Penerbit Akademia Baru

Journal of Advanced Research Design

Journal homepage:
https://akademiabaru.com/submit/index.php/ard
ISSN: 2289-7984

Development of Machine Vision System for Automatic Size Detection

Amli Omar Ismail[1], Uswah Khairuddin[1,*]

[1] Department of Mechanical Precision Engineering, Malaysia-Japan International Institute of Technology, Universiti Teknologi Malaysia

ARTICLE INFO

Article history:
Received 26 April 2021
Received in revised form 29 November 2021
Accepted 14 December 2021
Available online 27 December 2021

Keywords:
Logistic company; machine vision; bilateral filtration; canny edge detector

ABSTRACT

Everyday, logistic companies process millions of parcels for deliveries, transit movement, package sorting and storing. Automation of these processes will greatly reduce dependencies on human workers and increases efficiency of parcel handling. In order to automate package sorting and storing, a logistic company must also automate the measuring process of the package size *via* sensors or machine vision. This paper presents the development of a machine vision system for automatic box size detection using single camera vision and grid-less background. This system consists of a camera and a plain platform. The system combines canny edge detector algorithm for edge detection, Gaussian and Bilateral Filtration algorithm for pre-processing and statistical calculation for final dimension measurement. The system has 0.1 cm dimension resolution and achieved 94 % accuracy when determining width and height of boxes. The use of this single camera vision system may eliminate the use of physical detection sensors in box size measurement.

1. Introduction

E-commerce nowadays is becoming the world highlight. Online shopping trend shows rapid increase from the past few years. Due to this fact, logistics companies are under high demands due to increasing number of daily packaging and deliveries. The given situation is a big alarm to signal a massive change to drive industrial revolution and global businesses [1]. Logistics company is the one that will greatly affected due to this trait of changes. Increasing numbers of deliveries, transit movement, packaging handlings and numbers of workload are the things that should not be taken lightly.

Logistic process involves preparing, packaging, handling, storing and delivering goods or product either domestically or in international trade. Therefore, it is always a mistake to think logistic is a simple process. In contrast, it covers series of tasks starting from when retailers received goods from factories, until when it arrived at the recipient hands. The entire process itself is repetitive and tedious due to sheer number of packaging required to be handled and various size of packaging that are never consistent. Based on report from amazon, amazon handled about 15 million of boxes every day since the coronavirus pandemic outbreak [2]. This affected their total productivity and cost in overall view.

* *Corresponding author.*
E-mail address: uswah.kl@utm.my

Therefore, automation technologies are one of the best approaches to handle such demands. While restricted by financial constraints, the use of automation within production line will reduce the operational costs and profiting in the long run [3,4]. These technologies can replace workers to handle packaging, item sorting and warehousing that requires repetitive processes and high productivity. However, existing technologies today is still far from perfect. The complex nature of any automatic logistics systems makes the cost of such system high, which means only several logistic companies managed to acquire such technologies.

The common automation technologies that exist and used by logistic company today include robotic arms, automatic sorting and smart packaging process [4]. However, some of these machines still required some work to maximize its productivity and effectiveness.

Robotic arms can work at faster pace with high accuracy if only "its eye" provides a correct measurement of an object. Smart packaging can be fully automated only if all acquired boxes sizes and volumes are accurate and that manual supervision is not needed. These shows that if such accuracy can be achieved using some technology to automatically detect object sizes, manual works can be reduced while increasing productivity for whole process.

Size detection technologies have a long history in automation system as part of machine vision system that has been used in various fields such as agriculture [5-10], marine [11], manufacturing company [12] and smart system [13]. The purpose of the machine vision system used in those fields might differ yet the objective to automatically detect the object size from image taken is a similar trait. The objectives were also the same as it helps reduce cost and increases productivity.

In developed nation, size detection was implemented in agriculture field. This implementation was done after several research carried out to apply automation in detecting and monitoring crop sizes [7,9,10]. With implementation of object detection algorithm and colour image processing, number of harvest and yield quality can be determined by image captured. The quality of image also optimized using image enhancement to ensure that the crop within the image is the same as the real crop [8]. In other variation of machine vision system, crops size and mass also can be pre-determined by the aided of artificial intelligent.

In marine and manufacturing factory, size detection technology also been used as part of automation system. Due to colour image processing ability, with combination of image enhancement and pattern recognition, marine species, mass and volume can be determined by images [11]. In factory outlet, size detection system come together with other systems to be used in assembling parts and components [12]. This is not impossible to be done as data gained from image is relayed to another system.

Nowadays by using smartphones, size detection can also be executed due to development of a certain software that uses phone height and camera lens to calculate the size of object within images [13,14]. However, this is proven lack of accuracy and still required a lot of works to improvise. This is where this project came important as it target to have more accurate results, fast processing and easy to execute compared to existing system. Using simple set-up with single camera and enhanced algorithm, the system designed expected to automatically compute boxes volume with fastest rate and data gained can be relayed to other system.

2. Literature Review

2.1 Machine Vision System (MVS)

A MVS is an image-based technology that use images to automatically extract information by image processing. In the recent studies, MVS is used for automatically detect size of boxes [15]. As machine vision is a system, there are some techniques involved such as applying concept time-of-flight, laser triangulation and line structured light [16-18]. As various method and techniques exist, the MVS also may differ from one another based on system algorithm, equipment and material used, as well as type of camera. As to improve the vision-based result, several factors need to be considered such as range of detection, surrounding environment condition, performance of system and accuracy.

A study done apply the principle of time-of-flight from infrared light used in conjunction of image processing to obtain information of targeted object such as volume [16]. Based on the principle of time-of-flight, obtaining 3D information is done by calculating the time taken for infrared signal to travel towards the targeted object and reflected to the sensor. The latest generation of Kinect cameras, Azure Kinect DK used time-of-flight in its 3D depth sensor. Previously, Kinect cameras was used in several studies by applying time of flight light-based camera and extracted the 3D coordinate from apples [5]. However, even with some successful result, the use of Kinect cameras has several limitations. Firstly, it due to the infrared range of detection as accuracy in infrared sensor can only properly functioning at a very limited distance to obtain reading [15]. Then, surrounding condition such as environmental influences affect the performance of infrared reading.

Fig. 1. Azure Kinect DK

Another technique was laser triangulation, also used to detect objects for its size detection. As the name sound, this technique used laser illumination together with camera to obtain the 3D information of a targeted object. From recent study the laser triangulation proven to be a non-contact method with high processing speed [17]. However, the application of laser triangulation only used to detect the size of an object in regular shapes. If the irregular shape were used, the result might be different. This study combines the laser triangular technique to detect points on surface of boxes and combining deep learning to obtain the 3D information of boxes [17]. This is to enhance the detection system and provide better accuracy for volume obtained.

