

Traffic Sign Detection and Recognition using the AdaBoost and SIFT algorithm

Francisco A. da Silva^{1,2}, Antônio C. P. Nascimento¹,
Maria Stela Veludo de Paiva²

¹Universidade do Oeste Paulista – Unoeste/FIPP
Presidente Prudente/SP, Brasil

²Universidade de São Paulo – USP/EESC
São Carlos/SP, Brasil

chico@unoeste.br, acpn@unoeste.edu.br, mstela@sc.usp.br

Almir Olivette Artero³, Marco Antonio Piteri³,
Ricardo Luiz Barbosa⁴

³Universidade Estadual Paulista – Unesp/FCT
Presidente Prudente/SP, Brasil

⁴Universidade Estadual Paulista – Unesp
Campus de Sorocaba/SP, Brasil

{almir, piteri}@fct.unesp.br, rluisbarbosa@usp.br

Abstract—This paper presents the algorithm TSRAS (Traffic Sign Recognition using AdaBoost and SIFT) proposed for traffic sign detection and recognition using images from videos. Initially, the AdaBoost algorithm is used to detect the plates from a specific training for traffic signals, without considering all the information printed on the plates, but only the basic information, such as red contours, the presence or not of diagonal lines represent prohibition, yellow or red background. Different from several studies in the literature, this strategy reduces the amount of different plates needed for AdaBoost training. In addition, the HSV color space was used to eliminate regions incorrectly detected by AdaBoost. In the next, the algorithm proposed by Silva et al. [2] that uses the SIFT algorithm to extract keypoints in the candidate areas was applied. Finally, we performed the text recognition for the speed plates. The processing time of this new approach was 1/8 over the previous approach.

Keywords—traffic sign detection; traffic sign recognition; AdaBoost; SIFT

I. INTRODUÇÃO

O reconhecimento automático de placas de trânsito está se tornando uma área com muitas aplicações na indústria automotiva [3], por exemplo, possibilitando que um motorista seja avisado sobre ações impróprias e situações potencialmente perigosas, o que consiste em uma tarefa essencial para um sistema inteligente de auxílio a motoristas [4]. De fato, os textos e símbolos inseridos em um sinal de trânsito, geralmente, apresentam informação útil, como limite de velocidade, direção da via, e situações de trânsito, que ajudam os motoristas a dirigir com maior segurança [5].

Este trabalho apresenta uma metodologia para realizar a detecção e o reconhecimento de sinais de trânsito, a partir de quadros de vídeo, com potencial para operar em tempo real, pois utiliza o algoritmo AdaBoost [18], que tem sido muito usado para o reconhecimento de faces, para selecionar áreas candidatas contendo imagens de placas de sinalização de trânsito. Os sinais de trânsito tratados neste trabalho são as placas de sinalização afixadas em calçadas de ruas e avenidas.

As demais seções deste trabalho estão organizadas da seguinte maneira: Na Seção II são apresentados alguns trabalhos relacionados ao reconhecimento de sinais de trânsito; Na Seção III é apresentada uma breve revisão dos algoritmos

usados; Na Seção IV é apresentado o algoritmo TSRAS, proposto neste trabalho, que realiza a detecção e o reconhecimento das placas; Na Seção V são apresentados os resultados obtidos com a aplicação da metodologia proposta em uma sequência de quadros de vídeo, além de apresentar uma comparação com um trabalho anterior, que usa um conjunto de imagens de alta resolução; Por fim, na Seção VI são apresentadas as conclusões e trabalhos futuros.

II. TRABALHOS RELACIONADOS

Ao longo dos últimos anos foram propostas diversas abordagens para resolver o problema do reconhecimento de sinais de trânsito, como por exemplo: utilizando propriedades de cor dos sinais [6]; a partir da geometria das placas em imagens em tons de cinza [7]; baseada nas propriedades de forma dos sinais [8]; baseada em propriedades de cor e forma [9]; utilizando classificadores baseados em redes neurais [3]; analisando a curvatura da representação do sinal com contornos ativos (Snakes) [10]; usando *Support Vector Machine* (SVM) [11]; usando uma segmentação de sinais de trânsito baseado em conjuntos fuzzy [12].

