**Proceedings of the ASME 2017 Dynamic Systems and Control Conference**
**DSCC2017**
**October 11-13, 2017, Tysons, Virginia, USA**

# DSCC2017-5187

# TRAFFIC SIGN RECOGNITION IN AUTONOMOUS VEHICLES USING EDGE DETECTION

**Harish Vishwanathan**
Kettering University
Flint,Michigan,USA

**Diane L. Peters**
Kettering University
Flint,Michigan,USA

**James Z. Zhang**
Kettering University
Flint, Michigan,USA

## ABSTRACT

Autonomous vehicles have the potential to improve safety by eliminating human error in driving, as well as providing mobility to those who cannot safely drive. Such vehicles do require new technology to monitor their environment and ensure that they are operating safely. One such technology that will be necessary is the ability of the vehicle to recognize traffic situations and traffic signs. This can be accomplished by an appropriate implementation of edge detection methods. In this paper, we compare three different edge detection methods: Canny method, Sobel method and Zhang method. This comparison was conducted on both still pictures and on a video. When analyzing the video, which was taken on a clear day with an undamaged and clearly visible stop sign, all three methods performed equally well; the time at which the stop sign was identified, based on the edge map, was the same. The purpose of this comparison is to evaluate the performance of each of the three methods, in the context of the problem of identifying traffic signs. The methods are compared on still images of a stop sign under various conditions, in addition to the single video comparison. Based on the still image comparison, we conclude that Zhang's method (linear prediction) generates the best edge map, particularly when the images include snow, ice, rain or other factors and even at night vision.

*Index Terms*—Edge Detection , Autonomous vehicle, Traffic signs, Canny Method, Sobel Method, Zhang Method, STOP sign

## 1. INTRODUCTION

In 2015, the United States of America had the highest one year percentage increase in traffic deaths in half a century [1]. 38,300 people were killed on roads and roughly 4.4 million were reported to have sustained injuries that resulted in medical consultations. Every year the number of deaths due to road accidents has been increasing by 8%. Driver behavior and concentration have great impact on safety while driving a vehicle. Introduction of intelligent driving assistant systems in cars which adapt to road conditions, give lane departure warnings (LDW), parking assist, blind spot detection, and many more systems designed to increase safety have reduced the number of road accidents to a certain extent. Additional work on active safety systems is a high priority for automotive manufacturers, as is the development of fully autonomous vehicles. Autonomous vehicles that can plan and execute a route without driver input have the potential to drastically reduce accidents caused by driver fatigue, error, or inattention. Such vehicles would include functions such as the recognition of common traffic signs and signals, in order to properly respond to them.

An autonomous car [2] is a self-driving car that is capable of sensing its environment and navigating without human input. This autonomous vehicle has an advanced control system which interprets sensory information and uses a variety of techniques such as radar, LIDAR, GPS, odometry and computer vision to identify its path and to avoid obstacles. Cameras which capture video of a car's surroundings are an important part of the sensor suite, and the use of vision to detect signs has been a subject of research interest. The Canny method was used in [3] for edge detection, with a geometrical analysis carried out on the extracted edges. In [4], color thresholding and shape analysis was used to detect signs, with a neural network for the recognition problem. Another approach, the use of a neural network and genetic algorithm, was implemented in [5]. This research paper focuses on implementing edge detection algorithms in an autonomous vehicle for the purpose of detecting and recognizing traffic signs, specifically the STOP sign. By recognizing the sign, the car's control system would then be able to react appropriately.

This paper is structured as follows: in Section 2, a general discussion of edge detection algorithms is presented. Next, in Section 3, three different algorithms are discussed, including the mathematical basis for each. In Section 4, the image processing of the stop sign is explained. In Section 5 the application of the algorithms is presented for the recognition of a "STOP" sign in a video. This section also discusses the results of the application. Finally, Section 6 presents conclusions and outlines future work on this problem.

## 2. EDGE DETECTION

Edge detection is commonly known as the set of mathematical methods which aims at identifying points in a digital image at which the image brightness changes sharply or, more formally, has discontinuities. A set of curved lined segments termed edges are organized at which the image brightness changes sharply. In this application, image edges are defined in terms of pixel locations where gray – level changes take place. An edge is a sharp mutation at the gray level of images that characterize the boundary of objects. It reduces the unnecessary information while maintaining the structure of the image [6]. This research paper focuses on implementing edge detection methods in an autonomous vehicle, through which a vehicle would be able to detect a "STOP" sign by using the edges of the sign. Similar work has been done in many contexts, e.g., [7-12]; however, research on a comparison of various methods has not been conducted.