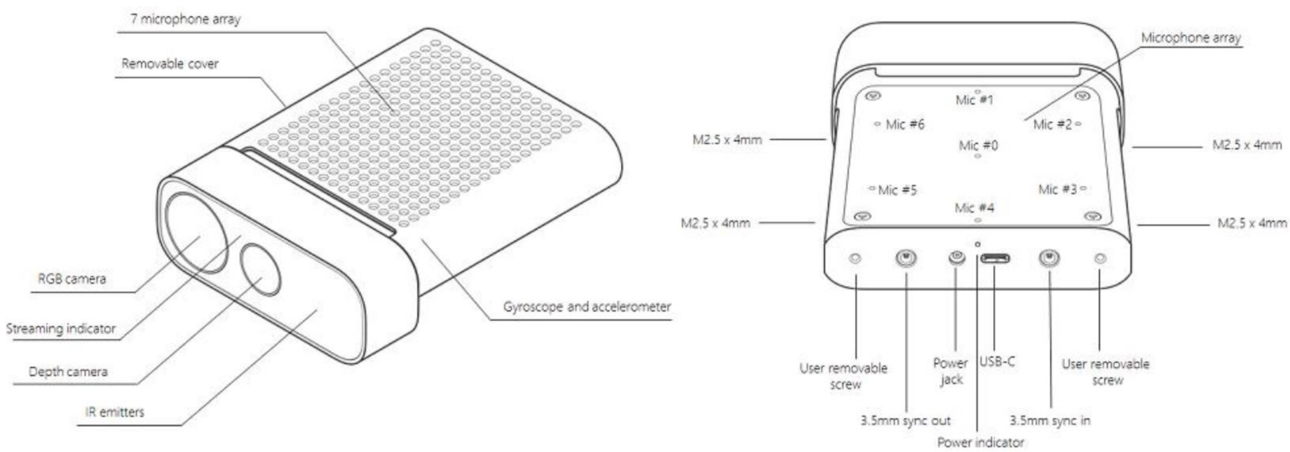


Fig. 2. Azure Kinect DK layout

A study shows the use of line structured light to measure the 3D depth of targeted object and compute the volume [19]. The process of obtaining 3D information is based on the pattern deformation projected on to an object [19]. The common procedure done was utilizing Kinect camera that comprises RGB camera and 3D depth sensors. Since stereo vision is relatively more expensive, this technique considered preferable as it responded with accurate 3D depth in a short time. On the other hand, due to limited range of detection in for the depth sensor to work, obtaining information at too close or far distance will be difficult to execute. Moreover, due to strict lighting condition, requirement for application at warehouse will be difficult to fulfil. While, comparing the time-of-flight principle to structured light approach, time-of-flight will be more reliable for the use in warehouse.

Table 1
Summary of vision-based system

Author	Vision-based system	Object	Details
Gongal *et al.*, [5]	RGB-HSV	Apple	3D Coordinate
Nyalala *et al.*, [7]	RGB-D	Tomato	3D Coordinate
Shahdib *et al.*, [16]	Time-Of-Flight	Box	3D Point
Tao *et al.*, [18]	Structured Line	Box	3D Depth
Tao *et al.*, [17]	Lina Triangulation	Box	3D Surface

2.3 Object Detection

Recent studies detection for the size of object was localizing the targeted object area for data extraction [19]. This can be considered as object-oriented data extraction that focused on the use of object within the image for obtaining 3D information. The method of splitting an image into multiple segments are called image segmentation process. The system will work with image segmentation algorithm to detect the targeted object and isolate it from the background. This is for having a new representation of targeted object so that it is easier to analyze. There are several different type of image as different image segmentation methods available. Based on existing system, the type of images mainly divided into two groups. First group utilized colour image to perform segmentation technique based on colour profile. Second group utilized image with depth information to perform the segmentation technique based on 3D depth information. The following sub-section will discuss various image segmentation techniques used for both colour images and depth images.

2.3.1 Color image

Digital image includes made up from tiny pixel in form of array. Each of pixel in digital image contained colour information, this image called colour image. The information within array acts as coordinate to the image. This will allow image segmentation technique to be applied through thresholding or edge detection by using colour information from the pixels for object segmentation. It is common to realize that most object images captured as in colour images in the RGB mode. RGB is a colour model space which is an additive colour model in which made up by three base that is red, green and blue light combined to produce a wide range of colour with different proportions. Other than most common, it considered as easier way to present the colour profile without needs of extra transformation. Nevertheless, to determine range of colour through RGB colour model required extra works.

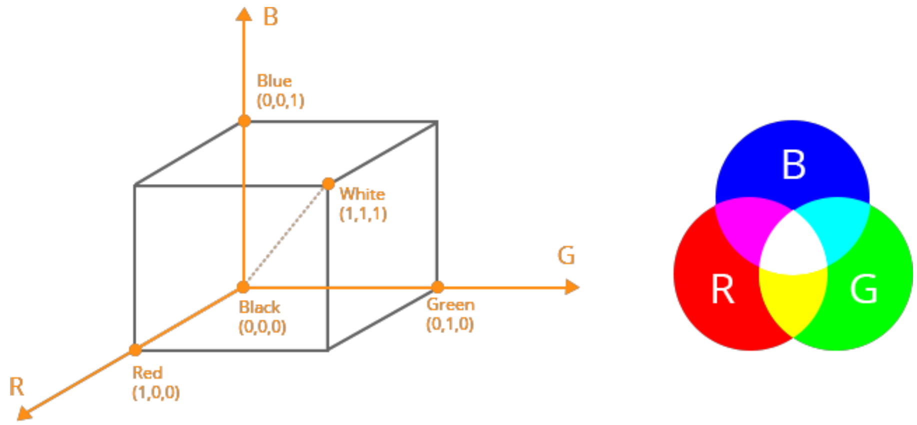


Fig. 3. Color model based on RGB color space

For this reason, based on recent studies, the HSV colour space is used to transform RGB colour model into the HSV colour model due to HSV sensitivity towards the changes external condition [12]. HSV colour model also known as HSB uses hue, saturation and value of brightness to define colour profile that is closer and well-matched with human visual perception. Hue represents the model colour portion, saturation represents amount of grey within specific colour and value represents the value of colour brightness. In some research, HSI colour model (Hue, saturation and intensity) used instead of the HSV colour model [5]. In general, HSV and HSI used same approach of representing colour with the only different in equations used to rearrange RGB colour.

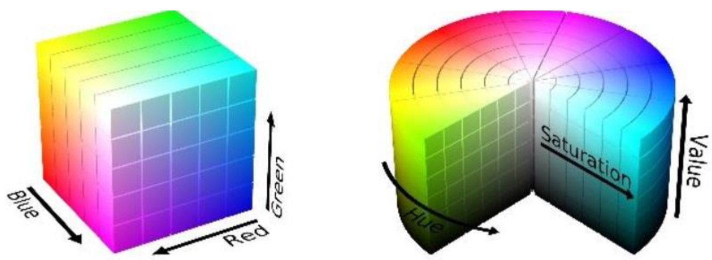


Fig. 4. Comparison RGB color space (left) and HSV color space (right)

Before image acquired is processed, image enhancement is done to improve the quality of image and the closest to real object [8]. There are various image pre-processing techniques are used for image enhancement such as reducing noise and increasing contrast. Most common technique done is converting the image into grayscale before the image enhancement [5,7]. This is because luminance is far more significant for determining the visual characteristics from image. Through grayscale process, each pixel only contains information on light intensity and represented as a range of gray shades with no apparent. These prove to be a fruitful result due to keep the coding and processing time at minimal while simplified image complex image in 3D pixel into 1D pixel.

In the recent studies image pre-processing algorithm were utilized by amplifying the image colour difference between the targeted object and background areas [5,12]. Some of those technique used are Histogram Equalizer and Gamma Transform [5,12]. Applying contrast within the image possible by Histogram Equalizer [5]. Gamma Transform technique act to control the overexposes or over-darks area [12]. While to reduce noise at a minimum and to improve the quality of images, several filtering algorithms can be applied to do the job. In a study Weiner filtering used to eliminate noise through minimization of the mean-square error [5]. Thresholding or Binarization is an act to transform the images from grayscale to binary image. In binary image, pixel value consists only either in zero as black or one as white. Most common technique used for thresholding is Otsu Thresholding to spread the pixel of image into foreground and background [5,11].

