Journal of Simulation and Analysis of Novel Technologies in Mechanical Engineering
14 (3) (2022) 0039~0046

Research article

Road detection by image processing, using neural network

Ahmad Keshavarzi*, Mehdi Keshavarz, Alireza Moradi

Department of Mechanical Engineering, Khomeinishahr Branch, Islamic Azad University, Khomeinishahr, Iran.

*keshavarzi@iaukhsh.ac.ir

(Manuscript Received --- 13 June 2022; Revised --- 31 Oct. 2022; Accepted --- 6 Nov. 2022)

Abstract

One of the methods in road detection is image processing, which is detected according to the contents of the image captured by the car camera. Much research has been done in this area by researchers, who have used a variety of methods to better process the road and detect the direction of vehicle movement. Using the photos taken by the camera installed on the car and their preprocessing, such as removing inefficient corners and turning it into a gray photo during various stages and applying a goblin filter on the photos and extracting the tallest diagram in order to detect the direction of the road. In this research, standard data processing such as line slope for neural network input and LVQ neural network are used to process road images and detect road direction based on three directions: straight, left and right. IAfter several experiments, it was found that the slope of the line in the full photo could not represent the route of the road, therefore, the photos were divided into two, three and four parts and another effective parameter in the system. the seven layers were used to start the neural network training and each time the number of layers increased and the processing results were compared. The best answers were obtained for the number of layers above 70, which is a small difference between the photos divided into three and Four were observed and with a difference of 2%, the four-part photos gave us the best answer.

Keywords: Image processing, Neural network, Road detection, Automatic driver

1- Introduction

Significant advances in automated driving (ADS) systems have been made in the last decade. The use of this system in order to reduce and control traffic and ease of transportation has made this issue one of the major goals of large automotive companies [1-3]. An automated driving system is a complex combination of different components that can be defined as a system for understanding, deciding and acting in a car in which a human uses a mechatronic system instead of driving [3]. Smart car technology is one of the most important tools in urban areas today. This activity has been seriously developing for 15 years [4]. Completely independent driving until 2021/2022 with security level 4 or 5 requires the use of a set of sensor systems. Today's systems use a number of radar and camera systems for semi-independent driving. The design of high-resolution

LIDAR systems with a range of up to 300 meters is still in the pre-development stage [5]. Most car manufacturers today assume that all three sensor systems are required to drive completely independently. As mentioned earlier, there are three main groups of sensor systems: camera-based, radar, and lid-based systems. Although ultrasonic sensors are now available for parking, they are less important for independent driving. Camera and radar systems are currently at levels 1 and 2 and are required for all different levels of automation. Video images provide the most detail to the driver and are very suitable as an input parameter for automatic driving. Depending on the field of image processing, the images created in each issue have special features that pay attention to these features to solve the problem. Relevant is very important, so the proper use to detect the road and the direction of movement of the car, first images must be obtained continuously from the movement of the car and in the first stage pre-processing is done and then the image feature is extracted, which is the image processing stage. And then we analyze the images by extracting the feature of each image using a neural network. So we need to look at the type of feature extracted to use the neural network and how the information enters this system, which is very important using the methods used by others and helps us to get the answer faster. Which will be mentioned in future chapters.

Many methods for intelligent road detection have been proposed in the articles [1, 6-10]. There are many challenges and problems in identifying roads and paths, especially on roads that are suitable without laying or even combing, and even worse, dirt and mountainous roads that make it difficult for humans to identify the right path, so for systems Intelligent has a more difficulty that has been studied and researched a lot and several methods can be offered for it [2, 11-17].

Neural networks are computer programs made up of hundreds, thousands, or millions of artificial brain cells that, like the human brain, perform learning and behavioral function. Behind a neural network is the simulation (simple yet stable copying) of many brain cells connected inside a computer to perform learning, pattern recognition, and human decision-making.

Due to the high flexibility of neural networks and many variables that can be used in this method, it has been used to detect the road [11]. In a 2011 study by the University of Edinburgh, McDonald [10] used neural networks on a predetermined road, and after three laps (neural network training), the car was fully automated using neural networks. Follow the path completely. Fernando et al. [11] also identified the road using an oval neural network and tree processing. they used path neural network and image processing and histogram analysis of images as well as color space change to detect the path.

Using image processing and neural networks, we detect the road by a camera installed on the car. In short, the image taken by the camera installed on the car is considered as input and by performing experimental processes on the photo such as selecting the desired view area and cropping the photos, using the zoom filter, changing the color space, changing the number of pixels and extracting the feature vector. As input to the network and with proper network training, which is the driver's reactions while driving, we prepare the system for smarter and better road detection. According to the algorithm presented in SAE J3016 and the use of road detection system, this work is in the minor intelligent category.

