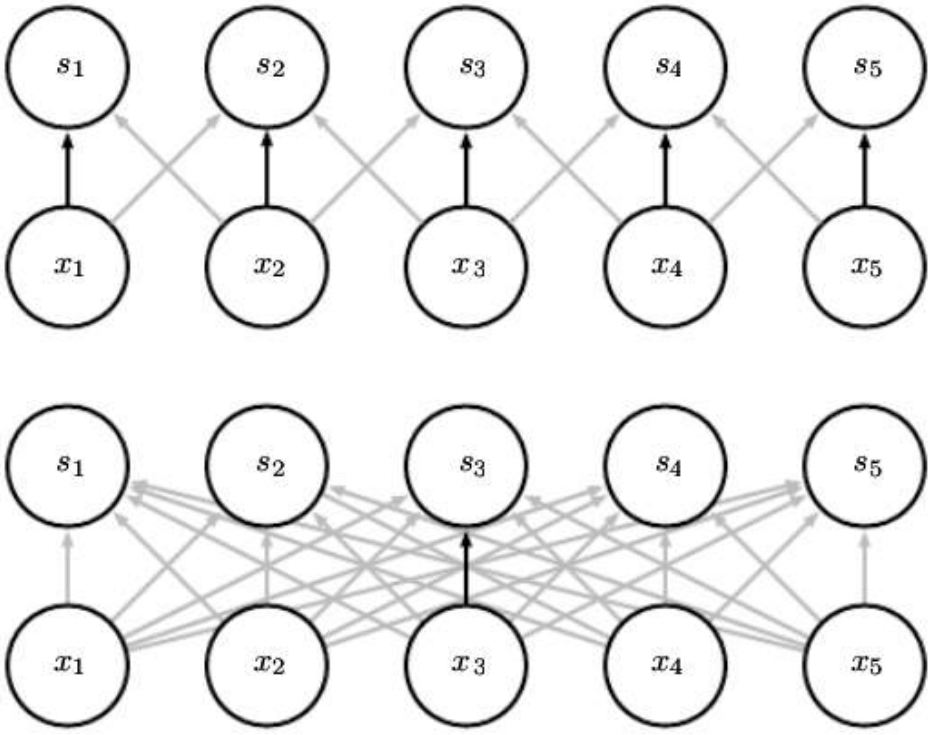
Também chamado de Estágio de Detecção (GOODFELLOW; BENGIO; COURVILLE, 2016, p. 339), essa etapa do modelo aplica funções não-lineares de ativação sobre o resultado da operação anterior. Arquiteturas tradicionais de MLP utilizam funções bastante difundidas na literatura: Sigmóide (Equação (20)) e Tangente Hiperbólica (*Tanh*, Equação (21)).

$$f(z) = \frac{1}{1 + e^z} \tag{20}$$

$$f(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} \tag{21}$$

Com o aumento exponencial no número de unidades e o concebimento do *Deep Learning*, Redes Neurais Convolucionais têm evitado a utilização de não-linearidades tradicionais, já que a sua suavidade faz com que os modelos demorem a convergir. A maioria dos

Figura 30: Exemplos de compartilhamento de parâmetros. Em camadas conectadas totalmente (abaixo), o peso (flecha em destaque) é responsável apenas por relacionar x_3 a s_3, o que não acontece em camadas conectadas por convolução (acima).



Fonte: GOODFELLOW; BENGIO; COURVILLE (2016, p. 338).

trabalhos atuais tem substituido essa operação pela função de retificação – batizado como *ReLU*, do inglês *Rectified Linear Unit* (Equação (22)). Segundo LeCun (2015), "ReLUs geralmente aprendem muito mais rápido em redes com muitas camadas, permitindo o treinamento de redes supervisionadas profundas sem nenhuma técnica de pré-aprendizado sem supervisão".

$$f(z) = max(0, z) \tag{22}$$

Além disso, a função de retificação linear mantém a propriedade de esparsidade da operação de convolução (GLOROT; BORDES; BENGIO, 2011). A esparsidade é respeitada quando $a \leq 0$, fazendo com que essa propriedade escale no modelo com o número de unidades de cada camada. A função Sigmoidal, por outro lado, tende a gerar valores não-nulos, resultando numa representação densa. Em 2015, essa era a função não-linear mais comum em *Deep Learning* (LECUN; BENGIO; HINTON, 2015).

5.1.1.3 Estágio de Pooling

Uma função de *pooling* é responsável por gerar a saída da camada de convolução por meio de um resumo estatístico sobre regiões das saídas dos estágios anteriores. Segundo Goodfellow *et al.* (2016, p. 342), essa operação sobre regiões espaciais faz com que a saída seja uma representação invariante a pequenas translações na entrada. Ainda, *Pooling* sobre saídas de diferentes convoluções parametrizadas faz com que os parâmetros possam aprender para qual tipo de translação eles devem se tornar invariantes.

Uma das principais vantagem da CNN é a possibilidade de realizar as mesmas opera-

ções para entradas que são, geralmente, de tamanho diferente. Essa propriedade é alcançada com a operações de pooling de tamanho variável. Por exemplo, seja uma camada C contendo apenas a operação de pooling definida por

$$s = (max(Q_0), max(Q_1), max(Q_2), max(Q_3)) \tag{23}$$

onde s é uma tupla contendo os valores máximos dos quatro quadrantes de uma imagem ($Q_{[0,3]}$). A camada não terá problemas em realizar a operação e obter os valores de s mesmo que o tamanho das imagens de entrada sejam diferentes – imagens de 32×32 ou 128×128 continuam tendo quatro quadrantes e, por consequência, um valor máximo em cada um deles.

Na maioria dos trabalhos, esse estágio é utilizado para reduzir a dimensionalidade das camadas subsequentes, a fim de selecionar apenas os parâmetros mais importantes em tarefas de classificação e reconhecimento. Boureal *et al.* (2010) apresenta uma análise teórica sobre a operação de *pooling*, discutindo tópicos como, por exemplo, quais métodos utilizar. Mais tarde outros trabalhos apresentam alternativas para a seleção do método dinamicamente (BOUREAU et al., 2011) ou abordagens aprendizagem do método de pooling (JIA; HUANG; DARRELL, 2012; MALINOWSKI; FRITZ, 2013). Atualmente, a operação mais utilizada nessa etapa em CNNs é chamada *max-pooling*, dado pelo valor máximo em cada região espacial específica (LECUN; BENGIO; HINTON, 2015).

5.1.2 Hiperparâmetros e Otimização

A saída de uma camada de convolução é um conjunto de imagens, chamados mapas. Em ambos os modelos que serão apresentados nesse capítulo, a saída da última camada de convolução é rearranjada como um vetor unidimensional de neurônios e conectado – todos com todos – a uma MLP, cuja última camada implementa a função de ativação *softmax* (Equação 24, (BISHOP, 2006)), uma função exponencial normalizada geralmente utilizada para regressão logística de múltiplas classes.

$$P(C_k|\phi) = y_k(\phi) = \frac{e^{a_k}}{\sum_j e^{a_j}} \tag{24}$$

onde as ativações a_k são dadas por

$$a_k = w_k^T \phi \tag{25}$$

Bishop (2006) utiliza a Equação (24) para determinar a probabilidade de que a entrada pertença à classe C_k dado o vetor de features ϕ – valores de saída da camada de neurônios anterior – multiplicado pela matriz de pesos w_k, baseado no Teorema de Bayes.

A maioria dos modelos de *Deep Learning* utiliza como parâmetro de aprendizagem a otimização de alguma função. Otimização se refere geralmente à tarefa de encontrar uma solução aproximada que minimiza ou maximiza uma função $f(x)$ ao mover x no universo de entrada. A função à qual se deseja minimizar/maximizar é chamada função objetivo. Nesse problema, em acordo com outros problemas de classificação de imagens, o critério de aprendizagem será a minimização do erro médio na camada de saída. O cálculo do erro utiliza a função de entropia cruzada (BISHOP, 2006, p. 209).

Seja um conjunto de entradas e saídas dado por ϕ_n, T_n, sendo T_n uma matriz $N \times K$ contendo 1 na posição k da classe desejada e zeros nas outras $K - 1$ posições, para cada N. Seja também $y_{nk} = y_k(\phi_n)$, onde Y é a matriz $N \times K$ com os valores de *softmax*

calculados para cada classe em cada entrada ϕ_n. A função de erro por entropia cruzada é dada por

$$E(w_1, \ldots, w_k) = -\sum_{n=1}^{N}\sum_{k=1}^{K} t_{nk} \ln y_n k \tag{26}$$

e possibilitando o cálculo do erro médio por

$$\overline{E}(W) = \frac{E(W)}{K} \tag{27}$$

O problema de minimização de uma função compreende uma área gigantesca da matemática e a solução mais clássica para esse tipo de problema é chamado de Método do Gradiente, criada por Augustin Louis Cauchy em 1847 (GOODFELLOW; BENGIO; COURVILLE, 2016, p. 83). O algoritmo move x, com pequenos passos, na direção contrária do gradiente de $f(x)$ para encontrar o mínimo global. Porém, esse método encontra problemas na minimização de funções complexas (com muitos mínimos locais), efeito que pode ser reduzido com o uso de diferentes exemplos de treinamento, por consequência diferentes pontos de partida.

No treinamento de redes neurais que utilizam $min(\overline{E})$ como função objetivo, por exemplo, os pesos são atualizados por

$$w^{(\tau+1)} = w^{(\tau)} - \eta\nabla\overline{E}(W) \tag{28}$$

onde η a taxa de aprendizado, ∇ o gradiente da função e τ é o passo de treinamento. Quando essa técnica é utilizada para atualizar os pesos durante o processo de treinamento para cada unidade diferente do conjunto de treinamento, ela é chamada de Gradiente Descendente Estocástico (SGD, do inglês *Stochastic Gradient Descent*).