Due to thresholding, image can easily be segmented to have an object silhouette by removing the background [11,20]. Then, with help of edge detection algorithm, targeted object boundary can be determined. Edge detection technique is a feature used to identify the shape and size features of targeted object by detecting the change in brightness from the image. Hough Transform is a technique used for feature extraction from image to separate the shape features through circles or line detected [15]. Circular Hough Transform, a variation of Hough Transform is used for detecting a circular object in the image [5]. By this phase, the targeted object is fully separated from the background and can be forwarded to size estimation process. However, depending on the studies being done, there are systems favored to analyze the background instead of targeted object [15].

2.3.2 Depth image

Depth images, also known as depth maps are representation of 3D images that contain the depth information on the distance of object from a relative point. In general, depth images can be taken using camera that is coupled with depth sensors such as Kinect generation camera [7,19,21]. The colour intensity represent the distance of the object surface facing the camera from a given point of the camera installed. Supposedly, depth image comes in grayscale form from which the source is generated but the colour code were added to allow effective visualization of depth and the distance clearer. To perform image processing to enable localization of targeted object from the background profile, segmentation techniques should be used based on the depth profile from the pixel. Common techniques used are background subtraction and morphological operations.

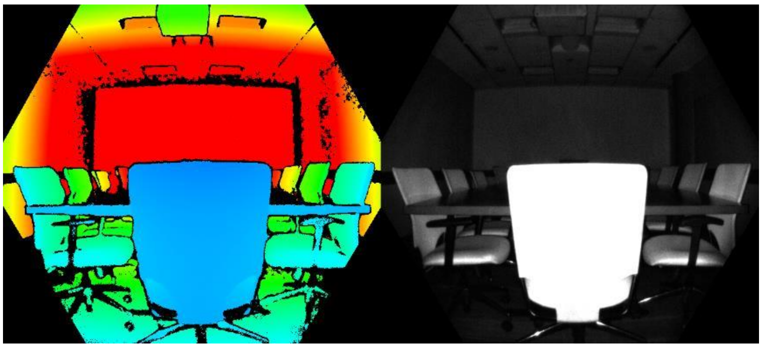


Fig. 5. Depth Image (left) and IR Image (right)

Recently, a study was done to develop box size estimation using a single perspective image [22]. As the system targeted to use only a single perspective image which mean the camera is at fixed position, the targeted object position and its direction plays a significant role in order to guarantee the visibility of three edges that are to be computed. Based on the study, a method called RANSAC was used after obtaining the 3D point (edges) to identify the inliers of the planes [22]. Through RANSAC, parameter estimation was done to determine the outlier. Then, foreground and background were separated using background subtraction. This was followed by the clique method to determine the adjacent point to find the plane of the object and the background. With this, the object can be segmented from the background plane. This is one of the methods that is applicable to measure size of boxes by utilizing depth image which is less complex. On the other hand, accuracy is a problem which position and the direction must be carefully taken into consideration.

Then again, in recent study distance thresholding was performed right away after image acquisition [19]. Distance thresholding technique is used to acquire the region of interest (ROI) from the targeted object by doing thresholding based on distance of object in depth image. This meant to set an area for the targeted object size to be automatically detected based on the ROI. Although this technique is a good example, but the complexity of the system is difficult to be adjusted.

Some studies chose to remove noise from the resultant depth images for smoother result [19]. A method called Gaussian kernel filter was introduced to be used for smoothing the image by removing details by blurring and reduce noise within depth image [7]. Again, morphological operation can be used to eliminate holes from image to gain a clearer image result [7]. Then, the binarization is done using Otsu's Thresholding to produce the binary image. This image can be processed for size estimation process.

2.4 Size Estimation

As a continuation from previous step, object size can be determined using either 2D features or 3D features of the images. For 2D features, object size can be computed using major axis, radial distance and area of the object surface. Meanwhile, for 3D features, utilized object surface area and volume based on object.

The recent study used Grids counting method to compute the volume of object with box shape [15]. The size detection for this system works by combining the mathematical equation and grids size to estimate the box volume. The platform with constant size grids was prepared with black background and white grids at the centre of the platform. The grids must cover large area that is enough to cover the object size. The size of grids was computed into the system as a preparation before measurement process [15]. The object silhouettes of the boxes then was measured by estimating the numbers of grids. Based on the number of grids and knowing the real size of the grids measurement, the volume of boxes can be calculated. This method allows the size of boxes to be fully measured by calculating the grids. To ensure maximum accuracy, all points must be within the grids area.

In another research, the size of object was automatically detected by the number of pixels and its size [5]. Opposite to previous method that focused on background preparation, this technique focused on the pixel profile of the camera such as the pixel size, coordinate of pixel and even the distance between camera and image. The relationship between the subject of pixel profile is analyzed and translated into real object size. This method is almost related to previous studies that calculates the size by using mathematical equations to represent the size of reference subject [15]. Then, by correlating the pixel size, pixel coordinates and the distance of image from camera, a regression model is produced. Lastly, using the number of pixels on major axis of the targeted object and multiplies with pixel size, through mathematical equation, the size can be detected [5].

Object size can also be detected by co-registration of images [7]. Image registration is a technique used in image processing to allow image datasets to be converted into coordinates. Thereby, a 3D coordinate for the pixels of the image can be produced and the targeted object can be translated into a coordinate system. The common method used is representation by mathematical equations to find the Euclidean distance between two specific pixels from the image [7]. The straight-line between two points in an Euclidean space is called Euclidean distance. In several research, pixels located at two different points such as vertices of boxes are chosen. Hence, the length of the edge can be obtained through Euclidean distance. Lastly, as the Euclidean distance on three different axes are obtained, volume of the boxes can be computed automatically.

Table 2
Summary of vision-based system

Author	Type of Image	Object	Method	Accuracy (%)
Gongal *et al.*, [5]	Colour image	Apple	Pixel Size	84.8
Tay and Khairuddin [15]	Colour image	Box	Grid Counting	94.3
Nyalala *et al.*, [7]	Depth image	Tomato	Numerical Integration	96.3
Tao *et al.*, [18]	Depth image	Box	Image Registration	99.0
Tao *et al.*, [17]	Depth image	Box	Image Registration	99.3
Quintino *et al.*, [21]	Depth image	Box	Euclidean	84.4

The background area of an image leaves an empty spot that is usually missed out when extracting and acquiring information from the image. If the dataset is attached to a platform that served as a background, instead of limiting the data extraction on the object, the background can also be used to provide data for extraction.

It is undeniable for the accuracy of the final result of existing size detection technique is heavily depended on the segmentation process. This process is easily affected by the change in the environmental condition such as lighting factor. For this reason, a proper condition must be fixed along the process for the experiment to be carried out successfully.

3. Methodology

3.1 List of Equipment and Apparatuses

3.1.1 Equipment

The equipment used consist of hardware and material for setting up the platform and camera installation. As provided in Table 3, there were several equipment used for setting up the system. These equipment were selected based on literature search done. Also provided are the cost needed for each related item**.**

