Speed Limit Sign Detection: Literature Review

# Academic Literature

Our group is focused on employing computer vision and machine techniques to detect speed limit and read the limits. When researching on this topic, we found scholars creating traffic sign recognition systems (TSR) that focus on differentiating the various signs and a few of them take it a step further by identifying the numbers on the speed limit signs. The main issue we faced when researching on this topic is that many researchers used a European traffic sign dataset to build their programs. The research papers that did use a USA traffic sign dataset, used the LISA Traffic Sign Dataset. Therefore, our group will also be using this dataset to create our computer vision driven program.

The researchers employed a variety of computer vision techniques, including edge detection using Hough transform to detect rectangles, Integral & Aggregate Channel features, Fast Radial Symmetry (FRS), thresholding for determining the actual speed limit values, and many more. The first important thing we learned from these articles is that due to speed limit signs being monochrome with a white background, it’s hard to detect them with color-based approaches and produces mixed results. The most common traffic sign detector created to date is the Fast Radial Symmetry detector. It’s essentially a voting-based detector that works directly on the edges of an image that can be extended to find rectangles. We noticed that some researchers used computer vision-based detection techniques for detecting the signs such as the previously mentioned shape detection algorithm FSR, Hough Transform, Histogram of Oriented Gradients (HOG), and then used modeling techniques such as neural networks and SVM to create the recognition algorithm that reads the text on the signs. Based on the research, we think that the best approach for our project is to use a shape/ edge-based detection algorithm to detect the speed limit signs and then either create a neural network (using TensorFlow or scikit-learn) or a machine learning algorithm such as SVM, to train a model to determine the limit values.

# Publicly Available Data, Tools, and Code

## Programming Language – Python

Python is an interpreted high-level programming language for general-purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, and a syntax that allows programmers to express concepts in fewer lines of code, notably using significant white space. It provides constructs that enable clear programming on both small and large scales.

Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has large and comprehensive standard libraries.

Python has many image intensive libraries like Python Imaging Library, VTK and Maya 3D Visualization Toolkits, Numeric Python, Scientific Python and many other tools available for numeric and scientific applications.

## Data

Our team is using the LISA Traffic Sign Dataset (<http://cvrr.ucsd.edu/LISA/lisa-traffic-sign-dataset.html>). This dataset contains is a set of videos and annotated frames containing US traffic signs. Each sign is annotated with sign type, position, size, occluded (yes/no), on side road (yes/no), all of which are in plain text .csv files. Additionally, it includes a set of Python tools to handle the annotations and easily extract relevant signs from the dataset.

## Code

Many people have taken a gander on creating their own traffic sign detection program. Below are a list of links that we believe will be useful when creating our own program:

1. <https://towardsdatascience.com/recognizing-traffic-signs-with-over-98-accuracy-using-deep-learning-86737aedc2ab>
2. <https://medium.com/@waleedka/traffic-sign-recognition-with-tensorflow-629dffc391a6>
3. <https://universalflowuniversity.com/Books/Computer%20Programming/Computer%20Vision%20and%20Image%20Processing/OpenCV%20with%20Python%20Blueprints_%20Design%20and%20develop%20advanced%20computer%20vision%20projects%20using%20OpenCV%20with%20Python.pdf>
4. <https://github.com/topics/traffic-sign-recognition>

# Industry Solutions

## Tesla

On a broad spectrum, in 2017, with version 8.0 of its software, Tesla employs a video camera, image-processing software, and radar that uses a crowdsourcing system called “fleet learning” to navigate driving on the highway. Before this update, radar wasn’t used as part of the primary system for autopiloting. Looking into the hardware side, “Eight surround cameras provide 360 degrees of visibility around the car at up to 250 meters of range. Twelve updated ultrasonic sensors complement this vision, allowing for detection of both hard and soft objects at nearly twice the distance of the prior system. A forward-facing radar with enhanced processing provides additional data about the world on a redundant wavelength that is able to see through heavy rain, fog, dust and even the car ahead.” The onboard computer runs a “...Tesla-developed neural net for vision, sonar and radar processing software… provides a view of the world that a driver alone cannot access, seeing in every direction simultaneously, and on wavelengths that go far beyond the human senses.”

