

Distance measurement between objects in an image

A study on shape detection
ENGI-9804: Project Proposal
Sourav Sarker (souravs@mun.ca)

Proposal: This project will demonstrate how to measure the distance between the object in an image using image processing methods. Although this is a classical image processing project in the future, this project can be expanded to "distance measurement for self-driving cars using the stereo camera"¹, which can be useful for traffic control and driving safety.

In this project, before detecting the distance first, we should gather some information about the objects, i.e. dimensions and the coordinates, then find an object as a quarter and measure the gaps between a quarter and other purposes.

Scope of the project:

This project will use the following image processing methods:

- ◆ Grayscale conversion for preprocessing the image
- ◆ Gaussian filter as a smoothing technique
- ◆ Canny edge detector as an edge detection method
- ◆ Dilation and erosion as morphological operator
- ◆ External contour detection

Steps of the Project:

- ◆ Object detection:
 - We will order the four (x, y) coordinates of the bounding box for each object clockwise to perform the corner matching of the objects.
 - For the edge detection process, after converting to the grayscale image, we will use a gaussian filter to smooth the image by removing noise, which helps to get the intensity gradients of the image. A dilation followed by an erosion operator will help to close any gaps in the edge map.
 - We will apply an external contour detection method, as in this project we will only deal with the objects' outlines.
- ◆ Dimension calculation:
 - We will calculate the object's dimension by performing calibration using a reference object, which can be found based on the placement of the object (the usual location will top-left corner in an image). After that, we will apply a sorting process of object contour and use it to get "pixel per metric"², which will allow us to count number of pixel fit into inches. The "pixel per metric" is calculated by finding the Euclidean distance between the midpoint³ formula:
$$M = \left(\frac{x_1 + x_2}{2}, \frac{y_1 + y_2}{2} \right)$$
- ◆ Distance measurement: In the previous steps, we got sorted coordinates of the bounding box, midpoint and pixels per metric. We can now easily get by calculating the Euclidean distances⁴ between the reference object and other objects in inches dividing by "pixel per metric."

References:

1. Yasir Dawood Salman, Ku Ruhana Ku-Mahamud, & Eiji Kamioka. (2017). Distance measurement for self-driving cars using stereo camera in Zulikha, J. & N. H. Zakaria (Eds.), Proceedings of the 6th International Conference of Computing & Informatics (pp 235-242). Sintok: School of Computing
2. <https://www.pyimagesearch.com/2016/03/28/measuring-size-of-objects-in-an-image-with-opencv/>
3. <https://www.purplemath.com/modules/midpoint.htm>
4. Meshram, Sameer. (2016). Re: Anybody is having an idea about how can we measure the real distance between the objects in the Image, if we know the height of the camera position?. Retrieved from:
https://www.researchgate.net/post/Anybody_is_having_an_idea_about_how_can_we_measure_the_real_distance_between_the_objects_in_the_Image_if_we_know_the_height_of_the_camera_position/56fb8d8a404854eba107a363/citation/download.