Real-Time Car Detection-Based Depth Estimation Using Mono Camera

I. Mohamed Elzayat[1], M. Ahmed Saad[1], M. Mohamed Mostafa[1], R. Mahmoud Hassan[1],
Hossam Abd El Munim[2], Maged Ghoneima[3], M. Saeed Darweesh[4,5] and Hassan Mostafa[1,4]

[1]Electronics and Electrical Communications Engineering Department, Faculty of Engineering, Cairo University, Egypt.
[2]Computer and Systems Engineering Department, Faculty of Engineering, Ain Shams University, Egypt.
[3]Mechatronics Engineering Department, Ain-Shams University, Egypt.
[4]Nanotechnology Department, Zewail City of Science and Technology, Egypt
[5]Electronics and Communications Engineering Department, Institute of Aviation Engineering and Technology, Egypt.

***Abstract*—Object depth estimation is the cornerstone of many visual analytics systems. In recent years there is a considerable progress has been made in this area, while robust, efficient, and precise depth estimation in the real-world video remains a challenge. The approach utilized in this presented paper is to estimate the distance of surrounding cars using a mono camera. Using YOLO (You Only Look Once) in the detection process, by generating a boundary box surrounding the object,then an inversion proportional correlation between the distance and the boundary box's dimensions (height, width) is ascertained. Getting the exact equation between the studied variables; the dependent variables are the distance, and independent variable is the height and width of YOLO boundary box. In the regression model, multiple regression techniques were acclimated to evade heteroskedasticity and multi-collinearity problems.Achieving a real-time detection with a 23 FPS (Frame Per Second) and depth estimation accuracy 80.4%.**

I. Introduction

Self-Driving Vehicles also Known as autonomous Vehicles has taken a huge interest worldwide. Several major automobile manufacturers have set targets to launch commercially available fully autonomous vehicles. By 2021 Ford will have a fully autonomous vehicle in a commercial operation [1]. Self-Driving Vehicles have five core components: Computer Vision, Sensor Fusion, Localization, Path, and control [2]. Computer Vision means how the camera used the images to figure out what the world around us looks like. While sensor fusion is how the data is incorporated from other sensors like Light Imaging Detection and Ranging (LiDAR), radars and the ultrasounds to get a richer understanding of the surrounding environment [2].

A stereo vision system has two cameras located at a known distance and take pictures of the scene at the same time. Using the geometry of the cameras, we can apply algorithms and create the geometry of the environment. However, these algorithms are so complicated and single cameras are already available in series production performing numerous vision-based driver assistance algorithms such as intelligent headlight control and night view [3].

There were many attempts to estimate the in-path distances of the objects using a single forward-looking camera mounted on the dashboard based on the camera properties and geometry applied to the input images, also the relation between 2D and actual 3D view of the image is estimated. The depth estimation technique is to get the vanishing point that assumed to be the center of the image when the camera axis is parallel to the optical plan. The distance is estimated depending on the positions of the bottom of the object (Y-axis). The relation between the Y and the distance is reverse proportion. The classification technique and the detection is done by the Support Vector Machine (SVM). However, its weaknesses are the processing time of the classification and detection and it depended on only one parameter which is not enough to get accurate distances in all cases [4].

While the work presented in [5] is fixated on depth evaluation utilizing a single camera. The algorithm utilizes the markers to update its orientation and to compute the distance. However, the algorithm is capable of engendering the output in just a few seconds with virtually 91%, it's not engendering a real-time algorithm which is the main core in autonomous systems. Additionally, the work in [5] depends on the simple regression technique by utilizing the covered area without taking the consideration of the partial quandary which will be discussed later, while this work depends on the multiple regression technique and compare it with different simple regression techiques taking in consideration all the possible test cases.

This paper proposes a new algorithm to estimate the distance using only a mono camera. The idea of the presented work is based on collecting data from this mono camera which its parameters are: 29mm focal length, f2.2 lens's aperture, with a 12-megapixel resolution. Then the range must be measured by using accurate measurement tools. The cues which can be used are the size of the object and its position.