The method of identifying discontinuities in a 1D signal is known as 'Step Detection' and similarly finding signal discontinuities over a period of time is known as 'Change Detection'. In edge detection, step edge detectors have been a crucial part of many image processing and computer vision systems. The edge detection process serves to extract the structural information of an object in a picture by eliminating irrelevant information in that picture. There are many approaches to edge detection, with different requirements and benefits; however, there are some common pre-requisites for all methods. There are three major steps in any edge detection process, as depicted in Figure 1.

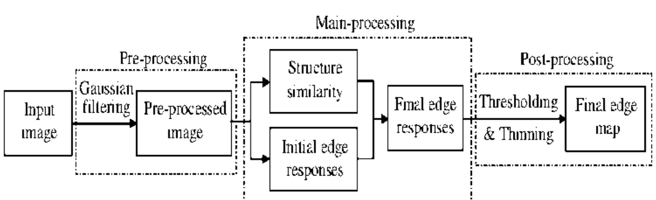


**Figure 1: Flow Chart of Input Image Converted to Final Edge Map**

As shown in Figure 1, an input image is converted to an edge map in three stages namely pre-processing, main-processing and post-processing. Preprocessed images are further processed to generate final edge responses in the main processing stream followed by thresholding and thinning edge to create the final edge map in post-processing stage. The final mapped image varies with the edge detection methods used and also based on the noises and filter coefficients. Figure 2 shows the edge map of the image 'LENA' which is taken from [13].

*Figure 2: Final Mapped edge of image LENA [13]*

In the same manner, for the STOP signs the autonomous vehicle is fitted with a camera which will capture the images while it is in motion. This video can be processed as a series of still images, which can be processed as outlined in Figure 1. The processed images can then be examined to see whether the edges detected give an appropriate octagon shape to indicate a stop sign. When it detects the octagon shape, the vehicle can be programmed to stop.

As there are multiple edge detection methods, the question of which is best suited for this application is an important one. In this work, we compare three methods for edge detection: the Canny method [13], which was used in [3], Zhang method [14] and Sobel Method [15]. The parameters of these detectors must be chosen to yield outwardly engaging results. Despite the fact that there are a few techniques to quantitatively quantify the visual nature of edge detectors, none are widely accepted by analysts [16-18].

## 3. TYPES OF EDGE DETECTION

In one dimension, we can characterize the position of a step edge in space with one position coordinate. But in two dimension edge is said to have even an orientation with the position coordinate. For 2- D input signal $x(i,j)$, the output $y(i,j)$ of the digital filter can be expressed as the 2-D convolution summation [19]

$$y(m,n) = \sum_{i=0}^{N1}\sum_{j=0}^{N2} x(i,j)h(m-i,n-j) \quad (1)$$

$$= \sum_{i=0}^{N1}\sum_{j=0}^{N2} x(m-i,n-j)h(i,j) \quad (2)$$

where $(N_1, N_2)$ are the orders of a nonrecursive 2-D digital filter. The most essential for detecting the edges in images is 2-D filtering. To enhance the edge of the image, a matrix H, known as the convolution mask is used. With respect to the different detectors, H can represent a different approximation of the image gradient or either the first or second derivative of the "Gaussian smoothing kernel" given by [20]

$$G(i,j) = e^{-(i^2+j^2)/2\sigma^2} \tag{3}$$

Where $(i,j)$ are the coordinates of the discrete image and $\sigma$ is the standard derivation of the associated Gaussian probability distribution, the images are assumed to have additive white Gaussian noise.