Kingma e Ba (2015) apresentam uma modificação para o método SGD, chamada *Adam*, capaz de calcular a taxa de aprendizado de maneira adaptativa durante o processo de aprendizado, utilizando os momentos do gradiente. Seja o gradiente $g_\tau = \nabla\overline{E}$ no passo de aprendizado τ, as estimativas do primeiro e segundo momentos ($\hat{m}_\tau$ e $\hat{v}_\tau$, respectivamente) são definidos por

$$\begin{aligned} m_\tau &= \beta_1 \cdot m_{\tau-1} + (1-\beta_1)\cdot g_\tau \\ \hat{m}_\tau &= \frac{m_\tau}{1-\beta_1^\tau} \end{aligned} \tag{29}$$

$$\begin{aligned} v_\tau &= \beta_2 \cdot v_{\tau-1} + (1-\beta_2)\cdot g_\tau^2 \\ \hat{v}_\tau &= \frac{v_\tau}{1-\beta_2^\tau} \end{aligned} \tag{30}$$

onde g_τ^2 é a potência de dois ponto-a-ponto de g_τ, β_1 e β_2 são hiperparâmetros configuráveis e β_1^τ e β_2^τ são β_1 e β_2 na potência de τ. O valor dos pesos é então atualizado de acordo com

$$w_t = w_{t-1} - \frac{\eta\cdot\hat{m}_\tau}{\sqrt{\hat{v}_\tau}+\epsilon} \tag{31}$$

onde η é a taxa de aprendizagem inicial, e ϵ outro hiperparâmetro utilizado para prevenir divisão por zero. Após vários testes, os autores recomendam a utilização de alfa $\eta = 0,001$, $\beta_1 = 0,9$, $\beta_2 = 0,999$ e $\epsilon = 10^{-8}$. Os valores de β são utilizados para balancear as influências do gradiente e dos momentos no passo dado em direção ao mínimo. Com

exceção de η, que foi adaptado durante os experimentos, o restante dos parâmetros foram mantidos por padrão.

O treinamento de modelos de aprendizagem de máquina é o estágio em que o algoritmo tenta, na busca pela solução ótima de uma função (nesse caso, o erro mínimo), se ajustar ao conjunto de dados que lhe é apresentado. Entretanto, esse modelo deve ser capaz de generalizar o universo de interesse para obter performance similar com dados que nunca lhe foram apresentados.

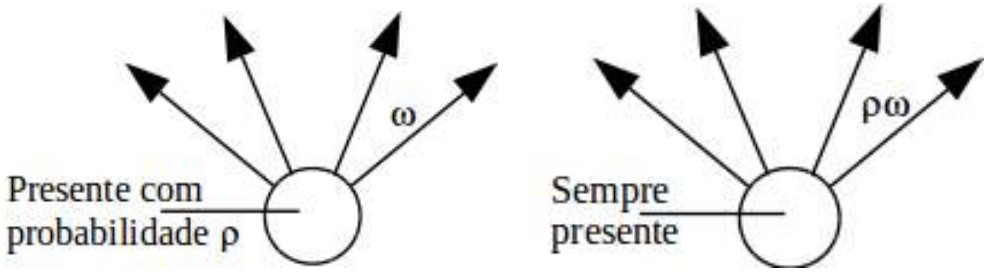
Quando um modelo se adapta demasiadamente ao conjunto de treinamento mas não é capaz de apresentar a mesma eficiência com um conjunto diferente de dados, ocorre o problema conhecido como *overfitting* (em tradução livre, sobreajuste). *Overfitting* ocorre porque o modelo passou a memorizar os dados de entrada, ao invés de aprender a generalizar o padrão de tendências do conjunto.

Diferentes técnicas tentam prevenir um modelo de se sobreajustar ao conjunto de entradas. Uma das principais é a produção de novos exemplos, realizando um aumento artificial no número de dados com operações que não modificam o padrão nos conjuntos mas é capaz de gerar unidades distintas para reforçar a variedade do conjunto.

Outros métodos trabalham sobre o modelo. Srivastava *et al.* (2014) apresentaram uma estratégia para previnir o *overfitting* em modelos de aprendizagem profunda. O *Dropout*, nome dado à técnica, aleatoriamente elimina unidades de algumas camadas da rede durante o treinamento, forçando que o modelo encontre novos caminhos para otimizar a função objetivo.

Na prática, a técnica distribui um valor de probabilidade para os neurônios da rede estarem ativos durante o treinamento. Ao encerrar esse estágio, o valor de probabilidade de cada unidade é repassado para as suas conexões, escalando o valor dos pesos, como mostra a Figura 31.

Figura 31: Técnica *Dropout*. **(a):** Durante o treinamento o neurônio possui uma probabilidade ρ de estar ativo. **(b):** No teste essa probabilidade é passada às conexões da unidade.



(a) *Droupout* durante o treinamento. (b) *Dropout* durante o teste.

Fonte: SRIVASTAVA; HINTON; KRIZHEVSKY; SUTSKEVER; SALAKHUTDINOV (2014).

Os modelos construídos para solucionar o problema de classificação de sinalização vertical de trânsito, ao qual se propõe esse capítulo, utilizam todas as técnicas apresentadas até o momento. As próximas seções vão apresentar alguns parâmetros de construção e decisões de implementação feitas durante os experimentos.

5.2 Experimentos e Resultados

CNNs são o estado-da-arte em diversos problemas de reconhecimento de imagens. Esse trabalho se propõe a apresentar dois modelos de Redes Neurais Convolucionais curtas para, concomitantemente, detectar e classificar placas de sinalização vertical em regiões de interesse extraídas de cenários de trânsito brasileiros e alemães.

Em ambos os experimentos que serão discutidos nas subseções a seguir, as estruturas de CNN recebem como entrada um recorte da imagem original contendo uma possível placa de sinalização. Essa região é uma MSER do algoritmo de extração proposto.

No entanto, MSERs não possuem uma caixa de seleção de contorno (*bounding box*) necessariamente quadrada. Sabe-se no entanto, que em situações ideais, uma placa de sinalização possui essa *bounding box* de lados iguais. Por isso a informação de proporção entre os lados de uma MSER é utilizada para realizar uma filtragem prévia. Caixas de seleção com um dos lados 2 vezes maior que o outro são eliminadas e não avançam para a classificação.

Para o restante das regiões, o lado maior é utilizado como métrica para todos os lados, permitindo a extração de caixas de seleção quadradas, mesmo que partes do ambiente ao redor da placa apareçam na imagem. Essa decisão permite que as CNNs ganhem robustez ao entorno do objeto de interesse durante o aprendizado.

As implementações foram feitas utilizando a linguagem de programação Python 2.7 e a biblioteca de código aberto para *Machine Learning* do Google, TensorFlow. Um número expressivo de arquiteturas, com diferentes configurações, foram gerados manualmente e testados, porém apenas os dois modelos melhor classificados serão discutidos nas próximas seções.

5.2.1 Reconhecendo Sinais do BRTSD

Como já discutido, o conjunto de imagens BRTSD ainda não possui anotações de classes e regiões da imagem que contém placas de sinalização. Durante a sua criação, foi possível notar que, além disso, as classes presentes na base de dados estão totalmente desbalanceadas. Isso significa que podem haver um número muito maior de objetos de uma classe específica em relação às outras.

Por isso, tomou-se a decisão de realizar uma macroclassificação desses objetos – como ocorre no problema de detecção do GTSDB – em seis superclasses definidas de acordo com a estrutura física da sinalização brasileira de trânsito, apresentadas na Figura 32.

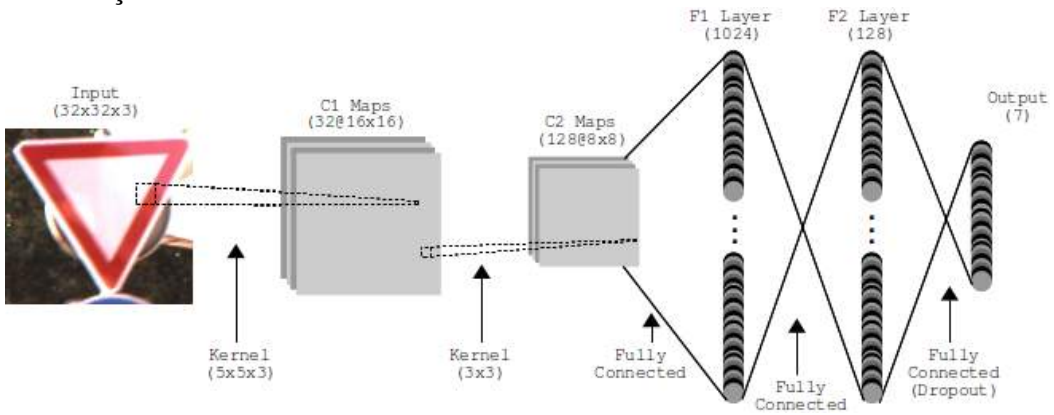
Figura 32: Superclasses do BRTSD.

Fonte: Elaborado pelo autor.

O modelo construído para esse problema combina duas camadas de convolução (com 32 filtros de $5 \times 5 \times 3$ e 128 filtros de 3×3 respectivamente) e operações *max-pooling* para regiões de 2×2. A saída da última camada de convolução é rearranjada e conectada (todos-com-todos) a um MLP com duas camadas ocultas de 1024 e 128 neurônios, respectivamente. Esta útlima camada mantém um processo de *dropout* e a sua saída é conectada à camada de classificação (contendo 7 neurônios, para as seis superclasses e uma classe negativa). A entrada é uma imagem RGB de 32×32, obrigando uma operação

de redimensionamento de todas as regiões extraídas. A Figura 33 apresenta o modelo, batizado como BRCNN.

Figura 33: Modelo de Rede Neural Convolucional utilizado no reconhecimento de placas de sinalização da base BRTSD.



Fonte: Elaborado pelo autor.

O algoritmo de extração foi capaz de selecionar 3.185 placas de sinalização corretamente. Para o treinamento, esse conjunto foi utilizado apenas para gerar um conjunto de 29.602 imagens artificialmente modificadas, com transformações aleatórias de brilho, contraste, rotação e translação. Uma média de 4.933 imagens em cada superclasse positiva. Para representar a classe negativa, 11.902 regiões que não continham placas mas foram extraídas pelo estágio anterior foram selecionadas.