The neural network is used to get the size of the object which is related to the classification and the detection based on YOLO network [6]. Regression is used to get the relation since we have the following inputs: object size and its distance. It is a big challenge to make a real-time system so once the detection is done the distance is estimated, which is embedded in the same process (End to End Process).

The paper is organized as follows: Section II describes the

978-1-5386-8167-1/18/$31.00 ©2018 IEEE

proposed depth estimation technique. In Section III the results are shown and the Conclusion is shown in Section IV.

II. Proposed Depth Estimation Technique

Automatic detection and verification of objects in images is a central challenge in computer vision and pattern analysis research. A new depth estimation algorithm is proposed in this paper, by fusing YOLO in detection and Multiple regression in estimation in one single process. The whole system is end-to-end which means once the image enters the network the object is detected and the distance is estimated in one process.

The main objective of this algorithm is to be used in Autonomous cars, a real-time processing being maintained, using YOLO which is, an object detection system targeting real-time processing.

While multiple regression, used to learn more about the relationship between several independent or predictor variables and a dependent or criterion variable is utilized to describe the relationship between the distance of the object and different parameters of the boundary box such as width, height or both. Equation [1] is the general form of the multiple regression model.

$$Y = \beta 0 + \beta 1X1 + \beta 2X2 + .. + \beta kXk + e. \tag{1}$$

Where β's are the estimated slope of the regression of Y on X, X is the independent variable, and e is the regression constant.

The problem of estimating the distance of the object can be solved by taking the advantage of the boundary box which generated from the neural network. Predicated on the box dimensions, the distance can be estimated by different approaches such as: the width approach in which the distance accumulated is based on only one parameter which is the width of the car, while the height approach in which the distance accumulated is based on only one parameter which is the height of the car, or by the multiple regression approach in which the distance accumulated is based on more than one parameter which are both the width and the height of the car.

To achieve the functional form of the regression model two fitting tools were utilized Minitab [7] and STATA [8] tools. The statistical model used is a multiple regression which is made to indicate the relationship between the width, height, and distance of the object. To get the equation accumulating data of authentic distance of cars is required then, gathered more than one hundred pictures for various cars in various positions, and different orientations. Thereafter, R Squared and adjusted R Squared techniques were used to test the goodness of the equation of the model which will be discussed in details in the following subsection.

A. *R Squared and Adjusted R Squared*

R squared shows how well terms (data points) fit a curve or line. While adjusted R squared additionally designates how well terms fit a curve or line, but adjusts for the number of terms in a model. If more and more useless variables to a model are added, adjusted r-squared will decrement and vice versa.

There is one main distinction between R squared and the adjusted R squared: R squared assumes that every single variable expounds the variation in the dependent variable. While the adjusted R Squared gives the percentage of variation explicated by only the independent variables that authentically affect the dependent variable. The higher adjusted R squared model was selected with the following equation.

$$\begin{aligned} Distance = 2021.256 - 1.276714 \cdot height - 0.6042361 \cdot width \\ + 0.0004751 \cdot hieght \cdot width \end{aligned} \tag{2}$$

All the coefficients in Equation [2] are obtained from STATA after the data is applied. A detailed discussion about each term of the equation of this model, and how each one of them affects the model is described below.

1) Width Term: The width term existence is consequential in the equation, but if it is utilized individually, the equation will not be valid in two paramount cases:

- Sizably voluminous sized cars have astronomically immense boundary boxes, so they appear more proximate than diminutive sized cars with minute boundary boxes.
- The width transmuting with car orientation as it gets more sizably voluminous as the car peregrinates to a side and we require the parameter only depends on the car itself, not its orientation.

2) Height Term: If the height term is used individually, it will cover the two cases mentioned above and it will be much better than the width. However, the equation will not be valid as well in case of the partial detection problem. If a part is captured from the car, the boundary box will appear smaller and the distance will be falsely far. When the box is partially truncated due to the proximity of the upcoming car, the authentic value of the real value of height will be shrunk, thus the distance will be widely far.

