

Estudo e Desenvolvimento de um Sistema de Reconhecimento de Sinais de Trânsito em Java

Study and Development of a Traffic Sign Detection and Recognition System in Java

RESUMO

Rafael Olivera Araujo Ogawa
rafael.ogawa@alunos.utfpr.edu.br
Universidade Tecnológica Federal do Paraná, Curitiba, Paraná, Brasil

Ionildo José Sanches
isanches@utfpr.edu.br
Universidade Tecnológica Federal do Paraná, Curitiba, Paraná, Brasil

Sistemas de reconhecimento de placas de trânsito podem ser usados para auxiliar e alertar os motoristas de sinais que não tenham sido vistos previamente, estes sistemas reduzem o número de acidentes e fatalidades. O objetivo deste trabalho consiste no desenvolvimento de um sistema de reconhecimento de placas de trânsito em imagens. O sistema desenvolvido utiliza a biblioteca OpenCV (*Open Source Computer Vision*), uma biblioteca de código fonte aberto implementada para aplicações no campo de aprendizado de máquina e visão computacional. Foi estudado também técnicas de aprendizagem profunda para gerar modelos computacionais para estender os tipos de placas a serem reconhecidas, utilizando o DeepLearning4J. Após a realização do treinamento foi possível criar um sistema capaz de detectar inicialmente placas de Parada Obrigatória. Os resultados preliminares apontaram para uma taxa de acerto de 68 por cento. Mais imagens positivas e negativas serão utilizadas no treinamento visando melhorar os índices de acertos, e novos testes serão realizados utilizando aprendizagem profunda.

PALAVRAS-CHAVE: Placas de trânsito. OpenCV. Aprendizagem profunda.

Recebido: 19 ago. 2019.

Aprovado: 01 out. 2019.

Direito autorial: Este trabalho está licenciado sob os termos da Licença Creative Commons-Atribuição 4.0 Internacional.



ABSTRACT

Traffic sign recognition systems can be used to assist and alert drivers of unseen signs, these systems reduce the number of accidents and fatalities. The aim of this work is the development of a traffic sign recognition system in images. The developed system uses OpenCV (*Open Source Computer Vision*) library, an open source library implemented for applications in the field of machine learning and computer vision. Deep learning techniques to generate computational models to extend the types of signs to be recognized using DeepLearning4J. After the training, it was possible to create a system capable of initially detecting Stop signs. Preliminary results pointed to a hit rate of 68 percent. More positive and negative images will be used in training to improve hit rates, and new tests will be conducted using deep learning.

KEYWORDS: Traffic sign. OpenCV. Deep Learning.

INTRODUÇÃO

Com o aumento contínuo do desenvolvimento de carros autônomos vem a necessidade de desenvolver sistemas de reconhecimento para placas de trânsito, além do uso do sistema em carros autônomos, é possível utilizar o reconhecimento de placas de trânsito para auxiliar e alertar os sinais que os motoristas não tenham percebido. Este sistema pode ser introduzido ao ADAS (*Advanced Driver Assistance Systems*), que são sistemas avançados de assistência ao motorista, aumentando a segurança do automóvel e consequentemente a segurança do tráfego nas cidades e rodovias.

A maioria dos acidentes de trânsito ocorre por erros humanos (BROOKHUIS; DE WAARD; JANSSEN, 2019), e um dos motivos para ocorrência dos acidentes é não respeitar a sinalização de trânsito, como por exemplo, as placas de parada obrigatória. Aproximadamente 1,35 milhão de pessoas morrem em decorrência de acidentes de trânsito (OMS, 2019), entre 20 e 50 milhões de pessoas sofrem lesões não fatais de acordo com a Organização Mundial da Saúde (OMS).

Os dados apresentados acima são uma das principais razões para os avanços tecnológicos nos veículos autônomos, um sistema do tipo ADAS pode reduzir o número de acidentes no trânsito, já que tais sistemas auxiliam os motoristas a terem uma condução mais segura. A proposta do ADAS é reduzir ou até mesmo eliminar os erros que são causados pelo motorista, fornecendo avisos, instruções e alertas em tempo real, este tipo de sistema também pode ser descrito como um Sistema de Suporte ao Motorista (DSS – *Driver Support Systems*).

Entre os sistemas inclusos no DSS pode-se citar a adaptação inteligente de velocidade (ISA), que consistem em um sistema para garantir que o veículo não exceda o limite de velocidade da via, a redução da velocidade pode ser feita automaticamente ou o motorista pode ser alertado para reduzir a velocidade. Este sistema pode utilizar uma tecnologia de reconhecimento óptico, em que um conjunto de câmeras reconhece os sinais de velocidade e as marcações na estrada, os dados coletados são comparados com a velocidade atual do veículo e assim é possível alertar o motorista ou diminuir a velocidade automaticamente (VAA; ASSUM; ELVIK, 2012).