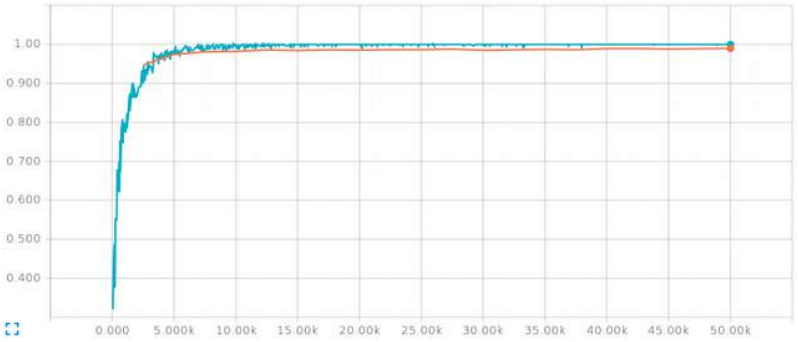
O treinamento, utilizando o algoritmo de otimização *Adam*, foi feito com 70% dessa base aumentada (cerca de 30.000 imagens – sendo os outros 30% para validação). Ruído branco foi adicionado aleatoriamente às imagens durante o treinamento. Rodando em um computador pessoal, com sistema operacional Linux, processador i7-3770K (4GHz), 32 GB de memória RAM e sem GPU, essa etapa levou 17 horas para realizar 50.000 rodadas de treinamento em conjuntos, também construídos de maneira aleatória, com 128 imagens cada. A evolução na acurácia e no erro do modelo nos conjuntos de treinamento e validação é apresentada na Figura 34.

Ao final, o modelo alcançou uma precisão de 99.77% na classificação de todas as regiões extraídas pela etapa anterior, com uma micro-média da área-sob-a-curva (AUC) de Precisão-Recall de 99.78% e uma macro-média de 94.75%. Dos 3.185 sinais de trânsito extraídos, 48 foram rotulados com a classe errada, e um falso positivo é classificado como placa de sinalização a cada 10 imagens do BRTSD. A Figura 35 apresenta a matriz de confusão normalizada para esse experimento, sendo as linhas representando a classe à qual a região pertence e a as colunas representando a classe na qual ela foi classificada.

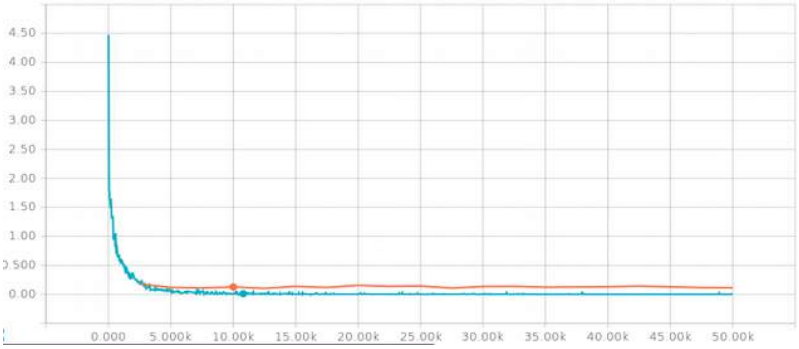
5.2.2 Reconhecendo Sinais do GTSDB

As bases GTSRB e GTSDB possuem 43 classes de sinalização alemã de trânsito, porém apenas 38 delas são coloridas e poderiam ser extraídas corretamente pelo estágio de seleção proposto (apresentadas na Figura 36). Por isso, como solução para o problema de reconhecimento do GTSRB (e como consequência, reconhecimento das regiões extraídas do GTSDB), decidiu-se aplicar um modelo CNN para, simultaneamente, detectar placas de sinalização e classificá-las corretamente em uma das classes coloridas, ao mesmo tempo em que elimina falsos positivos com uma classe negativa.

Figura 34: Gráficos de evolução da acurácia e erro do modelo BRCNN durante o treinamento para os conjuntos de treinamento (linha azul) e teste (linha alaranjada).



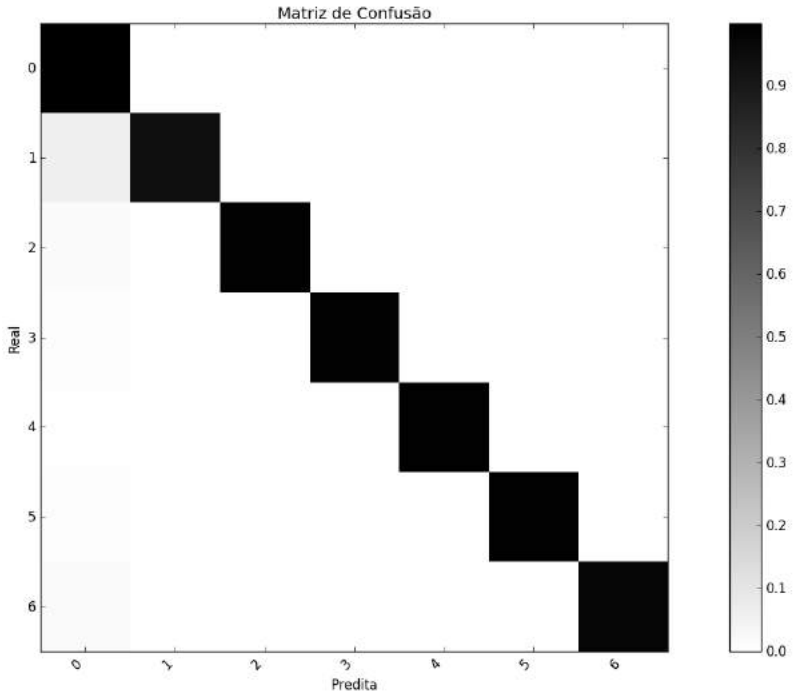
(a) Evolução da acurácia do modelo.



(b) Evolução do erro do modelo.

Fonte: Elaborado pelo autor.

Figura 35: Matriz de confusão para o modelo de classificação validado nas regiões extraídas da base BRTSD.



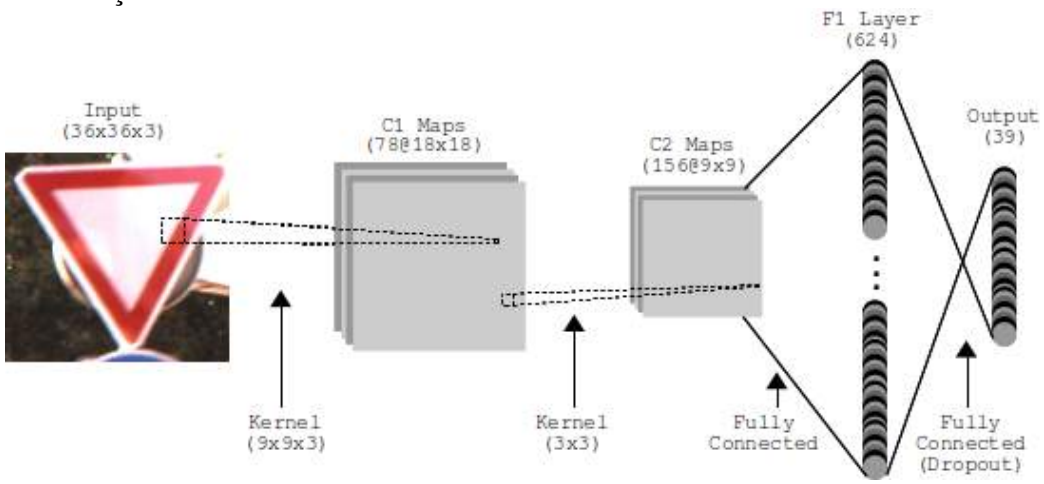
Fonte: Elaborado pelo autor.

Figura 36: Classes vermelhas e azuis do GTSRB.

Fonte: Elaborado pelo autor.

A CNN aplicada nesse caso é composta, também, por duas camadas de convolução (com 78 filtros de $9 \times 9 \times 3$ e 156 de 3×3, respectivamente) e operações *max-pooling* para regiões de 2×2. A saída da última camada de convolução é rearranjada e conectada (todos-com-todos) a uma única camada MLP de 624 neurônios. Esta camada mantém um processo de *dropout* e a sua saída é conectada à camada de classificação (contendo 39 neurônios, para as 38 classes coloridas GTSDB/GTSRB e uma classe negativa). A entrada é uma imagem RGB de 36×36, obrigando uma operação de redimensionamento de todas as regiões extraídas. A Figura 37 apresenta o modelo, batizado como GTSCNN.

Figura 37: Modelo de Rede Neural Convolucional utilizado para classificação de placas de sinalização na base GTSRB/GTSDB.



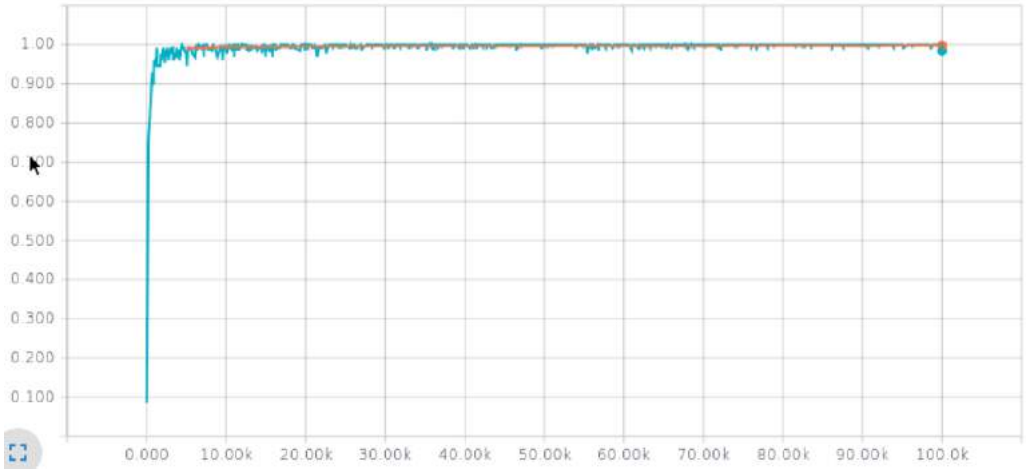
Fonte: Elaborado pelo autor.

O algoritmo de extração foi capaz de selecionar 1.018 placas de sinalização corretamente. Entretanto, nenhuma delas será utilizada para gerar o conjunto de treinamento. Em vez disso, foi utilizado o conjunto de treinamento da base GTSRB, que contem X imagens para as classes coloridas. Para corrigir as pequenas diferenças que podem desbalancear o treinamento, foram aplicadas transformações aleatórias de brilho, contraste, rotação e translação às classes, obtendo uma média de 2.088 imagens em cada superclasse positiva. Para representar a classe negativa, 5.553 regiões que não continham placas mas foram extraídas pelo estágio anterior foram selecionadas.