The obtained multiple regression model is integrated into our neural network (YOLO) to be processed in the real-time detection to know how far cars are going whenever they are moving. The distance will be estimated for each detected car and the user could visually perceive the value in meters above the boundary box that bounding the detected car.

3) Interaction Term: The presence of a significant interaction indicates that the effect of one predictor variable on the response variable is different at various values of the other predictor variable. It is tested by integrating a term to the model in which the two predictor variables are multiplied.

III. Results and Discussion

The performance of the proposed system depends on a real-time detection utilizing YOLO and its run on Jetson TX2 Kit [10] which is the fastest and the most power-efficient

TABLE I
COMPARISON BETWEEN MULTIPLE REGRESSION MODELS

	Model equation	Adjusted R Squared	Test Cases
Model 1	$Distance = 1618.83 - 0.5369597 \cdot height - 0.2773714 \cdot width$	55%	Failed
Model 2	$Distance = 1546.7 - 0.634 \cdot height - 0.0801 \cdot width$	69.47%	Failed
Model 3	$Distance = 0.72318(510160 \cdot width0.983821) + 0.1863(72085.6 \cdot height0.691721) + 159.7768$	76%	Failed
Model 4	$Distance = 2048 - 1.312 \cdot height - 0.6160 \cdot width + 0.000490 \cdot width \cdot height$	66.30	Succed
Model 5	$Distance = 1951.751 - 1.106 \cdot height - 0.4893 \cdot width + 0.00032 \cdot width \cdot height$	64.50%	Succed
Model 6	$Distance = 2021.256 - 1.276714 \cdot height - 0.6042361 \cdot width + 0.0004751 \cdot hieght \cdot width$	80.40%	Succed

TABLE II
FINAL COMPARISON BETWEEN THE THREE MODELS

	Model equation	Adjusted R Squared
Width	$(1.096 \cdot 10^5) \cdot width^{-7.24335644 \cdot 10^{-1}}$	55%
Height	$(7.20855715 \cdot 10^4) \cdot height^{-6.91721311 \cdot 10^{-1}}$	66%
Non linear multiple regression	$2021.256 - 1.276714 \cdot height - 0.6042361 \cdot width + 0.0004751 \cdot hieght \cdot width$	80%

embedded AI (Artificial Intelligence) computing device, built around an NVIDIA Pascal-family GPU and loaded with 8 GB of memory and 59.7 GB/s of memory bandwidth, as a trail for reaching 23 frames per second. As previously mentioned, power regression between the collected distances data and the width of the detected car is utilized to compose a general width approach which estimates the distance of any detected car. A precision of 55 % is achieved, However, the problem of side view cars which results in the erroneously estimated distance appears.

After applying the proposed width approach the output from YOLO shows an obvious difference between side and back view. The side view is appeared to be closer as shown in Fig. 1.

Consequently, height approach is applied. In the height approach, side view cars quandary is solved with better overall precision than the width approach with a percentage of 55%. As it is based on the height of the car which does not differ from the back or side view. In Fig. 2 the two cars are appeared at almost the same distance using the height approach.

Another problem appears which is the partial cars quandary where a part of the car is detected when it is so close. Therefore, it gives an erroneous more immensely colossal distance.

Consequently, multiple regression is implemented. In multiple regression, height and width approaches are cumulated to solve antecedent problems in which a part of the car is truncated vertically, the width will be the dominant parameter to estimate the distance and vice versa. Moreover, height solves both the side view problem and horizontally truncated cars. That is why the weight given to the height term is higher than the one given to the width. Fig. 3. shows that however, a part of the car is vertically truncated, the distance is correctly estimated. Results can be more authentic by integrating more parameters not only height and width.

