Obstacle Detection over Rails Using Hough Transform


Abstract—Category (4). Autonomous systems can assist humans in the important task of safe driving. Such systems can warn people about possible risks, take actions to avoid accidents or guide the vehicle without human supervision. Whether in cars or trains or ships the artificial vision algorithms offer an alternative for the design and implementation of autonomous driving systems. In railway scenarios cameras in front of the train can assist drivers with the identification of obstacles or strange objects on the rails. Multiple factors add huge complexity to this task. The changing conditions create a scene where background is hard to detect, lighting varies and process speed must be fast. This article describes a first approximation to the problem where using the Hough transform, the rails and area of interest are detected. On this area a systematic search is done for finding and delimiting possible obstacles. Our system accomplished a real time performance when employed in the analysis of recorded videos from the driver perspective. Using digital added obstacles our algorithm detects mostly all of them and warns if the objects over the rail can create a danger to the safety travel of the train.

Key Words—Obstacle detection, autonomous train driving, digital image processing, Hough transform.

I. INTRODUCTION

A ccomplish vehicle autonomous driving is a huge technical challenge which still lacks general solution. For several decades researches had explored different artificial vision methods for implementing automatic driving algorithms but many issues like changing lighting, changing background, process speed, etc., turns this problem into a complex task with no easy solution. Autonomous train driving is a particular case where the previous problems are present too but where the specific characteristics suggest the use of a different methodology for the design and implementation of vision algorithms.

One such feature is the rails presence along the complete route. Detecting the rails allow designing specialized algorithms that can find more easily obstacles using the relative monotony in the railway. This is why our first task was to create vision algorithms that can extract the rails in videos recorded from the train driver perspective.

Then tracking over each rail was done detecting possible obstacles in front of the train. Our system can perform a video analysis in real time offering a first approximation towards a supporting tool for autonomous train driving in the city.

The article structure is as follows: section II describes some related work in the field of artificial vision and automatic driver assistance strategies. Section III shows the proposed system and implementation details. Experiments and results are described in section IV. Conclusions are in Section V and the future work is detailed in section VI.

II. RELATED WORK

Mainly two strategies guide the implementation of autonomous train driving systems: Active systems depend on the emitting devices deployment and associated sensors in train's head, for example infrared lasers where the reflection beam is analyzed in order to detect obstacles or judge safety route. Works that follow this approach includes [1][2]. The main disadvantages of these kinds of systems are their difficult to identify obstacles boundaries, short range of action and their low accuracy on curve zones. In contrast passive systems normally use video cameras in front of the train and rely on image processing algorithms for obstacle detection. A complementary approach consist in mixing active and passive system and accomplish the fusion of the multiple sources of information (see for example [3][4][5]). In this work we will focus in passive system where the only source of data is the video obtained from a single camera.

Following this approach Ukai [6] used the Hough transform for tracking the rails and compute the vanishing point using a moving camera. The camera is dynamically adjusted in such way that this point is always near the center of the image. For obstacle detecting the optical flow is employed with several algorithms for mitigating the adverse effect of the moving background. In order to detect fixed objects any rail interruption on the images is labeled as a possible obstacle. A low luminosity lens in the camera and anti-blurring techniques complement the system.

In the work described in [7] the authors use clothoids segments for the rail detection. In each segment several algorithms are used for detecting obstacles, included: discontinuities in the lines (rails), gray scale variability...
between adjacent segments, optical flow between video frames and statistic of textures in the segments.

In [8] dynamic programming is used in order to extract the rails and the area between them but the obstacle detection problem is not considered. Calculating the image gradient using the Sobel operator the near driver segments of the rails are extracted. After that the Hough transform is employed for estimating the rails vanishing point and delimiting the area enclosed for the rails. The remaining segments in the upper part of the image are treated in a recursive way.

In [9] a projective transform is used for creating a new image where the rails become parallel or with a constant separation between them like in a birds view perspective. The Hough transform is employed on this image for tracking and detecting the railway. If the system fails to track the rails at any point the possibility of it being cover with an obstacle arise. In this system an intensive previous configuration is required for the projective transform. In the configuration phase manual operator must indicate through a complete train route where the rails are, which is the area between them and estimates near and far distances.

In [10] the rail detection is considered but not the obstacle identification. Using the inverse mapping transform [11] a bird eye view is compute. With this transformation the distance between rails is kept constant from the bottom to the upper part of image taken from the train driver perspective. Using Gaussian filters blurring is mitigated and every frame is segmented in ten consecutive parts. On every segment the rails detection task is initiated using a polynomial model. This model computes the distance between pair of lines and selects the most likely candidates for representing the rails. Information in the ten segments is mixed and the central imaginary line between rails is obtained. Using this information the area of interest is identified for future obstacle tracking.

All the previous works had problems identifying the rails in changing environmental conditions, scenes with low luminosity and in pronounced curve zones. The obstacle detection in front of the train is such hard problem that many author prefer not confront it, limiting their approaches to the rail and near area identification.