De acordo com um estudo realizado na Suécia, o número de acidentes envolvendo carros que continham o sistema ISA teve uma redução aproximada de 49%, e uma redução de até 12% no número de acidentes com vítimas dependendo do limite de velocidade da via em que o acidente ocorreu. Dentre os sistemas de suporte ao motorista também se pode citar, o controle de cruzeiro adaptativo (ACC), o controle eletrônico de estabilidade (ESC) e os alertas ao motorista em caso de fadiga ou sonolência.

Os sistemas citados acima precisam de uma correta sinalização na via para que funcionem corretamente, as condições das placas de trânsito podem afetar o funcionamento dos sistemas de assistência ao motorista, uma placa suja, depredada ou até mesmo alterações climáticas podem afetar o desempenho destes sistemas, durante este trabalho será apresentado algumas soluções encontradas para minimizar os erros ao identificar placas de trânsito.

Este trabalho consiste na elaboração de um sistema de reconhecimento de placas de trânsito utilizando o algoritmo proposto por Viola e Jones (VIOLA; JONES,

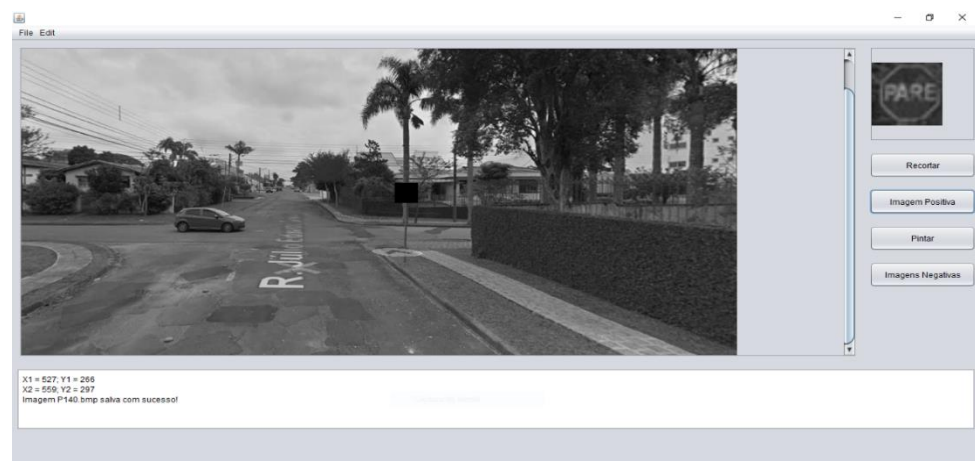
2004), utilizando aprendizado de máquina com as imagens adquiridas pelo Google Maps Street View de placas de trânsito, inicialmente utilizando apenas as placas de parada obrigatória, utilizando a linguagem Java para seu desenvolvimento. Além de um estudo sobre a biblioteca Deeplearning4j, utilizando redes neurais convolucionais.

METODOLOGIA

Para a elaboração deste trabalho foi feita uma aquisição de imagens de placas de trânsito, utilizando o Google Street View, que possibilita adquirir imagens de placas de trânsito em diversas cidades brasileiras, expostas em diferentes tipos de iluminação e nas mais diversas condições de conservação. Fornecendo as imagens em diversas situações, é possível expor o sistema a condições reais, melhorando a quantidade de placas detectadas. As imagens foram obtidas a partir da ferramenta de captura do Windows e posteriormente recortadas em um tamanho de 1010x 540 pixels com o objetivo de remover as barras de pesquisa e as outras informações presentes nas imagens.

Após a coleta de 200 imagens, foi elaborado um software utilizando a linguagem Java para recortar a figura positiva da imagem, neste caso a placa de parada obrigatória, e preencher o local em que a imagem foi recortada com um quadrado preto, como mostra a Figura 1. Em seguida a imagem resultante foi recortada em 100 imagens de diversos tamanhos para servirem como um conjunto de imagens negativas, que são utilizadas posteriormente no treinamento utilizando a biblioteca OpenCV (*Open Source Computer Vision*), que consiste de uma biblioteca de código fonte aberto implementada para aplicações no campo de aprendizado de máquina e visão computacional (OpenCV, 2014).

Figura 1 – Software para processamento e recorte das imagens



Fonte: Autoria própria (2019).

Antes de realizar o recorte da imagem, o software transforma a imagem para escala de cinza, diminuindo o processamento necessário para realizar o treinamento, após realizar o recorte da imagem e separar entre imagem positiva e imagens negativas, a imagem positiva é redimensionada pelo software para um tamanho de 100 x 100 pixels, logo todas as imagens positivas utilizadas durante este treinamento possuem o mesmo tamanho, como mostra a Figura 2.

As imagens negativas foram geradas em tamanhos aleatórios a partir da imagem original, retirando apenas a imagem positiva, alguns exemplos de imagens negativas podem ser vistas na Figura 3.