To do the project, first install the mobile phone on the car and then run the Open Camera software to adjust the settings so that one shot is taken every second. More than 500 photos were taken in this way to train the neural network. The process was such that the car was moving at a speed of about 20 kilometers per hour and tried to include all possible situations for the road in the photos taken. To increase the accuracy of the program at different times of the day, both shadowless and noise-free roads and shaded roads or any other item that is considered a noise road were considered. After this stage, the dysfunctional photos were removed and the images that expressed the general condition of the road were separated from them. We reduced the size of these photos to speed up the processing so that the image details were not lost. Here are some other steps you can take to begin the process of preparation for mediation.

In order to achieve the best result, various scenarios were examined on the photographs, which will be described in the following.

2- Select the desired range (ROI)

The image that the camera initially takes has inefficient data, which increases computations and thus slows down the system. If a photograph is considered, it can be roughly divided into two equal halves with a horizontal line (Fig. 1), the upper half is the horizon and the sky, and the lower half is the image of the road and road components. Thus, the upper half of the image is inefficient for road detection and must be removed before any processing

Fig. 1 Selected image in range

a) Convert RGB format to YCbCr

Images and videos are stored in real time in RGB color space, because it is based on the sensitivity of color recognition cells in the human visual system. In digital image processing, YCbCr color space is often used to take advantage of the low resolution capability of the human image system for color due to its luminosity. Therefore, RGB to YCbCr conversion is widely used in image and video processing (Fig. 2).

Fig. 2 Convert RGB image to YCbCr and extract Y component

According to the digital pixel shown in RGB format, 8 bits per sample, in which 0 and 255 represent black and white, can be used to obtain the YCbCr components, respectively, according to the following equations.

$$Y = 16 + \frac{65.738R}{256} + \frac{129.057G}{256} + \frac{25.064B}{256}$$

$$Cb = 128 - \frac{37.945R}{256} - \frac{74.494G}{256} + \frac{112.439B}{256} \quad (1)$$

$$Cr = 128 + \frac{112.439R}{256} - \frac{94.154G}{256} - \frac{18.285B}{256}$$

Fig. 3 Original image, Y component image in YCbCr format

b) Apply Hough conversion on the image

Huff Converter is designed to detect lines. This conversion from the parametric representation form of a line

$$\rho = x\cos(\theta) + y\ \sin(\theta) \quad (2)$$

The variable ρ is the distance from the origin to the direction of the line vector, θ is also the angle between the axis x and this vector. Standard goal conversion (SHT) is a parameter in the matrix space whose rows and columns correspond to the values of ρ and θ, respectively. Hough conversion elements accumulate values. First, each component takes a value of zero. Then for each non-background point in the image, the value of ρ is calculated for each θ. The value ρ gets to the nearest allowed row in the half-rendered conversion. The aggregate components increase. At the end of this process Q belongs to SHT, That is, the Q-points on the XY are on a line marked with ρ-θ (Fig. 4).

c) Draw lines for Huff conversion values for the image

In this section, we draw the lines extracted by the houghpeaks command using the houghlines command. These lines are shown in green images.

d) Extract the highest line

Assuming the longest line is on the road. The longest line will be obtained among the lines drawn in the previous section. In the image, the longest line of each section is shown in red. (Fig. 5).

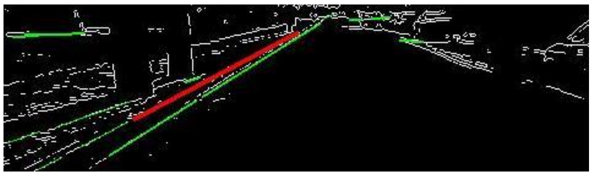

Fig. 4 Draw hough lines and extract the longest line for a straight path

e) Character extraction for images

Given that a line can be drawn with a slope and a point from it, we select the following two parameters as the characteristics of each image.

