Evaluation of Single and Dual Image Object Detection through Image Segmentation Using ResNet18 in Robotic Vision Applications

Phichitphon Chotikunnan 1, Tasawan Puttasakul 2*, Rawiphon Chotikunnan 3, Benjamas Panomruttanarug 4, Manas Sangworasit 5, Anuchart Srisiriwat 6

1, 2, 3, 5 College of Biomedical Engineering, Rangsit University, Pathum Thani, Thailand
4 Department of Control and Instrumentation Engineering, King Mongkut’s University of Technology Thonburi, Bangkok, Thailand
6 Department of Electrical Engineering, Pathumwan Institute of Technology, Bangkok, Thailand

Email: 1 phichitphon.c@rsu.ac.th, 2 tasawan.p@rsu.ac.th, 3 rawiphon.c@rsu.ac.th, 4 benjamas.pan@kmutt.ac.th, 5 manas.s@rsu.ac.th, 6 anuchart@pit.ac.th

*Corresponding Author

Abstract—This study presents a method for enhancing the accuracy of object detection in industrial automation applications using ResNet18-based image segmentation. The objective is to extract object images from the background image accurately and efficiently. The study includes three experiments, RGB to grayscale conversion, single image processing, and dual image processing. The results of the experiments show that dual image processing is superior to both RGB to grayscale conversion and single image processing techniques in accurately identifying object edges, determining CG values, and cutting background images and gripper heads. The program achieved a 100% success rate for objects located in the workpiece tray, while also identifying the color and shape of the object using ResNet-18. However, single image processing may have advantages in certain scenarios with sufficient image information and favorable lighting conditions. Both methods have limitations, and future research could focus on further improvements and optimization of these methods, including separating objects into boxes of each type and converting image coordinate data into robot working area coordinates. Overall, this study provides valuable insights into the strengths and limitations of different object recognition techniques for industrial automation applications.

Keywords—Robotic vision; Dual design; Image segmentation; Object Detection; ResNet18.

I. INTRODUCTION

Robotic arms have experienced significant development and widespread adoption across various industries, including automotive, electronics, pharmaceuticals, food processing, and aerospace. These versatile machines have evolved over the years, with advances in computer-controlled electrical systems, sensors, artificial intelligence, and machine vision enhancing their capabilities. Recently, collaborative robots, or cobots, have emerged as a new category designed to work safely alongside humans in shared workspaces. The integration of machine learning and AI has further expanded the applications of robotic arms into fields such as healthcare, where they assist in surgery, and space exploration, where they perform maintenance tasks on the international space station. The continuous advancement of technology promises even greater potential for robotic arms in the future.

Robotic vision technology offers a wide range of applications across various domains, making it a vital component of modern industries. In agriculture, it is employed for tasks such as detecting different fruit types using faster R-CNN systems, estimating fruit quantity and ripeness, evaluating crop harvests, and identifying plant diseases [1]-[9]. Besides agriculture, robotic vision technology is utilized for facial recognition [12]-[18], classroom location, public identity verification, and payment processing [19]. It can also determine if a person is wearing a mask [20]. Moreover, this technology is extensively used for object detection in diverse areas, including survey research, passenger detection in buses, and traffic management applications [21]-[29].

Having been in use for an extended period, this technology incorporates various techniques like CNN or R-CNN, multiscale image blocking, deep learning, and machine learning [23]-[33]. Image segmentation, a critical process in robotic vision technology, involves separating objects within an image by identifying specific points or coordinates of interest [34]-[36]. This technique has multiple applications across different fields, such as topographic surveys [37]-[39], medical research [40]-[43], and plant disease recognition [46], among others.

To achieve accurate image segmentation, multilevel thresholding techniques based on energy curves with harmony search algorithms or CNN techniques [44], as well as deep convolutional neural networks (DCNNs) [45], can be used. Examples of image segmentation applications include pathological image analysis [47], segmentation of lungs in chest X-ray images [48], brain tumor classification [49], COVID-19 diagnosis [50], and crack detection using image processing [51].

Expert systems, a type of artificial intelligence control system, are crucial in many fields, including medicine, engineering, and finance [52]-[54]. They simulate human decision-making processes by using a knowledge base and a set of rules to make decisions, making them valuable in situations requiring accurate and consistent decisions based
on specific criteria. Expert systems are primarily used in diagnosis [55]-[59] and can be combined with other systems, such as fuzzy expert systems [60], [61], fuzzy neural network expert systems [62], and neural network expert systems [63], to enhance their performance and diversity.