Figura 2 – Exemplos de imagem positiva



Fonte: Autoria própria (2019)

Figura 3 – Exemplo de imagem negativa



Fonte: Autoria própria (2019)

Partindo com um banco de imagens composto de 200 imagens positivas e 200 imagens negativas, foi feito o primeiro treinamento utilizando o algoritmo de Viola e Jones, que faz parte da biblioteca OpenCv. Para a realização do treinamento foram utilizadas as imagens positivas, compostas pelo objeto que se deseja detectar, e as imagens negativas compostas pelas áreas de fundo, após separar as imagens, o próximo passo é criar o vetor de imagens positivas (.vec), a partir da função *createsamples* da biblioteca OpenCV (VIOLA; JONES, 2004). Em seguida o treinamento é iniciado a partir da função *haartraing*, que também pertence a biblioteca do OpenCV, que gera um xml contendo o treinamento com as imagens de placas de trânsito. Este mesmo treinamento também foi realizado para uma um banco de imagens contendo 400 imagens positivas e negativas, e variando o tamanho da imagem positiva fornecida para o treinamento.

Um dos parâmetros para realizar o treinamento é a quantia de estágios, que afeta a duração e a qualidade do treinamento, para este trabalho, foram utilizados 15 estágios, para aumentar a quantidade de estágios seria necessário aumentar o banco de imagens.

A aprendizagem profunda (*Deep learning*) foi estudada a partir da biblioteca Deeplearning4j, desenvolvida para a linguagem Java, com o objetivo de utilizar a biblioteca para estender os tipos de placas a serem identificados pelo sistema. O estudo foi realizado a partir de uma Rede Neural Convolucional, CNN (*Convolutional Neural Network*), amplamente aplicada em treinamentos para reconhecimento e detecção de imagens. Uma rede neural é um modelo matemático baseada na conexão entre neurônios, em que as conexões se estabelecem em camadas apenas entre os neurônios adjacentes (SHUSTANOV; YAKIMOV, 2017).

RESULTADOS E DISCUSSÕES

A Tabela 1 contém os resultados obtidos no treinamento utilizando o OpenCV, utilizando um banco com 200 e 400 imagens e alterando o tamanho das imagens positivas. Os xml foram testados utilizando um banco de imagens contendo 100 fotos que não foram utilizadas durante o treinamento.

Tabela 1 – Primeiros Resultados

xml	Verdadeiro positivo	Falso Positivo
200 imagens de 24x24	56	2
200 imagens de 50x50	62	148
400 imagens 24x24	60	2
400 imagens 50x50	68	20

Fonte: Autoria própria (2019).

É possível verificar que o número de acertos aumentou conforme o número de imagens fornecidas para o treinamento, o maior número de acertos foi obtido utilizando um conjunto de 400 imagens de tamanho 50x50, um exemplo pode ser visto na Figura 4.

Figura 4 – Identificação da placa



Fonte: Autoria própria (2019).

Porém a quantidade de falsos positivos foi muito elevada em relação ao outro treinamento que utilizou 400 imagens, este problema pode ser visto na Figura 5. E foi contornado fazendo uma verificação adicional, verificando se dentro da área em que o programa considerava a existência de uma imagem positiva, a presença predominante da cor vermelha a partir das componentes matiz, saturação e valor do sistema de cores HSV (*Hue Saturation Value*). Com isso foi possível diminuir o número de falsos positivos de 20 para 4 em um banco de 100 imagens.

Figura 5 – Falso positivo



Fonte: Autoria própria (2019).

CONCLUSÕES

Neste trabalho foi feito um treinamento para identificação de placas de trânsito, utilizando a biblioteca OpenCV a partir da linguagem Java, focando na identificação das placas de parada obrigatória neste primeiro trabalho, o treinamento realizado apresentou resultados satisfatórios de identificação das placas. É necessário aumentar a quantidade de imagens positivas nos próximos trabalhos para garantir uma maior taxa de acerto. Na sequência, é necessário continuar os estudos utilizando a aprendizagem de máquina utilizando o Deeplearning4j para estender a quantidade de placas que podem ser identificadas.

REFERÊNCIAS

BROOKHUIS, Karel A.; DE WAARD, Dick; JANSSEN, Wiel H. Behavioural impacts of advanced driver assistance systems—an overview. *European Journal of Transport and Infrastructure Research*, v. 1, n. 3, 2019.

OMS, Organização Mundial de Saúde, Folha Informativa – Acidentes de Trânsito. Disponível em :
https://www.paho.org/bra/index.php?option=com_content&view=article&id=5147:acidentes-de-transito-folha-informativa&Itemid=779. Acesso em: 10 Ago. 2019.

SHUSTANOV, Alexander; YAKIMOV, Pavel. CNN design for real-time traffic sign recognition. *Procedia engineering*, v. 201, p. 718-725, 2017.

VAA, Truls; ASSUM, Terje; ELVIK, Rune. Driver support systems: Estimating road safety effects at varying levels of implementation. *TØI Report*, n. 1202/2012, 2012.

VIOLA, Paul; JONES, Michael J. Robust Real-Time Face Detection. In: *International Journal of Computer Vision*. Netherlands: CVPR, pp.137-154, 2004.