



Obstacle Extraction based on Region Extraction on a Monocular Video

Pramodh K.P.
ISE department, PESIT
Bangalore-560085

Prasad Bhagwat
ISE department, PESIT
Bangalore-560085

Shashank P.S.
ISE department, PESIT
Bangalore-560085

Mamatha H.R., PhD
ISE department, PESIT
Bangalore-560085

ABSTRACT

Ever since the advent of computers, they has proven themselves extremely useful for automating routine parts of human life. With improvements in computer vision techniques, Unmanned Ground Vehicles (UGV) are a hot topic of research. In order to build such a system, one of the main objectives is to locate and identify things present in its surroundings. Obstacle Extraction is one such field which deals with detection of obstacles in front of it. In this project, an obstacle extraction system is built, by first extracting the region of interest, the road. This region of interest is then merged with the foreground obtained from the background modeling. Two different background modeling techniques are implemented, one based on GMM, and the other based on Bayesian color estimation. Further, a lane detection system has been implemented, which helps in detecting the current lane that the UGV is moving in. This is implemented based on two techniques, one based on Hough transform and the other based on Contours. The second method was also extended to determine the direction of the curve ahead of the vehicle.

Keywords

GMM, Otsu Thresholding, Canny Edge Detection, Contours, Mahalanobis Distance, Recursive Bayesian, Hough Transform.

1. INTRODUCTION

In recent years, there has been a lot of development in Unmanned Ground Vehicle (UGV) in cars and other means of road transportation. UGVs are systems that can automatically move around in the environment. UGVs generally sense their surroundings using techniques such as RADAR, LIDAR, GPS and computer vision. Techniques that use sensors are quite efficient in detecting the surroundings, in the sense that it requires very less post processing. Some sensors are tailor made to detect depth of the obstacles from its position. However, appearance based detection techniques usually use several visual cues to segment the pixels in an image. Therefore, while appearance-based techniques don't have information about depth, sensor-based techniques lack in color information of the pixels.

UGV use mounted cameras in order to sense the surroundings based on computer vision. There are two kinds of cameras usually used in such applications, 1) Stereo-vision cameras and 2) Monocular cameras. Stereo-vision systems use two cameras, which increase both the cost and processing time of the system. While the former, has the added advantage of multiple image (from different angles) of the same object, the latter, however, is cheaper and faster. Hence, in this project explores implementations based only on monocular camera. Our objective in this project is to find the limits of a

monocular camera, in 1) Obstacle extraction and 2) Lane detection in urban roads during a real time environment.

There is a slight difference between the words “detection” and “extraction”. Detection refers to obtaining a boundary around the position of the object. However, the clear shape of the obstacle is not extracted. Extraction refers to obtaining an outline of the object's shape and not just a boundary of it. In this project, a technique for “obstacle extraction” is implemented, however, the terms detection and obstacle are interchangeably used.

In the context of ground vehicles, objects protruding from the ground plane are considered obstacles. Obstacle extraction is the process of extracting the obstacles on the ground plane and separating it from the noise (For e.g., trees, pavements) in the surroundings. Obstacles can either be moving or static. It's important to devise a technique adaptable to various kinds of situations, than implement an algorithm for each separately. There are various techniques of extracting obstacles. Some are based on optical cues, while others are based on mathematical models. The more general properties the method is based on, the more adaptable it will be. In order to classify obstacles as true positives and false positives, the region of interest (road, in this case) is extracted.

Lane detection is the process of identifying the lane in which the UGV is travelling. Lane detection using computer vision is usually done using two approaches. 1) Model based approach and 2) Pattern based approach. In this project, two different pattern-based approaches are implemented.

2. HISTORY

Numerous works of research are done in the field of obstacle extraction for ground-based autonomous vehicles. Usually, better techniques are available for obstacle extraction, which uses stereo-vision cameras or multiple sensors (LIDAR, RADAR) etc. However, few techniques are available for detection using monocular camera, although it's perfectly capable of performing the task in many scenarios.

One of the techniques implement a optical-flow based tracking obstacles [1]. It is known that optical flow computation is quite expensive and is unsuitable for real-time environments. Hence, author proposes an approach, in which the optical flow is previously modeled and is compared with the sequence of images. This method also needs huge calculation time and fails to identify obstacles which are static. A second method is based on a monocular moving camera, which identifies obstacles based on shape [2] or symmetry [3]. These techniques are too specific such that, it can be applied only to pedestrians or cars, and are not easily customizable for complicated scenarios. In the paper [2], the authors build a pedestrian detection system, which is based on

the shape of the pedestrians, and hence complete pedestrian should be present in the picture. Further, they use a stereo-vision technique to detect the depth of the images.