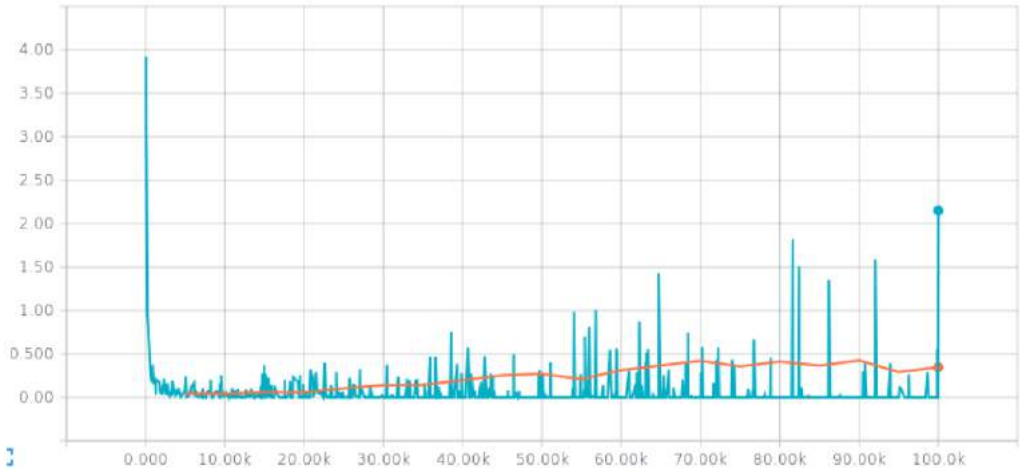
O treinamento, utilizando o algoritmo de otimização *Adam*, foi feito com 90% dessa base aumentada (cerca de 76.000 imagens – sendo os 10% para validação). Ruído branco

foi adicionado aleatoriamente às imagens durante o treinamento. Rodando em um computador pessoal, com sistema operacional Linux, processador i7-3770K (4GHz), 32 GB de memória RAM e sem GPU, essa etapa levou 23 horas para realizar 100.000 rodadas de treinamento em conjuntos, também construídos de maneira aleatória, com 128 imagens cada. A evolução na acurácia e no erro do modelo nos conjuntos de treinamento e validação é apresentada na Figura 38.

Figura 38: Gráficos de evolução da acurácia e erro do modelo GTSCNN durante o treinamento para os conjuntos de treinamento (linha azul) e teste (linha alaranjada).



(a) Evolução da acurácia do modelo.



(b) Evolução do erro do modelo.

Fonte: Elaborado pelo autor.

Para demonstrar a performance dessa abordagem, o modelo GTSCNN foi aplicado para as classes coloridas no conjunto de testes da base GTSRB. Essa base possui apenas regiões positivas (que representam placas de sinalização), mas que nunca foram utilizadas durante o treinamento. A Tabela 3 compara os resultados do nosso método para as 11.580 imagens das 38 classes coloridas do GTSDB com valores de AUC de precisão-recall de trabalhos estado-da-arte.

A Tabela 3 demonstra que essa abordagem possui resultados competitivos com o estado da arte e é capaz de classificar cada região em apenas 1.46ms. A Figura 39 apresenta a matriz de confusão normalizada para esse experimento, sendo as linhas representando a

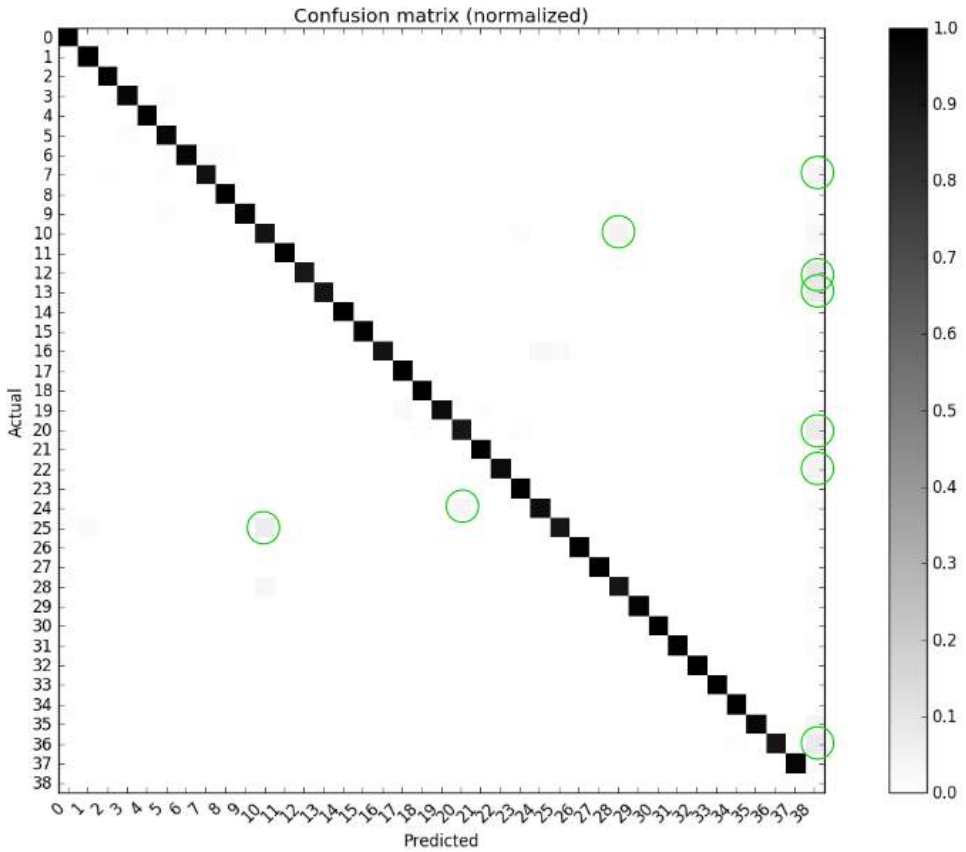
Tabela 3: Comaparação entre os resultados de classificação no conjunto de dados de teste da base GTSRB.

Método	Precisão	Tempo (ms)
(CIRESAN et al., 2012)	99.46%	11.4
(WANG et al., 2013a)	**99.52%**	40
(YANG et al., 2016)	97.75%	3
Proposto	97.13%	**1.46**

Fonte: Elaborado pelo autor.

classe à qual a região pertence e a as colunas representando a classe na qual ela foi classificada. As anotações, regiões em tom de cinza fora da diagonal principal, representam as 332 placas de sinalização que foram classificadas erroneamente (aproximadamente 2.87% do número de regiões coloridas do conjunto de testes da base GTSRB).

Figura 39: Matriz de confusão para o modelo de classificação validado no conjunto de teste da base GTSRB, com anotações de classificações errôneas visíveis.



Fonte: Elaborado pelo autor.

O objetivo principal dessa abordagem, porém, é comprovar a possibilidade de realizar tanto a detecção, quanto a classificação, sem precisar de nenhuma outro modelo intermediário para eliminar falsos positivos, reduzindo o custo computacional e a taxa de erro dos sistemas tradicionais. Por isso, esse modelo foi utilizado para classificar todas as 1.018 regiões extraídas corretamente do GTSDB, juntamente com os mais de 47.000 objetos falsos selecionados.

Antes de exibir resultados, é válido citar que o *benchmark* GTSDB é utilizado apenas

para o problema de detecção, considerado correto a partir do momento em que um modelo é capaz de dizer que a região selecionada da imagem é uma das superclasses do problema (proibição, perigo ou obrigação) ou não é uma placa de sinalização. Utilizando o modelo apresentado, essa resposta só pode ser obtida depois de classificar a região e uma das subclasses, já que a proposta é enviar diretamente todos as MSERs para a CNN que é otimizada para resolver um problema mais complexo que o de detecção.

Entretanto, para permitir uma comparação com outros trabalhos estado-da-arte, os valores de AUC de precisão-recall de cada subclasse do GTSRB foram agrupados nas superclasses do GTSDB. Apesar de não ser uma métrica favorável ao modelo proposto, demonstra que o algoritmo tem potêncial para competir com diversos trabalhos da literatura, mesmo que o foco destes seja resolver unicamente o problema de detecção. O número para classes que não pertencem a nenhuma das superclasses (outras placas do sistema alemão de sinalização de trânsito) também é agrupado e apresentado na Tabela 4.

Tabela 4: Comparação de resultados para o problema de detecção na base GTSDB.

	Proibitório	Perigo	Mandatório	Tempo (ms)
(WANG et al., 2013b)	100%	99,91%	100%	1122∼1232
(LIANG et al., 2013)	100%	98,85%	92%	400∼1000
(SALTI et al., 2013)	99,98%	98,72%	95,76%	571∼1667
(YANG et al., 2016)	99,29%	99,73%	97,62%	∼162
(LIU; CHANG; CHEN, 2014)	100%	99,20%	98,57%	∼192
(WANG et al., 2015)	99,87%	95,72%	91,14%	227∼301
Proposto	97,20%	92,44%	87,21%	∼151[1]

Fonte: Adaptado pelo autor.

Os resultados nessa métrica foram, claramente, influenciados pela taxa de falsos negativos durante a extração. Além disso, apesar de o modelo não ter sido treinado para macro-classificação nas superclasses do GTSDB (o que é, obviamente, uma tarefa mais simples), o método apresenta resultados competitivos e executa todo o processo em um tempo menor que qualquer outro na literatura.

Enquanto que o método proposto é capaz de extrair e classificar regiões em aproximadamente 0.15 segundos, Yang *et al.* (YANG et al., 2016) leva 0.162 segundos para realizar apenas extração e detecção, utilizando uma máquina equivalente (Intel 4-core 3.1 GHz CPU, 4G RAM). Os outro autores todos utilizaram um computador desktop high-end da época, (processadores Intel Quad Core) com pequenas variações de frequência de clock.

O sucesso do modelo é visualizado, porém, quando a técnica é aplicada para a finalidade à qual foi projetada: classificar as regiões selecionadas dos cenários de trânsito. O modelo obteve acurácia de 99.35% na classificação das mais de 40.000 regiões selecionadas nas 900 imagens do GTSDB, com uma taxa de precisão-recall de 99.68% (micro-média).