Segundo Chen e Hsieh [5], é difícil detectar os sinais de trânsito diretamente a partir de vídeos, devido a diferentes alterações de condições ambientais (nebulosidade, iluminação, sombras, etc), o que ocasiona sinais de trânsito com diferentes mudanças na sua aparência. Além disso, para a câmera montada na frente de um carro em movimento, seus efeitos de perspectiva fazem o sinal de trânsito ter diferentes tamanhos, formas, contrastes e borramentos (ocasionado pelo deslocamento do carro). Em alguns casos, a placa com o sinal de trânsito pode ficar totalmente ou parcialmente encoberta por outros objetos como árvores, veículos, etc.

O trabalho de Silva et al. [2] apresenta como proposta para o reconhecimento de placas de sinalização de trânsito, o uso do algoritmo SIFT, para detectar pontos chave nas imagens e, em seguida, compara os descritores destes pontos com os descritores obtidos em templates contendo os sinais de trânsito. Por fim, aplica um algoritmo de reconhecimento de caracteres para reconhecer textos, tais como os limites de velocidade. Este algoritmo apresenta bons resultados, entretanto, como precisa aplicar o algoritmo SIFT na imagem inteira, a sua aplicação em tempo real se torna impraticável. De fato, este é o maior

diferencial da abordagem proposta neste trabalho, que sugere aplicar o algoritmo AdaBoost na imagem inteira e, então, usar o SIFT apenas nas regiões candidatas, detectadas pelo AdaBoost. A vantagem desta proposta é que o AdaBoost possui um tempo de processamento muito pequeno, sendo indicado para aplicações em tempo real.

III. REFERENCIAL TEÓRICO

O detector de Viola e Jones [1] baseia-se em uma cascata de classificadores boosted Haar-like, que combina dois conceitos: (i) classificadores Haar-like e (ii) AdaBoost [18]. Os classificadores Haar-like são construídos com características retangulares simples, que representam as diferenças de somas de pixels específicos em uma imagem. Cada característica é emparelhada com um limiar, e a decisão do classificador é determinada por uma comparação do valor da característica com o limiar [13]. Esta simplicidade constitui um diferencial importante para minimizar o tempo de processamento. O algoritmo AdaBoost usa pesos em uma combinação linear de classificadores fracos, formando um classificador forte [5].

Viola e Jones seguem nesta linha, agrupando vários classificadores fortes, construídos por AdaBoost, em uma cascata que apresenta um processamento muito rápido. O classificador forte na primeira fase da cascata é escolhido de modo a descartar falsos positivos e, ao mesmo tempo, preservar quase todos os verdadeiros positivos do conjunto de treino. O processo de detecção de faces é realizado por uma janela deslizante de detecção através da imagem. Dentro da janela, a resposta da cascata é calculada. Depois de completar uma passagem sobre a imagem, o tamanho da janela de detecção é aumentado por algum fator (1.2 no OpenCV, significando que a escala da janela será aumentada em 20%). O tamanho da janela é aumentado até um tamanho pré-definido ser alcançado. Aumentando a janela de detecção por uma percentagem menor produz as melhores taxas de detecção, mas aumenta o número de falsos positivos e também o tempo total de processamento [13].

O sistema de detecção de faces proposto por Viola e Jones [1] utiliza características simples para classificar as imagens, baseadas nas funções de Haar. As características correspondem basicamente a retângulos que possuem duas regiões, sendo uma clara e outra escura. O cálculo do valor correspondente à característica é feito subtraindo o somatório da intensidade dos pixels da região escura do total acumulado pela intensidade dos pixels da região clara. Este cálculo pode ser feito rapidamente utilizando a representação de imagem integral. Na Fig. 1 são apresentadas as características básicas de Haar que foram utilizadas por Viola e Jones [1].



Fig. 1. Características Básicas de Haar.