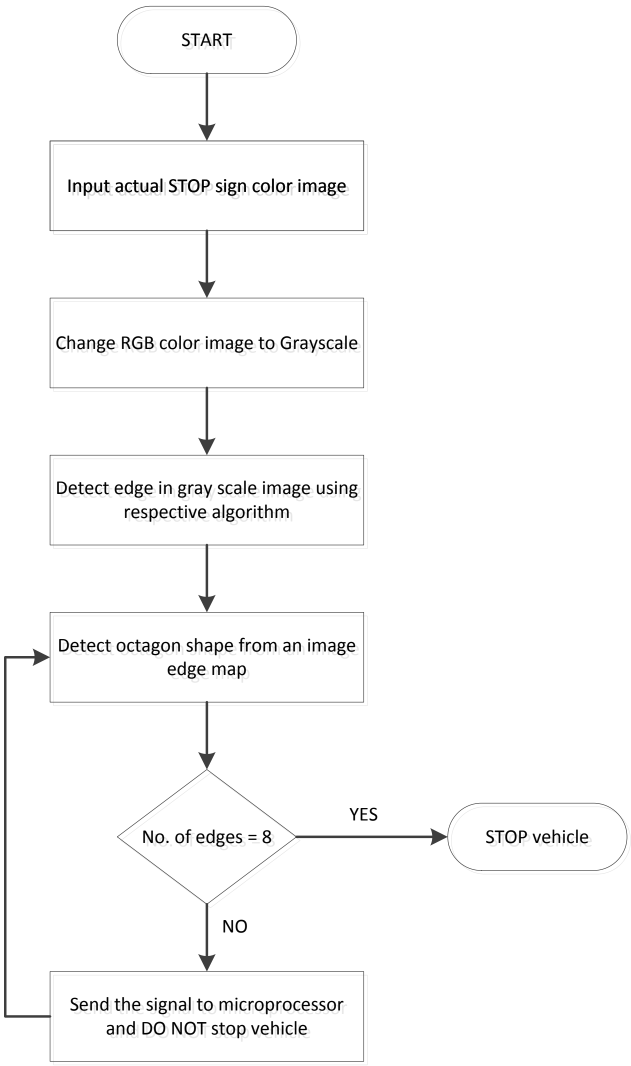


**Figure 3: Application and working of Edge Detection method for STOP sign**

*Canny's Method*

Canny's method of edge detection is a commonly used edge detection [13]. In this method the main criteria is to lower the error rate, the distance between the points marked by the detector and the center of the true edge is minimized. In other words the points marked as edge points by the operator should be as close as possible to the center of the true edge (Good localization). To detect edges of a particular orientation, we create a two dimensional mask for the orientation by convolving a linear edge detection function aligned normal to the edge direction with a project function parallel to the edge direction. If we convolve an image with an operator $G_n$ which is the first derivative of Gaussian smoothing Kernel from equation in some direction $n$ as [13]

$$G_n = \frac{\partial G}{\partial n} = n \cdot \nabla G \tag{4}$$

Where $n$, is oriental normal to the direction of an edge to be detected, a robust estimate for the smoothed gradient direction is given by

$$n = \frac{\nabla(G * u\,)}{|\nabla(G * u\,)|} \tag{5}$$

where * denotes convolution. This is a very good estimator for edge normal direction for steps, since a smoothed step has strong gradient normal to the edge. An edge point is defied to be a local maximum in the direction of $n$ of the operator $G_n$ applied to the discrete image$u(i,j)$. At a local maximum, we have

$$\frac{\partial}{\partial n}\, G_n * u = 0 \tag{6}$$

And substituting for $G_n$ from equation and associating Gaussian convolution, we get

$$\frac{\partial^2}{\partial n^2}\, G * u = 0 \tag{7}$$

Now the magnitude of the strength of the edges $g(i,j)$ by utilizing the convolving mask with the discrete image u as

$$g(i,j) = |G_n * u| \tag{8}$$

Finally to obtain the edge map $e(i,j)$ of the image, the Canny detector utilizes thresholding with hysteresis. In this method we select edges on $g(i,j)$ which are above high threshold $\tau_H$, secondly we select points with strong responses and are above low threshold $\tau_L$. such that $\tau_H > \tau_L$.

Hence, for the Canny detector, we are required to adjust three parameters to achieve the best results, namely, the high threshold $\tau_H$, the low threshold $\tau_L$ and the standard deviation, $\sigma$.

*Zhang's Method*

Zhang's method of edge detection follows the principle of linear prediction [14], a well-known filtering method used to predict the future values of a signal using the past and the present values. The main idea behind this linear prediction method is to optimize filter coefficients in order to minimize the prediction errors. Because edge points in an image are sharp changes in intensity, therefore, large prediction errors occur at these points compared with other points where intensity changes are gradual. Hence, the pixels with large prediction errors compared with a pre-defined threshold are the desired edge points.

