Available online at www.ijpe-online.com

International Journal of Performability Engineering

vol. 16, no. 10, October 2020, pp. 1588-1597
DOI: 10.23940/ijpe.20.10.p10.15881597

A Survey of the Inadequacies in Traffic Sign Recognition Systems for Autonomous Vehicles

Angelica F. Magnussen[a], Nathan Le[b], Linghuan Hu[c], and W. Eric Wong[c,*]

[a]*College of Engineering, University of Texas at Arlington, Arlington, TX, 76019, USA*
[b]*College of Science and Engineering, Texas A&M University, Corpus Christi, TX, 78412, USA*
[c]*Computer Science Department, University of Texas at Dallas, Richardson, TX, 75080, USA*

Abstract

Traffic sign recognition systems are crucial for autonomous vehicles. They assist autonomous driving systems by collecting road-related information, such as speed limits, stop signs, etc., that are necessary for safe driving. However, as evidenced by recent autonomous vehicle crashes and recognition system failure-related studies, there are serious concerns about the inadequacies of the traffic sign recognition systems and their used techniques. In response to the industrial needs and to help practitioners improve the reliability and safety of the traffic sign recognition systems, this paper discusses the general architectural outline of traffic sign recognition systems and the challenges that must be overcome, in order for traffic sign recognition systems to be safe and reliable. An in-depth discussion of various solutions is given to provide practitioners valuable insight into the improvement of traffic sign recognition systems.

Keywords: autonomous vehicles; traffic sign recognition system; software safety

(Submitted on July 15, 2020; Revised on August 17, 2020; Accepted on September 15, 2020)

© 2020 Totem Publisher, Inc. All rights reserved.

1. Introduction

For the past few decades, autonomous vehicles (AVs) have been a popular topic explored within the research community due to the benefits that they offer to our society. According to a 2015 survey by the National Highway Traffic Safety Administration, human error accounts for 94% of vehicular crashes involving at least one vehicle [1]. With the use of AVs, accident rates can be reduced profoundly, alleviating traffic-related deaths and injuries. However, before AVs can be widely deployed and adopted, they must be soundly designed and extensively tested to ensure the safety of passengers and the environment. Although there have been AVs developed by the commercial sector, those AVs may still make faulty decisions and, subsequently, partake in the wrong actions. Despite the media-cultivated hype of AVs, the safety of current AVs has been seriously concerned as evidenced by several recent accidents caused by AVs. In March 2018, an AV accident occurred when an automated Uber test vehicle fatally hit a woman. Because the automated Uber test vehicle lacked the capability to classify an object as a pedestrian unless that object, or pedestrian, was near a crosswalk, it failed to respond appropriately [2]. This incident could have been avoided if the Uber AV system could successfully classify the woman as a pedestrian, which would order the test car to stop earlier. In May 2016, a Model S Tesla car in self-driving mode collided with the side of a white tractor-trailer that was making a left turn on a highway, causing the death of the driver in the accident [3]. The self-driving car's failure to detect the tractor-trailer was due to Tesla's autopilot sensors' inability to detect the white tractor-trailer against a brightly-lit sky. Because the body of the trailer was mistaken, likely because of the high reflectivity of the color white, the AV system failed to apply the brakes. This implies that AV recognition systems are not yet suited or prepared for diverse weather conditions.

In each of the cases listed above, it has become apparent that AVs may not be ready to handle scenarios that could occur in the real world. This paper prompts the following questions: are modern traffic sign recognition (TSR) systems for AVs adequate enough for the real world? If not, how may their inadequacies be addressed so that the TSR system can perform correctly throughout the vast and diverse scenarios that are prevalent in the real world? To answer these questions, this paper

* Corresponding author.
E-mail address: ewong@utdallas.edu

examines the existing approaches used for TSR in modern AVs, which could lead to system failures. We then discuss the challenges of TSR systems in AVs, followed by an in-depth discussion on potential solutions to various issues. The remainder of the paper is organized as follows: Section 2 presents the background of TSR systems. Section 3 shows the challenges that need to be addressed in order to make TSR systems more reliable and safer. In Section 4, we provide an in-depth discussion of the potential solutions, followed by the conclusions and future work in Section 5.

2. Background of Traffic Sign Recognition Systems

To answer the questions prompted by this paper in Section 1, one must know the basics of how a TSR system functions. Many TSR systems [4-11] apply a two-part methodology: 1) traffic sign detection and 2) traffic sign recognition. Most TSR systems follow a procedure similar to the one shown as follows:

The first step of the process is traffic sign detection, and it is often achieved with the use of a camera that is integrated into the AV. The camera uses the frame of a video as the input, and the TSR system detects any regions of interest (ROIs) in the frames using traffic sign detection techniques such as slide window [4]. Traffic signs or ROIs can be found using a variety of traffic sign detection techniques. As previously mentioned, one technique used for detection is slide window. Slide window is a two-dimensional (2D) technique that detects patterns in 2D images by using a window, smaller than the dimensions of the camera-taken or input image, to scan or slide across the entire surface of the input image. Areas that are identified to feature predefined patterns are labeled as ROIs or highly-probable traffic sign candidates. Another technique used for detection is light detection and ranging (LiDAR) [12-14]. LiDAR utilizes laser lights and laser sensors to determine the distance or depth objects are from the AV. With three-dimensional (3D) modeling, LiDAR can identify the locations of traffic signs that are placed in front of the AV based on the shape of its 3D-constructed models [14]. Transfer learning is another technique that has been used in traffic sign detection [8]. Used in machine learning models, the transfer learning technique learns by storing knowledge gained from previous tasks. In other words, tasks are performed by applying previously gathered experience. Multilayer perceptron (MLP) neural networks have also been used to detect patterns in 3D images [4]. However, MLP has been found to be inferior to the convolutional neural network (CNN) approach due to the redundant architecture of MLP [15]. Other deep learning (DL)-based techniques that have been used for traffic sign detection include single shot detector (SSD), faster region convolutional neural network (FRCNN), and you only look once (YOLO) [16]. In the study [16], a comparison of each of these techniques was conducted. Each technique was found to have an advantage over the other techniques within a specific area. In summary, the survey determined that there are no specific methods that are superior to all others in all applications. As compared to FRCNN and YOLO, SSD was found to have the highest frame per second (FPS) score and an ability to balance computation time and accuracy. FRCNN had the lowest FPS, but it was found to be the most accurate at the cost of a longer computation time. YOLO was found to be the quickest but less accurate when detecting objects or ROIs [16].

