Computer Vision Studies
Volume 2, Issue 1, No. 1
Received 17 February 2023; Accepted 2 March, 2023
Published online 20 March 2023

Object size measurement and camera distance evaluation for electronic components using Fixed-Position camera

Minh Long Hoang
Department of Engineering and Architecture, University of Parma, Parma, Italy

Abstract. This article works on applying Open-Source Computer Vision Library (OpenCV) minimum area rectangle to measure electronics component dimension. A rotative contour covers the considered objects with the detected width and length. The pixel and real-world unit ratio are identified with a reference object for other device size accomplishment. The experiment contains Arduino UNO, microchip ESP32-WROOM, Inertial Measurement Unit (IMU) sensor, and a 9 V battery. The approach shows the less complicated way to achieve the appropriate results, with an absolute error of less than 3mm. The distance between the camera and object is also calculated based on the relationship between camera parameters and actual object height. The research concentrates on the size measurement of electronic components and the distance estimation from the object to the monitoring camera.

Keywords. Computer vision, size measurement, electronics, camera distance evaluation.

1. Introduction

Nowadays, computer vision [1, 2] has played a significant role in the electronic field. The industry's demand for object size tracking has increased to validate production quality, such as in conveyor systems. This research works on the circumstance where the camera is mounted in a fixed position to proceed the size measurement with object detection. Various approaches are utilized for object measurement with computer vision. Deep learning [3, 4] is applied with the You Only Look Once (YOLO) algorithm [5, 15-18] or Convolutional Neural Network (CNN) [6, 19-21] for the object recognition and measurement. Although the obtained results have sufficient accuracy, many custom datasets must train and test the model. It requires high time consumption and robust Central Processing Unit (CPU), increasing system complexity.

Meanwhile, OpenCV is a library of programming functions mainly for real-time computer vision [7]. OpenCV Minimum Area Rectangle is a simple but effective technique for object size measurement. The bounding area is automatically drawn on the electronic components with the object size in pixel (px). To find the relationship between pixel and real-world dimension (mm), camera calibration is processed with a reference object size (ESP32-WROOM) detection in px and its actual size to seek out the ratio between them. From this stage, all other object size will be converted from px to mm, depending on this ratio. In addition, this article also depicts the distance evaluation between object and camera, which is extremely useful in many situations.

The paper is organized as follows: the 1st part concerns the technical principle of object measurement and distance evaluation. In the experimental part, the size of 4 electronic components: a microcontroller board, a microcontroller chip, an IMU sensor [8, 9, 22], and a 9V battery are detected. Finally, the conclusion is drawn out.

Figure 1. Microcontroller contour detection by OpenCV system

2. Technical principle

2.1. Object measurement

Thresholding is a segmentation technique in computer vision that separates the foreground such as the interested object from the image's background. A threshold (T) is set to distinguish pixel intensity between dark and bright colors. T can be set manually, but there is a problem of the manual insert which may not be sufficient for the full image. Due to variations in lighting conditions, shadowing, etc., the manual T works for a certain part of the image but fails on the different segments.

Therefore, adaptive thresholding is applied in this case. In the arithmetic mean, each pixel in the neighborhood contributes equally to computing T. By applying adaptive thresholding, threshold local regions of the input image (rather than using a global value of T).

$$T = mean\ (LSR) - C \tag{1}$$

Where:

- T is threshold.
- LSR the local sub-region of the image.
- C is some constant to adjust T.

Here, the pixel neighborhood size is 31, which means the grayscale pixel intensity value of each 31×31 sub-region in the image to compute our threshold value T. In addition, C is 10, added or subtracted from the mean value to improve the discrimination between the object and background.

With the support of OpenCV, the minimum area rectangle technique generates the bounding rectangle with the minimum area surrounding the electronic devices that can also rotate to fit the best with the objects, as shown in Figure 1. At this point, the width and height of the concerned objects are acquired in px.

To accomplish the relationship between the pixel and the real-world unit, the real height of an object is measured, then the ratio will be calculated as below:

$$ratio = \frac{real\ size\ (mm)}{pixel\ size\ (px)} \tag{2}$$

With this ratio, other electronic components can be calculated:

$$real\ size\ (mm) = pixel\ size\ (px) * ratio \tag{3}$$

2.2. Camera distance evaluation

As explained in [10], once the real size of the object is estimated, the distance between the object can be calculated based on the following Equation:

$$D(mm) = \frac{f\ (mm) * real\ height\ (mm) * image\ height\ (px)}{object\ height\ (px) * sensor\ height\ (mm)} \tag{4}$$

Where:

- D is the distance between the camera and object.
- f is the focal length of camera.
- object height is detected by image processing with OpenCV.
- sensor height is vertical dimension of the sensor's active pixel area of camera.