Let $x[n]$ be the predicted signal value and $x[n-k]$ be the past values, the linear predictor of order p is given as

$$x[n] = -\sum_{k=1}^{p} a_p(k)x[n-k] \tag{9}$$

Where $a_p(k)$ are the predictor coefficients. If *e[n]* is the prediction error, the error generated by the estimate is

$$e[n] = x[n] - \hat{x}\,[n] = \sum_{k=0}^{p} a_p(k)x[n-k] \tag{10}$$

The MMSE (Minimum Mean Square Error) of e[n] is derived as follows

$$E\{x[n]x[n-l]\} + E\left\{\sum_{k=1}^{p} a_p(k)x[n-k]\,x[n-l]\right\} = 0 \tag{11}$$

where E is the expectation operator. From the equation above it can be obtained that

$$r_{xx}[l] = -\sum_{k=1}^{p} a_p(k)r_{xx}[l-k] \tag{12}$$

Where $l$=1, 2, 3…p. 'p' being the number of filter coefficients.

By solving the filter coefficients and by following the orthogonality principle, the MMSE can be expressed as

$$\min\{\epsilon_p^f\} = \min\{E[|e[n]|^2]\} = r_{xx}(0) + \sum_{k=1}^{p} a_p(k)r_{xx}(-k\ \ ) \tag{13}$$

Where $r_{xx}(-k)$ is the conjugate of the $k^{th}$ autocorrelation value of $x[n]$.

Consider an image I, its edge detection using Zhang's method along positive and negative directions of x and y to generate predicted images that are given as [14]

$$I_{x,error} = I'_{+x} - I'_{-x} \tag{14}$$

$$I_{y,error} = I'_{+y} - I'_{-y} \tag{15}$$

respectively. The values or the errors are effectively doubled because the direction of detection generates the same errors with opposite signs. The edges of the image after amplifying the errors can be determined by [5]

$$I_{edge} = \sqrt{I_{x,error}^2 + I_{y,error}^2} \tag{16}$$

The amplification of errors makes the threshold of separating edges from background less critical because the errors are suppressed by linear prediction in the areas of the signal where changes are gradual.

*Sobel Method*

The Sobel method is the edge detection method which follows the principle of image gradient [15], if the discrete image $u(i,j)$ is defined as a function $f(x,y)$ where $f$ represents the light intensity of the image at position$(x,y)$. In this condition $x\ and\ y$ are two variables. This method used two kernels each of order (3, 3) named as $G_x$ and $G_y$ which are convolved with the original image to calculate approximations of the derivatives, one along the horizontal changes and one along the vertical direction.

Sobel method uses two kernels that approximate the gradient $\nabla f(x,y)$ of the continuous function $(x,y)$. For the derivative of the continuous functions Sobel method uses the approximations [20]

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * u \tag{17}$$

$$G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & 2 & +1 \end{bmatrix} * u \tag{18}$$

At each point in the image, the resulting gradient approximations can be combined to give the gradient magnitude, using:

$$G(i,j) = \sqrt{(G_x)^2 + \left(G_y\right)^2} \tag{19}$$

If $G(i,j)$ exceeds some threshold τ the pixel location $(i,j)$ is declared as an edge location. This edge location constitutes the edge map $e(i,j)$ which is defined by the thresholding operation as [21]

$$e(i,j) = \{1, \quad (i,j)\ \in I_g$$

where

$$I_g = \{(i,j): G(i,j) > \tau\ \} \tag{20}$$

Sobel detectors are based on measures of the image gradient. The pixel location $(i,j)$ is declared an edge location if the magnitude of the gradient of the image exceeds some threshold τ. In this method one must select the parameter threshold τ for best results.

## 4. IMAGE PROCESSING

The majority of the edge detection algorithms contain four stages: filtering, enhancement, detection and location. At filtering stage our purpose is to suppress the noises of the images while retaining the real edges, at enhancement stage we find the filter that has a good response to the detection of edge points. It is a process that removes from a signal some unwanted components or features. Filtering means removing some frequencies and not others in order to suppress interfering signals and reduce background noise, at the detection stage we can detect the real edges by the Threshold methods and finally at the location stage we can detect the orientation and location of the edges [6].

For image processing the image is captured using a wireless camera which will be able to capture either images continuously of a traffic sign or a video of the car approaching towards the traffic sign, specifically, a STOP sign. This captured image is transmitted to a laptop or PC using the wireless transmitter in the case of a continuous video it is converted to a series of image frames based on the number of seconds using CV2 software [22]. An image is converted to a matrix and then various operation is performed to get desired results and values, for instance the captured image is converted to a grey scale from RGB color after filtering and based on the coding the eight edges (octagon shape) of a STOP sign are located and the edges of the STOP sign are mapped and sent to the microprocessor to stop the vehicle. If the eight edges are not found, the codes run the entire series of image frames until it finds a potential STOP sign edges.