The next step in a TSR system entails the classification or recognition of traffic signs. In this step, all ROIs are fed, one-by-one, into a classifier. Based on the shape, color, words, digits, and/or other characteristics found in the image of the traffic sign, the traffic sign is categorized. Traffic sign classification processes can vary by the techniques (e.g., CNN) or tools used (e.g., TensorFlow). The authors of the publications [4-6, 8, 17] used TensorFlow as a tool to create their proposed CNN-based TSR system. Machine learning (ML) classifiers for traffic sign recognition in TSR systems have been also used [6-7, 18]. These ML classification algorithms include K-nearest neighbor (KNN), support vector machine (SVM), and random forest (RF) [7]. KNN uses a database, stored in memory, of previously classified objects and those objects' related data [19]. Such data is then plotted in a multidimensional space or grid with the unclassified object plotted as well. The unclassified object, or traffic sign, is then classified based on the vote of its nearest k data points. SVM classifiers use at least one decision threshold to determine a traffic sign's category. Using a decision plane to separate objects belonging to different classes, SVM correctly classifies objects based on training data examples or traffic sign datasets. The classification algorithm, RF, constructs multiple decision trees and classifies based on the popular votes among all decision trees.

In addition to the ML-based approaches, the convolutional neural network (CNN) is another popular traffic sign classification approach [8, 10-11, 14, 16, 17, 20-22]. CNN, renowned for its use in image classification, is a type of neural network model with the ability to recognize complex patterns and abstract representations in images. CNN consists of input, output, and hidden layers. Hidden layers consist of a series of convolution layers that involve mathematical models to pass on results to successive layers. CNN uses hidden layers to extract features from an image's raw pixel data. Despite an increase in time needed to train the model, the addition of more hidden layers to a CNN enables the model to identify more sophisticated features. A benefit to the use of a CNN-based traffic sign classifier is its resilience to images with low resolution or occlusions [5]. Although DL-based classification methodologies have a high accuracy rate, they are highly complex and require a long processing time [10].

3. Challenges for TSR Systems

In this section, we show the challenges that need to be overcome in order to ensure that TSR systems are safe and reliable for real-world settings.

3.1. Traffic Sign Anomalies

Traffic signs are essential for safe driving. Without them, the occurrence of catastrophic accidents would dramatically rise due to a lack of order, guidance, and hazard warnings that traffic signs communicate. However, the task of successfully developing an accurate TSR system is difficult as it must be trained to recognize traffic signs that are partially occluded by objects, in deplorable conditions due to poor maintenance, and barely visible due to extreme weather. Traffic sign anomalies have shown to be great factors and challenges for TSR systems [12]. These anomalies often cause TSR systems to experience difficulties when detecting and classifying traffic signs. Some anomalies are related to weather, environmental conditions, visibility, noticeability, position, and altercation. Lengyel et al. [12] categorized some anomalies by the visibility, clarity, recognizability, and position properties of a traffic sign that they affect. For clarification purposes, the definitions of visibility, clarity, recognizability, and position in relation to traffic sign anomalies will be explained in accordance with the definitions provided in [12]. Visibility refers to the portions of a traffic sign that are visually exposed and not occluded. For example, a road sign covered by a tree or a large bush would have low visibility. Clarity refers to the quality of the traffic sign image that is captured by the camera of the AV. An image is clean or has a high clarity when the traffic sign within the image is not being blocked by fog, rain, dust, or other similar factors. Likewise, an image is dirty or has a low clarity when the contents of the image are made hazy and ill-defined, most commonly due to the weather at which the picture was taken. Recognizability refers to the features of the traffic sign that are observed. The recognizability of road signs includes observing the size, form, deformity, altercation, colors, and, if present, stickers on a traffic sign in order to correctly distinguish it and categorize it by its traffic sign type. Finally, the position is used to refer to where, in the camera frame, a traffic sign is located. This becomes relevant when determining whether the detected sign is at the right height or is too far, in order to be reliably recognized from the AV. We classify the traffic sign anomalies into two categories: 1) anomalies caused by physical conditions and 2) anomalies caused by occlusion.

The first category refers to the anomalies caused by physical conditions that could affect the visibility, clarity, recognizability, and position of traffic signs. In the worst scenarios, TSR systems may misclassify the sign or even fail to detect it. For example, the alteration of traffic signs, whether through the use of stickers, paint, or another medium, could cause traffic signs to have low visibility and low recognizability. In [23], the impact alteration on the accuracy of TSR systems through a series of experiments was measured. In the experiments, graffiti or stickers were added onto traffic signs. A TSR system was then placed with the task of classifying these signs. When stickers forming the words "love" and "hate" were added to stop signs, the AV misclassified the stop signs 73% of the time. When randomly placed stickers were added to stop signs, the AV mistook the signs for a 45-mph speed limit sign in every test. In the same study, right turn signs with a faded arrow were incorrectly classified by the AV 66% of the time. These results become especially alarming when considering the findings of a 2016 study [24] that measured the extent of traffic sign vandalism in the state of Utah. The study concluded that the "initial analysis of the data showed that almost 7% of 97,314 measured signs were damaged, of which at least 22% of the damages were intentionally caused by humans." From these results, it can be estimated that about 1.54% of road signs in the state of Utah have been vandalized. This includes the application of graffiti, stickers, markers, or another medium that diminishes the recognizability of traffic signs.