A GTSCNN foi capaz classificar as 1.018 placas de sinalização extraídas pelo estágio anterior, mesmo sem ter recebido nenhuma delas como instância de treinamento. Além disso, a técnica gera apenas um falso positivo a cada seis imagens do *benchmark*. Esse resultado mostra que, mesmo que treinado com as imagens geradas manualmente no GTSRB, o modelo conseguiu generalizar o problema e ser capaz de classificar os recortes positivos feitos automaticamente utilizando segmentação Fuzzy e o algoritmo MSER,

[1] Tempo de processamento do sistema todo.

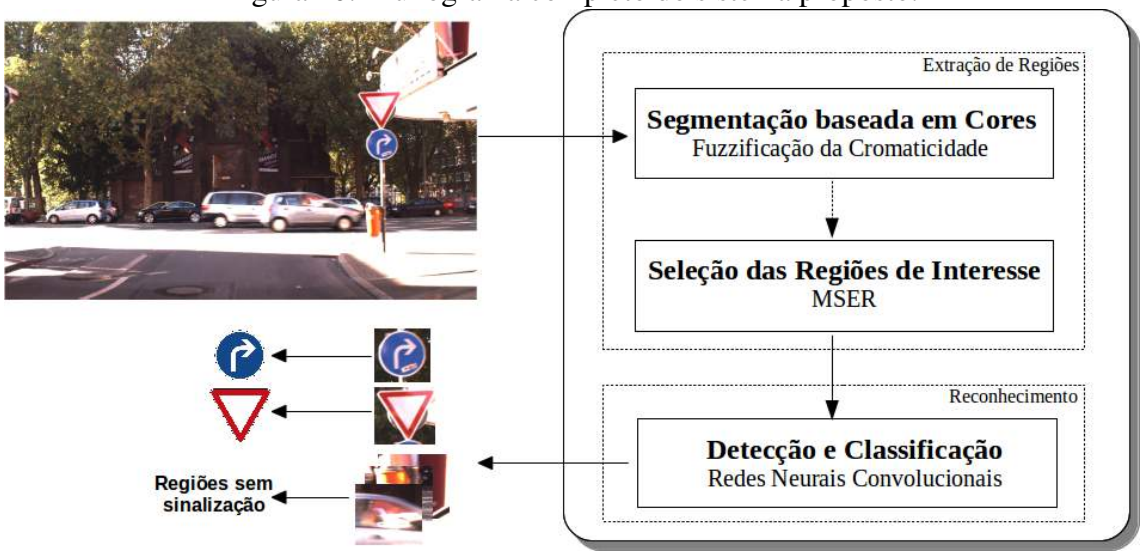
com uma precisão acima do que a abordagem proposta por Yang *et al.* (2016), enquanto realiza todo o processo em menos tempo.

6 CONCLUSÕES

Detecção e reconhecimento de placas de sinalização em imagens é um dos principais assuntos da literatura na área de Sistemas Inteligentes de Transporte. Máquinas capazes de identificar e compreender leis de trânsito podem ser o próximo passo para os Sistemas Avançados de Assistência ao Motorista e, com certeza, serão obrigatórios em veículos autônomos no modelo atual de sinalização.

Até o momento, a literatura tem apresentado diversos métodos para esse problema, com foco principal na seleção de regiões baseado nas cores e formas pré-estabelecidas dos objetos de interesse. Esse trabalho apresentou uma abordagem para ambos os problemas: detecção e reconhecimento, propondo um *dataflow* muito mais próximo de um modelo *end-to-end*, capaz de reduzir o custo computacional dessa tarefa (Figura 40).

Figura 40: Fluxograma completo do sistema proposto.



Fonte: Elaborado pelo autor.

O processo estuda diferentes abordagens de segmentação e inclui uma nova ferramenta de segmentação de imagens baseada em cores: a Fuzzificação da Cromaticidade. Essa técnica utiliza o conhecimento prévio do comportamento das cores de interesse em ambiente externo e a sua representação no espaço de cores HSV para determinar se um píxel possui aquela cor, atribuindo diferentes graus de certeza.

Considerado uma espécie de filtro de limiarização, esse método é facilmente modelado para qualquer problema baseado em cores e pode ser implementado utilizando tabelas de consulta. Dessa forma, é possível reduzir o custo computacional e até mesmo

projetar hardware específico para computação paralela. Além disso, a complexidade de espaço associada pode ser reduzida utilizando uma representação do espaço de cores RGB com um número menor de bits, sem afetar drasticamente a sua eficiência.

Ao gerar imagens em tons de cinza (representando os graus de certeza da cor de interese), essa técnica permite o uso do algoritmo MSER, capaz de selecionar objetos conectados em destaque em uma imagem. Com uma implementação de complexidade computacional linear, o MSER pode ser usado em aplicações de tempo real. Além disso, todo o processo de extração de regiões de interesse pode ser ajustado ao se modificar os parâmetros de configuração de ambos os algoritmos. Entretanto, o abrandamento desses parâmetros pode causar um aumento no número de regiões selecionadas por imagem, causando, por consequência, um aumento no tempo de processamento.

O *dataflow* se propõe a simplificar os sistemas de TSDR ao solucionar ambos os problemas de detecção e classificação utilizando uma única Rede Neural Convolucional curta, capaz de rotular os objetos selecionados ao mesmo tempo que exclui falsos positivos. A ideia é seguir em direção à uma aplicação totalmente *end-to-end* para veículos autônomos.

Para validar a abordagem, dois *benchmarks* foram utilizados. Esse trabalho apresentou o BRTSD, uma base com mais de 2.000 imagens de alta resolução retiradas manualmente pelo Google Street View, em diferentes cidades do Brasil. Apesar de não possuir uma anotação precisa, essa base serviu para demonstrar qualitativamente o funcionamento das técnicas. Para uma comparação quantitativa com outros trabalhos da literatura, foi utilizada as bases GTSDB/GTSRB, com cenários de trânsito e placas de sinalização alemãs.

É possível afirmar ainda que a base criada nesse trablaho se mostra mais desafiadora que o GTSDB, já que a maioria das placas de sinalização são geralmente menores que as observadas na base alemã. A Figura 41 compara os tamanhos das placas detectadas pelo método proposto com os tamanhos das placas presentes nas 900 imagens do GTSDB. Além disso, o conjunto cobre uma grande variedade de situações de iluminação, condições climáticas e degradação por envelhecimento ou vandalismo, além da poluição visual em centros urbanos.

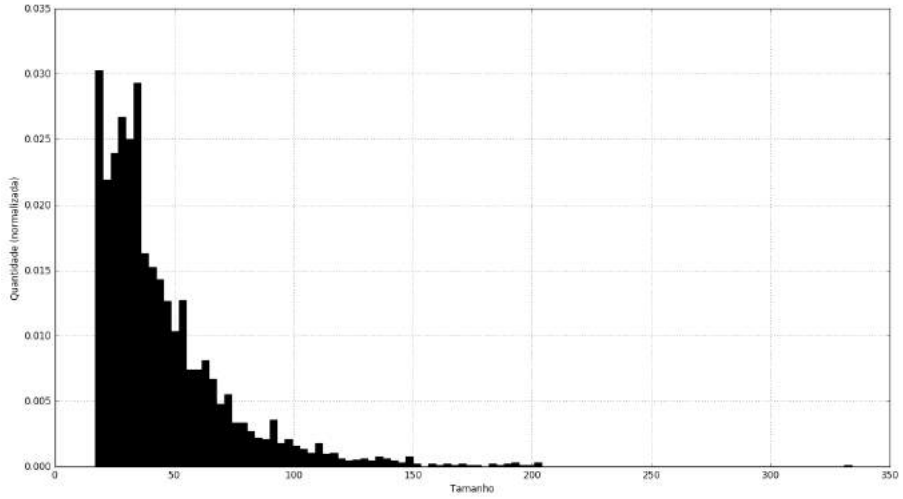
O método de extração proposto foi capaz de selecionar corretamente 94% das placas de sinalização presentes nas 900 imagens do GTSDB. Para cada imagem, esse estágio realiza a segmentação nas cores vermelha e azul e extrai as MSERs de cada uma das imagens segmentadas, em cerca de 70 milisegundos utilizando um desktop Linux.

Ambos os modelos de CNN apresentados para reconhecer placas de sinalização brasileiras e alemãs apresentaram alta eficiência. Dos 3.185 sinais de trânsito extraídos nas imagens do BRTSD, apenas 48 foram classificados erroneamente, enquanto que um falso positivo era rotulado a cada 10 imagens.

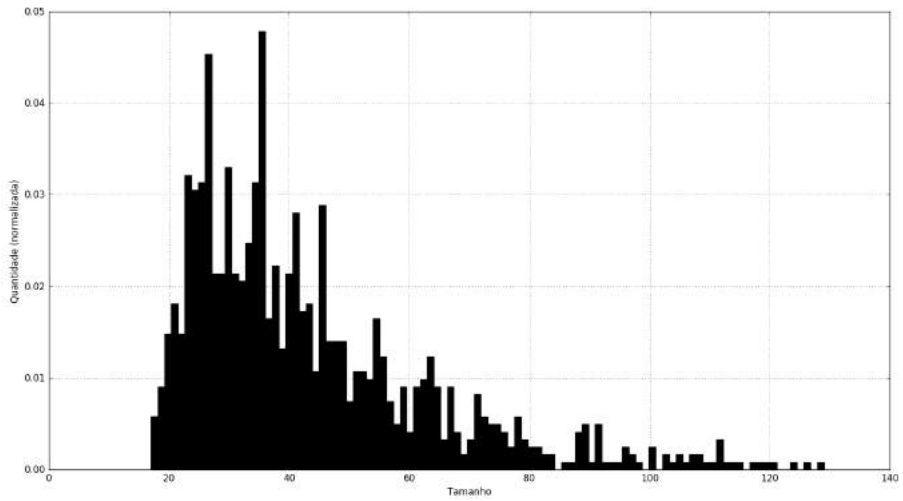
Para as classes coloridas do GTSRB, o modelo construído alcançou uma taxa de AUC de precisão-recall de 97.13%, sendo competitivo com o estado-da-arte enquanto é capaz de classificar um objeto a cada 1.46ms. Além disso, o modelo alcançou resultados quase perfeitos na classificação das regiões extraídas pelo estágio antecessor, sendo capaz de classificar todas as 1.018 placas de sinalização selecionadas, enquanto gerava, em média, seis falsos positivos a cada cenário.