Another technique, called GOLD [4], uses a stereo-vision camera to find the depth image of the obstacles and the obstacles are detected by two or more images of the obstacles at different angles. However, this technique introduces a lot of overhead in processing, due to multiple images. It also becomes costlier to implement. It is also tightly coupled with the camera calibration as mentioned by the authors. In [5], an obstacle detection and depth estimation system is proposed for a Android-based mobile platform. The author proposes a IPM method for depth estimation and histogram processing for obstacle detection.

The approach in [6], road region extraction, is a part of obstacle extraction technique. The authors, in this paper propose a technique to detect road quite efficiently using illuminant-invariant technique. This technique is robust in detecting just the road, and not the vehicles on it. It is also capable of detecting roads on different lighting conditions, or in presence of noise, such as shadows from the trees. This approach needs camera calibration to be done at the beginning to obtain a particular required parameter.

Table 2: Existing techniques

Theme	Technique
optical flow	Theoretical optical flow model, Separation from ground plane
Shape based Technique	Looks for vertical edges, symmetry.
Symmetry property of Snake models	Based on symmetry

3. DESCRIPTION

The obstacles can be classified as false positives and true positives. In order to recognize the true obstacles and ignore the noise and surrounding environment, a region of interest is determined, inside which obstacles are found. In ground vehicles, the region of interest would be the road region. This region usually is the bottom third part of the image. Elements such as shadows, change in color of road are some of the interesting problems in road region extraction

3.1 System Design

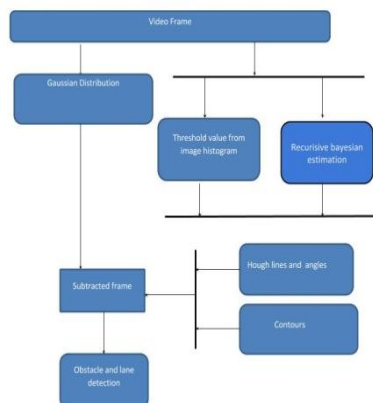


Figure 1.0 System Design

3.2 Background Modeling

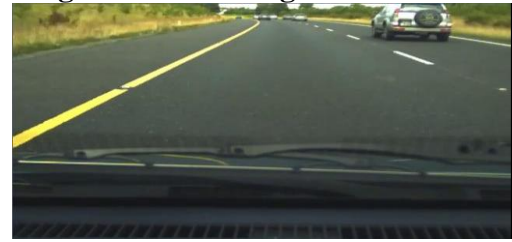


Figure 2.0 Video frame

Background subtraction is the process of separating out foreground objects from the background in a series of video frames. It is a major preprocessing step in many vision based applications. The fundamental logic for background subtraction is called the “Frame difference method”, in which the current frame is subtracted from the reference frame, to get the foreground of the current frame.

There are two kinds of backgrounds, either static or dynamic. For example, visitors counter where a static camera, takes the number of visitors entering or leaving the room, or a traffic camera extracting information about the vehicles etc. In such cases the moving objects are separated as foreground and the static parts of the video are separated as background. However, in dynamic scenarios such as, a vehicle mounted camera moving on the road, it is difficult to segment the video into foreground and background.

In dynamic scenarios, the “reference frame” mentioned earlier, also needs to keep changing along with the current frame. Hence, a simple subtraction of two frames would not give sufficient results. In such cases, advanced recursive techniques such as Gaussian Mixture model can be used to get better results.

In this technique, it is assumed that every pixel’s intensity values can be modeled using a Gaussian mixture model. A simple heuristic determines which intensities are most probably of the background. Then the pixels which do not match to these are called the foreground pixels. The foreground pixels are grouped using 2D connected component analysis.

A time series of pixels is called “pixel process”. At any point in time, for a particular pixel (x0, y0) its history is known. So, a pixel process for a static background / static object would remain fairly constant. However, the pixel process should be short enough to know, any new static object that is added, would soon be a background. Hence, the history of the pixel process is updated and modeled based on such factors, using K Gaussian distributions. So, the probability of observing the current pixel value would be

$$X_1, \dots, X_t = \{V(x_0, y_0, i) : 1 \leq i \leq t\} \quad \text{Eq. 1}$$

Denotes a set of pixel process for t = 1 to t = t.

$$P(X_t) = \sum_{i=1}^K \omega_{i,t} N(X_t | \mu_{i,t}, \Sigma(i, t)) \quad \text{Eq. 2}$$

Probability of a Gaussian at time t, where N is

$$N(X_t | \mu_{i,t}, \Sigma(i, t)) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma_{i,t}|^{1/2}} \exp\left(-\frac{1}{2}(X_t - \mu_{i,t})^T \Sigma_{i,t}^{-1} (X_t - \mu_{i,t})\right) \quad \text{Eq. 3}$$

To classify a mixture as background process, heuristically, it should be the processes with most supporting evidence and least variance. This mixture model takes mean and variance of



the distribution as its parameters and performs the above operations on it.