In [25], a survey of the condition of traffic signs in Utah was conducted. A strong association between traffic sign damage and its retroreflectivity was found. In some cases, the damage of traffic signs was so profound that it violated the minimum retroreflectivity standards outlined by the manual on uniform traffic control devices (MUTCD), a document issued by the U.S. Federal Highway Administration that specifies traffic signs standards. The study concluded that the effects of sign vandalism on retroreflectivity greatly depends on the extent of the damage. It was also found that a sign's reflectivity is reduced when that traffic sign is vandalized, cracked, faded, or peeled. To elaborate, faded and cracked signs had the highest rate of retroreflective failure, and signs surveyed with their legend peeling off were also likely to fail to convey their message to the road users. Furthermore, it was also discovered that the actual luminance of bent traffic signs under nighttime conditions may be lower than the MUTCD requirements.

The traffic sign's position, distance, height, and rotation could also impact the manner in which a TSR system detects and classifies it. For example, a circular sign may be rotated to appear as an ellipse rather than a circle [9]. Any numbers or letters visible on the sign would also appear deformed. Poorly positioned traffic signs may also fail to provide sufficient luminance for the sign to be visible to cameras. A traffic sign stationed too close to the ground may have nearby objects obstruct it; TSR systems may, consequently, fail to detect it [12]. Considering these factors, it can be concluded that vandalized or bent traffic signs may fail to be detected by TSR systems, and the risk grows when the AV is driving in nighttime conditions. In addition, traffic signs that are scratched, dented, peeling, bent, or cut could also diminish TSR systems' accuracy as edges, shapes, and objects could be mistakenly derived from damage sections even if the damage is subtle. These traffic sign pseudo-features would then potentially go on to lead classifiers, such as CNN-based ones, to a misclassification.

The second category refers to the anomalies that are caused by the occlusion of traffic signs. The condition of the environment could impact the quality of a traffic sign image that is captured by the camera of an AV in real-time. The most profound elements that impact traffic sign visibility, clarity, and recognizability are objects and weather [5, 9, 12, 13, 26-27]. These conditions need to be considered when TSR systems are designed. One of the most critical issues in AV development is the poor performance of TSR systems under adverse weather conditions (e.g., rain, snow, fog, and hail). Zang et al. [13] discussed examples that explained why weather conditions play a key factor in poor performance for AVs. For example, rain and fog degrade the functions of cameras and LiDAR, which can consequently cause sensors to signal inaccurate information. In effect, the AV can partake in inaccurate decisions and result in a crash. Snow and hail may also obstruct important features of traffic signs. The impact of snowy conditions on camera is also discussed. If the camera is unshielded, ice can easily damage the camera. Hail can damage the lens. Frost can cover the camera's lens and block the view. Rainy conditions can cause electrical and optical malfunctions if the camera is not designed to be waterproof. Raindrops can change the camera focus, causing parts of the picture to be out of focus. Foggy and dusty conditions can decrease the clarity of traffic signs and increase the difficulty of pattern edge recognition. Another challenge that TSR systems face is the partial occlusion of traffic signs in captured images. In the specific case of speed limit signs, digits must be accurately identified so that AVs can adjust their speed. However, in severe cases, some digits are occluded. In such occlusion cases, there is not enough information to interpret what the speed is. In [5], the challenge of occlusion in images of speed limit signs was elaborated on, and a proposed DL-based TSR system that handles instances of occlusion traffic sign images was presented.

3.2. Image Quality-Related Issues

The quality of the images captured by cameras is another challenge for TSR systems. In [9-10, 28], motion blur in captured images and video was mentioned as a major challenge in traffic sign recognition. Motion blur causes sharp edges in the captured frame to weaken. This makes it difficult for TSR systems to detect and classify traffic signs. Because edges are blurred and shapes are morphed, TSR systems could experience a decrease in accuracy if adequate image pre-processing methods are not used. Furthermore, image noise can alter the clarity of the image that a camera captures [18, 27]. Noise is a variation of brightness or color information in an image, and it is produced by the image sensor and circuitry. Noise can appear as black and white pixels randomly distributed throughout an image in a manner in which the image appears blurry. In [18], the noise was referred to as salt and pepper noise. Image noise can alter the image and make the detection and classification difficult for the system.

3.3. System Efficiency and Accuracy

Accuracy is always a major concern for TSR systems. False positives or false negatives can occur if a TSR system wrongly classifies an object within a camera's frame as an ROI, or if it fails to label a traffic sign present in the environment as an ROI. In the advent of a false positive or false negative, an AV may take risky actions and cause a collision. In [4], an example in which a false positive may occur was presented. A camera falsely detects a speed limit sign that is on the back of a van as a sticker. While many works have proposed TSR systems that have a high accuracy rating, they must not neglect to consider the speed and efficiency of their TSR system. For systems dependent on increased memory storage or enhanced performance (e.g., systems incorporating the KNN algorithm), the system will experience a decrease in computation speed as a substantial amount of data is accumulated and stored in the system.

3.4. Lack of Standard Benchmark for Evaluation

In [4, 6, 17, 20], TSR systems were proposed, trained with an acquired dataset of traffic sign images, and then evaluated. However, there is a lack of standard benchmark for evaluating TSR systems, and many of these studies tested their TSR system with a different dataset of traffic sign images. Despite each system claiming to possess a specific accuracy, with many exceeding 95%, these evaluated accuracy scores are not a fair metric to compare the TSR systems from each other. Varying on the diversity of the dataset used for testing, some testing datasets may not test all scenarios (e.g., rain, snow, night, day, fog, etc.) or may not test those scenarios to the extent that they should be (e.g., light fog, intermediate fog, and heavy fog). A system's testing dataset could also have images that are homogeneous to each other and homogeneous to the images used in the training dataset. In such a scenario, limited traffic sign conditions are tested, and the homogeneity properties of the images used for testing and training will biasedly skew the accuracy rate upwards. An illustration of how a TSR system's evaluated accuracy can vary upon the testing dataset used was presented in [6]. In the study, the proposed TSR system was evaluated on two testing datasets. When the system was tested using the German traffic sign database (GTSDB), a higher accuracy of 98.2% was obtained as opposed to the 95% accuracy score that was obtained when the dataset of Italian traffic signs (DITS) was used.