Our work deals with the railway identification through the Hough transform and initiate the obstacle finding using artificial vision algorithms. Autonomous train driving is still an open research area with not given general solution. Our work is a first exploration in the topic where we initiate the design and implementation of signal processing methods in order to extract the rails, the area of interest near them and the detection of obstacles in front of the train. This is the first local initiative in this important and relevant area of research.

In the next section we will describe the proposed system with its processing phases and the resources used for adjusting and testing the system.

III. METHODOLOGY

A. Motivation

Fig. 1 shows an image where the rails are capture from the train driver perspective.

![Fig. 1: Typical railway appearance from the driver perspective](image)

We can say that the rails look like lines projecting from the bottom of the image to the horizon and they seem to converge in a far point known as the vanishing point. In the near to driver zone the rails appearance is quite monotonous in all the route while in the horizon do exist greater variability mainly in curve sections. The presence of the rails can be used as a first input for computer vision algorithms. To delimit an area of interest is the first task to start the search for obstacles and decide if the way is free and can be safely traveled. In this case the rails, area between them and the close outer area represent the zone of interest. One strategy for detecting the rails consists in extracting the lines in the image and selecting the most likely candidates to rails using criteria such as length, angle and position in the image. In this paper we use the Hough transform for detecting in each video frame the rails and start the delimitation of the area of interest for searching obstacles.

B. Hough Transform

The Hough transform [12] is an artificial vision algorithm proposed for detecting lines in images. The parameters m and b (line-intercept) were initially used according to (1).

\[ y = mx + b \]  

Every point \((m, b)\) in the space of parameters represents a complete line in the original image space. Nevertheless the absence of boundaries in the parameters \(m\) and \(b\) poses serious difficulties at the implementation of the algorithm. Cuda and Hart [13] improved the algorithm using the alternative line representation given by the expression in (2).

\[ x \cdot \cos \theta + y \cdot \sin \theta = \rho \]
In this equation the parameters ($\rho$ and $\theta$) represent respectively the length of the normal line from the origin to the original line and the angle of the normal to the axis of abscissas (See Fig. 2).

![Fig. 2: Line Parameterization with Bounded Parameters rho and theta](image)

With this representation each point in the $xy$ plane maps to a sinusoid in the parameter space $\rho \theta$. If two points are collinear in the $xy$ plane in the parameter space their sinusoids will intersect in a given point ($\rho$, $\theta$) that defines the line according to (2).

The algorithm general idea is to find the points ($\rho$, $\theta$) in the space parameters where many sinusoids intercept. These points will correspond to lines in the original image. In order to make the algorithm computationally efficient the parameter space is divided using a grid. The precision in the lines found is directly related with the grid’s division size. Each division in the grid is represented with an accumulator. Every $xy$ point in the original image generates a series of points (a sinusoid) according to (2). The accumulators are updated if these sinusoid points fall into the corresponding grid divisions. After covering all the points in the original image those accumulators that reach highest values will represent lines with its parameters given by the division position in the grid (See Fig. 3).

C. System Diagram

Fig. 4 shows a general diagram of the proposed system. Although the input to the systems is a video recorded from the train driver perspective currently we process the video in a frame to frame basis. The first task is a preprocessing stage where a filter for improving contours is applied. Then a gray scale conversion and dimensioning reduction phase take place. As previously said the Hough transform was the mechanism used to find the lines representing the rails. We found that using a smaller image could improve the process speed without affecting precision considerably. So the rail detection task resizes the image and applies a rectangular mask for eliminating irrelevant background objects. After a Canny edge process and a closer the Hough algorithm find candidate lines. Applying several criteria like length and position the lines that more likely represent the rails are selected.

![Fig. 3: (a) Binary image with a quasi-line example. (b) Accumulators representation after Hough transform over image in (a), marked intersection point gives line parameters](image)

In relation with the obstacle detection stage the first task is to use the Canny algorithm. A closer follows and then the contours in the image are identified and stressed. Small and disconnected objects are then eliminated. After filling the contours we start a systematic search using the rails as guide. An independent tracking in each rail is conducted by using a small dynamic area that moves from the bottom of the rail to the upper part. En each analysis step the area and centroid are compute. When the metrics are far from the expected values the selected area grows and modified its centroid with the intention of enclosing the strange object (See Fig. 5).

IV. RESOURCES

Due to the difficulty of using real videos where obstacles appear in front of the train, initially we choose to use freely available recordings on the Internet. We searched high resolution videos recorded from the driver’s perspective where the route followed had no problems. After that the obstacles were digitally added through the superimposition of Chroma key videos. For this task we use the Kdenlive [14] software. All the system implementation was accomplished using the C++ libraries of OpenCv [15].