Table 3
List of equipment

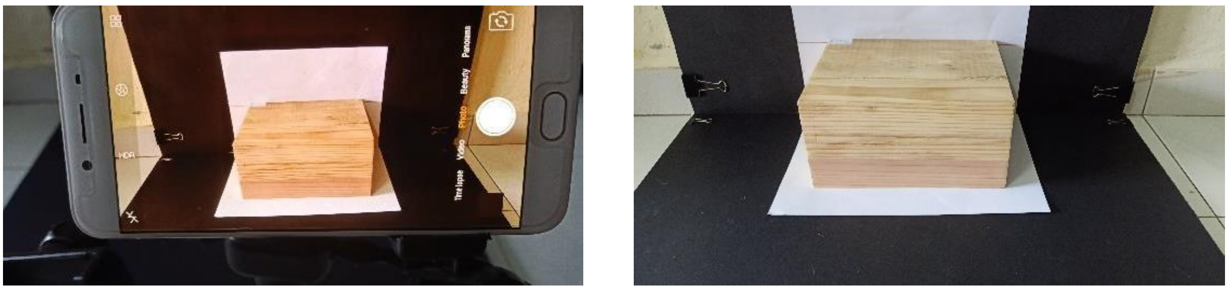
#	Item	Quantity	Model info	Price
1	Camera	1		
	a) Phone Camera	1	Oppo R9s 16 MP, f/1.7,1/2.8", 1.12µm, dual PDAF 4K@30fps, 1080p@30fps	N/A
2	Platform	2	1 with a reference line and the other is plain. Dark colored is used	N/A
3	Sample Boxes	30	Vary with sizes. (According to database attachment) Max: 30cm x 30 cm x 30 cm	N/A
4	Raspberry PI Set	2		
	a) Raspberry Pi 4	1	One set with charger, fan,casing. Raspberry PI 4 model B	RM 405.00
	b) Raspberry Pi Camera module	1	8MP NoiR Camera Module	RM 94.90

3.1.2 Apparatuses

To perform manual calculation as to obtain the actual volume, 30 cm L-square were used to measure length, height and width of the boxes.

Fig. 6. 30 cm and 90-degree L-Square ruler set-up of equipment

The platform was supposed to be made from a black background with A4 paper at centre of the platform. The reference line is at 10.5 cm from surface of platform. Then smartphone with holder were placed at 50 cm from platform with 60-degree elevation from surface facing the platform at height of 30 cm. Oppo R9s smartphone (Android) were used for image quality. The equipment set-up can be seen as shown in following Figure 7.



(a) **(b)**

Fig. 7. View of (a) Platform after setup from phone and (b) Platform view

However, due to errors found during training phase, the background changed to yellow with white B4 paper placed at centre of the platform. Then the reference line was drawn at folded part between horizontal and vertical B4 paper. The limit line was also drawn at height of 25 cm from surface of platform. The camera was installed at distance 71 cm from platform. Oppo R9s smartphone camera was applied for data acquisition. The angle of camera was adjusted until the entire platform area could be capture in the image.

3.2 Database Set

The values obtained refers to the 30 sample of boxes that was manually measured using L-square ruler. The boxes share similar traits as the colour of boxes were the same for all 30 boxes. This is to ensure a controlled variable. All measurements were taken with one decimal place in cm unit. The following Table 4 lists all the measurement taken. However due to circumstances, not all sample boxes were available, only five boxes from these 30 boxes were used but the orientation for placement varied. The list of dimension measurements was included and shown in Table 4.

Table 4
List of size of boxes

Sample	Length (cm)	Width (cm)	Height (cm)	Volume (cm^3)
1	2.6	3.6	14.8	138.53
2	3.8	4.2	12.3	196.31
3	5.8	5.8	5.8	195.11
4	1.8	6.4	9	103.68
5	4	6	13	312
6	6.6	6.6	11.6	505.3
7	3.6	6.8	17.6	430.85
8	6.8	7	14	666.4
9	7	7.5	18.3	960.75
10	7.1	7.4	20	1050.8
11	2.3	7.7	17.2	304.61
12	7.3	7.5	26.1	1428.98
13	1.9	8.5	22.8	368.22
14	9.1	9.6	8.1	707.62
15	4.2	9.1	12.6	481.57
16	5.3	8	12.4	525.76
17	5.5	8.1	17.1	761.81
18	6.3	9.2	22.5	1304.3
19	4.9	9	17.6	776.16
20	6.2	9.5	17.9	1054.31
21	6.5	10.3	18.3	1225.19
22	1.8	10.2	16.8	308.45
23	2.7	10.5	13.3	377.06
24	3.3	11	21.5	780.45
25	5.3	11.2	15.5	902.72
26	6.1	13.8	17.2	1447.9
27	4.5	15.5	20.8	1450.8
28	5.2	16	22.3	1855.36
29	11.1	17.3	21.6	4147.85
30	5.4	18	21.2	2060.64

3.3 System Algorithm

This section discussed the system algorithm applied in this project that focussed on the background image instead of object for obtaining the information. The system began at image acquisition and end at performance analysis as shown in Figure 8.

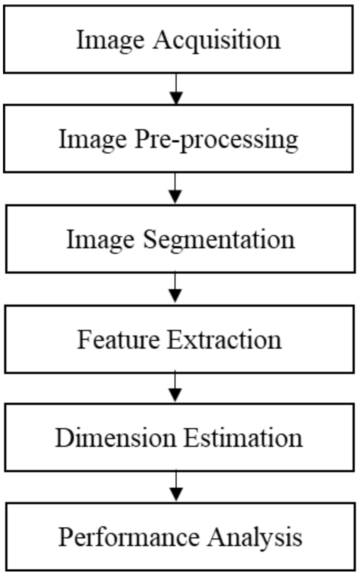


Fig. 8. Process flow design

The designed system started with image acquisition of colour image through smart phone. Image pre-processing was done to enhance image taken and to acquire better quality of image for segmentation of image from background. The image datasets were acquired with a white platform with a horizontal single reference line in background. As the segmentation was done and background image extracted, the data on height, length and width was calculated to compute the volume of boxes. Lastly, the results obtained from the system were evaluated together with manual measurement datasets to calculate the performance analysis for the system.

3.3.1 Size detection algorithm

Figure 9 shows the detailed framework of system algorithm was divided into three phases. The first phase can be considered as preparation phase where the background for platform was prepared and installed together with the platform. Then the camera was set at a fixed position. Next, the second phase was the experimental or program execution as this is where the image processing, segmentation and data extraction been carried out to find the volume size of boxes. Lastly, the validation phase of the system was to check and validate the result acquired from the system in comparison to manual measurement for performance analysis.