Com o intuito de melhorar ainda mais a etapa de detecção de objetos, Lienhart e Maydt [20] estenderam a quantidade de tipos possíveis de características retangulares, acrescentando características às previamente sugeridas, mas desta vez

rotacionadas em 45 graus. Com estas novas adições ao conjunto, consegue-se reduzir o número de falsos positivos em torno de 10%, aumentando ainda mais a eficiência do processo de detecção. Na Fig. 2 é apresentado o conjunto completo com as novas características retangulares sugeridas.

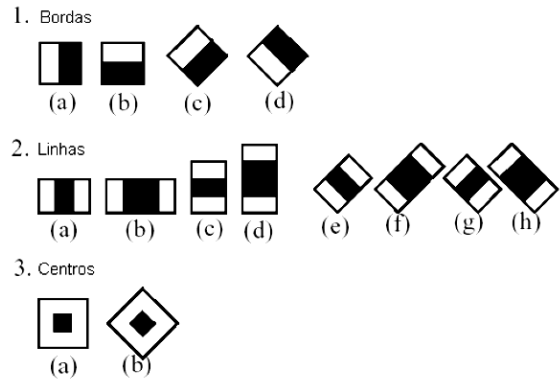


Fig. 2. Características Estendidas. Fonte: [20].

A. Algoritmo AdaBoost

Boosting é um método usado para aprimorar o desempenho de um algoritmo de aprendizado, não sendo empregado de forma isolada, mas sendo aplicado de forma combinada com outras técnicas, como Árvores de Decisão e Redes Neurais. O objetivo é transformar classificadores “fracos” em “fortes”, o que é feito combinando classificadores gerados por um mesmo algoritmo de aprendizado (algoritmo base ou fraco) e ajustando de acordo com os erros cometidos pelo classificador anterior. Deste modo, busca-se gerar um classificador melhor, com uma eficiência maior do que as obtidas pelos classificadores isolados [22]. O modelo Boosting, compreende uma família de algoritmos, dentre os quais o algoritmo Adaptive Boosting (AdaBoost) é o mais relevante e popular [22]. O funcionamento do algoritmo AdaBoost é descrito a seguir [21] [23]:

A primeira etapa consiste em um treinamento usando um conjunto de exemplos $S = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$, onde:

- Cada x_i representa m vetores de atributos do sistema, ou seja, um conjunto de dados referentes aos parâmetros (ou propriedades) analisados;
- Dentre todos os possíveis grupos de classificação previamente definidos para o sistema analisado, cada y_i representa o grupo de classificação associado ao x_i ;
- O número total de exemplos da amostra de treinamento é igual a m .

O algoritmo base, que trabalha junto ao AdaBoost, fornece a cada iteração, uma distribuição de pesos referentes a cada um dos dados de treinamento. Os classificadores iniciam com pesos associados (Equação 1) igualmente distribuídos.

$$D_0 = \frac{1}{m} \quad (1)$$

Em cada ciclo de aprendizagem, o algoritmo base gera uma hipótese h_i , baseada nos pesos atuais, com o objetivo de priorizar a classificação correta dos dados que apresentam os

maiores pesos associados. Deste modo, o algoritmo base busca a hipótese que minimiza o erro de treinamento e_t .

$$e_t = \sum_{i:h_t(x_i) \neq y_i} D_t(i) \tag{2}$$

Em seguida, esses pesos são reavaliados, de forma a aumentar aqueles que são relacionados aos dados incorretamente classificados e, em seguida, uma nova iteração se inicia. Por fim, ao serem realizadas todas as T iterações determinadas previamente, o AdaBoost combina todas as hipóteses intermediárias h_t de modo a gerar uma única hipótese final $H(x)$. O método para se obter a hipótese final $H(x)$ é a combinação ponderada das diversas saídas das iterações. Assim, uma votação ponderada das T hipóteses fracas é realizada, onde γ_t é a importância associada à hipótese h_t . O esquema mostrado na Fig. 3 resume o funcionamento do algoritmo AdaBoost.