Figure 2.1 after background modeling

3.3 Otsu Thresholding

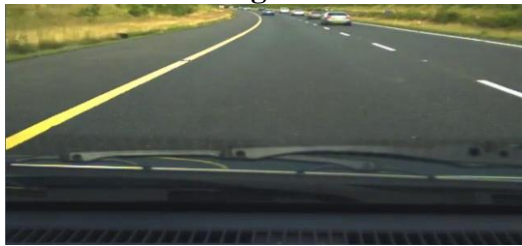


Figure 2.2 Video frame

Image intensity-based machine vision approaches are the most common methods to find the probable areas of road from moving images. The luminance of road pixels is usually close to the others, so it is possible to find an appropriate threshold value and segment the road in images. The parameters and threshold value used in each method are selected experimentally, which are sensitive to image acquisition conditions and lighting of the area when the images are taken.

In order to separate the road region from the background automatically, Otsu thresholding technique is applied to the images. In other words, Otsu's method is used to perform histogram-based image thresholding automatically, that assumes that the image to be verged contains two classes of pixels by using a bimodal histogram.

This method calculates the optimum threshold value that efficiently separates these two classes, so that their inter-class variance is maximal. The advantage of the proposed method comparing to the similar approaches is one do not have to remove sky region from the image.

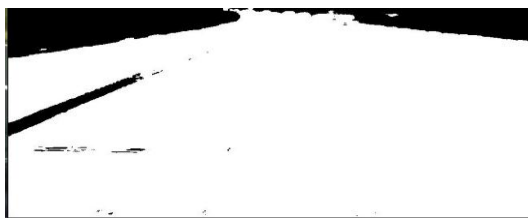


Figure 2.3 After Otsu algorithm

3.4 Recursive Bayesian Color Estimation

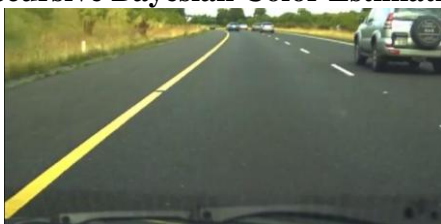


Figure 2.4 Video frame

Recursive Bayesian Color Estimator (RBCE) is a technique based on color properties of the road. In this technique, it is assumed that the only the bottom-third part of the image contains valuable information about the road, and the remaining parts are penalized/ignored based on its distance from the mean point. A mean point is calculated for the road pixel in the bottom part of the image. The distance of color pixel from this point is calculated with a function called Mahalanobis distance. The Mahalanobis distance is a function of mean and covariance of the point. It denotes the dissimilarity between the two points. Based on this the road and non-road regions are extracted.



Figure 2.5 Frame after recursive Bayesian color estimation

3.5 Lane detection

Detection of lanes on suburban/urban roads is another aspect which helps in working of an AGV. Typically, Lane detection is usually done using a boundary detection technique applied on the images captured by a camera mounted on the vehicle. There are many properties of a good lane detection technique:

The intersection point of the lines detected by Hough transformation can be further used for detection of horizon.

One, It should be capable of processing the painted and the unpainted roads, two, It should be capable of processing the painted and the unpainted roads.

3.6 Canny Edge and Hough transform technique



Figure 2.6 Video frame

The Hough line transformation technique was good at detecting lanes on straight lines. However, it could not identify the lanes in a curved road. Even though, a small part of a curved lane could be identified, it could not identify the complete lane till the horizon on curved roads. Hence, Hough Line transform technique fails on curved roads.



So, a modified Hough Line transform is implemented. The input image is first pre-processed using Canny Edge detector, in order to extract just the edges and ignore the noises in the image.

Hough Line transform is a technique used for identification of straight lines in the image. The technique was applied with a threshold (line length) of around 50-100 pixels. The unwanted lines detected by the transform are removed by filtering the lines based on the angle of its inclination. The horizontal lines are unimportant when doing lane detection, and hence the lines with an angle of 180 are removed. This parameter can be modified, for different road angles.