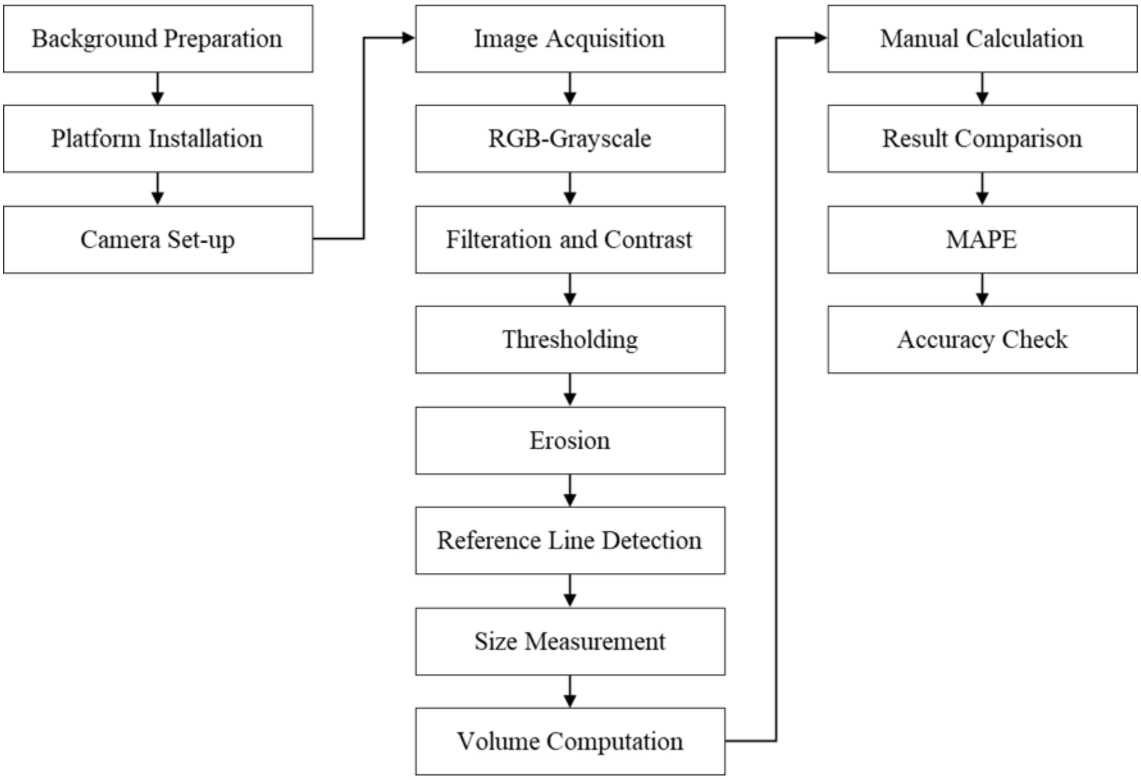


Fig. 9. Automatic size detection system workflow

3.3.2 Phase one algorithm

Figure 10 shows the workflow for phase one on system algorithm. For this system, a platform was designed with a single line acting as reference line to determine the width and height. The distances line drawn from one end to another of the platform was 35 cm. The line was drawn on the folded part of white B4 paper and using black ink to increase contrast during image acquisition. The white paper was then placed on top of yellow background (initially black background) and then attached to the wall and floor.

Colour image acquisition of boxes was done using smartphone Oppo R9s camera (16 MP Sony IMX398 Sensor, Sony, Nihonbashi, Tokyo, Japan). The camera was placed at 71 cm from the platform on a stand with height of 26 cm from floor level. The tilt angle was adjusted until all platform area was captured on the image. The surrounding condition was controlled such as lighting, humidity and temperature. White LED light was used to provide extra lighting for consistency of brightness from LED flash with temperature colour of 7300K.

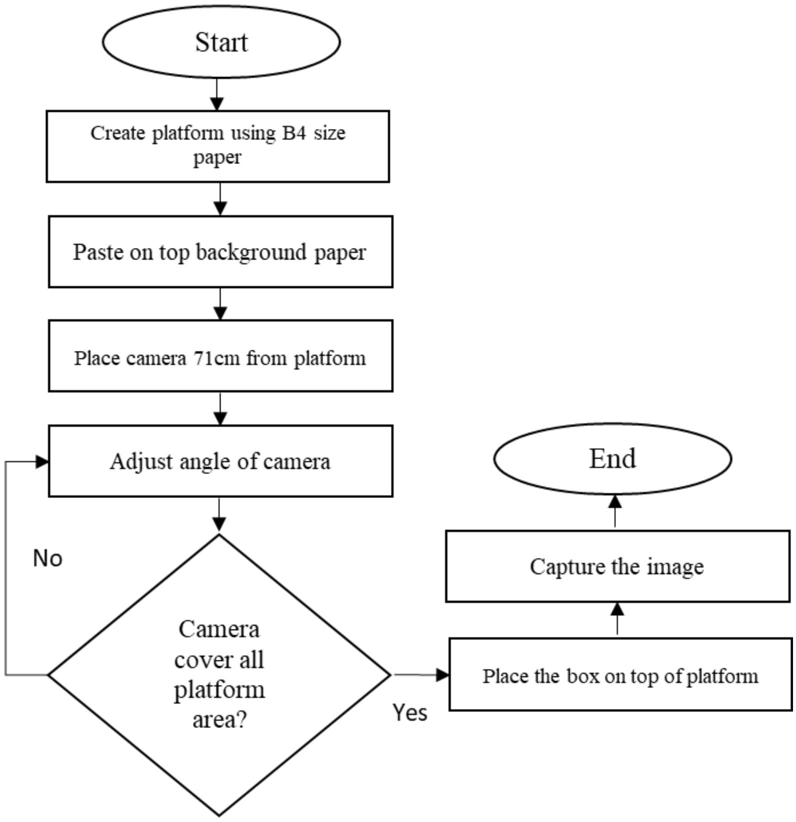


Fig. 10. Phase one system workflow

Figure 11 depicts the view from the photographer's perspective for the platform set-up. However, later a yellow-colored background was added to reduce error from surrounding background. Five boxes with different sizes and maximum dimension of 30 cm for each side were used as samples for the creation of dataset. Each dimension was manually measured by using L-square ruler to acquire the actual size of boxes. For the boxes, the placements and orientations were varied and recorded. For the images, it were transferred to an Intel Core i5-9300H CPU, 2.40GHz (Intel, Santa Clara, CA, USA), Microsoft Windows 10 Home laptop.

Fig. 11. Image acquisition system set-up

3.3.3 Phase two algorithm

The image processing techniques were used to determine the pixel_per_metric value of the reference line. Several techniques used were smoothening and noise filtering, edge detection, object segmentation and feature extraction. Figure 12 shows the detailed workflow on image processing during phase two.

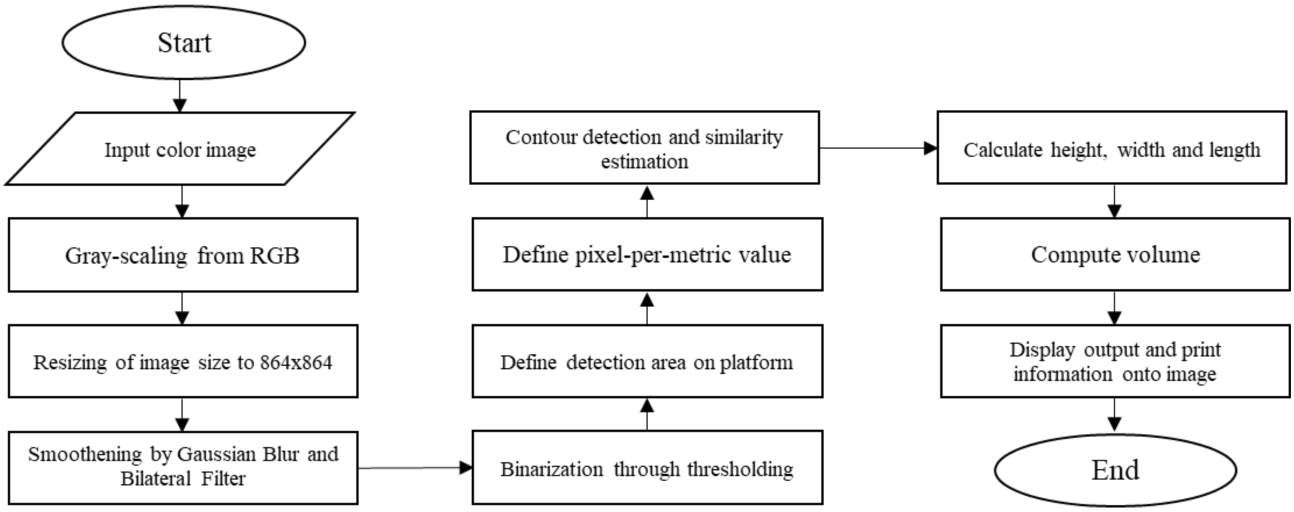
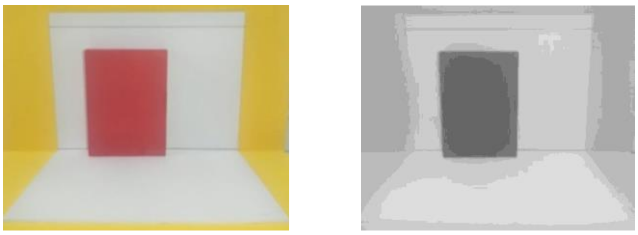


Fig. 12. Phase two system workflow

The system began with reading the input of colour image. Then the image was converted from three channel of RGB colour model into grayscale. The colour conversion was done to ease the image processing and reduce processing time. The following formula was used to change the information from RGB colour model into grayscale. The input image was originally at sizes of 3456 pixel x 3456 pixel that was then resized to 864 pixel x 864 pixel, one-fourth of original size of the image acquired.

$$y = 0.299R + 0.587G + 0.114B \quad (1)$$

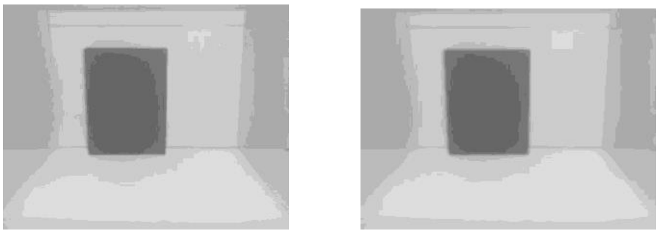
Where y is light intensity of image. While R, G and B is intensity of red, green and blue channel respectively from the original image. Figure 13 shows the conversion of image from colour image to grayscale image.