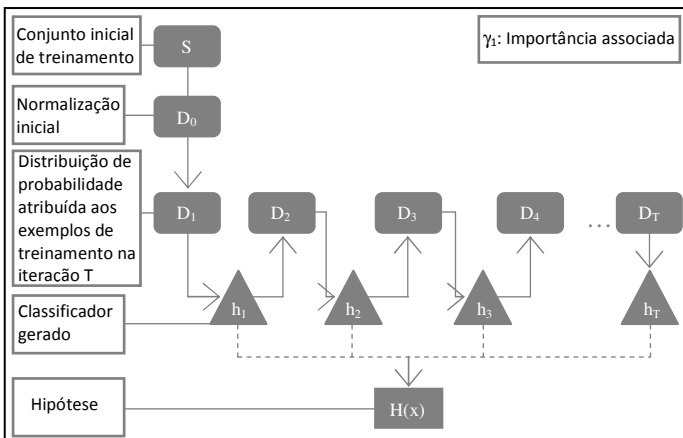


Fig. 3. Esquema do funcionamento do algoritmo AdaBoost.

Por alta velocidade de processamento e eficácia, o algoritmo AdaBoost é indicado neste trabalho para selecionar as áreas candidatas na imagem a terem uma placa de sinalização de trânsito, reduzindo as regiões a serem processadas pelo algoritmo SIFT, que apresenta tempo elevado de processamento.

IV. TSRAS - RECONHECIMENTO DE SINAIS DE TRÂNSITO USANDO ADABOOST E SIFT

O processo de reconhecimento de placas, geralmente, é dividido em duas etapas, que são a detecção e o reconhecimento. Neste trabalho, a detecção utiliza o algoritmo AdaBoost [1]. Apesar do AdaBoost ter sido usado originalmente na detecção de rostos, segundo Brkić [13], outras pesquisas têm aplicado com sucesso o detector em uma grande quantidade de outras classes de objetos. De fato, muitos autores já utilizam essa técnica para realizar o reconhecimento de placas de trânsito [3] [4] [5] [14] [15] [16] [17], sendo o principal diferencial deste trabalho, o uso de imagens de treinamento simplificadas, que omitem a maior parte das informações impressas nas placas, ou seja, com apenas as informações de bordas, como círculos vermelhos com ou sem as linhas diagonais que representam proibição, além de placas com fundo amarelo, ou placas com fundo vermelho. O objetivo desta simplificação é abranger a maior quantidade possível de diferentes tipos de sinais de trânsito,

sem a necessidade de se usar exemplares de cada um deles durante o treinamento. Em geral, os autores se limitam a uma pequena quantidade de placas diferentes, para se obter um tempo baixo de processamento.

Após a aplicação da etapa de detecção, que deveria resultar apenas regiões das imagens que possuem uma placa de sinalização de trânsito, ainda é possível existir algumas regiões que não possuem tais elementos. Para que essas regiões sejam rapidamente eliminadas do processo, evitando o seu processamento desnecessário com o SIFT, foi utilizado o espaço de cor HSV (*Hue Saturation Value*), para se verificar as cores predominantes nessas regiões e, assim, descartar regiões detectadas incorretamente.

Em seguida, as regiões candidatas resultantes são enviadas para a etapa de reconhecimento, de modo semelhante à abordagem sugerida por Silva et al. [2], entretanto, neste trabalho, apenas as regiões candidatas a terem uma placa de trânsito são processadas pelo algoritmo SIFT, que localiza os pontos chave e seus respectivos descritores e, então, os compara com os descritores dos pontos chaves extraídos de templates. Por fim, usa o mesmo algoritmo de reconhecimento de caracteres, usados por Silva et al. [2], para reconhecer os limites de velocidades.

V. EXPERIMENTOS REALIZADOS

Esta seção apresenta dois experimentos realizados com a proposta apresentada neste trabalho. O primeiro consistiu em aplicar o algoritmo TSRAS em um conjunto de 1.958 imagens, extraídas de um filme com nove minutos de duração, e uma taxa de captura de 29 quadros por segundo, sendo processados apenas os quadros múltiplos de oito (0, 8, 16, ..., 1.957). O segundo experimento consistiu em comparar os resultados obtidos com a abordagem proposta por Silva et al. [2] e os obtidos com o uso do algoritmo TSRAS.