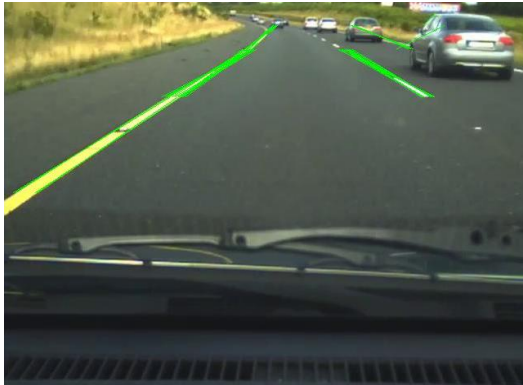


Figure 2.7 Video frames after Hough transformation

3.7 Contour-based Lane and curve-ahead detector



Figure 2.8 Video frame

The image is first given to a canny edge detector, with very less difference between minimum threshold and maximum threshold. This helps in removing all the lines that are short. And hence, only the lines corresponding lanes are extracted.

Contours on this image are found out and are filtered based on a minimum area. This eliminates all the smaller contours are only the lane lines are detected, including the curved lanes.

Further, a line fitting technique is used to detect straight lines tangential to the contours of the lanes. Based on the position of the maximum y-coordinate of the contour point with respect to the tangent, the lane is said to be either curving to the “right”, to the “left” or is “straight”. This is detected quite early in the frame, since contours can detect even the farther lines. Hence, this can be used as a curve-ahead predictor.

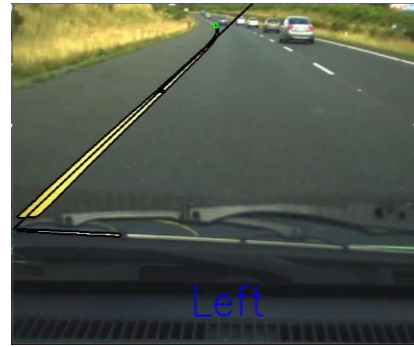


Figure 2.9 Video frames after contours processing

4. CONCLUSION

In this project, it has been shown that a monocular camera is moderately efficient in detecting obstacles. We were also able to detect lanes efficiently, as long as there were lane markers on the road. We could detect even curved lanes, without a mathematical model, and also detect the curve ahead in the road. Betterment of algorithms, and better cameras might provide even better outputs. However, the road detection algorithms work well only under constraints mentioned in the report above. On the other hand, Background modeling is not robust enough to handle some complicated situations such as, sudden illumination change and road color change.

5. ACKNOWLEDGEMENT

This paper consumed huge amount of work, research and dedication. We are grateful to Dr. Mamatha H R for providing necessary guidance concerning projects implementation. Without her superior knowledge and experience, the Project would like in quality of outcomes, and thus her support has been essential.

Nevertheless, we express our gratitude toward our families and friends for their kind co-operation and encouragement which help us in completion of this project.

6. REFERENCES

- [1] Christophe Braillon, Cedric Pradalier, Jim Crowley, Christian Laugier. Real-time Moving Obstacle Detection Using Optical Flow Models. Proc. of the IEEE Intelligent Vehicle Symp., Jun 2006, Tokyo (JP), France. pp.466-471
- [2] A. Broggi, M. Bertozzi, et.al , Shape-based Pedestrian Detection, Proceedings of the IEEE Intelligent Vehicles Symposium 2000, Dearbon (MI), USA, pp. 1-6
- [3] Shumin Liu, Yingping Huang and Renjie Zhang, On-Road Vehicle Recognition Using the Symmetry Property and Snake Models, pp. 1-9
- [4] Shaohua Quin et al., Road Region Estimation and Obstacle Extraction using Monocular camera, ICIC International 2013, Vol 9, No 9 September 2013, pp. 1-12
- [5] Razvan Itu, Radu Danescu, An Efficient Obstacle Awareness Application for Android Mobile Devices, ICCP IEEE 2014, pp. 1-7
- [6] José M. Álvarez and Antonio M.Lopez, IEEE Transactions on Intelligent Transportation Systems, Vol. 12, No. 1, March 2011, pp. 1-10



- [7] Arthur Miranda Neto, Alessandro Correa Victorino, Isabelle Fantoni, Janito Vaqueiro Ferreira. Real-Time Estimation of Drivable Image Area based on Monocular Vision. IEEE Intelligent Vehicles Symposium (IV), Jun 2013, Gold Coast, Australia. pp.63-68, 2013
- [8] Stauffer, Chris, Adaptive background mixture models for real-time tracking, IEEE CVPR 99, Vol 2 pp. 1-7
- [9] Duda, R. O. and P. E. Hart, "Use of the Hough Transformation to Detect Lines and Curves in Pictures," *Comm. ACM, Vol. 15*, pp. 11–15 (January, 1972)
- [10] Canny, J., *A Computational Approach to Edge Detection*, IEEE Trans. Pattern Analysis and Machine Intelligence, 8(6):679–698, 1986.
- [11] Video Dataset for lane detection and obstacle extraction <https://www.youtube.com/user/DiarmaidOCualain/playlists>