Although foreign traffic sign recognition benchmarks, such as the German traffic sign database (GTSDB) and Chinese traffic sign database (CTSDB) [22], have been established by the government of certain countries that are active in the research and development of AV technology, there is currently not a standard benchmark established for the United States of America. This causes a problem because, without a standard test dataset, TSR systems that are constructed for AVs traveling on U.S. roads cannot be compared to each other. For reasons mentioned before, the accuracy score and computation time scores obtained through testing without the presence of a standard U.S. benchmark dataset will render the evaluation results as an ineffective means of comparison [9].

4. Discussions of Existing Solutions

There are many existing methodologies that are implemented in various TSR systems intended for AV use. This section lists specific techniques, algorithms, methods, or tools that have been proposed for use in TSR systems. In this section, a few of these existing approaches are addressed, explained, and analyzed. In addition, some challenges of these methods will be presented, and potential solutions may be suggested.

4.1. Light Detection and Ranging

In the proposed TSR systems present in [12, 13, 14], light detection and ranging (LiDAR) was used to aid in traffic sign detection. LiDAR uses light lasers and laser sensors to determine what objects, based on a 3D constructed model of the environment, are in the vehicle's field of view and the distance that those objects are from the vehicle. LiDAR accomplishes this by pulsating light lasers into the environment and measuring the time it takes for that light to reflect off objects and return to the LiDAR device [13]. From several time measurements, the distance and form of surrounding objects can be determined. Specifically, distance is equal to half the round-trip time multiplied by the speed of light. The use of LiDAR can provide TSR systems outstanding angular resolution and highly accurate range measurements. Using 3D constructed models of objects found in front of an AV, LiDAR can be used to help detect the locations of traffic signs that the vehicle is approaching and the shapes of those traffic signs. Identifying the shape of a traffic sign using LiDAR could be an alternative to identify traffic sign shapes using image processing and a machine learning-based classifier.

Although LiDAR can be effective in gathering information in typical weather regardless of the level of sunlight available, it is not as accurate and reliable in adverse conditions [14]. The well-defined and clear attributes of 3D models may diminish as heavy rain or snow interfere with outgoing or returning pulsated lights [13]. In terms of modeling a traffic sign, conditions such as fog or rain could lead to problems related to blocking traffic sign edges partially or entirely, causing LiDAR devices fail to correctly and consistently model the traffic sign's shape.

4.2. Camera Tracking

When a traffic sign is occluded, whether by a weather element or an object, the edges of the sign become more difficult to detect. Hence, the shape of the sign or images conveyed on it may be incorrectly classified. Camera tracking can be used to alleviate this problem [7, 9, 13, 14]. After an ROI is identified, multiple images or frames of a video are taken of a detected traffic sign. For each video frame taken, the same traffic sign, or tracking target, is identified through processing. Considering the time of capture of the frames and the gradual change in location of the same traffic sign positioned within the frames, the position of the same traffic sign is tracked throughout time. Furthermore, a predictive tracking algorithm can be used to reconfirm the validity of future detected locations of the same traffic sign. By overlapping or comparing images of a traffic sign, previously occluded edges can be revealed, and edges detected from previous frames can be reaffirmed. Camera tracking approaches may include the techniques predictive tracking algorithm, key points tracking, and kernel-based tracking [9]. In [7], a predictive tracking algorithm was implemented by comparing the location of a traffic sign within the frames captured by cameras. It also implemented measures to ensure the validity of its tracking, two of which include resizing traffic sign regions to compute a similarity value (the sum of absolute difference in the pixel intensity of regions) and rejecting adjacent frames whose center distances between signs are less than a given threshold.

4.3. Resilient and Self-Cleaning Cameras

A TSR system that utilizes a resilient, self-cleaning, and transparent shield for its camera's lens could be a solution for certain weather and temperature conditions (e.g., condensation building on the camera lens, pebbles pelting the lens). In the case of rain, water droplets on the camera lens would make the focus of the camera frames blurry and unreliable. As mentioned before, cameras could also be severely damaged from the weather (e.g., ice formation). The camera would be better protected if a durable window encasement, with self-cleaning capabilities, shielded it from the elements. This approach requires the use of a robust self-cleaning mechanism (e.g., windshield wipers) to counter severe weather and other elements that could accumulate on the transparent encasing and obstruct the camera lens' sight. Effectively, this self-cleaning mechanism would wipe away substances (e.g., dirt, rain, snow) from the transparent encasement, hence preventing particle buildup. If a shielded camera is used instead of a bare, external-vehicle camera, then the layer of protection provided for it would keep camera feeds crisper under adverse weather conditions and deter safety hazards. In such cases, the shielded camera and camera lens would be protected from the risks associated with flying pebbles, mud, dust, hail, frost, and water droplets interfering with system-input footage or degrading the condition of the camera's components.

4.4. Convolutional Neural Networks

CNN has been a popular classification technique among the TSR system research community as illustrated by its use in the TSR systems proposed in the publications [5-6, 8, 17, 21]. Although CNN is effective and efficient, it is resourcefully demanding. The more layers that a CNN model has, the better filtering and accuracy that the CNN model can provide. However, those additional layers require more training and computation time. Although CNN has outperformed previous classification methods in the field of traffic sign recognition, CNN-based TSR systems are still in need of development and innovation. As mentioned before, CNN-based TSR systems have a positive feature of being able to work with low-resolution images and also with occlusions [5, 10]. Researchers have incorporated CNN with other modules or techniques to form novel TSR systems. Although the general trend is increased performance and resistance to diverse traffic sign anomalies, these deep learning-based systems may still face challenges, one of which is explained below.