•The highest line slope of each section

•Select a point on the line

Table 1: Slope information and one point of the highest Huff line for 35 road images

direction	Point height	Point width	Slope	direction	Point height	Point width	Slope
Go straight	135	393	-7.23692	Go straight	19	401	-30.1243
Go straight	135	114	14.39359	Left	19	445	1.041627
Go straight	135	114	1.762391	Go straight	106	110	1.709814
Go straight	135	386	-7.12502	Go straight	106	411	-29.9816
Go straight	135	328	3.945186	Go straight	106	373	-27.9169
Go straight	132	105	5.376705	Left	106	111	12.45825
Go straight	132	368	-0.62276	Go straight	106	349	-42.93
Go straight	132	383	0	Go straight	106	478	-19.6833
Go straight	132	190	0.868051	Go straight	124	159	13.17987
Go straight	132	254	-29.4275	Go straight	124	188	16.18921
Go straight	16	428	-29.4759	Go straight	124	129	19.02561
Go straight	16	459	-18.6495	Go straight	124	237	-0.89517
Left	16	459	-18.6495	Go straight	124	116	27.07208

As can be seen from the information in the table above, the slope of the highest line does not give accurate information about the direction of the road and in most cases its information is misleading. This method is not efficient for the intended destination. A way to solve this problem will be provided below.

To solve the problem of the previous step, this time the output of the image after applying the Zoebel filter is divided into two, three and four parts according to a vertical cut from the middle and each data set is given to the neural network for testing (Fig. 6).

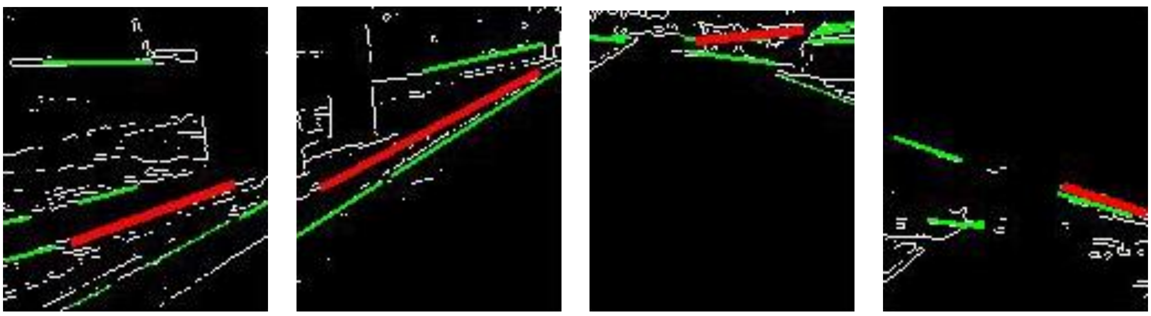

Fig. 5 Apply Hough conversion on images divided into four parts

3- Neural network testing

The information obtained from the analysis of road images is stored in an Excel file and the identity matrix vectors are considered to determine the direction of the road. Given that the number of outputs in question is 3; That is, the straight, left, or right path of the first vector is assigned to the left, the middle vector to the direct direction, and the third vector to the right, corresponding to a set of binary or triple images.

If we consider the identity matrix of 3 in 3, the first column of this matrix represents the deviation to the left, the second column represents the straight path and the third column represents the deviation to the right, which are called the first, second and third categories, respectively. From the available data, 60% is used for system training and 20% for network testing. These values are randomly selected by the software. According to the experimental information, to improve the network accuracy and reduce the error, the LVQ.1 network is executed with 5 repetitions and then the LVQ2.1 network is adjusted with 500 repetitions. The learning rate in this system is assumed to be 0.01 by default, if For five consecutive repetitions, if no improvement is made to the system, the network will be shut down. These are, in fact, the conditions for leaving the network, and if each of them is satisfied, the network will stop.

The only adjustable parameter in the network is the number of network layers, the values of which start from 7 layers. In this work, the optimal network is compared in terms of the number of available layers with 3 inputs and 4 inputs. Each figure contains 4 images, each of which is a four-by-four table. If the first column is considered, the first cell of this column indicates the number of images that are in the first category and have been correctly identified. The second cell of this column is the number of images that are in the first category but are categorized by mistake in the second category and the third cell of this column is the number of images that are in the first category but are categorized by error in the third category. The percentage displayed below each number is the ratio of the total number of input images to the network. The last row of this column shows the correct categorization function. The same can be done for the second and third columns of this row. The fourth column and fourth row houses also show the overall performance of the network. The neural network output based on the division of photos into three parts is shown in Fig. 6 andFig. 7 is shown the output based on the division of photos into four parts.

Fig. 6 Information table of optimal number of 90 layers with three inputs

Fig. 7 Information table of 100 layers with four inputs

In the Fig. 9, the red lines indicate the accuracy in percentage for the three and four inputs, respectively, which are marked in blue and red. In fact the error is optimized in terms of the number of network layers for 3 and 4 inputs, respectively.