In the case, the distance between the camera and the object is known, it is possible to calculate the object height without the ratio calibration:

$$real\ height = \frac{Distance * object\ height\ (px) * sensor\ height\ (mm)}{f\ (mm) * real\ height\ (mm) * image\ height\ (px)} \tag{5}$$

3. Experimental and data analysis

In the experiment, 4 electronic devices are utilized for the system validation:

- Arduino UNO board [11]
- Battey 9v.
- MPU 6050 sensor [12], [13].
- ESP32-WROOM [14].

The ESP32 is used as the reference object for image calibration. As shown in Figure 2, the developed system is able to detect the contour on the objects and estimate the pixel number of their width and height.

Table 1 and Table 2 report the object dimension in px and mm, respectively. The digital caliper measures each electronic component's real size as ground truth to compare with the system evaluation. The errors are stated in Table 3. With this technique, the errors are minor, less than 3 mm, which is in the acceptable range.

Table 1. Pixel value of object size

Parameters	Arduino	Battery	IMU sensor	ESP32 Wroom
Width (px)	212.36	107	63	71
Height (px)	307.61	196	81	97

Table 2. Value of object size in mm

Parameters	Arduino	Battery	IMU sensor	ESP32 Wroom
Width (mm)	55.82	28.12	16.56	18.66
Height (mm)	80.86	51.52	21.29	25.50

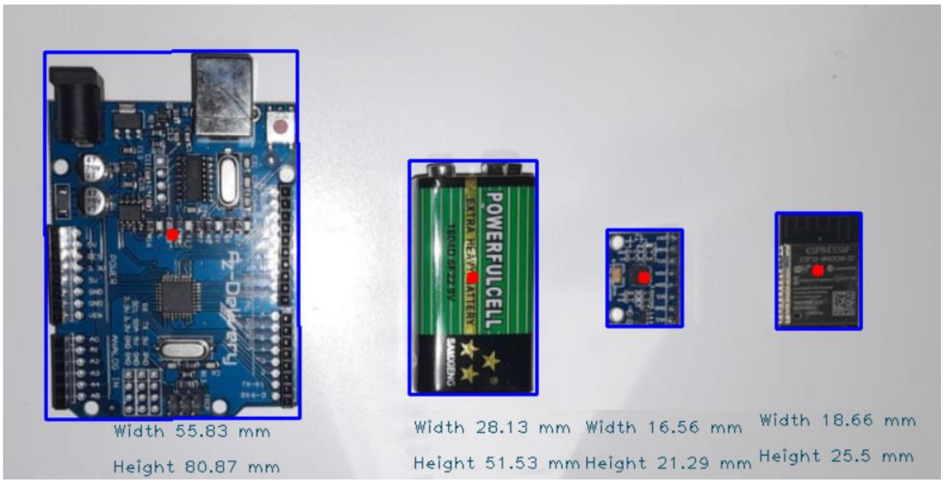

Figure 2. Contour and size detection

Table 3. Absolute error of size detection

Parameters	Arduino error	Battery error	IMU sensor error	ESP32 Wroom error
Width (mm)	2.43	2.85	0.56	0.65
Height (mm)	2.56	2.62	2.29	0

Moreover, the distance from objects to the camera is evaluated based on the ESP32 module height. Substituting the camera data from Table 4 into the Equation (3), the camera distance is approximately 264.35 mm.

Table 4. Camera data

Parameter	Value
Image width (px)	604
Image height (px)	1069
Focal length (mm)	5.23
Sensor width (mm)	3.66
Sensor height (mm)	2.74

4. Conclusion

This paper presented vision-based development for the electronics component based on OpenCV library, with the error of less than 3mm. To achieve more precise results, more complicated calibration or other robust AI approaches are required to proceed. However, the advantage of this method is simple, direct and sufficiently effective. Furthermore, another strong point of the concerned technique is to work directly via fast calibration without a large dataset. On the other hand, the distance between the camera and objects was evaluated, providing practical information about the camera position. In the future, more computer vision techniques will be researched deeply and compared with each other in terms of objective measurement.

References

[1] V. Lepetit, "On Computer Vision for Augmented Reality," 2008 International Symposium on Ubiquitous Virtual Reality, Gwangju, Korea (South), 2008, pp. 13-16, doi: 10.1109/ISUVR.2008.10.

[2] V. Wiley and T. Lucas, "Computer Vision and image processing: A paper review," International Journal of Artificial Intelligence Research, vol. 2, no. 1, p. 22, 2018.

[3] S. S. Ankad, B. Shivaprasad Raj, A. Nasreen, S. Mathur, P. Ramakanth Kumar and K. Sreelakshmi, "Object Size Measurement from CCTV footage using deep learning," 2021 IEEE International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS), Bangalore, India, 2021, pp. 1-5, doi:

[4] M. L. Hoang and A. Pietrosanto, "New Artificial Intelligence Approach to Inclination Measurement Based on MEMS Accelerometer," in IEEE Transactions on Artificial Intelligence, vol. 3, no. 1, pp. 67-77, Feb. 2022, doi: 10.1109/TAI.2021.3105494. 10.1109/CSITSS54238.2021.9683671.