Despite CNN's history of use in TSR systems, TSR systems still face a major problem that could occur when using a CNN-based classifier for the categorization of detected traffic signs. Due to the "one-pixel attack", misclassification errors can occur despite a subtle change of a camera-captured image. Su et al. [29] concluded that deep neural networks can easily lead to the wrong classification if a small change (e.g., change one pixel) is applied to an image. In a set of images, one pixel was strategically chosen and modified. Their results showed that 67.97% of the natural images in the Kaggle CIFAR-10 test dataset and 16.04% of the images in the ImageNet (ILSVRC 2012) dataset can be unsettled to at least one target class by modifying just one pixel with 74.03% and 22.91% confidence on average. In other words, this modification invoked unexpected classification errors in the classifier, causing the feature extraction and identification process utilized by CNN for accurate classification to fail.

It is concerning because this vulnerability can have an impact on various deep learning-based TSR systems. If a subtle change was to be made to an image of a traffic sign, either physically or digitally, an inability to accurately classify the sign could be invoked due to the location of that alteration. These minuscule classification-defining alterations could occur in the form of a sticker, marker dot, or even scratch. Digitally, pixels in the image could be altered due to corrupt camera signals in moments beyond the control of the system [30]. If a camera signal does become corrupt from the camera capturing stage to the image rendering state, then an accidental misclassification may occur. As shown in [29], even one pixel can break a deep learning classifier. In either of these scenarios, a subtle change could invoke the system to misclassify a traffic sign and consequently cause the AV to act dangerously.

4.5. Pre-Processing Techniques

To increase the performance of TSR systems, many studies suggest that effective and efficient image pre-processing techniques should be adopted [9-10, 18, 30-31]. The purpose of pre-processing is to edit the input image so that image noise can be reduced, edges can be enhanced, and contrast can be increased for ease of edge detection. The most common pre-processing techniques would include the use of filters. To smooth the image and reduce noise, the Wiener filter or an averaging filter can be used [18, 30]. On the occurrence that a signal becomes corrupt and renders a pixel as the incorrect hue (e.g., a blue pixel in the center of a stop sign) [30], the Wiener filter can effectively make this correction. Through the use of statistical equations, the Wiener filter uses the probability of occurrence to correct misfit pixels, thus smoothing the image. For general purpose smoothing, an averaging filter can be used to divide the image into miniature grids and convert all pixels within that grid to the computed mean value of that grid. Two popular averaging filters include the arithmetic mean filter and the geometric mean filter [30]. In addition, a Gaussian filter [32] can also be used to reduce image noise at the cost of blurring edges. This burred-edges effect can be corrected with the subsequent use of an edge sharpening technique. Image sharpening filters that can be used in the pre-processing stage include unsharp mask filtering (USM) type 1, where the image is sharpened at a single fixed sharpness value, and unsharp mask filtering (USM) type 2, where the sharpness of the image can be controlled.

Occasionally, traffic signs that are within an ROI are partially occluded either by objects (e.g., vegetation) or weather elements (e.g., resting or falling snow). When a sign is partially occluded, classification algorithms that use knowledge of universally-defined shapes and symmetry could be used by TSR systems to estimate the form of traffic signs. Part-based object detection techniques include local binary patterns (LBP), histogram of oriented gradient (HOG), and randomized Hough transform (RHT) [9]. LBP converts images to a solely black and white scheme by converting the image to grayscale and establishing pixel grids, typically 16 by 16 pixels, throughout an image. In each grid, the center pixel is compared with its surrounding pixels. If a surrounding pixel's value is greater than the center, it is converted to black; otherwise, it is converted to white. In the resulting image, edges will be represented by black pixels, and the spaces between edges are denoted in white. The purpose of LBP is to expose edges present in an image so shape patterns can be more easily observed. HOG also divides an image into several non-overlapping grids in an attempt to make edges within an image more apparent. In each grid, lines are identified based on the pattern of dark or bright pixels present in the grid. Lines found in the grid are converted to white, and pixels that are not a part of the lines are converted to black. RHT can also be used to identify shape features such as curves and lines within an image by analyzing an image and implementing a voting system for which the most probable edges are voted the highest.

In the real world, rotated or tilted signs can cause the shape of a traffic sign to appear morphed. For example, a circular sign that is tilted in a manner in which one side is protruded will have a stretched circle or ellipse appearance [9]. To aid in the shape classification of detected traffic signs, a technique that handles shape deformations can be used. These techniques include generalized Hough transform (GHT), accumulator array creation with filtering, scale-invariant feature transform (SIFT), speeded up robust features (SURF), oriented FAST and rotated BRIEF (ORB), and local energy-based shape histogram (LESH) [9]. GHT uses matching of universal shape templates to find shapes, whereas accumulator array creation with filtering can be used to handle deformed and disconnected shapes. SIFT and SURF can be used to handle size variation cases, and ORB is a popular rotation and scale-invariant feature. LESH can also be used as a scale-invariant shape-based feature for object detection.

Color spaces are another approach that can be used in the recognition of traffic signs. The RGB color space classifies pixels for human vision [10]. In the RGB color space, every pixel is assigned a red, blue, and green value between 0 to 255. Images taken by cameras are typically in the RGB color space. As road signs are primarily red or blue, researchers use the property of color to detect and recognize traffic signs [32]. However, when the color of a traffic sign is similar to the color of objects in the background, a problem emerges for systems that implement image segmentation based on color. To preserve the integrity of object borders observed in an image, images are converted to different color spaces and operated on [10]. Conversion to the hue saturation value (HSV) color space is often used because it offers a faster detection speed, is less affected by illumination, and has a preferable segmentation advantage. Therefore, images in the RGB color space that are converted to the HSV color space enable further image processing to identify edges, objects, and shapes with more precision and ease.