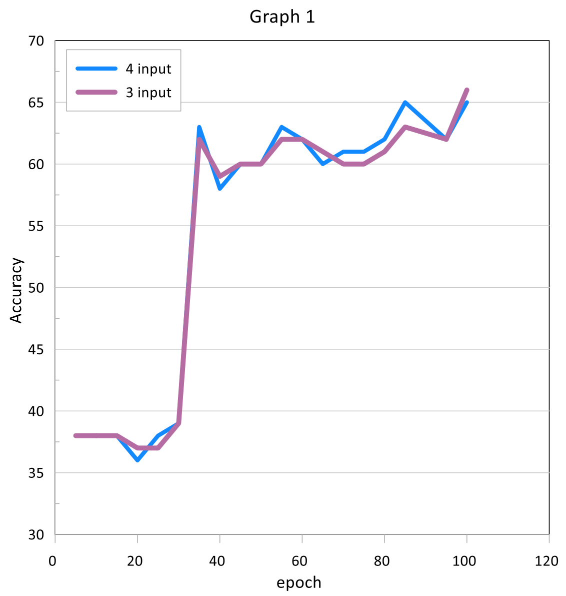

Fig. 9 Neural network accuracy based on the number of inputs

4- Conclusion

If we look at the case as a whole, we will not be able to achieve the ideal accuracy of road detection solely on the basis of road signs. It should also be noted that in consecutive photos taken from the road, the slope of the road index in the current photo does not change abruptly with the previous photo, and a maximum value can be considered for it if the difference between the slope of the current photo and the photo The previous one is more than a certain amount in that photo has been correctly detected due to the presence of slope noise and that photo should not be considered in the process of road detection calculations. Based on the above, it is recommended that you use other devices such as GPS and Lidar in addition to reduce system error as much as possible and the car to detect its route with maximum confidence.

References

[1] Chan, C.-Y. (2017). Advancements, prospects, and impacts of automated driving systems. *International journal of transportation science and technology, 6*(3), 208-216.

[2] Fritsch, J., Kuehnl, T., & Geiger, A. (2013). *A new performance measure and evaluation benchmark for road detection algorithms.* Paper presented at the 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013).

[3] Thorn, E., Kimmel, S. C., Chaka, M., & Hamilton, B. A. (2018). *A framework for automated driving system testable cases and scenarios*. Retrieved from

[4] Atiqur, R. Automated smart car parking system for smart cities demand employs internet of things technology. *Int J Inf & Commun Technol ISSN, 2252*(8776), 8776.

[5] Magnusson, C. P. (2021). *Unsettled Topics Concerning Coating Detection by LiDAR in Autonomous Vehicles* (2640-3536). Retrieved from

[6] Assidiq, A. A., Khalifa, O. O., Islam, M. R., & Khan, S. (2008). *Real time lane detection for autonomous vehicles.* Paper presented at the 2008 International Conference on Computer and Communication Engineering.

[7] Arce, F., Zamora, E., Hernández, G., & Sossa, H. (2017). Efficient lane detection based on artificial neural networks. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 4*, 13.

[8] Saha, A., Roy, D. D., Alam, T., & Deb, K. (2012). Automated road lane detection for intelligent vehicles. *Global Journal of Computer Science and Technology*.

[9] Van Winden, K., Biljecki, F., & Van der Spek, S. (2016). Automatic update of road attributes by mining GPS tracks. *Transactions in GIS, 20*(5), 664-683.

[10] Huval, B., Wang, T., Tandon, S., Kiske, J., Song, W., Pazhayampallil, J., . . . Cheng-Yue, R. (2015). An empirical evaluation of deep learning on highway driving. *arXiv preprint arXiv:1504.01716*.

[11] Pomerleau, D. A. (1989). *Alvinn: An autonomous land vehicle in a neural network*. Retrieved from

[12] Macdonald, I. D. G. (2011). A Simulated Autonomous Car. In: Citeseer.

[13] Foedisch, M., & Takeuchi, A. (2004). *Adaptive real-time road detection using neural networks.* Paper presented at the Proceedings. The 7th International IEEE Conference on Intelligent Transportation Systems (IEEE Cat. No. 04TH8749).

[14] Arce, F., Zamora, E., Sossa, H., & Barrón, R. (2016). *Dendrite morphological neural networks trained by differential evolution.* Paper presented at the 2016 IEEE Symposium Series on Computational Intelligence (SSCI).

[15] Arce, F., Zamora, E., Barrón, R., & Sossa, H. (2016). Dendrite morphological neurons trained by differential evolution. *Computational intelligence*.

[16] Zamora, E., & Sossa, H. (2017). Dendrite morphological neurons trained by stochastic gradient descent. *Neurocomputing, 260*, 420-431.

[17] Davidson, J. L., & Ritter, G. X. (1990). *Theory of morphological neural networks.* Paper presented at the Digital Optical Computing II.