A. Experimento I

Para a realização do treinamento AdaBoost foram utilizadas 5.300 imagens positivas, ou seja, imagens que contém placas de trânsito, distribuídas em quatro classes, conforme mostra a Tabela I, que também apresenta o número de estágios usados durante o treinamento (trabalhos encontrados na literatura [1] [5] [22] sugerem de 20 à 30 estágios) e o tamanho das imagens. Também foram usadas 11.200 imagens negativas, ou seja, que não possuem placas de trânsito. As imagens foram obtidas percorrendo um trajeto de aproximadamente 4.000 metros, com uma câmera de vídeo JVC Full HD GZ-EX210BUB, fixada sobre o teto de um veículo, e transitando a uma velocidade média de 30 km/h.

TABELA I. CLASSES DE PLACAS USADAS NO TREINAMENTO.

	Positivas	Negativas	Estágios	Tamanho
Circular	2.892	6.000	25	24 x 24
Alerta	608	1200	20	24 x 24
Pare	900	2.000	20	24 x 24
Preferência	950	2.000	20	24 x 24

A Fig. 4(a) mostra alguns exemplos de imagens positivas, enquanto que em (b) tem-se as mesmas placas sem a

informação referente ao sinal de trânsito, com apenas as bordas circulares, sem ou com as linhas diagonais de proibição, e uma placa apenas com seu fundo amarelo. Essas placas sem a informação do sinal, representam exemplos de imagens que foram usadas no treinamento AdaBoost, adotado na etapa de detecção, cujo objetivo é identificar as regiões da imagem que podem ter uma placa de trânsito.

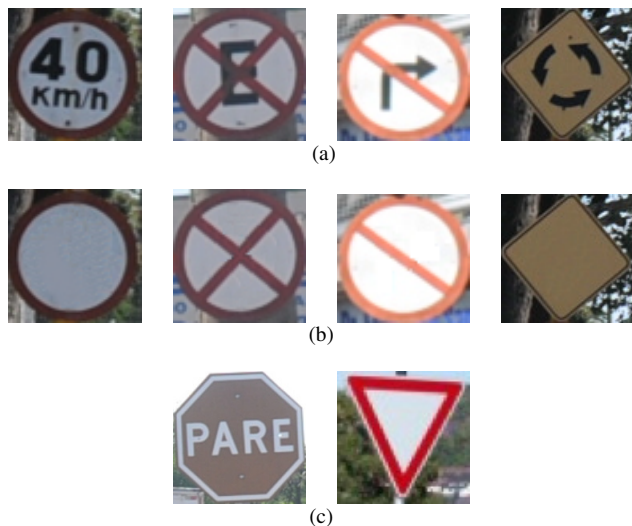


Fig. 4. Exemplos de imagens positivas: a) originais; b) usadas no treinamento, apenas com as informações básicas das placas; c) usadas diretamente no treinamento.

A Fig. 5 apresenta alguns exemplos de imagens negativas, usadas no treinamento, ou seja, imagens que não possuem placas de sinalização.



Fig. 5. Exemplos de imagens negativas (não possuem placas de trânsito) usadas no treinamento.

Os templates usados para se obter os pontos chave neste experimento são apresentados na Fig. 6. Estes templates foram escolhidos por serem os sinais de trânsito mais encontrados no trajeto percorrido.



Fig. 6. Templates das placas de trânsito extraídos do vídeo.

A Fig. 7 apresenta três exemplos de resultados obtidos com a metodologia apresentada neste trabalho, com o template à esquerda e o reconhecimento da placa de sinalização na imagem à direita.



Fig. 7. Resultados do processamento do vídeo.

B. Experimento II

Esta seção apresenta os resultados de um experimento com 60 imagens com resolução de 2144 x 1424 pixels, obtidas a partir de uma câmara Nikon D300S instalada sobre um veículo (Sistema de Mapeamento Móvel). As imagens foram obtidas com o veículo em movimento em condições normais de tráfego. Também é feita uma comparação com os resultados obtidos no trabalho de Silva et al. [2].

A Fig. 8 apresenta os *templates* usados neste experimento.