Edge detection produces edges of an image, which highlights the intensity changes. The boundaries of the frames tend to produce sudden changes in the image intensity. For example, different frames are usually different colors or hues and this causes the image intensity to change as we move from one frame to another [23]. In addition, different surfaces of an image receive different amount of light, which again produces intensity changes [24]. Therefore the intensity boundary information that we extract from an image will tend to indicate object boundaries.

## 5. SIMULATION AND RESULTS

In the last decade, road sign detection has attracted great attention as it requires automatic road sign detection especially for images with cluttered background. STOP sign detection is difficult due to reasons such as large component of non-rigidity and textural difference, detection is also made difficult due to dust, snow or even ice. These features increase the variability of road sign patterns. As discussed before, change in light source distribution can also cause a significant change in the appearance of the STOP sign image.

Unlike other approaches to traffic sign detection, the system proposed herein does not depend on color information, but rather uses machine learning by detecting the edges of a STOP sign. This yields reliable classification and robust traffic sign detection under various illumination and weather conditions.

An important part of the edge detector evaluation is that its result should correlate with visual perception. To compare the three major edge detection methods and compare the quality of edge map, we process the three methods on a “STOP” sign under various conditions.

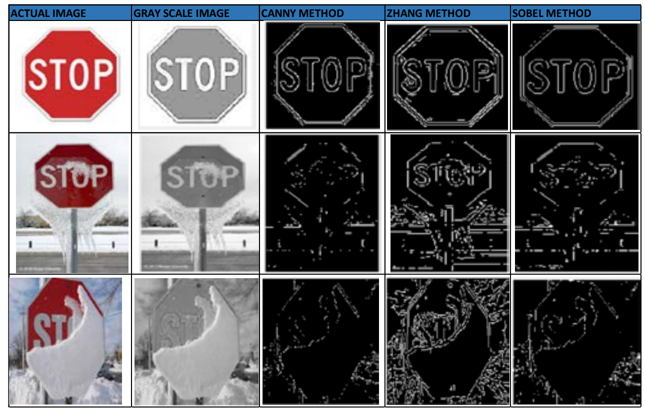


**Figure 4: Comparison of STOP sign using three edge detection methods**

Figure 4 shows the comparison of STOP sign using edge detection methods at normal conditions, covered with ice and during snow, where an image is converted to a grey scale image and edge is mapped in respective methods. It is evident that during ideal conditions all the three methods are effective at edge detection and provide equal and better results, but under various conditions like snow and ice Zhang’s edge detection method (Linear Prediction method) provides better result when compared to the other two methods. Figure 4 shows that the resulting edge quality can vary greatly with the edge detectors internal parameters, it is seen that Zhang’s method produced the most appealing edge maps in the sense that edges are smooth and consistent where Canny and Sobel produce edge maps with a more patchy look. The areas for feature extraction are very well defined and relatively independent from adjacent edges. Well extracted edges representing desired features and a good suppression of neighboring edges will certainly provide a higher detection ratio.

According to Zhang’s method, in order to accurately extract these features, the edge map need to clearly contain the necessary features with information surrounding the features be suppressed to minimize measurement errors. During snow and icy conditions Canny and Sobel methods could not detect desired edges (octagon shape) accurately as the definitions are not clean, but Zhang’s method extracts the eight edges accurately in spite of various types of noises in the image.

 

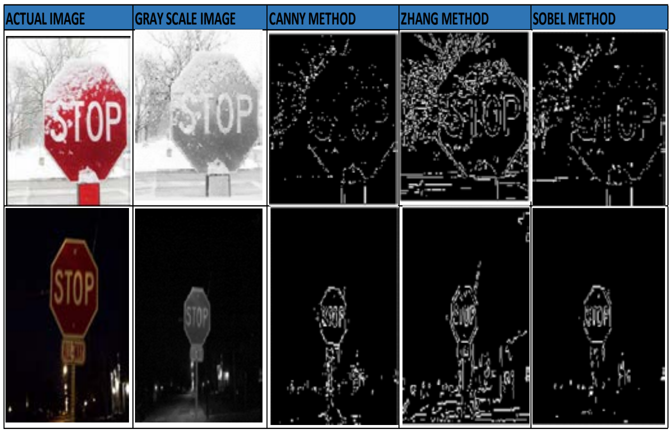


**Figure 5: STOP sign detection using three edge detection methods during snow storm and at night**

Figure 5 shows the STOP sign detection using three edge detection methods during a snow storm and at night. It is evident that even during a snow storm or at night vision Zhang method extracts the eight edges of a STOP sign more accurately than the other two methods. Hence, Zhang method yields a more reliable classification in spite of receiving less amount of light and intensity and provides a more robust traffic sign detection under various illumination and weather conditions.