Thresholding is used for detecting the borders of traffic signs within an image and, in terms of recognition, detecting the shapes, letters, or digits present on a traffic sign. A popular thresholding technique for image segmentation is Otsu's method [9, 32]. Otsu's method consists of thresholding for image segmentation. Each threshold has its own bitmask. Each mask will contain new images, some containing only the background and others containing only the objects [32]. The optimal threshold is the masking threshold for which the largest variance is found. This would be the threshold function and the optimal thresholds found by Otsu for the value, saturation, and hue channels. The values of the channels enable traffic road signs to be extracted, or differentiated from the background, within an image. In order to obtain a perfect segmentation that separates all panels from the background, a merging of the results obtained in hue, value, and saturation is needed. Binary image operators can be used to make this work, as they are used to perform automatic thresholding to a binary image from the shape of the image histogram.

4.6. Post-Processing Techniques

When designing a TSR system, adding post-processing techniques to the TSR system can also improve the accuracy or efficiency of detection or recognition. Post-processing techniques are essentially installed to enhance the TSR system over time. In [9], various post-processing techniques used to improve the efficiency of traffic sign detection were given. An example of a post-processing technique that has been used to improve the detection of ROI or traffic signs is false negative detector (FND). FND calculates the efficiency of a detector based on the number of false negatives generated. The backpropagation algorithm (BPA) is another technique used for post-processing [11]. BPA is used in machine learning-based classifiers to improve the future classification accuracy of traffic signs.

4.7. Pre-Determinant Elimination at Object Detection

A TSR system could decrease its sign recognition speed if spatial neighborhood filtering (SNF) [6] is applied. In SNF, detected ROIs are assigned a probability of being a traffic sign based on their location. A probability threshold that is computed or programmer-defined is then applied. ROIs that are labeled with a value lower than the probability threshold are then ignored or eliminated from further pre-processing. This pre-determinant elimination could also decrease the number of false positives occurring. Hence, the accuracy of a TSR system may increase. According to [6], SNF can increase a TSR system's accuracy by 1-2% depending on the input dataset. To illustrate the practical use of SNF, the following example is provided: a red frisbee resting vertically against a pole that is on the side of a road will be pre-determined to not be a potential traffic sign candidate due to its unusual location within the camera frame. To accomplish this, a tracking algorithm will be needed to help with optimal feature selection to reduce false rate [7]. Tracking algorithms provide TSR systems with the ability to predict the future position of detected traffic signs that are moving within the frame of a camera. Because features have a high dimension of data and some of that data may not be useful for distinguishing an object between frames, tracking algorithms can be used to keep the recognition rate high and reduce the number of feature dimensions between frames, which will also reduce the processing time. In [7], a feature selection approach based on genetic algorithm (GA) is designed. The GA-based approach effectively reduces the dimension of features without significantly reducing the recognition rate. In the experimental results, GA reduces feature dimension up to 90% while maintaining the recognition performance. The tracking mechanism reduces the false rate of a proposed TSR system significantly, from 21.22% to 10.2%.

4.8. Improve Traffic Sign Recognition via Testing Using a Standard Benchmark and Simulation

As mentioned before, a standard traffic sign dataset needs to be established as a benchmark for developed TSR systems intended for use within a specific country (e.g., the United States). The advent of a benchmark would enable all TSR systems intended for use in that country to have their accuracy rate fairly measured and established as a reliable metric for TSR system comparison. However, benchmarks must be carefully designed and comprehensive of traffic sign anomalies, both common and uncommon, that are present in the real world. Benchmarks that are properly designed will ensure that TSR systems are extensively tested and will, consequently, surface any vulnerabilities. In the study [27], a TSR classifier developed by Majumdar et al. was proposed and then evaluated. After the CNN-based classifier is trained to recognize traffic signs in diverse weather conditions, the system is evaluated. To the surprise of the developers, the classifier, when tested, only performed well when presented with images of fog intensity levels that were less than 20% or higher than 40%. It can be inferred that the classifier developed by Majumdar et al. could have increased its accuracy rate if it had been trained with images representative of all fog intensities. Therefore, benchmarks should convey traffic signs under all types of conditions, both anomalies and weather, that affect the image quality of traffic signs in addition to conveying those conditions under different intensities. In this manner, benchmarks will serve as a means to expose system vulnerabilities, especially the ones less apparent, in TSR systems.

When faced with the task of acquiring a wide and diverse image dataset of traffic signs for TSR system testing, it is unreasonable to wait for separate instances, each at a varied intensity, of hazardous weather to naturally occur. Therefore, TSR systems need to be tested more promptly and can be tested more promptly within a graphically-realistic computer simulation [27]. In a graphically-realistic computer-generated simulation, clarity-depreciating conditions can be parameterized and controlled. This is to say that varying intensities of fog, snow, and rain can be accurately and realistically simulated with the use of a powerful physics engine and a photo-realistic renderer, such as those used in modern video game engines (e.g., Unity and Unreal Engine). Non-weather related driving conditions could also be tested, such as a muddy camera, water droplets morphing a camera's input images, motion blur, flying insects, up-kicked dirt, or varying degrees of lightness and darkness, to a shocking degree of realism.

5. Conclusion and Future Work

Modern AVs are capable of handling driving and navigation tasks, but their TSR systems may not be accurate and reliable enough for real-world settings. Recent AV crash accidents show that AV technology still needs to be further improved to ensure safety. In this paper, we summarize the challenges of TSR systems and provide an in-depth discussion of the existing approaches to provide practitioners valuable insight into the improvement of safety and reliability of TSR systems. Overall, we observed that the performance of the TSR systems can be improved by adopting various approaches, such as advanced sensor technology, pre-processing, post-processing, testing-based, and simulation-based techniques. In the future, we will work on each specific challenge and propose a cost-effective approach for improving the performance of TSR systems.

Acknowledgements

This work is supported by the US National Science Foundation (REU Award No. 1757828 and I/UCRC Award 1822137).