Link: <https://www.tesla.com/autopilot>



## NVIDIA

NVIDIA provides hardware and data solutions to suppliers and automobile manufacturers for implementing computer vision into their vehicles. For example, its DRIVE AGX “... is a scalable, open autonomous vehicle computing platform that serves as the brain for autonomous vehicles.” NVIDIA’s DRIVE software “... enables key self-driving functionalities such as sensor fusion and perception.” This open full-stack solution features frameworks, source packages, compilers, and libraries for developing applications for autonomous driving and user experience.

Link: <https://www.nvidia.com/en-us/self-driving-cars/drive-platform/>

## RISP Vision

RISP Vision is a company that focuses on image processing and computer vision. In the automotive field, they have created a traffic signs detections software. The software detects a road sign at a distance and using machine learning and two classifiers, it determines if it’s a speed limit sign and if so, what that limit is.

Link: <https://www.rsipvision.com/road-signs-detection/>

## Vaisala

Vaisala has developed a RoadAI system that monitors signs and traffic with video recording. The system goes through a video feed and detects a sign many times. It then uses a video based, 3D point cloud system to localize all detection to a single position, then filters out any incorrect detections using computer vision.

Link: <https://www.vaisala.com/en/blog/2018-09/video-analysis-computer-vision-ultimate-format-traffic-sign-inventories>

# Bibliography

1. Balali, Vahid, Armin Ashouri Rad, and Mani Golparvar-Fard. "Detection, classification, and mapping of US traffic signs using google street view images for roadway inventory management." *Visualization in Engineering* 3.1 (2015): 15. <https://link.springer.com/article/10.1186/s40327-015-0027-1#citeas>.
2. Dewan, Prachi, et al. "An Overview of Traffic Signs Recognition Methods." *International Journal of Computer Applications (0975–8887) Volume* (2017). <https://pdfs.semanticscholar.org/5fd7/eca5432e4e4038dfc2ad425fa5f3e8739f0b.pdf>.
3. Islam, Kh Tohidul, and Ram Gopal Raj. “Real-Time (Vision-Based) Road Sign Recognition Using an Artificial Neural Network.” Ed. Simon X. Yang. *Sensors (Basel, Switzerland)* 17.4 (2017): 853. *PMC*. Web. 18 Oct. 2018. <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5424730/>.
4. Javidi, Bahram, et al. "Automated detection and analysis of speed limit signs." *University of Connecticut, Tech. Rep. JHR*(2002): 02-285. <http://docs.trb.org/00824628.pdf>.
5. Keller, Christoph Gustav, et al. "Real-time recognition of US speed signs." Intelligent Vehicles Symposium, 2008 IEEE. IEEE, 2008. <https://lmb.informatik.uni-freiburg.de/people/bahlmann/data/ke_sp_ba_ba_gi_iv08.pdf>.
6. Miyata, Shigeharu. "Automatic Recognition of Speed Limits on Speed-Limit Signs by Using Machine Learning." *Journal of Imaging* 3.3 (2017): 25. <http://www.mdpi.com/2313-433X/3/3/25/pdf>.
7. Møgelmose, Andreas. *Traffic Sign Detection Using Computer Vision*. 18 May 2012, <https://projekter.aau.dk/projekter/files/63218113/report.pdf>.
8. Møgelmose, Andreas, Dongran Liu, and Mohan Manubhai Trivedi. "Detection of US traffic signs." *IEEE Transactions on Intelligent Transportation Systems* 16.6 (2015): 3116-3125. <http://cvrr.ucsd.edu/publications/2015/signsITSTrans2015.pdf>.
9. Møgelmose, Andreas, Mohan M. Trivedi, and Thomas B. Moeslund. "Vision-Based Traffic Sign Detection and Analysis for Intelligent Driver Assistance Systems: Perspectives and Survey." *IEEE Trans. Intelligent Transportation Systems* 13.4 (2012): 1484-1497. <http://vbn.aau.dk/ws/files/72574480/survey.pdf>.
10. Stallkamp, Johannes, et al. "Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition." *Neural networks* 32 (2012): 323-332. <https://www.ini.rub.de/upload/file/1470692859_c57fac98ca9d02ac701c/stallkampetal_gtsrb_nn_si2012.pdf>.