Boa parte desse resultado se deve ao uso de técnicas do estado-da-arte para otimização e controle das camadas da CNN: *dropout* e *Adam*. Acredita-se que o número, já pequeno de falsos positivos, poderia ser reduzido ainda mais com uma abordagem de rastreamento, utilizando o resultado da classificação da mesma região em diferentes quadros como um *feedback* para o retreinamento online do modelo.

Figura 41: Histograma de tamanhos das placas das bases BRTSD e GTSDB.



(a) Placas da BRTSD.



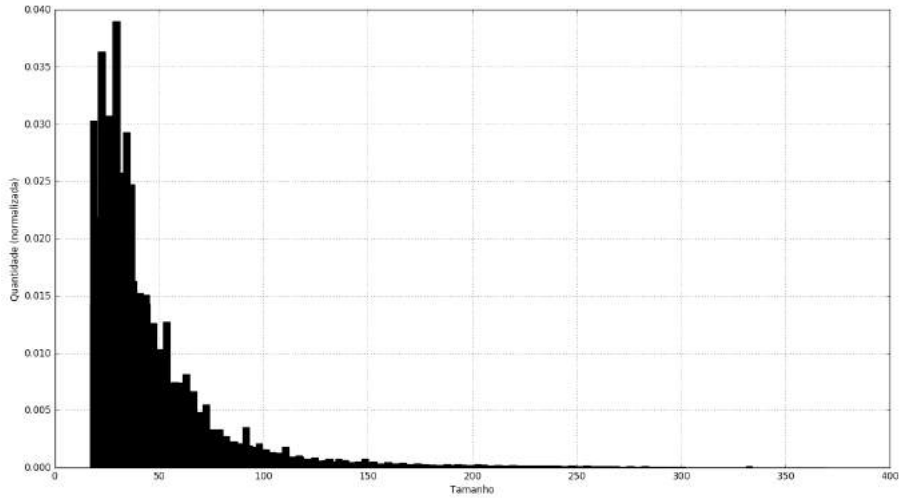
(b) Placas da GTSDB.

Fonte: Elaborado pelo autor.

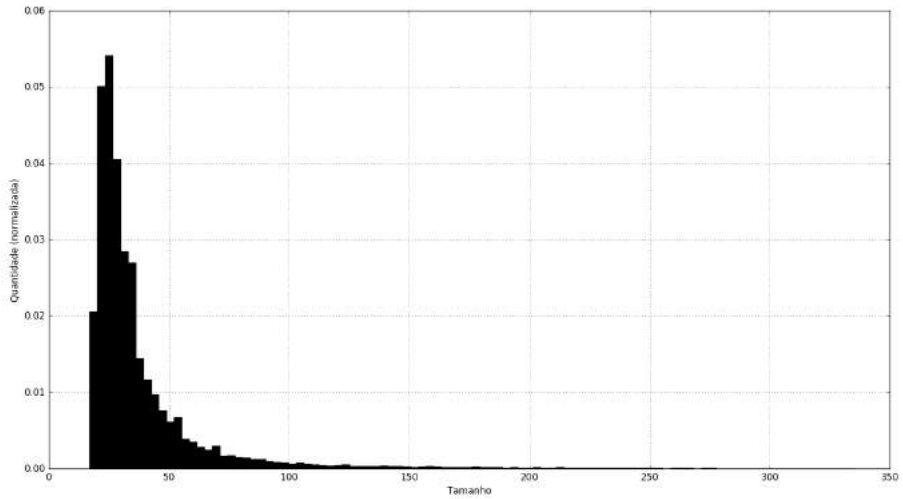
O tempo de processamento também pode ser reduzido com técnicas de redução de espaço de busca ou ao aumentar o tamanho mínimo das regiões de interesse. Dessa forma, o sistema apenas buscaria placas que estão mais próximas ao veículo, porém o número de MSERs extraídas seria reduzido drasticamente, levando a um número muito menor de regiões para classificação. A Figura 42 apresenta estatísticas de tamanhos de MSERs selecionados em ambas as bases BRTSD e GTSDB e que foram classificadas corretamente como não-placas. Cerca de 70% delas possui um tamanho de no máximo 36×36 píxeis no GTSDB, enquanto que na base brasileira, 70% das MSERs tem um tamanho de no máximo 45×45.

Desenhado com o intuito de integrar um sistema de tempo real, esse módulo (sem otimizações de implementação e multiprocessamento) é capaz de processar, em média, seis imagens de alta definição por segundo em um computador pessoal *high-end*. Isso permite que um veículo equipado com essa tecnologia e navegando em uma via pública a 60km/h seja capaz de detectar e reconhecer placas de sinalização a cada 2,78 metros.

Figura 42: Histograma de tamanhos das MSERs negativas selecionadas pelo método nas bases BRTSD e GTSDB.



(a) MSERs negativas da BRTSD.



(b) MSERs negativas da GTSDB.

Fonte: Elaborado pelo autor.

Além disso, a capacidade de reconhecer objetos pequenos (36×36 píxeis, cerca de 0.1 % da área total da imagem) permite detectar placas de sinalização em longas distâncias. O tamanho mínimo pode ser reajustado para aumentar a precisão da técnica, reduzir o tempo de processamento (ao eliminar falsos positivos pequenos, como mostra a Figura 42) e encaixar o sistema aos requisitos de tempo real. Além disso, sistemas embarcados comerciais geralmente utilizam hardware dedicados, aumentando a eficiência da implementação.

O estudo de técnicas de rastreamento, a expansão e anotação da base de dados construída e o estudo da substituição das informações de cores por informação de profundidade na segmentação de objetos são tópicos a serem discutidos em trabalhos futuros.

REFERÊNCIAS

ABUKHAIT, J. et al. Road sign detection and shape recognition invariant to sign defects. In: IEEE INTERNATIONAL CONFERENCE ON ELECTRO/INFORMATION TECHNOLOGY, 2012, Indianapolis. **Proceedings...** Piscataway: IEEE, 2012. p.1–6.

ADAPTEVA. **The Parallella Board**. Disponível em: <https://www.parallella.org/>. Acesso em: 19 set. 2016.

AVNET. **ZedBoard**. Disponível em: <http://zedboard.org/product/zedboard>. Acesso em: 19 set. 2016.

BAYER, B. **Color imaging array**. US3971065 A, 20 jul. 1976. Disponível em: <https://www.google.com/patents/US3971065>. Acesso em: 20 dez. 2016.

BISHOP, C. M. **Pattern Recognition and Machine Learning**. 1st. ed. New York: Springer US, 2006.

BOUREAU, Y. L. et al. Ask the locals: multi-way local pooling for image recognition. In: INTERNATIONAL CONFERENCE ON COMPUTER VISION, 13., 2011, Barcelona. **Proceedings...** Piscataway: IEEE, 2011. p.2651–2658.

BOUREAU, Y.-L.; PONCE, J.; LECUN, Y. A Theoretical Analysis of Feature Pooling in Visual Recognition. In: INTERNATIONAL CONFERENCE ON MACHINE LEARNING, 27., 2010, Haifa. **Proceedings...** [S.l.]: IMLS, 2010. p.111–118.

BUI-MINH, T. et al. Two algorithms for detection of mutually occluding traffic signs. In: INTERNATIONAL CONFERENCE ON CONTROL, AUTOMATION AND INFORMATION SCIENCES (ICCAIS), 1., 2012, Ho Chi Minh City. **Proceedings...** Piscataway: IEEE, 2012. p.120–125.

CENTRAL INTELLIGENCE AGENCY. **The World Factbook**. Disponível em: <https://www.cia.gov/library/publications/the-world-factbook/geos/br.html>. Acesso em: 02 jun. 2016.

CHEN, L. et al. Traffic sign detection and recognition for intelligent vehicle. In: IEEE INTELLIGENT VEHICLES SYMPOSIUM, 2011, Baden-Baden. **Proceedings...** Piscataway: IEEE, 2011. n.4, p.908–913.

CHENG, H. D. et al. Color image segmentation: advances and prospects. **Pattern Recognition**, Amsterdam, v.34, n.12, p.2259–2281, 2001.

CHOI, S. et al. **Advanced driver-assistance systems**: challenges and opportunities ahead. Disponível em <http://www.mckinsey.com/industries/semiconductors/our-insights/advanced-driver-assistance-systems-challenges-and-opportunities-ahead>. Acesso em: 21 ago. 2016.

CINTULA, P.; FERMULLER, C. G.; NOGUERA, C. Fuzzy Logic. In: ZALTA, E. N. (Ed.). **Stanford Encyclopedia of Philosophy**. Winter ed. Stanford: The Metaphysics Research Lab, Stanford University, 2016.

CIRESAN, D. et al. A committee of neural networks for traffic sign classification. In: INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS (IJCNN'11), 2011, San Jose. **Proceedings...** Piscataway: IEEE, 2011. v.1, n.1, p.1918–1921.

CIRESAN, D. et al. Multi-column deep neural network for traffic sign classification. **Neural Networks**, Amsterdam, v.32, p.333–338, Aug. 2012.

CONFEDERAÇÃO NACIONAL DO TRANSPORTE. **Pesquisa CNT de Rodovias 2016**. Brasília, 2016. 1 p. Disponível em: <http://pesquisarodoviascms.cnt.org.br //PDFs/boletim_pesquisa_cnt_rodovias_2016.pdf>. Acesso em: 25 maio 2016.

CONSELHO NACIONAL DE TRÂNSITO. **Manual Brasileiro de Sinalização de Trânsito**. 2^a ed. Brasília, 2007. 220 p. v.I.

CONSELHO NACIONAL DE TRÂNSITO. **Manual Brasileiro de Sinalização de Trânsito**. 1^a ed. Brasília, 2007. 218 p. v.II.

CUN, Y. L. et al. Handwritten digit recognition: applications of neural network chips and automatic learning. **IEEE Communications Magazine**, Piscataway, v.27, n.11, p.41–46, Nov. 1989.

DALAL, N.; TRIGGS, B. Histograms of oriented gradients for human detection. In: IEEE COMPUTER SOCIETY CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION, 17., 2005, San Diego. **Proceedings...** Piscataway: IEEE, 2005. v.1, p.886–893.

DE LA ESCALERA, A.; ARMINGOL, J. M.; MATA, M. Traffic sign recognition and analysis for intelligent vehicles. **Image and Vision Computing**, Amsterdam, v.21, n.3, p.247–258, Mar. 2003.