(a) (b)

Fig. 13. (a) RGB colour image (b) Grayscale image

Next, another image pre-processing was done which was filtering. Previously, there was another process that was carried out which was adaptive contrast (CLAHE). However, using CLAHE caused error during binarization of image. Therefore, it was replaced with another filtering process which was bilateral filter. Bilateral filter was done after Gaussian blur process to remove noise and enhance smoothness of image. The results of these processes are shown in Figure 14.



(a) (b)

Fig. 14. Image pre-processing (a) Grayscale image (b) Blurred and filtered image

The first image processing for segmentation in this algorithm was binarization, also known as thresholding. Image segmentation was applied to identify ROI through pixel value. Thresholding was done by setting the threshold value, for this research threshold value was at 135. Pixel with value higher than threshold value turned into 1, while lower than threshold value became 0. The formula representing the binarization process can be seen as following:

$$dst(x,y) = \begin{cases} 255, if\ src(x,y) > thresholding\ value \\ 0, otherwise \end{cases} \quad (2)$$

After binarization was successfully done, the box was segmented out from the platform. The result is as shown in Figure 15.

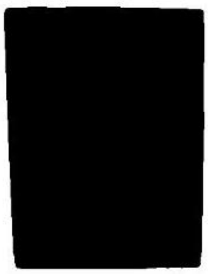


Fig. 15. Binarization process

Originally, a morphology technique was supposed to be used to enhance the output from the segmentation process of binarization and to produce the shape of segmented area. However, due to complexity of controlling the kernel and success rate of this method, this method was replaced with Canny edge detection to reduce the size of memory needed to execute the algorithm for faster processing. Through the use of Canny edge detection algorithm, edges of the box were perfectly detected and shown as in Figure 16. Lastly, from the results obtained, contour detection and triangle similarity method were used to compute the dimension information and volume of boxes within the images.

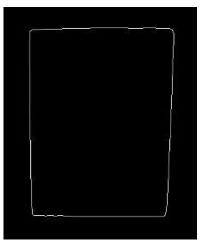


Fig. 16. Edge detection

To make use of triangle similarity and contour detection for feature extraction, it required ratio of pixel to the metric unit. This can be used as reference unit to determine the real metric value on length, width and height within the images in dataset. The method of calculation the value of pixel per metric, computing the width and height can be expressed as shown following formulae:

$$Pixels\_per\_metric = \frac{535pixel}{35cm} = 15.3\text{pixel/cm} \tag{3}$$

$$width = \frac{1}{pixels\_per\_metric}(w) \tag{4}$$

$$height = \frac{1}{pixels\_per\_metric}(h) \tag{5}$$

The width and height represent metric unit while the w and h represent the pixel unit of object. Through the given formula the width and height can be calculated for all 40 images. For the dimension, length and triangle similarity method was used by having image focal distance using the reference line. This can be expressed as following:

$$Focal\ distance = \frac{h}{height}(71cm) \tag{6}$$

As the focal distance was obtained, translated height was used to determine the distance of object from the camera. The distance between the platform and camera was fixed at 71 cm. Thus, the length can easily be determined by subtraction of total distance and distance of object to camera. At this point, the volume can be calculated. The output will be printed on the images as shown in Figure 17.

$$height' = \frac{1}{pixels\_per\_metric}(h - 7) \tag{7}$$

$$distance\ from\ camera = \frac{height(Focal\ distance)}{h'} \tag{8}$$

$$length = 71cm - distance\ from\ camera \tag{9}$$

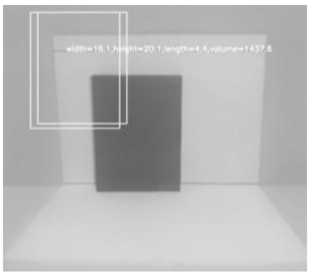


Fig. 17. Output image with dimension information

3.7 Phase Three

This final phase was designed to evaluate the performance and accuracy of the size detection algorithm upon the results collected from all 40 images within dataset. The detected sizes were compared to the manually measured size of boxes. The differences were then calculated and presented as percentage error of the system.

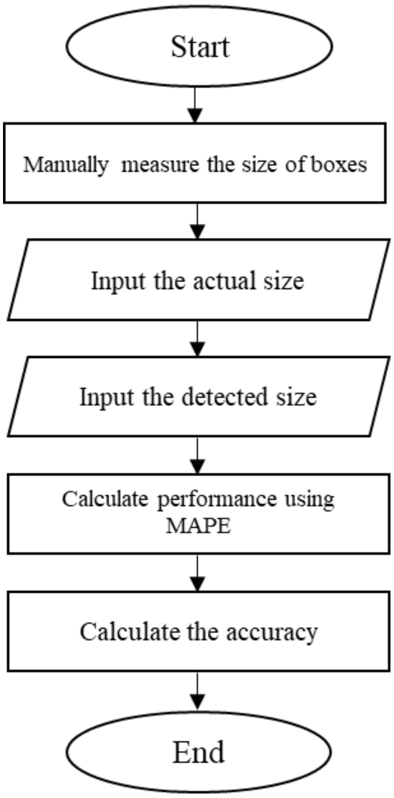


Fig. 18. Phase three system workflow

Regarding the performance analysis of this system, the Mean Absolute Percentage Error (MAPE) was calculated by the end of the project. This is to obtain the percentage for overall system performance based on the value gained from the system in comparison to the actual value of the boxes. The formula that was used is as the following:

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{A_t - F_t}{A_t}\right| \quad (10)$$

Where, n is number of boxes, A_t is actual volume of boxes and F_t is calculated volume of boxes obtained from the system.

4. Result and Discussion

The results from manual measurement and through size detection algorithm were recorded and tabulated into the following Table 4 as actual and detected size respectively. From the results, it can be observed that length had largest error at 136 % while detection on width and height both shows at acceptable level, 7 % and 10 % respectively. The box volume obtained through manual measurement and algorithm also tabulated into Table 4. It is shown that most of volume computed produce error results.