Fig. 8. Placas de sinais de trânsito usadas como template.

A Tabela II mostra o resultado obtido aplicando a metodologia proposta, sendo a primeira coluna referente aos índices numéricos existentes na Fig. 8.

TABELA II. RESULTADOS NO RECONHECIMENTO DAS PLACAS.

Templates	Metodologia proposta por Silva et al. [2]			Metodologia proposta neste trabalho		
	Acertos	Erros	Não reconhecidas	Acertos	Erros	Não reconhecidas
1	4 (100%)	0 (0%)	0 (0%)	3 (75%)	0 (0%)	1 (25%)
2	3 (100%)	0 (0%)	0 (0%)	2 (67%)	0 (0%)	1 (33%)
3	14 (73,68%)	0 (0%)	5 (26,32%)	16 (84,21%)	0 (0%)	3 (15,79%)
4	0 (0%)	0 (0%)	1 (100%)	0 (0%)	0 (0%)	1 (100%)
5	4 (100%)	0 (0%)	0 (0%)	3 (75%)	1 (25%)	0 (0%)
6	5 (100%)	0 (0%)	0 (0%)	3 (60%)	1 (20%)	1 (20%)
7	7 (77,78%)	0 (0%)	2 (22,22%)	5 (55,56%)	0 (0%)	4 (44,44%)
8	8 (100%)	0 (0%)	0 (0%)	8 (100%)	0 (0%)	0 (0%)
9	4 (66,67%)	0 (0%)	2 (33,33%)	6 (100%)	0 (0%)	0 (0%)
10	1 (100%)	0 (0%)	0 (0%)	1 (100%)	0 (0%)	0 (0%)
11	2 (100%)	0 (0%)	0 (0%)	2 (100%)	0 (0%)	0 (0%)
12	6 (100%)	0 (0%)	0 (0%)	5 (83,33%)	0 (0%)	1 (16,67%)
13	1 (100%)	0 (0%)	0 (0%)	1 (100%)	0 (0%)	0 (0%)
14	3 (75%)	0 (0%)	1 (25%)	3 (100%)	0 (0%)	1 (25%)

VI. CONCLUSÕES

O reconhecimento de placas de sinalização de trânsito é uma tarefa importante na construção de sistemas de apoio ao tráfego de veículos e, fundamental na implementação de veículos autônomos, porém, a implementação de algoritmos que apresentem tempos de processamento razoáveis continua sendo um grande desafio. O trabalho de Silva et al. [2] apresenta taxas de acerto razoáveis, entretanto, a aplicação do algoritmo SIFT em todas as imagens torna o tempo de processamento inadequado para aplicações em tempo real. Assim, a adaptação do algoritmo AdaBoost para selecionar as áreas com maior potencial para conter placas de sinalização, e a aplicação do algoritmo SIFT, apenas nestas áreas, conseguiu manter a qualidade dos resultados e, ainda, diminuiu sensivelmente o tempo de processamento, o que indica que esta é uma alternativa que merece atenção, na solução deste problema.

REFERÊNCIAS

- [1] P. Viola and M.J. Jones, "Robust real-time object detection", In: Second International Workshop on Statistical and Computational Theories of Vision – Modeling, Learning, Computing and Sampling, July 13, 2001.
- [2] F.A. Silva, A.O. Artero, M.S.V. Paiva e R.L. Barbosa, "Uma Metodologia para Detectar e Reconhecer Placas de Sinalização de Trânsito", In: VIII Workshop de Visão Computacional (WVC 2012). Anais... Goiás: UFG, 2012.
- [3] A. Martinović, G. Glavaš, M. Juribašić, D. Sutić and Z. Kalafatić, "Real-time Detection and Recognition of Traffic Signs", Proc. 33rd Inter. Convention on Information and Communication Technology, Electronics and Microelectronics – MIPRO 2010, v. III, 2010. p. 247-252.
- [4] X. Baró, S. Escalera, J. Vitrià, O. Pujol and P. Radeva, "Traffic Sign Recognition using Evolutionary Adaboost Detection and Forest-ECOC classification", In: IEEE Trans. on Intelligent Transportation Systems. 2009. p. 113-126.
- [5] S.-Y. Chen and J.-W. Hsieh, "Boosted road sign detection and recognition", In: International Conference on Machine Learning and Cybernetics, Proceedings... v. 7, pp. 3823–3826, July 2008.

