 ISSN : 0976-8491 (Online) | ISSN : 2229-4333 (Print)

Autonomous Vehicle With Monocular Vision for Object Detection Using Haar-cascade Classifiers

[1]Tuneer Guha, [2]Sarthak Kataria, [3]Shivam Goel, [4]Netra Patil
[1,2,3,4]Dept. of Computer Engineering, Bharati Vidyapeeth University, College of Engg., Pune, Maharashtra, India

Abstract
The biggest challenge in case of an autonomous driving system is proper integration of all the functions to be carried out for a safe driving experience. Detection of obstacles and prevention of collision plays an important role in road safety in case of such vehicles. The use of Haar-feature based cascade classifiers provides a light weight solution to object detection with extensive use of computer vision. With the help of monocular vision, image processing algorithms (Gaussian blur), identifying brightest spots (e.g. break lights of a car) and distance measuring algorithms, quick collision preventing actions can be taken. Multilayer Perceptron (MLP-ANN) can be used to classify objects and paths to create a simple decision making data structure for proper navigation. This paper illustrates on an intelligent self-driving vehicle that uses all the above methods tied together to drive on a predefined path. The monocular vision camera takes in image inputs that are processed and the above algorithms are applied on them. This acts as input to the ANN that recognizes the road direction. The road direction is classified into 4 classes- left, right, forward and reverse. The neural network then takes the correct decisions based on the road direction and obstacles that lie on it.

Keywords
Autonomous Vehicle, Artificial Neural Network, Multilayer Perceptron, Edge Detection, Gaussian Filter, Object Detection, Internet of Things.

I. Introduction

The idea of Autonomous vehicle might seem far-fetched and impossibly futuristic but the growing amount of research done in this field is certain to take our hands off the wheel and show the true power of open source automation. Autonomous vehicles have followed a long evolutionary path that stretches back to Leonardo da Vinci and beyond. Many major companies like NVidia, Tesla, Google, Ford etc. are extensively rushing to build autonomous vehicles for the rapidly changing consumer market.

Autonomous vehicles rely entirely on object detection and decision making for proper movement across paths without causing collisions and potentially fatal accidents. For this reason, the precision required to be achieved on these fields needs to be immaculate. Precise controlling and proper navigation need to be a priority for achieving a good, workable and reliable system. The basic functioning of autonomous vehicles involves an image processing system, a system to identify the point of interest and obstacles in the images captured, an algorithm for measuring distance, an Artificial neural network for decision making, and hardware to control movement. All these systems work together by firstly capturing image, compressing it, finding areas of interest, using object detecting algorithms to recognize objects in the recognized path and then taking the correct decision using decision trees to stop or keep moving. Monocular vision is a sketchy subject in case of measuring distances but a method proposed by Chu, Ji, Guo, li and Wang (2004) describes a geometric model of detecting distance is significantly accurate.

II. Approach

This model of autonomous driving involves a set of independent systems for movement ANN based navigation and object detection; the MLP-ANN based navigation utilizes computer vision to make decisions and the Object detection is done with the help of OpenCV which is an open source framework for computer vision.

A. Approach for MLP-ANN Based Navigation

Artificial neural networks a basically crude electronic models which are based on the structure of neurons of a brain. ANNs do a promising job in promising patters, processing images, and other dynamic tasks with a much higher success rate.

A multilayer perceptron neural network is a feedforward artificial neural network model that directs a set of input data onto a manually trained set of appropriate outputs. An MLP is basically a directed graph consisting of multiple layers of nodes. Every node other than the input node is a neuron and each node has a nonlinear activation function. Backpropagation is a supervised learning technique that is specifically used for training the network. MLP is basically a modification of standard linear erceptron and can distinguish data that is not linearly separable.

For the proposed system a three level MLP network is being used with an input layer (with 38,400 inputs) a hidden layer (with 32 inputs) and an output layer (with 4 outputs). The number of nodes in the hidden layer is an arbitrary number, the number of inputs is 320*120 i.e. the number of probable inputs. The number of outputs is 4, one for each direction of movement.