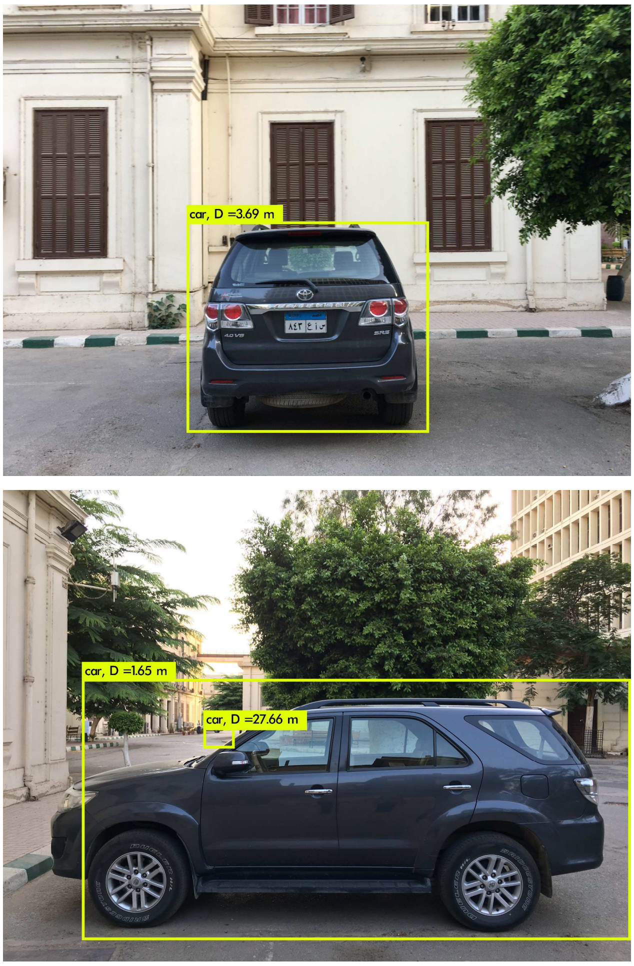

Fig. 1. The distance at 3.69m from back view and 1.65 from side view "side view problem"

Table 1 shows the six linear and non-linear multiple regression models estimated describing the relationship between the three parameters. Adjusted R squared technique was used to differentiate between them besides applying the test cases: the side car and the partial one to see the result for each of them.

Table 2 shows a final comparison between the three approaches (width, height, and non-linear multiple regression) where the multiple regression considered to be the best.