Table 4
Actual and detected size of boxes and percentage error

Sample	Length			Width			Height		
	Actual size (cm)	Detected size (cm)	Percentage error	Actual size (cm)	Detected size (cm)	Percentage error	Actual size (cm)	Detected size (cm)	Percentage error
1	6.90	25.60	2.71	17.70	16.70	0.06	3.50	4.10	0.17
2	13.70	47.50	2.47	6.90	6.10	0.12	3.50	3.90	0.11
3	3.50	4.50	0.29	6.90	6.60	0.04	17.70	17.00	0.04
4	6.90	8.40	0.22	3.50	2.40	0.31	17.70	17.10	0.03
5	17.70	37.40	1.11	3.50	3.50	0.00	6.90	6.80	0.01
6	3.50	8.20	1.34	17.70	16.70	0.06	6.90	7.00	0.01
7	6.90	24.50	2.55	17.70	17.90	0.01	3.50	4.10	0.17
8	3.50	7.70	1.20	17.70	17.10	0.03	6.90	6.90	0.00
9	3.50	4.00	0.14	6.90	7.00	0.01	17.70	16.90	0.05
10	6.90	7.80	0.13	3.50	4.20	0.20	17.70	16.90	0.05
11	17.70	36.60	1.07	3.50	4.20	0.20	6.90	6.70	0.03
12	17.70	45.90	1.59	6.90	7.30	0.06	3.50	4.00	0.14
13	17.70	45.60	1.58	6.90	7.40	0.07	3.50	4.10	0.17
14	14.10	31.80	1.26	6.80	6.80	0.00	7.00	7.00	0.00
15	7.00	18.70	1.67	14.10	13.70	0.03	6.80	6.70	0.01
16	6.80	9.80	0.44	14.10	7.20	0.49	7.00	13.50	0.93
17	7.00	18.40	1.63	14.10	14.60	0.04	6.80	6.60	0.03
18	14.10	30.80	1.18	7.00	7.80	0.11	6.80	6.90	0.01
19	14.10	30.80	1.18	7.00	7.90	0.13	6.80	6.90	0.01
20	6.80	9.10	0.34	7.00	7.30	0.04	14.10	13.30	0.06
21	13.40	44.10	2.29	10.50	10.40	0.01	2.70	3.30	0.22
22	13.40	43.70	2.26	10.50	10.30	0.02	2.70	3.40	0.26
23	10.50	39.50	2.76	13.40	12.70	0.05	2.70	3.30	0.22
24	2.70	4.00	0.48	13.40	13.00	0.03	10.50	10.30	0.02
25	2.70	4.50	0.67	13.40	13.10	0.02	10.50	10.20	0.03
26	13.40	43.60	2.25	10.50	10.80	0.03	2.70	3.30	0.22
27	10.50	41.00	2.90	13.40	14.10	0.05	2.70	3.10	0.15
28	2.70	4.50	0.67	13.40	13.00	0.03	10.50	10.10	0.04
29	12.70	36.90	1.91	9.20	9.10	0.01	4.20	4.60	0.10
30	9.20	28.90	2.14	12.70	12.20	0.04	4.20	4.60	0.10
31	4.20	9.20	1.19	12.70	12.60	0.01	9.20	8.80	0.04
32	12.70	36.60	1.88	9.20	10.00	0.09	4.20	4.50	0.07
33	9.20	28.50	2.10	12.70	13.30	0.05	4.20	4.60	0.10
34	4.20	8.40	1.00	12.70	12.50	0.02	9.20	8.80	0.04
35	15.50	41.20	1.66	20.80	20.40	0.02	4.40	4.70	0.07
36	20.80	47.70	1.29	15.50	14.80	0.05	4.40	4.70	0.07
37	4.40	4.20	0.05	15.50	15.10	0.03	20.80	20.00	0.04
38	15.50	40.60	1.62	20.80	23.80	0.14	4.40	4.70	0.07
39	20.80	47.60	1.29	15.50	17.60	0.14	4.40	4.70	0.07
40	4.40	4.40	0.00	15.50	16.10	0.04	20.80	20.10	0.03
MAPE	1.36			0.07			0.10		

Table 5
Actual and computed volume for the boxes and percentage error

Sample	Volume		
	Actual Size (cm)	Detected Size (cm)	Percentage Error
1	427.50	1735.30	3.06
2	427.50	1144.00	1.68
3	427.50	501.50	0.17
4	427.50	345.80	0.19
5	427.50	897.40	1.10
6	427.50	955.80	1.24
7	427.50	1778.80	3.16
8	427.50	907.20	1.12
9	427.50	472.20	0.10
10	427.50	559.10	0.31
11	427.50	1030.00	1.41
12	427.50	1328.80	2.11
13	427.50	1366.00	2.20
14	671.20	1510.40	1.25
15	671.20	1712.30	1.55
16	671.20	952.60	0.42
17	671.20	1775.10	1.64
18	671.20	1656.80	1.47
19	671.20	1670.60	1.49
20	671.20	889.20	0.32
21	379.90	1526.60	3.02
22	379.90	1533.50	3.04
23	379.90	1646.30	3.33
24	379.90	536.40	0.41
25	379.90	595.20	0.57
26	379.90	1567.20	3.13
27	379.90	1807.10	3.76
28	379.90	592.00	0.56
29	490.70	1531.90	2.12
30	490.70	1640.60	2.34
31	490.70	1021.40	1.08
32	490.70	1651.00	2.36
33	490.70	1765.40	2.60
34	490.70	916.40	0.87
35	1418.60	3956.20	1.79
36	1418.60	3329.80	1.35
37	1418.60	1281.00	0.10
38	1418.60	4549.70	2.21
39	1418.60	3936.10	1.77
40	1418.60	1437.60	0.01
MAPE	1.56		

Based on the results, it can be said that the errors originated from the detected length using the algorithm. From 40 images, only two images showed error lest than 10 %. It was found that the major source of error was inaccurate mathematical formula representation for the system in computing the length of boxes. The use of horizontal instead of vertical reference line was the main cause of error for this system. This was due to the pixel per metric ration done only applicable for the horizontal components of the image which was height and width. However, when computing the length, the ratio value became irrelevant as the nearer the object to the camera, the ratio of pixel per metric also increases. However due to the static reference line changes in length, it caused inconsistent relative reference. Therefore, the nearer the object to the horizontal platform, the more accurate the result will be computed. This was the case with the two images that showed less than 10 % error. This can be seen in Figure 19.

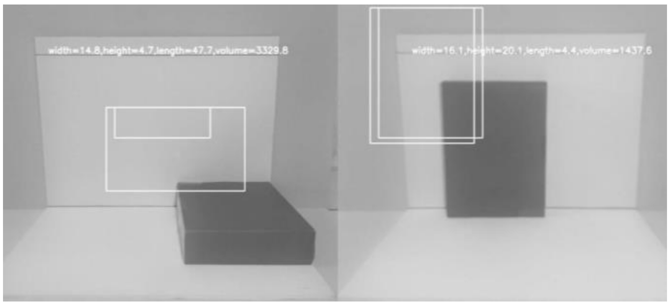


Fig. 19. Error in length detection

4.1 Optimization

The box placement on the platform played an important role for this system. The boxes were placed either at the side or center of the platform. 21 images using placement at the center and the remaining applied at the side placement. The result of detection in respect of placement of object on platform for each dimension is tabulated in Table 6.

Table 6
Comparison between object placement orientation and percentage error

Object placement	Dimension		
	Length	Width	Height
Side	1.41	0.07	0.13
Center	1.32	0.07	0.07

When computing the length through placement at the side, the error showed 9 % more compared to placement at center of platform. The height obtained through placement at the side also showed more 6 % at 13 % error compared to placement at center of platform. However, there is no difference when computing the width. This is due to the width dimension was calculated in parallel to the reference line and to each other. Both showed errors at 7 %. It can be said that placement at the center of platform provided more accurate results compared to the side.

Based on Figure 20 it depicted the top image as original set-up of platform before optimization. Originally, the background chosen was black coloured with the platform only made of white A4 paper. However, it was difficult to be applied during size detection as it caused many errors. The boxes also originally were not covered with coloured paper. However, during line detection, the system detected the wood mark as line and caused excessive error.