Neural networks play a crucial role in processing visual signals, with convolutional neural networks (CNNs) being widely employed in various applications such as dental image diagnostics [64], agriculture [65], and chest disease examination [66]. Residual neural networks, also known as ResNets [76]-[78], are a specific type of artificial neural network used in diverse applications, including image processing and disease detection. Due to their versatility, ResNets are popular across numerous fields and have multiple implementations, such as ResNet-18 [78], [79], ResNet-34 [80], ResNet-50 [81], and ResNet-101 [82]. The integration of these technologies can also extend to the development of multi-robotic arm systems [32] and mobile robots capable of facial detection [33]. This combination has led to the creation of increasingly accurate systems while streamlining the design process through the use of various tools and functions. As a result, these advancements continue to contribute to the ongoing progress of robotic vision technology.

This study focuses on the development and implementation of image preparation techniques for both single and dual image object detection using image segmentation, with potential applications in the realm of robotic vision systems. By leveraging the ResNet18 neural network for object detection based on color and shape, the research explores two image preparation procedures: single image preparation, which uses an expert system to validate the RGB color scale, and dual image preparation, which compares data differences between two images for more accurate object area determination before image segmentation. Additionally, the study calculates the center of gravity (CG) value of the image. The experimental results showcase the effectiveness of these approaches in collaboration with ResNet18 for identifying various object types, emphasizing the significance of this research as a case study in object detection through image segmentation and its potential impact on robotic vision applications.

II. METHOD

A. Robotic Vision

This research presents a robotic vision design, illustrated in Fig. 1 as a block diagram, that employs a collaborative structure between computer and robots. The design includes a computer system equipped with a camera to detect the location of an object, and the location information is then transmitted to the MATLAB program for image processing and object separation. The robot's operating system can be controlled by exchanging data with a computer. The Arduino microcontroller board can operate in real-time by connecting to the MATLAB & Simulink program. While other research studies have proposed various designs for the system, such as micro-robotics [83], Microgrid [84], mini drone [85], DC motor control [86], [87], BLDC motor [88], lane-keeping control [89], and hexapod robot [90], none of them have focused on image processing. Nonetheless, these studies illustrate the potential applications of robotic vision design in different areas of robotics.

This study presents a sample design for the robot's working area, as shown in Fig. 2 and Fig. 3 illustrates the various positioning areas of the robot from a top view, which includes the camera, the robot, the workpiece area, and the workbox. To minimize outside interference, the camera is installed in a closed room, and the room is illuminated with white neon lights. The camera is positioned at a height of 65 cm above the workpiece tray, which measures 28 cm by 18 cm.

Fig. 1. Block diagram of the overall system.

Fig. 2. Overview of the work area.

Fig. 3. Overview of the top view area.

Fig. 4 shows different robots that are compatible with the robotic vision system. These robots include the mechanical arm robot [91], [92], scala robot, and cartesian robot [93]-[97]. Additionally, industrial motors can be integrated into the robotic vision system. Furthermore, mobile robots can also be applied to the machine vision system [98], [99].
The research design's main objective is to enable the robot to detect objects in the work area and automatically pick them up to place them in a workpiece box. The workpiece box may be divided based on the object's color or shape. The entire system design comprises three parts, as illustrated in Fig. 5, which are explained in detail below:

1. Computer vision, which involves image processing, object coordinate identification, object color separation, and object shape separation.
2. Robot trajectory design.
3. Object retrieval and placement.

However, this research will only present part 1 of the design.

**B. Hardware Implementations**

This section describes the components utilized in the image processing phase of this research, including the HD webcam and a notebook ASUS TUF gaming FX506LI-HN091T.

The Logitech HD webcam C170 was chosen as the camera for this research due to its cost-effectiveness. This camera is primarily designed for video calling and offers seamless communication with friends and family through most major instant messengers. Logitech fluid crystal technology enhances the camera's video quality, providing smoother video, sharper pictures, richer colors, and clearer sound in real-world conditions. The camera has a standard resolution of 640x480 pixels. However, video capture can be set to 1024x768 pixels, and photos can be captured at up to 5 MP. The camera connects via Hi-Speed USB 2.0, as certified by the manufacturer. The camera body is presented in Fig. 6.

![Logitech HD webcam C170](image)

The second piece of equipment used in this research is the notebook ASUS TUF gaming FX506LI-HN091T, which features an intel core i7-10870H CPU, a GTX 1650TI 4GB GDDR6 video card, a 15.6” Full HD IPS 144Hz adaptive sync display, 16 GB DDR4 2933 MHz RAM, and a 512 GB SSD M.2. This equipment was utilized for image processing operations and controlling the robot through the MATLAB R2022b program, as illustrated in Fig. 7.