While incorporating these various edge detection methods in vehicle using a camera at normal conditions, it is observed that all the three methods detect the edges accurately and identify the octagon shape at the same time without any time delay. Figure 6 shows the experimental simulation done at vehicle level.

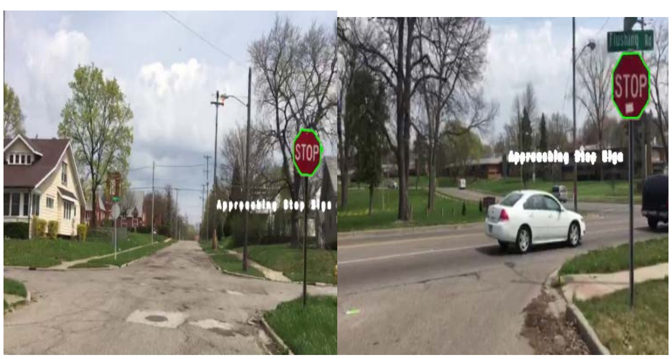


**Figure 6: Linear Prediction method identifying stop sign in a vehicle at normal conditions**

The "STOP" sign is detected using the radius and length criteria, the coordinates in the image frame are specified in order to detect the octagon shape (8 edges). It follows the same algorithm as shown in Figure 3. Due to the noise in the images, trees and clouds in the octagon shape were also detected (false positive) but was not considered as the "STOP" sign. When the actual "STOP" sign is detected "Approaching stop sign" text flashes in the image frame and vehicle comes to stop as soon as the signal is processed and sent to the ECU (Electronic Control Unit). Figure 7 shows the false positive which was detected during the experiment.

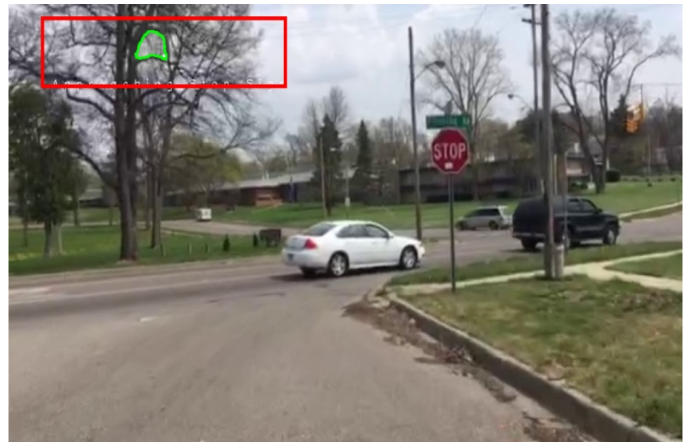


**Figure 7: False Positive detected in the image frame during simulation**

Running the same programs in MATLAB considering the entire frame for STOP sign detection also produces few false positives.

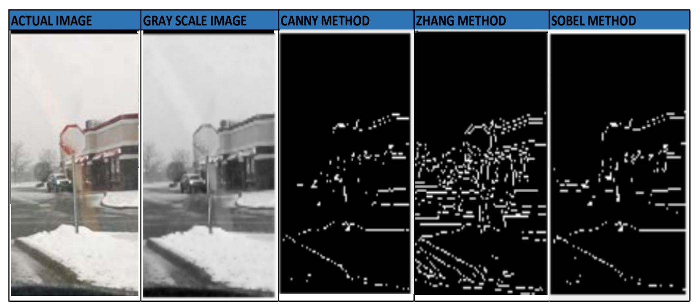


**Figure 8: Considering the entire frame for STOP sign detection (edge detection) during snow.**

Edge mapping done by Zhang's method in Figure 8 has better accuracy in detecting the STOP sign when compared to the other two methods but at the same time it also produces various edges while considering the entire frame resulting in false positive which needs to be controlled by either monitoring the filter coefficient or by setting the angle and direction of detection of the STOP sign (eight edges).