- [6] I.M. Creusen, R.G.J. Wijnhoven, E. Herbschleb and P.H.N de With, "Color Exploitation in Hog-Based Traffic Sign Detection", Proc. IEEE 17th International Conference on Image Processing. Hong Kong, 2010, pp. 2669-2672.
- [7] G. Loy and A. Zelinsky, "Fast Radial Symmetry for Detecting Points of Interest", IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 25, no. 8, 2003, pp. 959-973.
- [8] R. Belaroussi and J.-P. Tarel, "A real-time road sign detection using Bilateral Chinese Transform", Proc. International Symposium on Visual Computing, ISVC, 2009, pp. 1161-1170.
- [9] A. Arlicot, B. Soheilian and N. Paparoditis, "Circular Road Sign Extraction from Street Level Images Using Colour, Shape and Texture Database Maps". In: Stilla U, Rottensteiner F, Paparoditis N (Eds) CMRT09. IAPRS, Vol. 38, France, 2009.
- [10] C.F. Paulo and P.L. Correia, "Traffic Sign Recognition Based on Pictogram Contours", Ninth Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS'2008), Klagenfurt, Austria, May 2008, pp. 67-70.
- [11] N. Barnes, A. Zelinsky, "Real-Time Radial Symmetry for Speed Sign Detection", Proc. Intelligent Vehicles Symposium, 2004, pp. 566-571.
- [12] H. Fleyeh, "Road and Traffic Sign Color Detection and Segmentation - A Fuzzy Approach", MVA2005 IAPR Conference on Machine Vision Applications, Tsukuba Science City, Japan, May 16-18, 2005, pp. 124-127
- [13] K. Brkić, "An overview of traffic sign detection methods", Department of Electronics, Microelectronics, Computer and Intelligent Systems Faculty of Electrical Engineering and Computing Unska 3, 10000 Zagreb, Croatia, Tech. Rep., 2010.
- [14] X. Baró and J. Vitria, "Fast traffic sign detection on greyscale images", Recent Advances in Artificial Intelligence Research and Development, p. 69-76, October 2004.
- [15] R. Timofte, K. Zimmermann and L. Van Gool, "Multi-view traffic sign detection, recognition, and 3d localization", Snowbird, Utah, 2009, pp. 69-76.
- [16] A. Ihara, H. Fujiyoshi, M. Takagi, H. Kumon and Y. Tamatsu, "Improved matching accuracy in Traffic Sign Recognition by using different deature subspaces", Machine Vision Applications (MVA2009), 3-26, p. 130-133, 2009.
- [17] K. Brkić, A. Pinz, and S. Šegvić, "Traffic sign detection as a component of an automated traffic infrastructure inventory system", Stainz, Austria, 2009.
- [18] Y. Freund and R.E. Schapire, "A Short Introduction to Boosting", Journal of Japanese Society for Artificial Intelligence, 1999, p. 771-780.
- [19] T. Acharya, A.K. Ray, "Image processing: principles and applications", John Wiley and Sons, 2005. 428 p.
- [20] R. Lienhart, and J. Maydt, "An extended set of Haar-like features for rapid object detection", In IEEE ICIP 2002, 2002, pp. 900-903.
- [21] Y. Freund and R.E. Schapire, "Experiments with a New Boosting Algorithm", Proc. Thirteenth International Conference on Machine Learning, p. 148-156, 1996.
- [22] B.B. Chaves, "Estudo do Algoritmo AdaBoost de Aprendizagem de Máquina aplicado a Sensores e Sistemas Embarcados". 2012. 119 f. (Mestrado em Engenharia Mecânica) – Escola Politécnica da Universidade de São Paulo, São Paulo, 2012.
- [23] Oza, N. C. Online Ensemble Learning. Tese de Doutorado, Berkeley: University of California, 2001.