![Asus TUF Gaming FX506LI-HN091T](image)

**III. IMAGE PROCESSING AND IMAGE SEGMENTATION**

**A. Overview of the Image processing Procedure**

This section provides an overview of the computer vision system's functionality, which can be separated into internal components as shown in Fig. 8.
The image obtained from image processing is then framed and stored in the ResNet-18 system to sort the images according to the inspection target and display the results of the running and ending the program.

This section describes the program flowchart for image processing, as depicted in Fig. 8. The first step to activate the camera in MATLAB is to install the MATLAB support package for USB webcams, as shown in Fig. 9. Next, the webcam object is initialized using the command "cam = webcam", and the video stream can be previewed using "preview(cam)" as shown in Fig. 10. To capture an image from the camera, the "snapshot(cam)" command is used to store the image as RGB image data in a program variable.

To reduce the image size and processing time, the image is cropped to select the area of interest in image processing. In this research, the coordinates (252, 323) and (800, 663) are used, as shown in Fig. 11. Finally, the objects are placed in the workpiece placement area, as illustrated in Fig. 12 and Fig. 13.

Image processing and segmentation are commonly used techniques to identify objects within images. Image processing involves manipulating the original image to enhance its features and make it easier to analyze, such as by converting it to black and white or adjusting its contrast. Image segmentation is the process of dividing the image into meaningful segments or regions, which can involve removing some parts of the image that are not relevant to the analysis from the original image. The effectiveness of image segmentation techniques depends on the specific characteristics of the image being analyzed. For instance, similar research has been applied to image thresholding for the highway visual tracking system [100].

During the preparation process for image analysis, color images are usually converted to grayscale and modified to become black and white. The resulting image is then used for image segmentation to identify the coordinates or locations of the target object. As seen in Fig. 14, the Logitech HD webcam C170 produces images of moderate quality that may appear unclear, but they are still useful for identifying the coordinates or targets of objects within the defined work area. To perform RGB to grayscale conversion in Image Processing, the original color image is first converted to grayscale using the 'rgb2gray' function. After that, the 'im2bw' function is applied to the grayscale image using a threshold value to convert it into a binary image. However, image segmentation may not always result in the complete identification of all objects in the image. In this example, only three out of the ten expected objects were successfully identified.
value to 2,700 and the maximum value to 10,000 and saves the index value in the bboxes variable. Using the command [areas, centroids, bboxes] = step(blob_analysis, binary_img), where the binary_img variable is a predesigned black and white image, the resulting bboxes variable provides the value of areas(𝑥_объект, 𝑦_объект), which returns the 𝑥_объект, 𝑦_объект coordinates of the top corner of the object within the image, and centroids(𝑥_объект, 𝑦_объект), which is the length of the object within the image itself.

IV. PREPARATION PROCESS OF IMAGE PROCESSING

In this section, we will discuss the preparation process for image processing, as presented in the paper. The preparation process includes two conditions for image processing: the preparation process of single image processing and the preparation process of dual image processing.

A. Preparation Process of Single Image Processing

This section focuses on the design of the preparation process for single image processing, which involves designing the rules of an expert system presented as equation as in (3).

\[
\text{IF } (RGB \text{ of image input } \geq \min \text{ of } RGB \text{ in background}) \text{ and } (RGB \text{ of image input } \leq \max \text{ of } RGB \text{ in background}) \text{ THEN } (RGB \text{ of image input is 0}).
\]

The rules outlined in (3) can be translated into a program code, as shown in (4).

\[
\text{IF } (\text{input } R_{xy} \geq \min \text{ of } R_{xy} \text{ in background}) \text{ and } (\text{input } G_{xy} \geq \min \text{ of } G_{xy} \text{ in background}) \text{ and } (\text{input } B_{xy} \geq \min \text{ of } B_{xy} \text{ in background}) \text{ and } (\text{input } R_{xy} \leq \max \text{ of } R_{xy} \text{ in background}) \text{ and } (\text{input } G_{xy} \leq \max \text{ of } G_{xy} \text{ in background}) \text{ and } (\text{input } B_{xy} \leq \max \text{ of } B_{xy} \text{ in background}) \text{ THEN } (R_{xy} \text{ of image input is 0}) \text{ and } (G_{xy} \text{ of image input is 0}) \text{ and } (B_{xy} \text{ of image input is 0}).
\]
Fig. 17 shows a sample image in which the RGB color value of the floor of the object placement was measured. The measurement revealed that the plates had RGB color values of R ranging from 105-180, G ranging from 115-180, and B ranging from 95-180.