However there are also situations when the STOP signs are situated at the left side of a road due to which we cannot set the direction of the detection, Figure 9 shows one such situation.


**Figure 9: STOP sign located at the left side of a road**

Edge mapping of the frame when the STOP sign is situated at the left side of the road is explained in Figure 10. There are still some challenges, which requires further intensive research in order to eliminate the false positive while considering the entire frame in order to detect the eight edges of a STOP sign. One approach might be to require a stricter criterion for detection of a STOP sign in a non-standard position, since it is less likely to occur; this is a subject of future research.

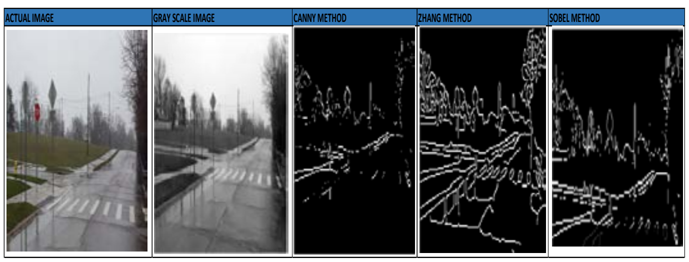


**Figure 10: Edge mapping of a frame where STOP sign is situated at the left side of a road**

There are also situations when the STOP signs are disoriented or damaged. Figure 11 shows one such situation when a STOP sign is disoriented due to the pole being bent, which might have been caused due to an accident or bad weather conditions. It is very evident that even during these conditions the STOP sign is detected and the edges are mapped using Zhang's method.

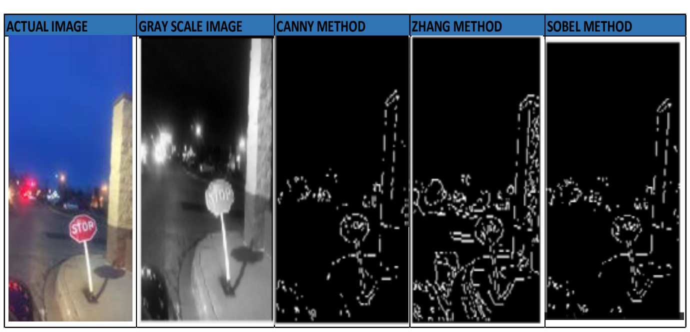


**Figure 11: Edge Detection of a disoriented STOP sign**

## 6. CONCLUSIONS AND FUTURE WORK

From various simulations, we can conclude that Zhang's method (Linear Prediction) maps the edges best under various conditions, at normal conditions all the three methods invariably provide good result and maps out edge at the same time. Overall, the results show that linear prediction method is less sensitive to the conditions of the original images than the Canny and Sobel methods. This did not affect the results using the video, since it was recorded under "ideal" conditions.

Zhang's method is considered the preferred method when compared to Canny and Sobel method. In spite of various noise in the frame, the edges are mapped and detected clearly in Zhang's method when compared with the other two. All three methods identify the STOP sign almost at the same time and map out edges clearly at ideal conditions, whereas during snow or rain the Canny method maps out edges in the background caused by the snow or rain, which hinders identification of the STOP sign. In the case of the Sobel method, the STOP sign is not identified when the frame is covered with snow or rain. Based on these observations, Zhang's method is the best method at various conditions since it maps out the STOP sign edges accurately under a variety of conditions.

In future work, additional videos will be analyzed, using non-ideal conditions. Such conditions could include branches obstructing a clear view of the stop sign, different lighting and weather conditions such as bright sun, fog, rain, snow, or darkness. The sign could also have snow or sleet on it, large amounts of dirt, or some form of minor damage.

Future work will also include the detection and recognition of different types of signs, including some that convey different information, such as speed limit signs. Detection and recognition of stoplights is also an important problem, which could be addressed using edge detection. These problems will all be studied to determine which edge detection method is most appropriate under different conditions, and how to best tune the methods.

As the work is carried out, successful implementation of the edge detection methods will be integrated into a larger program of research into autonomous vehicles, as an input to a vehicle control system.

## ACKNOWLEDGMENTS

The authors would like to thank Kettering University Office of Sponsored Research for funding the research, Govindrajan Chebrolu [MS Computer Science, Georgia Tech] who helped with eclipse, python and CV2 software and Dr. Rebecca Reck of Kettering University for her valuable advice and input throughout the course of this work.