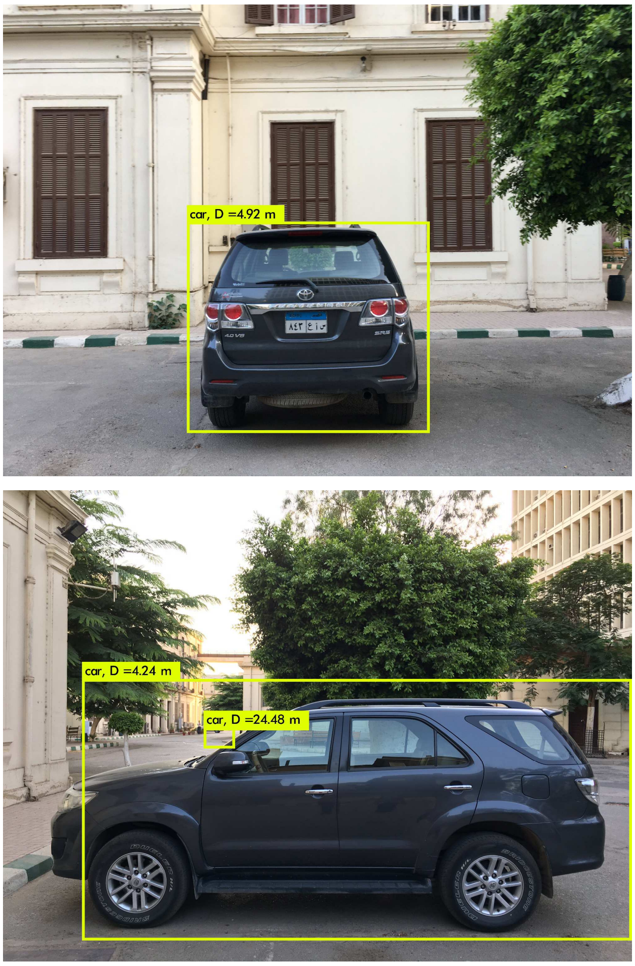

Fig. 2. The distance around 4m for both side and back views

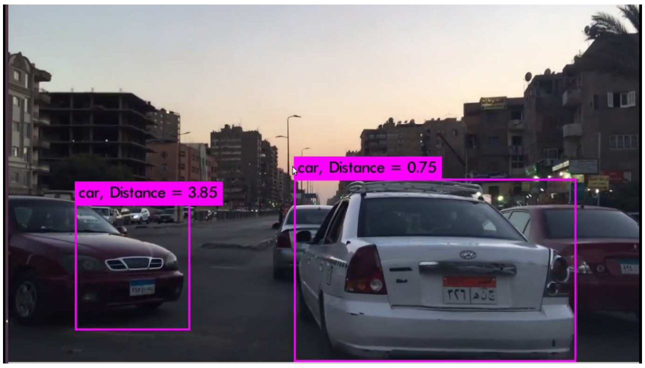

Fig. 3. The distance is 0.75m "Partial problem is solved in multiple regression"

IV. Conclusion

The main target of this paper is to build a module in a self-driving car, so distances of surrounding cars are estimated through different approaches. Depth estimation algorithm is engendered to detect the cars utilizing an only mono camera, In which boundary box is engendered from YOLO by finding the cognition between the distance and the dimensions of the boundary box.

The regression technique is utilized to fit the distance with the width and the height of the car with different linear and non-linear models. But there is a quandary of the width changing with car orientation as it gets more sizably voluminous as the car moves to the side view. Thus, the model with lower weight for width and highest R squared has been opted for.

Acknowledgement

This research was partially funded by IHUB at Ain Shams University and ONE Lab at Cairo University and Zewail City of Science and Technology.

References

[1] "Ford Corporate". [Online]. Available: https://corporate.ford.com/innovation/autonomous-2021.html. [Accessed: 08-Aug-2018].
[2] T.Litman,"Autonomous vehicle implementation predictions: Implications for Transport Planning," Victoria Transport Policy Institute, vol. 28, 2017.
[3] Stereo vision images processing for real-time object distance and size measurements Kuala Lumpur, Malaysia, July 2012.
[4] Depth Estimation Using Monocular Camera .Apoorva Joglekar, Devika Joshi, Richa Khemani, Smita Nair, Shashikant Sahare.
[5] Depth Estimation from a Single Camera Image using Power Fit Muhammad Umair Akhlaq, Umer Izhar and Umar Shahbaz. April 2014.
[6] J. Redmon et al., "You Only Look Once: Real-Time Object Detection," [EB/OL].
[7] Minitab, Minitab, Inc. [Online]. Available: http://www.minitab.com/. [Accessed: 08-Aug-2018].
[8] Stata: Data Analysis and Statistical Software. [Online]. Available: https://www.stata.com/.
[9] J. Cohen and J. Cohen,"Applied multiple regression/correlation analysis for the behavioral sciences," Mahwah, NJ: L. Erlbaum Associates, 2003.
[10] "CEO Jensen Huang takes the stage at SIGGRAPH 2018," NVIDIA. [Online]. Available: http://www.nvidia.com/page/home.html. [Accessed: 08-Aug-2018].
[11] L. Ott, M. Longnecker, and J. D. Draper, "An introduction to statistical methods and data analysis," Boston, MA: Cengage Learning, 2016.
[12] Y. Hochberg, G. Weiss, and S. Hart, On Graphical Procedures for Multiple Comparisons, Journal of the American Statistical Association, vol. 77, no. 380, pp. 767772, 1982.
[13] Ott, M. Longnecker, and J. D. Draper, An introduction to statistical methods and data analysis. Boston, MA: Cengage Learning, 2016.
[14] S. Wolfram, The mathematica book. Champaign, IL: Wolfram Media, 2004.
[15] W. I. King, The Annals of Mathematical Statistics, The Annals of Mathematical Statistics, vol. 1, no. 1, pp. 12, 1930.