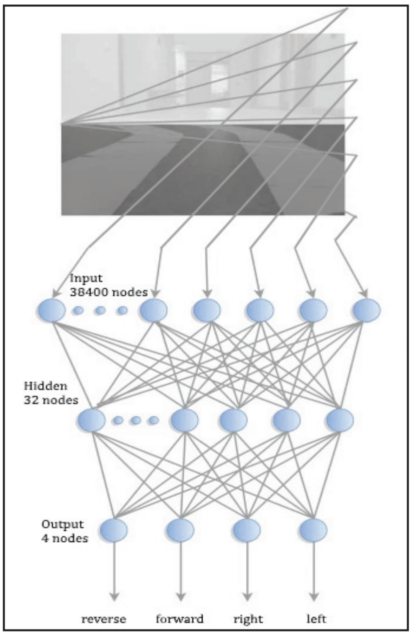

Fig. 1: Multilayer Perceptron Artificial Neural Network

Collection of training data is a step by step process. A set of progressive frames are captured and converted to an n/p array. This array is then paired with a label which is basically the human input and all these images are then saved into a lightweight database. The neural network is trained in Open CV using the back propagation method. Once the training is completed the weights for each input node (labels for the captured images previously) is stored into an XML file. To see the predictions work, the neural network is loaded with this XML file.

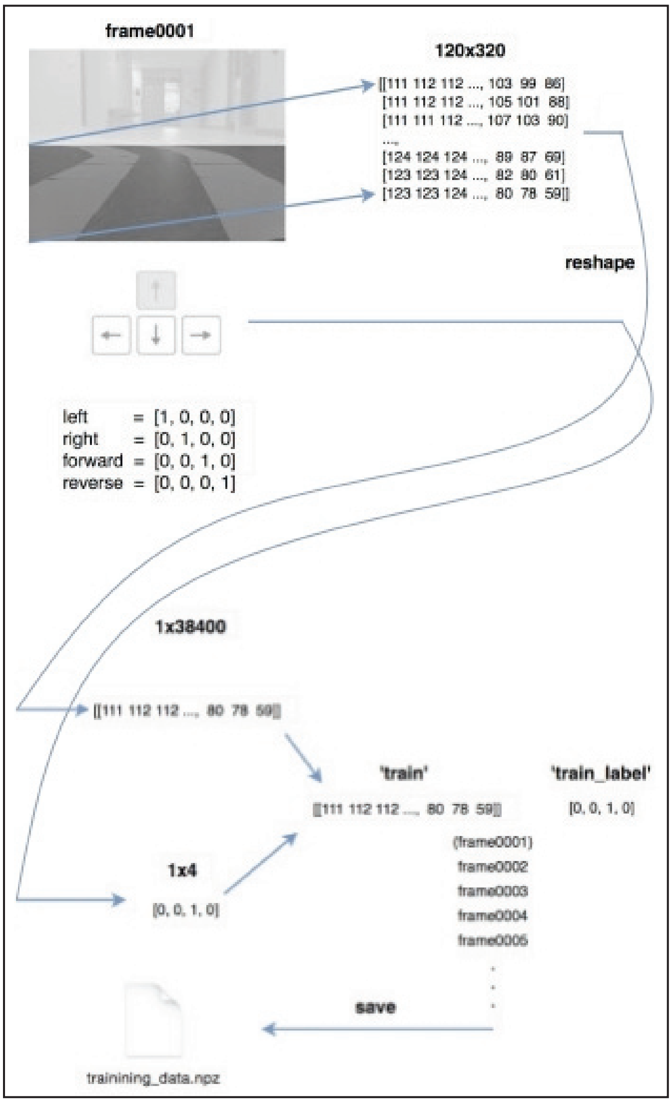

Fig. 2: Collection of Training Data

B. Approach for Object Detection Using Haar-cascade Object Classifier

Shape detection using Haar feature based cascade classifiers is significant and simple enough to be implemented into a small and economical system. Each object in this requires its own classifier and requires to be trained individually.