DEPARTAMENTO NACIONAL DE INFRAESTRUTURA DE TRANSPORTES. **Estatísticas de Acidentes**. Brasília, 2016. Disponível em: <http://www.dnit.gov.br/rodovias/operacoes-rodoviarias/estatisticas-de-acidentes>. Acesso em: 02 jun. 2016.

DICKMANNS, E. D. Vehicles capable of dynamic vision. In: INTERNATIONAL JOINT CONFERENCE ON ARTIFICAL INTELLIGENCE, 15., 1997, Nagoya, Japan. **Proceedings...** San Francisco: Morgan Kaufmann Publishers Inc., 1997. p.1577–1592.

DUBOIS, E. **The Structure and Properties of Color Spaces and the Representation of Color Images**. 1 ed. Austin: Morgan & Claypool Publishers, 2009. (Synthesis Lectures on Image, Video, and Multimedia Processing).

ECONOMIC COMISSION FOR EUROPE. **Report of the sixty-eighth session of the Working Party on Road Traffic Safety**. Geneva: UN, 2014. 11 p.

ESCALERA, A. de la et al. Road traffic sign detection and classification. **IEEE Transactions on Industrial Electronics**, Piscataway, v.44, n.6, p.848–859, 1997.

ESTABLE, S. et al. A real-time traffic sign recognition system. In: INTELLIGENT VEHICLES SYMPOSIUM, 1994, Paris. **Proceedings...** Piscataway: IEEE, 1994. p.213–218.

FAIRCHILD, M. D. **Color Appearance Models**. 3rd. ed. New York: Wiley, 2013. (The Wiley-IS&T Series in Imaging Science and Technology).

FLEYEH, H. Shadow And Highlight Invariant Colour Segmentation Algorithm For Traffic Signs. In: CONFERENCE ON CYBERNETICS AND INTELLIGENT SYSTEMS, 2., 2006, Bangkok. **Proceedings...** Piscataway: IEEE, 2006. p.1–7.

FLEYEH, H.; BISWAS, R.; BHUIYAN, N. U. An Adaptive Approach to Detect Warning Traffic Signs using SOM and Windowed Hough Transform. In: IASTED INTERNATIONAL CONFERENCE ON SIGNAL AND IMAGE PROCESSING AND APPLICATIONS (SIPA'11), 2011, Crete. **Proceedings...** Calgary: Actapress, 2011.

FLEYEH, H. et al. Invariant road sign recognition with fuzzy artmap and zernike moments. In: IEEE INTELLIGENT VEHICLES SYMPOSIUM, 2007, Istanbul. **Proceedings...** Piscataway: IEEE, 2007. p.31–36.

FLEYEH, H.; ROCH, J. Benchmark Evaluation of Hog Descriptors As Features for Classification of Traffic Signs. **International Journal for Traffic and Transport Engineering**, Belgrade, v.3, n.4, p.448–464, 2013.

GLOROT, X.; BORDES, A.; BENGIO, Y. Deep sparse rectifier neural networks. In: INTERNATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE AND STATISTICS, 14., 2011, Fort Lauderdale. **Proceedings...** [S.l.]: JMLR, 2011. v.15, p.315–323.

GONZALEZ, R.; WOODS, R. **Digital Image Processing**. 3rd. ed. Upper Saddle River: Pearson Prentice Hall, 2008.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep Learning**. 1st. ed. Cambridge: MIT Press, 2016.

GRANLUND, G. H. Fourier Preprocessing for Hand Print Character Recognition. **IEEE Transactions on Computers**, Washington, v.C-21, n.2, p.195–201, 1972.

GREENHALGH, J.; MIRMEHDI, M. Real-Time Detection and Recognition of Road Traffic Signs. **IEEE Transactions on Intelligent Transportation Systems**, Piscataway, v.13, n.4, p.1498–1506, 2012.

HAAG, C. A arca humana num diúvio de dados. **Pesquisa FAPESP**, São Paulo, p.4, Jun. 2011.

HAN, Y.; ORUKLU, E. Real-time traffic sign recognition based on Zynq FPGA and ARM SoCs. In: IEEE INTERNATIONAL CONFERENCE ON ELECTRO/INFORMATION TECHNOLOGY (EIT'14), 2014, Milwaukee. **Proceedings...** Piscataway: IEEE, 2014. p.373–376.

HAWKINS, A. J. . Uber's self-driving truck company just completed its first shipment: 50,000 cans of budweiser. **The Verge**, [S.l.], Oct. 2016. Disponível em <http://www.theverge.com/2016/10/25/13381246/otto-self-driving-truck-budweiser-first-shipment-uber>. Acesso em: 03 nov. 2016.

HOUBEN, S. et al. Detection of traffic signs in real-world images: the german traffic sign detection benchmark. In: INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS (IJCNN'13), 2013, Dallas. **Proceedings...** Piscataway: IEEE, 2013. p.1–8.

HUYNH-THE, T.; NGUYEN THANH, H.; TRAN CONG, H. Traffic Sign Recognition using Multi-class morphological detection. In: INTERNATIONAL CONFERENCE ON ADVANCED TECHNOLOGIES FOR COMMUNICATIONS, 7., 2014, Hanoi. **Proceedings...** Piscataway: IEEE, 2014. p.274–279.

INTEL. **The Internet of Things is on the Road to Autonomous Driving**. Disponível em: <http://www.intel.com/content/www/us/en/internet-of-things/infographics/iot-autonomous-driving-infographic.html>. Acesso em: 20 out. 2016.

JIA, Y.; HUANG, C.; DARRELL, T. Beyond spatial pyramids: receptive field learning for pooled image features. In: IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION (CVPR), 25., 2012, Providence. **Proceedings...** Piscataway: IEEE, 2012. p.3370–3377.

JIN, J.; FU, K.; ZHANG, C. Traffic Sign Recognition With Hinge Loss Trained Convolutional Neural Networks. **IEEE Transactions on Intelligent Transportation Systems**, Piscataway, v.15, n.5, p.1991–2000, Oct. 2014.

JOCHEM, T. et al. PANS: a portable navigation platform. In: INTELLIGENT VEHICLES SYMPOSIUM, 1995, Detroid, USA. **Proceedings...** Piscataway: IEEE, 1995. p.107–112.

JOCHEM, T. M.; POMERLEAU, D. **No Hands Across America**. Disponível em: <http://www.cs.cmu.edu/afs/cs/usr/tjochem/www/nhaa/nhaa_home_page.html>. Acesso em: 08 março 2016.

KELLER, C. G. et al. Real-time recognition of U.S. speed signs. In: IEEE INTELLIGENT VEHICLES SYMPOSIUM, 2008, Eindhoven. **Proceedings...** Piscataway: IEEE, 2008. p.518–523.

KHAN, J. F.; BHUIYAN, S. M. a.; ADHAMI, R. R. Image Segmentation and Shape Analysis for Road-Sign Detection. **IEEE Transactions on Intelligent Transportation Systems**, Piscataway, v.12, n.1, p.83–96, Mar. 2011.

KINGMA, D. P.; BA, J. Adam: A method for stochastic optimization. In: INTERNATIONAL CONFERENCE FOR LEARNING REPRESENTATIONS, 3., 2015, San Diego. **Proceedings...** [S.l.]: CBLS, 2015. p.1–15.

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. In: ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS, 25., 2012, Stateline. **Proceedings...** La Jolla: Neural Information Processing Systems Foundation, 2012. p.1097–1105.

LARSSON, F.; FELSBERG, M. Using Fourier Descriptors and Spatial Models for Traffic Sign Recognition. In: SCANDINAVIAN CONFERENCE ON IMAGE ANALYSIS, 17., 2011, Ystad. **Proceedings...** Berlin: Springer Berlin Heidelberg, 2011. v.6688, p.238–249.

LARSSON, F.; FELSBERG, M.; FORSSEN, P. E. Correlating fourier descriptors of local patches for road sign recognition. **IET Computer Vision**, Stevenage, v.5, n.4, p.244–254, 2011.

LAUER, M. Grand Cooperative Driving Challenge 2011. **IEEE Intelligent Transportation Systems Magazine**, Piscataway, v.3, n.3, p.38–40, 2011.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. **Nature**, London, v.521, n.7553, p.436–444, May. 2015.

LIANG, M. et al. Traffic sign detection by ROI extraction and histogram features-based recognition. In: INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS (IJCNN'13), 2013, Dallas. **Proceedings...** Piscataway: IEEE, 2013. p.1–8.

LIU, C.; CHANG, F.; CHEN, Z. Rapid Multiclass Traffic Sign Detection in High-Resolution Images. **IEEE Transactions on Intelligent Transportation Systems**, Piscataway, v.15, n.6, p.2394–2403, Dec 2014.

LOY, G.; BARNES, N. Fast shape-based road sign detection for a driver assistance system. In: IEEE/RSJ INTERNATIONAL CONFERENCE ON INTELLIGENT ROBOTS AND SYSTEMS, 17., 2004, Sendai. **Proceedings...** Piscataway: IEEE, 2004. p.70–75.

LUCCHESE, L.; MITRAY, S. K. Color image segmentation: a state-of-the-art survey. **Proceedings., Indian National Science Academy (INSA-A)**, Delhi, v.67, p.207–221, 2001.

LUO, R.; POTLAPALLI, H. Landmark recognition using projection learning for mobile robot navigation. In: IEEE INTERNATIONAL CONFERENCE ON NEURAL NETWORKS (ICNN'94), 4., 1994, Orlando. **Proceedings...** Piscataway: IEEE, 1994. p.2703–2708.

LUO, R.; POTLAPALLI, H.; HISLOP, D. Translation And Scale Invariant Landmark Recognition Using Receptive Field Neural Networks. In: IEEE/RSJ INTERNATIONAL CONFERENCE ON INTELLIGENT ROBOTS AND SYSTEMS, 5., 1992, Raleigh. **Proceedings...** Piscataway: IEEE, 1992. p.527–533.

LUO, R.; POTLAPALLI, H.; HISLOP, D. Natural scene segmentation using fractal based autocorrelation. In: INTERNATIONAL CONFERENCE ON INDUSTRIAL ELECTRONICS, CONTROL, INSTRUMENTATION, AND AUTOMATION, 18., 1992, San Diego. **Proceedings...** Piscataway: IEEE, 1992. p.700–705.