V. Results

Our system was tested using several modified videos from the Internet but the video used for measuring the system performance shows the route from city of Landskrona to Ramlösa (Sweden) [16]. We were granted permission for modification of the original video from the author. This video was recorded from the driver perspective with favorable weather conditions, at day, without turnouts and with splendid visibility. Using 7 minutes of recording we added a total of 20 digital obstacles of different nature, shape and obstruction trajectory.
The system successfully identified 19 of the 20 obstacles, but still has a high occurrence of false positives (12). Most false detections were due to the presence of objects near the road but actually do not represent real risk for the train travel, these includes signs, pass levels, bridges over the rail, some platforms, etc.

The real time performance of the system reached rates around 60 fps using image resolutions of 640x480 pixels. The precision measure in the video, using $fp = 12$ y $tp = 19$, was as shown in (3).

$$\text{Precisión} = \frac{tp}{tp + fp} = 0.61 \quad (3)$$

With only a false negative (fn) the sensibility reached was very high as shown in (4).

$$\text{Sensibilidad} = \frac{tp}{tp + fn} = 0.95 \quad (4)$$

Figs. 6-9 show some examples of positive obstacle identification. One of the false positives found is displayed on Fig. 10, where a signal is erroneously classified as an obstacle due its proximity to the rail.

The system successfully identified 19 of the 20 obstacles, but still has a high occurrence of false positives (12). Most false detections were due to the presence of objects near the road but actually do not represent real risk for the train travel, these includes signs, pass levels, bridges over the rail, some platforms, etc.

The real time performance of the system reached rates around 60 fps using image resolutions of 640x480 pixels. The precision measure in the video, using $fp = 12$ y $tp = 19$, was as shown in (3).

$$\text{Precisión} = \frac{tp}{tp + fp} = 0.61 \quad (3)$$

With only a false negative (fn) the sensibility reached was very high as shown in (4).

$$\text{Sensibilidad} = \frac{tp}{tp + fn} = 0.95 \quad (4)$$

Figs. 6-9 show some examples of positive obstacle identification. One of the false positives found is displayed on Fig. 10, where a signal is erroneously classified as an obstacle due its proximity to the rail.
VII. Future Work

Addressing this work showed us that there exist the potential to build support systems for automatic train driving. In order to continue in the pursuit of this objective multiple strategies can be followed to improve the system. Here we list some of them.

Correct identification of the rails is an important step for the system. In this work we used the Hough transform for line detection. However, when the road is curved the method is quite inaccurate in the remote areas of the rails. To compensate for this it is necessary to evaluate alternative strategies to capture more accurately the inclinations of the rails and find a more appropriate vanishing point according to the actual scene. A candidate method is the use of clothoids or parabolic segments that can much better fit to the rail.

Although our system uses as input a video obtained from the driver's perspective all processing is done frame by frame. Using temporal information can be especially valuable for analysis and tracking of obstacles, where their trajectory, persistence and consistency in the scene can provide a clearer perspective of the potential risk to the train travel. At this time we have tackled the task of using information from consecutive frames to improve tracking of objects and their possible collision paths.

In the current implementation obstacle detection relies heavily on the proximity of the object to the rails. To create more robust systems to detect obstacles on a collision course is a highly complex task to be explored. The changing nature of the background makes that tracking algorithms fail in detecting moving objects. The exploration of algorithms that mitigates the apparent background motion is pending attention.

Many objects in the railway can affect the monotony of the rails images and the nearby area without posing a real danger to the train course. This includes turnouts, signs, pass levels, tunnels, bridges, etc. Recognition of these objects is essential for a reliable behavior of a train driver assistance system. Learning and identification of benign object provides interesting challenges for computer vision algorithms.

Another important aspect that should be explored is the robustness of the algorithms to changes in light and weather conditions. Driving at night or amidst storms creates a vast number of difficulties for automated driving systems based on vision. It is possible that the creation of strategies based on adaptive parameters can provide a solution but its use is currently complex and difficult to implement.

VIII. Acknowledgments

The author Luis Fonseca would like to thank to Universidad de Antioquia and COLCIENCIAS for their support in the Jovenes Investigadores 2011 Project.
REFERENCES


Luis Fonseca received the B.S. degree in electronic engineering from the Universidad de Antioquia, Colombia and wrote his thesis at Technical University Munich, Germany in 2011. Currently, he is doing a research internship at Universidad de Antioquia, Colombia. His research interests include image processing, machine learning, video analysis and computer vision in general.

Jonny Alexander Uribe received his B.S. degree in electronic engineering in 2009 and his Master degree in Engineering in 2012 both from the Universidad de Antioquia, Colombia. Currently he is a Ph.D. candidate at Universidad de Antioquia. His research interest includes personalization in mobile applications, machine learning and digital signal processing.

Jesus Francisco Vargas was born in Bogotá, Colombia in 1978. Received the Electronic Engineering degree in 2001 and his master degree in Industrial Automation in 2003, both from the Universidad Nacional de Colombia – Manizales. His PhD degree was awarded at the Technology Center for Innovation in Communications (CETIC, University of the Palmas de Gran Canaria, Spain). Since 2004 he is Professor at the Department of Electronic Engineering in Universidad de Antioquia. Its main research interests include digital processing signals and pattern recognition.