References

1. S. Singh, "Critical Reasons for Crashes Investigated in the National Motor Vehicle Crash Causation Survey," *Traffic Safety Facts Crash Stats*, *Report No. DOT HS 812 115*, National Highway Traffic Safety Administration, Washington, DC, February 2015, (https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/812115)
2. P. McCausland, "Self-Driving Uber Car that Hit and Killed Woman Did Not Recognize that Pedestrians Jaywalk," NBCNews.com, *NBCUniversal News Group*, 11 Nov. 2019, (www.nbcnews.com/tech/tech-news/self-driving-uber-car-hit-killed-woman-did-not-recognize-n1079281)
3. B. Vlasic and Neal E. Boudette, "Self-Driving Tesla Was Involved in Fatal Crash, U.S. Says," *The New York Times*, 30 June 2016, (www.nytimes.com/2016/07/01/business/self-driving-tesla-fatal-crash-investigation.html)
4. D. R. Bruno and F. Santos Osorio, "A Comparison of Traffic Signs Detection Methods in 2D and 3D Images for the Benefit of the Navigation of Autonomous Vehicles," in *Proceedings of 2018 Latin American Robotic Symposium*, *2018 Brazilian Symposium on Robotics (SBR) and 2018 Workshop on Robotics in Education* (*WRE*), pp. 26-32, João Pessoa, 2018, doi: 10.1109/LARS/SBR/WRE.2018.00015
5. D. R. Bruno and F. S. Osorio, "Image Classification System based on Deep Learning Applied to the Recognition of Traffic Signs for Intelligent Robotic Vehicle Navigation Purposes," in *Proceedings of 2017 Latin American Robotics Symposium* (*LARS*) *and 2017 Brazilian Symposium on Robotics* (*SBR*), pp. 1-6, Curitiba, 2017, doi: 10.1109/SBR-LARS-R.2017.8215287
6. A. Youssef, D. Albani, D. Nardi, and D. D. Bloisi, "Fast Traffic Sign Recognition using Color Segmentation and Deep Convolutional Networks," in *Proceedings of International Conference on Advanced Concepts for Intelligent Vision Systems*, pp. 205-216, 2016, doi:10.1007/978-3-319-48680-2_19
7. L. K. Wahyono and K. Jo, "Traffic Sign Recognition and Tracking for a Vision-based Autonomous Vehicle using Optimally Selected Features," in *Proceedings of 2015 54th Annual Conference of the Society of Instrument and Control Engineers of Japan* (*SICE*), pp. 1419-1422, Hangzhou, 2015, doi: 10.1109/SICE.2015.7285415
8. R. Kulkarni, S. Dhavalikar, and S. Bangar, "Traffic Light Detection and Recognition for Self Driving Cars using Deep Learning," in *Proceedings of 2018 Fourth International Conference on Computing Communication Control and Automation* (*ICCUBEA*), pp. 1-4, Pune, India, 2018, doi: 10.1109/ICCUBEA.2018.8697819
9. B. Sanyal, R. K. Mohapatra, and R. Dash, "Traffic Sign Recognition: A Survey," in *Proceedings of 2020 International Conference on Artificial Intelligence and Signal Processing* (*AISP*), pp. 1-6, Amaravati, India, 2020, doi: 10.1109/AISP48273.2020.9072976
10. J. W. Cao, C. X. Song, S. L. Peng, F. Xiao, and S. X. Song, "Improved Traffic Sign Detection and Recognition Algorithm for Intelligent Vehicles," *Sensors*, Vol. 19, No. 18, pp. 4021, 2019, doi:10.3390/s19184021
11. J. Rezgui, A. Hbaieb, L. Chaari, and J. Maryland, "Traffic Sign Recognition using Neural Networks Useful for Autonomous Vehicles," in *Proceedings of 2019 International Conference on Smart Applications*, *Communications and Networking* (*SmartNets*), pp. 1-6, Sharm El Sheik, Egypt, 2019, doi: 10.1109/SmartNets48225.2019.9069770
12. H. Lengyel and Z. Szalay, "Traffic Sign Anomalies and Their Effects to the Highly Automated and Autonomous Vehicles," in *Proceedings of 35th International Colloquium on Advanced Manufacturing and Repair Technologies in Vehicle Industry*, pp. 193-204, Zielona Góra, 2018
13. S. Zang, M. Ding, D. Smith, P. Tyler, T. Rakotoarivelo, and M. A. Kaafar, "The Impact of Adverse Weather Conditions on Autonomous Vehicles: How Rain, Snow, Fog, and Hail Affect the Performance of a Self-Driving Car," *IEEE Vehicular Technology Magazine*, Vol. 14, No. 2, pp. 103-111, June 2019, doi: 10.1109/MVT.2019.2892497
14. X. Zhao, P. Sun, Z. Xu, H. Min, and H. Yu, "Fusion of 3D LIDAR and Camera Data for Object Detection in Autonomous Vehicle Applications," *IEEE Sensors Journal*, Vol. 20, No. 9, pp. 4901-4913, 1 May, 2020, doi: 10.1109/JSEN.2020.2966034
15. Uniqtech, "Multilayer Perceptron (MLP) vs Convolutional Neural Network in Deep Learning," *Medium*, Data Science Bootcamp, 13 June 2019, (medium.com/data-science-bootcamp/multilayer-perceptron-mlp-vs-convolutional-neural-network-in-deep-learning-c890f487a8f1)
16. S. P. Kaarmukilan, S. Poddar, and A. Thomas K., "FPGA based Deep Learning Models for Object Detection and Recognition Comparison of Object Detection Comparison of Object Detection Models using FPGA," in *Proceedings of 2020 Fourth International Conference on Computing Methodologies and Communication* (*ICCMC*), pp. 471-474, Erode, India, 2020, doi: 10.1109/ICCMC48092.2020.ICCMC-00088
17. V. Swaminathan, S. Arora, R. Bansal, and R. Rajalakshmi, "Autonomous Driving System with Road Sign Recognition using Convolutional Neural Networks," in *Proceedings of 2019 International Conference on Computational Intelligence in Data Science* (*ICCIDS*), pp. 1-4, Chennai, India, 2019, doi: 10.1109/ICCIDS.2019.8862152
18. A. V. Deshpande and M. M. Subashini, "A Survey of Different Image Processing Methods for the Design and Development of an Efficient Traffic Sign Board Recognition System," in *Proceedings of 2017 2nd IEEE International Conference on Recent Trends in Electronics*, *Information and Communication Technology* (*RTEICT*), pp. 340-343, Bangalore, 2017, doi: 10.1109/RTEICT.2017.8256612
19. S. Ray, "A Quick Review of Machine Learning Algorithms," in *Proceedings of 2019 International Conference on Machine Learning*, *Big Data*, *Cloud and Parallel Computing* (*COMITCon*), pp. 35-39, Faridabad, India, 2019, doi: 10.1109/COMITCon.2019.8862451