MALINOWSKI, M.; FRITZ, M. Learnable pooling regions for image classification. In: INTERNATIONAL CONFERENCE ON LEARNING REPRESENTATIONS, 2., 2013, Scottsdale. **Proceedings...** [S.l.]: CBLS, 2013. p.1–10.

MANDAL, S. N.; CHOUDHURY, J. P.; CHAUDHURI, S. R. B. In Search of Suitable Fuzzy Membership Function in Prediction of Time Series Data. **International Journal of Computer Science Issues**, Mahebourg, v.9, n.3, p.293–302, 2012.

MATAS, J. et al. Robust Wide Baseline Stereo from Maximally Stable Extremal Regions. In: BRITISH MACHINE VISION CONFERENCE, 13., 2002, Cardiff. **Proceedings...** Durham: British Machine Vision Association, 2002. p.36.1–36.10.

MATHIAS, M. et al. Traffic sign recognition - How far are we from the solution? In: INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS (IJCNN'13), 2013, Dallas. **Proceedings...** Piscataway: IEEE, 2013. p.1–8.

MOBILEYE. **Autonomous Car Vision System**. Disponível em: <https://youtu.be/_dvyzAA1Cn8>. Acesso em: 02 dez. 2015.

MØGELMOSE, A.; TRIVEDI, M. M.; MOESLUND, T. B. Vision-based traffic sign detection and analysis for intelligent driver assistance systems: perspectives and survey. **IEEE Transactions on Intelligent Transportation Systems**, Piscataway, v.13, n.4, p.1484–1497, 2012.

NGUYEN, B. T.; RYONG, S. J.; KYU, K. J. Fast traffic sign detection under challenging conditions. In: INTERNATIONAL CONFERENCE ON AUDIO, LANGUAGE AND IMAGE PROCESSING (ICALIP'14), 4., 2014, Shanghai. **Proceedings...** IEEE, 2014. p.749–752.

NISTÉR, D.; STEWÉNIUS, H. Linear Time Maximally Stable Extremal Regions. In: EUROPEAN CONFERENCE ON COMPUTER VISION, 10., 2008, Marseille. **Proceedings...** Berlin: Springer Berlin Heidelberg, 2008. p.183–196.

NVIDIA. **Embedded System Solutions from NVIDIA**. Disponível em: <http://www.nvidia.com/object/embedded-systems.html>. Acesso em: 19 set. 2016.

ORGANIZAÇÃO MUNDIAL DA SAÚDE. **Relatório Global Sobre O Estado Da Segurança Viária 2015**. Genebra, 2015. 16 p. Disponível em: <http://www.who.int/violence_injury_prevention/road_safety_status/2015/Summary_GSRRS2015_POR.pdf>. Acesso em: 25 mai. 2016.

PAULA, M. B. D. **Visão computacional para veículos inteligentes usando câmeras embarcadas**. 2015. 102 p. Tese (Doutorado em Engenharia Elétrica) — Universidade Federal do Rio Grande do Sul, Porto Alegre, 2015.

PICCIOLI, G. et al. Robust road sign detection and recognition from image sequences. In: INTELLIGENT VEHICLES SYMPOSIUM, 1994, Paris. **Proceedings...** Piscataway: IEEE, 1994. p.278–283.

POLÍCIA RODOVIÁRIA FEDERAL. **Balanço de Atividades 2014**. Disponível em: <https://www.prf.gov.br/portal/noticias/nacionais/prf-balanco-de-atividades-2014>. Acesso em: 25 jun. 2016.

RUSSAKOVSKY, O. et al. ImageNet Large Scale Visual Recognition Challenge. **International Journal of Computer Vision**, New York, v.115, n.3, p.211–252, Dec. 2015.

SAE INTERNATIONAL. **J3016_201609**: Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles. Warrendale, 2016. 30 p.

SALTI, S. et al. A traffic sign detection pipeline based on interest region extraction. In: INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS (IJCNN'13), 2013, Dallas. **Proceedings...** Piscataway: IEEE, 2013. p.1–7.

SERMANET, P.; LECUN, Y. Traffic sign recognition with multi-scale convolutional networks. In: INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS (IJCNN'11), 2011, San Jose. **Proceedings...** Piscataway: IEEE, 2011. p.2809–2813.

SMITH, A. R. Color Gamut Transform Pairs. **SIGGRAPH Computer Graphics**, New York, v.12, n.3, p.12–19, 1978.

SOENDORO, D.; SUPRIANA, I. Traffic sign recognition with Color-based Method, shape-arc estimation and SVM. In: INTERNATIONAL CONFERENCE ON ELECTRICAL ENGINEERING AND INFORMATICS, 3., 2011, Bandung. **Proceedings...** Piscataway: IEEE, 2011. p.1–6.

SOUANI, C.; FAIEDH, H.; BESBES, K. Efficient algorithm for automatic road sign recognition and its hardware implementation. **Journal of Real-Time Image Processing**, Berlin, v.9, n.1, p.79–93, Mar. 2014.

SRIVASTAVA, N. et al. Dropout: a simple way to prevent neural networks from overfitting. **Journal of Machine Learning Research**, [S.l.], v.15, n.1, p.1929–1958, Jan. 2014.

STALLKAMP, J. et al. Man vs. computer: benchmarking machine learning algorithms for traffic sign recognition. **Neural Networks**, Amsterdam, v.32, p.323–332, 2012.

TESLA MOTORS. **A Tragic Loss**. Disponível em: <https://www.tesla.com/blog/tragic-loss>. Acesso em: 23 out. 2016.

THE GREATER GOOD. **Wired Magazine Illustrations**. Disponível em: <http://thegreatergood.cc/ Wired-Magazine>. Acesso em: 15 mar. 2016.

THRUN, S. et al. Stanley: the robot that won the darpa grand challenge. **Journal of Field Robotics**, Hoboken, v.23, n.9, p.661–692, Sept. 2006.

TIMOFTE, R.; GOOL, L. V. Sparse Representation Based Projections. In: PROCEDINGS., 22 BRITISH MACHINE VISION CONFERENCE, 2011, Dundee. **Anais...** Durham: British Machine Vision Association, 2011. p.61.1–61.12.

TIMOFTE, R.; ZIMMERMANN, K.; Van Gool, L. Multi-view traffic sign detection, recognition, and 3D localisation. **Machine Vision and Applications**, Berlin, v.25, n.3, p.633–647, Apr. 2014.

UNITED NATIONS. **Decade of Action for Road Safety 2011-2020**. Disponível em: <http://www.who.int/roadsafety/decade_of_action/en/>. Acesso em: 16 mai. 2016.

VANDERBILT, T. Autonomous Cars Through the Ages. **Wired**, New York, Feb. 2012. Disponível em: <https://www.wired.com/2012/02/autonomous-vehicle-history/>. Acesso em 05 out. 2015.

VINCENT, J. World's first self-driving taxi trial begins in Singapore. **The Verge**, [S.l.], Aug. 2016. Disponível em: <http://www.theverge.com/2016/8/25/12637822/self-driving-taxi-first-public-trial-singapore-nutonomy>. Acesso em: 03 nov. 2016.

VINCENT, L.; SOILLE, P. Watersheds in Digital Spaces: an efficient algorithm based on immersion simulations. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, Piscataway, v.13, n.6, p.583–598, Jun. 1991.

VIOLA, P.; JONES, M. Rapid object detection using a boosted cascade of simple features. In: IEEE COMPUTER SOCIETY CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION, 13., 2001, Kauai. **Proceedings...** Piscataway: IEEE, 2001. v.1, p.511–518.

VITABILE, S.; GENTILE, A.; SORBELLO, F. A neural network based automatic road signs recognizer. In: INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS (IJCNN'02), 2002, Honolulu. **Proceedings...** Piscataway: IEEE, 2002. v.3, p.2315–2320.

VOLVO CAR GROUP. **Volvo Cars presents a unique solution for integrating self-driving cars into real traffic**. Disponível em: <https://www.media.volvocars.com/global/en-gb/media/pressreleases/158276/volvo-cars-presents-a-unique-system-solution-for-integrating-self-driving-cars-into-real-traffic>. Acesso em: 06 set. 2016.

WAITE, S.; ORUKLU, E. FPGA-Based Traffic Sign Recognition for Advanced Driver Assistance Systems. **Journal of Transportation Technologies**, [S.l.], v.03, n.01, p.1–16, 2013.

WANG, D. et al. A saliency-based cascade method for fast traffic sign detection. In: IEEE INTELLIGENT VEHICLES SYMPOSIUM, 2015, Seoul. **Proceedings...** Piscataway: IEEE, 2015. n.Iv, p.180–185.

WANG, G. et al. A hierarchical method for traffic sign classification with support vector machines. In: INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS (IJCNN'13), 2013, Dallas. **Proceedings...** Piscataway: IEEE, 2013. p.1–6.

WANG, G. et al. A robust, coarse-to-fine traffic sign detection method. In: INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS (IJCNN'13), 2013, Dallas. **Proceedings...** Piscataway: IEEE, 2013. p.1–5.

XILINX. **MicroBlaze Soft Processor Core**. Disponível em <https://www.xilinx.com/products/design-tools/microblaze.html>. Acesso em: 19 set. 2016.

YANG, Y. et al. Towards Real-Time Traffic Sign Detection and Classification. **IEEE Transactions on Intelligent Transportation Systems**, Piscataway, v.17, n.7, p.2022–2031, 2016.

ZAKLOUTA, F.; STANCIULESCU, B. Segmentation masks for real-time traffic sign recognition using weighted HOG-based trees. In: INTERNATIONAL IEEE CONFERENCE ON INTELLIGENT TRANSPORTATION SYSTEMS (ITSC), 14., 2011, Washington. **Proceedings...** Piscataway: IEEE, 2011. p.1954–1959.

ZAKLOUTA, F.; STANCIULESCU, B.; HAMDOUN, O. Traffic sign classification using K-d trees and random forests. In: INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS (IJCNN'11), 2011, San Jose. **Proceedings...** Piscataway: IEEE, 2011. p.2151–2155.

ZHU, Z. et al. Traffic-Sign Detection and Classification in the Wild. In: IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION (CVPR), 29., 2016, Las Vegas. **Proceedings...** Piscataway: IEEE, 2016. p.2110–2118.