## REFERENCES

[1] 'Fatality Analysis Reporting System' 2012, fars.NHTSA.dot.gov Retrieved July 3, 2014.

[2] Lassa, Todd (January 2013). "The beginning of the end of driving". *Motor Trend*. Retrieved 1 September 2014.

[3] Piccioli, G., De Micheli, E., Parodi, P. & Campani, M. (1996). Robust method for road sign detection and recognition. *Image and Vision Computing*, *14*(3), 209-223.

[4] De La Escalera, A., Moreno, L. E., Salichs, M. A., & Armingol, J. M. (1997). Road traffic sign detection and classification. *IEEE transactions on industrial electronics*, *44*(6), 848-859.

[5] De la Escalera, A., Armingol, J. M., & Mata, M. (2003). Traffic sign recognition and analysis for intelligent vehicles. *Image and vision computing*, *21*(3), 247-258.

[6] Abbaszadeh, A., & Daneshvar, S. (2016). A review of different types of edge detection methods. *Journal of Current Research in Science*, (1), 871.

[7] Park, J. G., & Kim, K. J. (2013). Design of a visual perception model with edge-adaptive Gabor filter and support vector machine for traffic sign detection. *Expert Systems with Applications*, *40*(9), 3679-3687.

[8] Hossain, M. S., & Hyder, Z. (2015). Traffic road sign detection and recognition for automotive vehicles. *International Journal of Computer Applications*, *120*(24).

[9] Zadeh, M. M., Kasvand, T., & Suen, C. Y. (1998). Localization and recognition of traffic signs for automated vehicle control systems. In *Intelligent Systems & Advanced Manufacturing* (pp. 272-282). International Society for Optics and Photonics.

[10] de Saint Blancard, M. (1992). Road sign recognition: A study of vision-based decision making for road environment recognition. In *Vision-based vehicle guidance* (pp. 162-172). Springer New York.

[11] de la Escalera, A., Armingol, J. M., & Salichs, M. A. (2001). Traffic sign detection for driver support systems. In *International conference on field and service robotics*. Universidad Carlos III de Madrid, Leganés, Madrid, Spain.

[12] Piccioli, G., De Micheli, E., Parodi, P., & Campani, M. (1994, October). Robust road sign detection and recognition from image sequences. In *Proceedings of the Intelligent Vehicles' 94 Symposium,* (pp. 278-283). IEEE.

[13] Canny, J. (1986). A computational approach to ddge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-8(6): 679-698.

[14] Zhang, J. Z., et al. (2010). A novel image edge detection method using Linear Prediction. In *53rd IEEE International Midwest Symposium on Circuits and Systems*.

[15] I.E. Sobel, “Camera models and machine perception.” Ph.D. dissertation, Stanford University, Stanford, CA, USA, 1970.

[16] Peli, T., & Malah, D. (1982). A study of edge detection algorithms. *Computer graphics and image processing*, *20*(1), 1-21.

[17] Argyle, E., & Rosenfeld, A. (1971). Techniques for edge detection. *Proceedings of the IEEE*, *59*(2), 285-287.

[18] Kanungo, T., Jaisimha, M. Y., Palmer, J., & Haralick, R. M. (1993). A quantitative methodology for analyzing the performance of detection algorithms. In *Proceedings of the Fourth International Conference on Computer Visio.*, (pp. 247-252). IEEE.

[19] Hayes, M. H. (1996). *Statistical Digital Signal Processing and Modeling*. New York: J. Wiley & Sons. ISBN 978-0471594M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[20] Jain, A. K. (1989). *Fundamentals of Digital Image Processing*. Prentice-Hall, Inc..

[21] Teixeira, F. C. A., et al. (2009). An optimization method for edge-detector parameter tuning based on visual perception. *2009 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing*.

[22] Betgeri, N. M. & Ketaki, K. W. (2013). Image processing based traffic sign recognising system. *International Journal of Electronics Communication and Computer Engineering*, 4(6) ISSN (Online): 2249–071X.

[23] Ahirwal, B., Khadtare, M., & Mehta, R. (2007). FPGA based system for Color Space Transformation RGB to YIQ and YCbCr. In *International Conference on Intelligent and Advanced Systems*.

[24] Poornima, B., Ramadevi, Y. & Sridevi T. (2011). Threshold based edge detection algorithm. *IACSIT International Journal of Engineering and Technology*, 3(4).

 