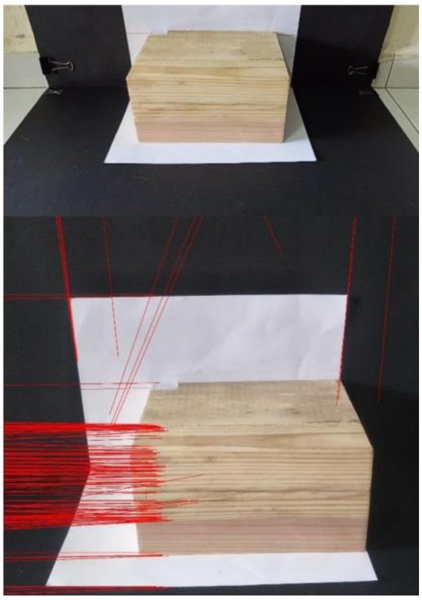


Fig. 20. (Top) Original platform set-up, (Bottom) Error due to the surface of boxes

Therefore, to reduce error in line detection and ease of computation and processing, the background was replaced with lighter coloured which is yellow. The platform was also replaced with B4 sized paper. The boxes were then given extra wrapping using red coloured paper to increase contrast and reduce error from the surface of boxes. Figure 21 below shows the latest set-up of platform.

Fig. 21. Finalized platform design

5. Conclusion

In this project, a MVS utilizing single camera was developed together with an algorithm to automatically detect box size volume measurement. For this stage of research, a controlled environment such as lighting condition, camera position and colour of boxes surface were taken into consideration to ensure quality results. The image acquisition step was carried out at the similar platform with controlled surrounding condition. Following the process flow, the image pre-processing was done by enhancing the image *via* gray-scaling and filter. Those stages include preparation before processing phase to extract information through edge detection and segmentation process. It was expected through the information extracted, the volume of the boxes could be computed automatically and be recorded. The accuracy of the system can be checked using MAPE. The system was targeted to achieve at least 90 % accuracy after optimization was carried out. The system managed to successfully detect area measurement of object with the least error.

However for volume detection, the algorithm still need some work to be refined for having more accurate result.

Acknowledgement
This research was not funded by any grant.

References

[1] M. Jeske, M. Grüner and F. Weiß, "Big Data in Logistics," DHL Customer Solutions & Innovation, Troisdorf, 2013.
[2] Holland, Frank, and P. HENNESSY. "Amazon is delivering nearly two-thirds of its own packages as e-commerce continues pandemic boom." *CNBC, August* 13 (2019).
[3] GD Express Carrier Bhd., "Annual Report 2019," GD Express, Petaling Jaya, 2019.
[4] Deutsche Post DHL Group, "2019 Financial Year," DHL Group, Bonn, 2019.
[5] Gongal, A., M. Karkee, and S. Amatya. "Apple fruit size estimation using a 3D machine vision system." *Information Processing in Agriculture* 5, no. 4 (2018): 498-503. https://doi.org/10.1016/j.inpa.2018.06.002
[6] Hu, Meng-Han, Qing-Li Dong, Pradeep K. Malakar, Bao-Lin Liu, and Ganesh K. Jaganathan. "Determining banana size based on computer vision." *International journal of food properties* 18, no. 3 (2015): 508-520. https://doi.org/10.1080/10942912.2013.833223
[7] Nyalala, Innocent, Cedric Okinda, Luke Nyalala, Nelson Makange, Qi Chao, Liu Chao, Khurram Yousaf, and Kunjie Chen. "Tomato volume and mass estimation using computer vision and machine learning algorithms: Cherry tomato model." *Journal of Food Engineering* 263 (2019): 288-298. https://doi.org/10.1016/j.jfoodeng.2019.07.012
[8] Wang, Zhibin, Kaiyi Wang, Feng Yang, Shouhui Pan, Yanyun Han, and Xiangyu Zhao. "Image enhancement for crop trait information acquisition system." *Information Processing in Agriculture* 5, no. 4 (2018): 433-442. https://doi.org/10.1016/j.inpa.2018.07.002
[9] Silwal, Abhisesh, Aleana Gongal, and Manoj Karkee. "Apple identification in field environment with over the row machine vision system." *Agricultural Engineering International: CIGR Journal* 16, no. 4 (2014): 66-75.
[10] Behera, Santi Kumari, Shrabani Sangita, Prabira Kumar Sethy, and Amiya Kumar Rath. "Image processing based detection & size estimation of fruit on mango tree canopies." *International Journal of Applied Engineering Research* 13, no. 4 (2018): 6-13.
[11] White, Darren J., Cato Svellingen, and Norval JC Strachan. "Automated measurement of species and length of fish by computer vision." *Fisheries Research* 80, no. 2-3 (2006): 203-210. https://doi.org/10.1016/j.fishres.2006.04.009
[12] Wang, Weiqiang, Yi Luo, Kun Yang, and Chunxue Shang. "Multi-angle automotive fuse box detection and assembly method based on machine vision." *Measurement* 145 (2019): 234-243. https://doi.org/10.1016/j.measurement.2019.05.100
[13] Smart Tools co., "Smart Measure," Suseong-gu, 2010.
[14] PotatotreeSoft, "ON 3D- CameraMeasure," 2013.
[15] Tay, Y. H., and U. Khairuddin. "Development of a Single Camera Machine Vision System for Automatic 3D Size Detection." In *IOP Conference Series: Materials Science and Engineering*, vol. 1051, no. 1, p. 012002. IOP Publishing, 2021. https://doi.org/10.1088/1757-899X/1051/1/012002
[16] Shahdib, Fayaz, Md Wali Ullah Bhuiyan, Md Kamrul Hasan, and Hasan Mahmud. "Obstacle detection and object size measurement for autonomous mobile robot using sensor." *International Journal of Computer Applications* 66, no. 9 (2013): 28-33.
[17] Peng, Tao, Zhijiang Zhang, Fansheng Chen, and Dan Zeng. "Dimension measurement and key point detection of boxes through laser-triangulation and deep learning-based techniques." *Applied Sciences* 10, no. 1 (2019): 26. https://doi.org/10.3390/app10010026
[18] Peng, Tao, Zhijiang Zhang, Yingjie Song, Fansheng Chen, and Dan Zeng. "Portable system for box volume measurement based on line-structured light vision and deep learning." *Sensors* 19, no. 18 (2019): 3921. https://doi.org/10.3390/s19183921
[19] Park, Ho-min, Arnout Van Messemac, and Wesley De Neveac. "Box-Scan: An efficient and effective algorithm for box dimension measurement in conveyor systems using a single RGB-D camera." In *Proceedings of the 7th IIAE International Conference on Industrial Application Engineering, Kitakyushu, Japan*, pp. 26-30. 2019. https://doi.org/10.12792/iciae2019.032
[20] Y. Mohd Mustafah, A. W. Azman and M. Ani, "Object Distance and Size Measurement Using Stereo Vision System," *Advanced Materials Research,* pp. 622-623, December 2012. https://doi.org/10.4028/www.scientific.net/AMR.622-623.1373

[21] Ferreira, Beatriz Quintino, Miguel Griné, Duarte Gameiro, João Paulo Costeira, and Beatriz Sousa Santos. "VOLUMNECT: measuring volumes with Kinect." In *Three-Dimensional Image Processing, Measurement (3DIPM), and Applications 2014*, vol. 9013, pp. 24-29. SPIE, 2014.

[22] Siswantoro, Joko, Anton Satria Prabuwono, and Azizi Abdulah. "Volume measurement of food product with irregular shape using computer vision and Monte Carlo method: a framework." *Procedia Technology* 11 (2013): 764-770. https://doi.org/10.1016/j.protcy.2013.12.256