OpenCV proves to be a good solution for training and detection in this contraption. The objects are classified using a set of cropped images which consist only of the region of interest of each object, these are called positive samples (consisting only the target object). Negative samples on the other hand were collected using random images which did not contain the object.

This stage requires extensive image processing and is therefore done in 5 distinct steps; the first stage involves capturing a greyscale image. Haar cascade classification is applied onto it and region of interest is discovered. Gaussian blur is then applied to this region of interest to reduce noise and to make the image cleaner. After this the brightest point in the Region of interest is detected and identified, the identification of brightest point will help in detecting traffic signals and break lights. The above data collected is then validated using previously defined classifiers and the appropriate action is taken.

Fig. 3: Image Processing Flowchart

The method of identifying an object based on its characteristics has 3 major steps:

- The primary area of interest is found by identifying the boundaries of a lane that is seen in the image captured by the camera, the blocking object is detected by identifying a difference in the primary area of interest of the captured image and that of the image that was used to train the system.
- Then a more precise area of difference is found by identifying the edges of the object by using a pre-trained classifier (haar-cascade) and a Gaussian blur function.
- The distance of the object from the vehicle is found by continuously tracking it and calculating the distance using a monocular distance formula.

It is very important to find the target object as soon as possible when the vehicle is moving at a high speed.
The object being detected in the image might be small due to its distance and hence detection is done on the basis of identification of basic characteristics of the said object.

III. Distance Measurement

This project utilizes a geometric model for calculating distance from an obstacle using monocular vision.

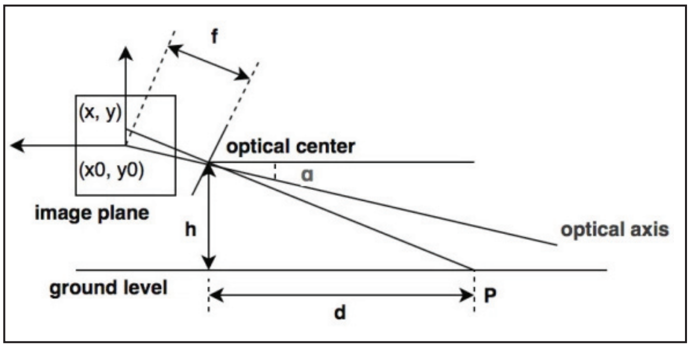

Fig. 4: Geometric Model for Measuring Distance

P is a point on the target object; d is the distance from optical center to the point P. Based on the geometric relationship described above, (1) shows how to calculate the distance d. In (1), f is the focal length of the camera; ∂ is the angle of the camera tilt; h is the perpendicular height of the optical center; (x_0, y_0) refers to the intersection point of image plane and optical axis; (x, y) refers to projection of point P on the image plane. Suppose O_1 (u_0, v_0) is the camera coordinate of the point of intersection of the optical axis and image plane, also suppose the physical dimension of a pixel corresponding to the x-axis and the y-axis on the image plane are dx and dy. Then:

$$d = h/\tan(\partial + \arctan((y - y_0)/f)) \quad (1)$$

$$u = \frac{x}{dx} + u_0 \quad v = \frac{y}{dy} + v_0 \quad (2)$$

v is the camera coordinates on y-axis and can be returned from the object detection process. All other parameters are camera's intrinsic parameters that can be retrieved from the camera matrix (In computer vision a camera matrix or (camera) projection matrix is a 3 X 4 matrix which describes the mapping of a pinhole camera from 3D points in the world to 2D points in an image).

V. Computer Vision and Opencv

This project extensively uses computer vision and that is achieved with the help of OpenCV. OpenCV is a library of programming functions for Computer vision.

This project uses OpenCV for two different approaches. Firstly, for object detection (with the help of Haar-cascade classifiers) and secondly, for distance measurement.