[5] X. Yu, T. W. Kuan, Y. Zhang and T. Yan, "YOLO v5 for SDSB Distant Tiny Object Detection," 2022 10th International Conference on Orange Technology (ICOT), Shanghai, China, 2022, pp. 1-4, doi: 10.1109/ICOT56925.2022.10008164.

[6] R. Xin, J. Zhang and Y. Shao, "Complex network classification with convolutional neural network," in Tsinghua Science and Technology, vol. 25, no. 4, pp. 447-457, Aug. 2020, doi: 10.26599/TST.2019.9010055.

[7] A. Zelinsky, "Learning opencv---computer vision with the opencv library (Bradski, G.R. et al.; 2008) [on the shelf]," IEEE Robotics & Automation Magazine, vol. 16, no. 3, pp. 100–100, 2009.

[8] M. L. Hoang, A. A. Nkembi, and P. L. Pham, "Real-time risk assessment detection for weak people by parallel training logical execution of a supervised learning system based on an IOT wearable MEMS accelerometer," Sensors, vol. 23, no. 3, p. 1516, 2023.

[9] M. L. Hoang and A. Pietrosanto, "A New Technique on Vibration Optimization of Industrial Inclinometer for MEMS Accelerometer Without Sensor Fusion," in IEEE Access, vol. 9, pp. 20295-20304, 2021, doi: 10.1109/ACCESS.2021.3054825.

[10] Wayne Fulton, "Www.scantips.com," Calculator to compute the Distance or Size of an Object in a photo Image. [Online]. Available: https://www.scantips.com/lights/subjectdistance.html. [Accessed: 16-Feb-2023].

[11] T. A. Team, "Uno R3: Arduino documentation," Arduino Documentation | Arduino Documentation. [Online]. Available: https://www.arduino.cc/en/Main/ArduinoBoardUno. [Accessed: 16-Feb-2023].

[12] MPU-6000/MPU-6050 Product Specification Document Number: PSMPU-6000A-00, August, 2013.

[13] M. L. Hoang, M. Carratù, V. Paciello and A. Pietrosanto, "Noise Attenuation on IMU Measurement for Drone Balance by Sensor Fusion," 2021 IEEE International Instrumentation and Measurement Technology Conference (I2MTC), Glasgow, United Kingdom, 2021, pp. 1-6, doi: 10.1109/I2MTC50364.2021.9460041.

[14] Espressif Systems, ESP32-WROOM-32 Datasheet, 2022.

[15] M. Mahendru and S. K. Dubey, "Real Time Object Detection with Audio Feedback using Yolo vs. Yolo_v3," 2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, 2021, pp. 734-740, doi: 10.1109/Confluence51648.2021.9377064.

[16] Y. -S. Poon, C. -C. Lin, Y. -H. Liu and C. -P. Fan, "YOLO-Based Deep Learning Design for In-Cabin Monitoring System with Fisheye-Lens Camera," 2022 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, 2022, pp. 1-4, doi: 10.1109/ICCE53296.2022.9730235.

[17] X. Wang, X. Jiang, Z. Xia and X. Feng, "Underwater Object Detection Based on Enhanced YOLO," 2022 International Conference on Image Processing and Media Computing (ICIPMC), Xi'an, China, 2022, pp. 17-21, doi: 10.1109/ICIPMC55686.2022.00012.

[18] X. Wang, X. Jiang, Z. Xia and X. Feng, "Underwater Object Detection Based on Enhanced YOLO," 2022 International Conference on Image Processing and Media Computing (ICIPMC), Xi'an, China, 2022, pp. 17-21, doi: 10.1109/ICIPMC55686.2022.00012.

[19] F. A. Uçkun, H. Özer, E. Nurbaş and E. Onat, "Direction Finding Using Convolutional Neural Networks and Convolutional Recurrent Neural Networks," 2020 28th Signal Processing and Communications Applications Conference (SIU), Gaziantep, Turkey, 2020, pp. 1-4, doi: 10.1109/SIU49456.2020.9302448.

[20] M. Bersali, A. Rachedi and H. Bouarfa, "Convolutional neural network for relays selection in the Internet of Vehicles," 2021 International Conference on Forthcoming Networks and Sustainability in AIoT Era (FoNeS-AIoT), Nicosia, Turkey, 2021, pp. 57-61, doi: 10.1109/FoNeS-AIoT54873.2021.00022.

[21] P. Samudre, P. Shende and V. Jaiswal, "Optimizing Performance of Convolutional Neural Network Using Computing Technique," 2019 IEEE 5th International Conference for Convergence in Technology (I2CT), Bombay, India, 2019, pp. 1-4, doi: 10.1109/I2CT45611.2019.9033876.

[22] M. L. Hoang and A. Pietrosanto, "Yaw/Heading optimization by drift elimination on MEMS gyroscope," Sensors and Actuators A: Physical, vol. 325, p. 112691, Jul. 2021, doi: https://doi.org/10.1016/j.sna.2021.112691.