20. R. Hasegawa, Y. Iwamoto, and Y. Chen, "Robust Detection and Recognition of Japanese Traffic Sign in the Complex Scenes based on Deep Learning," in *Proceedings of 2019 IEEE 8th Global Conference on Consumer Electronics* (*GCCE*), pp. 575-578, Osaka, Japan, 2019, doi: 10.1109/GCCE46687.2019.9015419
21. D. Tabernik and D. Skočaj, "Deep Learning for Large-Scale Traffic-Sign Detection and Recognition," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 21, No. 4, pp. 1427-1440, April 2020, doi: 10.1109/TITS.2019.2913588
22. Z. Zhu, J. Lu, R. R. Martin, and S. Hu, "An Optimization Approach for Localization Refinement of Candidate Traffic Signs," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 18, No. 11, pp. 3006-3016, November 2017, doi: 10.1109/TITS.2017.2665647
23. Eykholt, Kevin, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. "Robust physical-world attacks on deep learning visual classification." In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1625-1634. 2018.
24. M. Khalilikhah, K. Heaslip, and K. Hancock, "Traffic Sign Vandalism and Demographics of Local Population: A Case Study in Utah," *Journal of Traffic and Transportation Engineering* (*English Edition*), Vol. 3, No. 3, pp. 192-202, 2016, doi:10.1016/j.jtte.2015.11.001
25. M. Khalilikhah and K. Heaslip, "The Effects of Damage on Sign Visibility: An Assist in Traffic Sign Replacement," *Journal of Traffic and Transportation Engineering* (*English Edition*), Vol. 3, No. 6, pp. 571-581, 2016, doi:10.1016/j.jtte.2016.03.009
26. H. Lengyel and Z. Szalay, "Test Scenario for Road Sign Recognition Systems with Special Attention on Traffic Sign Anomalies," in *Proceedings of 2019 IEEE 19th International Symposium on Computational Intelligence and Informatics and 7th IEEE International Conference on Recent Achievements in Mechatronics*, *Automation*, *Computer Sciences and Robotics* (*CINTI-MACRo*), pp. 000193-000198, Szeged, Hungary, 2019, doi: 10.1109/CINTI-MACRo49179.2019.9105238
27. R. Majumdar, A. Mathur, M. Pirron, L. Stegner, and D. Zufferey, "Paracosm: A Language and Tool for Testing Autonomous Driving Systems," arXiv: 1902.01084, February 2019
28. L. Jöckel, M. Kläs, and S. Martínez-Fernández, "Safe Traffic Sign Recognition through Data Augmentation for Autonomous Vehicles Software," in *Proceedings of 2019 IEEE 19th International Conference on Software Quality*, *Reliability and Security Companion* (*QRS-C*), pp. 540-541, Sofia, Bulgaria, 2019, doi: 10.1109/QRS-C.2019.00114
29. J. Su, D. V. Vargas, and K. Sakurai, "One Pixel Attack for Fooling Deep Neural Networks," *IEEE Transactions on Evolutionary Computation*, Vol. 23, No. 5, pp. 828-841, October 2019, doi: 10.1109/TEVC.2019.2890858
30. A. V. Deshpande, R. Kannan, and M. M. Subashini, "Study of Various Image De-Noising Methods Used for the Purpose of Traffic Sign Board Recognition in an Intelligent Advanced Driver Assistance System," in *Proceedings of 2018 International Conference on Intelligent and Advanced System* (*ICIAS*), pp. 1-6, Kuala Lumpur, 2018, doi: 10.1109/ICIAS.2018.8540630
31. A. V. Deshpande and M. M. Subashini, "An Investigative Approach Towards Various Image Segmentation Algorithms Used for Traffic Sign Recognition," in *Proceedings of 2017 Fourth International Conference on Image Information Processing* (*ICIIP*), pp. 1-6, Shimla, 2017, doi: 10.1109/ICIIP.2017.8313693
32. A. Barodi, A. Bajit, M. Benbrahim, and A. Tamtaoui, "An Enhanced Approach in Detecting Object Applied to Automotive Traffic Roads Signs," in *Proceedings of 2020 IEEE 6th International Conference on Optimization and Applications* (*ICOA*), pp. 1-6, Beni Mellal, Morocco, 2020, doi: 10.1109/ICOA49421.2020.9094457

Angelica F. Magnussen is a B.S. student in computer science at the University of Texas at Arlington. Her research interests include software engineering, software testing, artificial intelligence, and distributed systems. Contact her at angelica.magnussen@mavs.uta.edu.

Nathan Le is a B.S. student in Computer Science at Texas A&M University Corpus Christi. His research interests include artificial intelligence and autonomous vehicles. Contact him at nle10@islander.tamucc.edu.

Linghuan Hu is a Ph.D. student in software engineering at the University of Texas at Dallas. His research interests include combinatorial testing and test generation. Hu received a M.S. in software engineering from the University of Texas at Dallas. Contact him at linghuan.hu@utdallas.edu.

W. Eric Wong is a professor and the director of the Advanced Research Center for Software Testing and Quality Assurance in Computer Science at the University of Texas at Dallas. His research focuses on helping practitioners improve the quality of software while reducing the cost of production. Wong received a Ph.D. in computer science from Purdue University, West Lafayette, Indiana. He is the Editor-in-chief of IEEE Transactions on Reliability and the corresponding author for this article. Contact him at ewong@utdallas.edu.