OpenCV acts a trainer as well as detector for object classification. Positive samples (containing specific images of the object) acquired using a camera module are cropped such that only the desired object in its various states is visible. Negative samples (without target object), on the other hand, are collected randomly. For example, consider a stop sign and a traffic light, positive sample of a traffic light contain equal number of red traffic lights and green traffic light and positive samples of a stop sign consist of the same in various intensities of light and different angles. The same negative sample datasets are used for both stop sign and traffic light training. Below are some positive and negative samples used in this project.

Fig. 5: Positive and Negative Samples

Table 1: Positive and Negative Samples

Type	Samples		
	Number of Positive samples	*Number of negative samples*	*Sample size*
Stop sign	20	400	25*25
Traffic lights	26	400	25*25

project, it is used to obtain a camera matrix from the camera module of the system. Camera matrix for a 5MP camera module is returned after calibration. Ideally, a_x and a_y have the same value. Difference of these two values will result in pixels that are not uniform in the image because they would not be square in shape. The matrix below indicates that the fixed focal length lens on the camera provides a reasonably good result in handling the distortion aspect of the image.

$$\begin{bmatrix} a_x = 331.7 & 0 & u_0 = 161.9 \\ 0 & a_y = 332.3 & v_0 = 119.8 \\ 0 & 0 & 1 \end{bmatrix}$$

The matrix returns values in pixels and h is measured in centimeters. By applying formula (3) (derived form (1) and (2) in the previous section), the physical distance d is calculated in centimeters.

Let $x_0 = y_0 = 0$, from (1) and (2):

$$d = h/\tan(\alpha + \arctan((v - v_0)/a_y)), (a_y = f/dy) \quad (3)$$

IV. Results

Prediction on the testing samples returns an accuracy of 93% compared to the predicted accuracy of 96% that the training samples returns. In an actual driving situation, predictions are generated about 10 times a second (streaming rate roughly 10 frames/s). Haar- cascade features by nature are rotation sensitive. Hence, for a considerable amount of accuracy multiple angles of the object must be captured.

For distance measurement aspect, an ultrasonic sensor is used in conjunction with the the ultrasonic sensor is only used to determine the distance to an obstacle in front of the RC car and provides accurate results when taking proper sensing angle and surface condition into considerations. On the other hand, Pi camera provides "good enough" measurement results. In fact, as long as we know the corresponding number to the actual distance, we know when to stop the car. Experimental results of detecting distance using a camera are shown as below:

Table 2: Accuracy of Results

Order	Distance						
	1	2	3	4	5	6	7
Actual value	15	20	25	30	35	40	45
Measured value	15.5	19.7	24.1	27.5	32	35.2	39.0

The accuracy of distance measurement using monocular vision approach could be influenced by the following factors:

- Errors in actual measurement of values,
- Variation in the bounding box of an object while detecting the object
- Errors in calibration of the camera,
- Non-uniform ratio between distance and camera coordinates, that is, the further the distance from the object, the more rapid is the change of camera coordinates, thus the greater the error.

Overall, the car could successfully navigate on the track with the ability to avoid front collision, and respond to the stop sign and the traffic light accordingly.

References

[1] Chu Jiangwei Ji Lisheng Guo Lie Libibing Wang Rongben et al,"Study on Method of Detecting Preceding Vehicle Based on Monocular Camera", IEEE Intelligent Vehicles Symposium University of Parma Pama, Italy, pp. 751-755, 2004.

[2] Y. L. Murphey, J. Chen, et al., DepthFinder,"A Real-time Depth Detection System for Aided Driving", Proceedings of the IEEE Intelligent Vehicles Symposium 2000, 2000.10, pp. 122-127.

[3] Firoz Mahmud, Al Arafat, Syed Tauhid Zuhori1 et al., "Intelligent Autonomous Vehicle Navigated by using Artificial Neural Network", 2012 7th International Conference on Electrical and Computer Engineering 20-22, pp. 105-107
[4] Richard Szeliski,"Computer Vision: Algorithms and Applications Vol. 11, pp. 1-7, 2010.
[5] Dave Anderson and George McNeill,"Artificial Neural Networks Technology", Kaman Sciences Corporation 258 Genesse Street, Utica, New York, pp. 6, 1992.