After cropping the original image in Fig. 18, the expert system method of (4) is used to modify the image. Following that, the image is converted to black and white using the conditions designed according to rule (4).

The im2bw(I, level) command is used to modify the image data. This command converts the grayscale image I to a binary image BW. It replaces all pixels in the input image with a luminance greater than the level with the value 1 (white) and replaces all other pixels with the value 0 (black).

The range of level is relative to the signal levels possible for the image's class. In this research, a level value of 0.2 corresponds to an intensity value halfway between the minimum and maximum value of the class of image files used in this research.

B. Preparation Process of Dual Image Processing

The preparation process of dual image processing is a proposed technique that utilizes the conditions of the original image before the work was placed and the image with the workpiece placed on it, as shown in Fig. 11 and Fig. 13. The technique involves comparing the RGB values in the x, y coordinates of both images to determine whether they are within the specified range in Fig. 11, while maintaining the size of the RGB colors in the image. If the conditions are not met, the values are set to 0. The expert system can be designed using (5).

\[
\text{IF (RGB of image input}} \leq [\text{RGB of background} + \text{range of upper}] \text{and (RGB of image input}} \geq [\text{RGB of background} - \text{range of lower})] \text{ WHEN (RGB of image input is 0).}
\]

The RGB color range of the floor of the object holder in the sample from Fig. 16 has been measured. The upper range can be set to 20 and the lower range can be set to 79, and both values are suitable for the experiments conducted in this study.
When cropping the original image to produce Fig. 19, the expert system method was applied using Equation (6) to modify the image. The resulting image was then converted to black and white using the im2bw function, with a level value of 0.2. This process was conducted in accordance with the conditions designed in rule (5) of the preparation process of dual image processing test, as well as the preparation process of a single image processing test.

This process was conducted in accordance with the conditions designed in rule (5) of the preparation process of dual image processing test, as well as the preparation process of a single image processing test.

V. OBJECT DETECTION USING RESNET-18

A. Architecture and Environment of the System

To identify objects or details within an image for inspection purposes, various techniques may be employed. In some cases, the use of optimized image processing techniques [101] can help determine the coordinates or areas to be focused on in the image. This can involve the use of CNN algorithms [102] or deep learning [103] to help extract the necessary information needed for object identification. Currently, many ready-made algorithms [104], [105] are available for image processing, which can be used to reduce design time for image analysis. The image classification system utilized in this study is based on convolutional neural network (CNN) and its architecture is presented in Fig. 20.

The system consists of two main components:

1. Feature extraction is an image processing technique that extracts the salient features of objects in the image, such as edges, contours, and creates a model. Then, the extracted feature information is taken as input and processed by the neural network in the following stage.

2. Neural network for classification is used to classify the images according to the model groups that have been trained.

In this research, the ResNet-18 architecture was utilized, which is a learning framework that enhances the effectiveness of deep residual network training. When working with a significant number of layers, vanishing gradients may become an issue. However, ResNet addresses this by using shortcut methods, bypassing certain network layers. The ResNet-18 model was implemented, and its structure is presented in Fig. 22, with the Add-On explorer depicted in Fig. 21.

B. System design of ResNet-18

The classification system's working design comprises the following steps:

1. The input consists of image data of workpieces that can be used for training, validation, testing, and prediction. The number of input images affects the processing because each sample of the workpiece block is processed separately. This research deals with two groups of workpiece blocks. Group 1 comprises two types, namely, triangular and rectangular. Group 2 includes six types, divided into six colors, four of which are blue, green, orange, red, verdant, and yellow. Each category in the two groups has a varying number of images. To equalize the number of images for each class in the training data, we introduced a technique called oversampling. Deep learning can become challenging if the training data's amount varies across classes, as shown in Fig. 23. To address this, an oversampling technique was used to balance the number of images for each class in the training data.

2. Convolution is a layer that processes input image data using a specified kernel size to obtain a feature map. This process detects straight, curved, and beveled edges in the images, allowing the system to create a model.

3. Max pooling is a layer that reduces the size of the feature map by half.

4. The fully-connected layer (FC) organizes the multidimensional data from the previous layer into a 1-dimensional vector (called Flatten) and feeds it to the neural network layer. In this layer, the input and weights act as

![Original Image](image1.jpg)

![Binary Image](image2.jpg)

Fig. 19. Outcome of employing the dual image technique in the image preparation process, converting an image to black and white.

Fig. 20. The architecture and procedure of convolutional neural networks.

Fig. 21. Add-On explorer of the deep learning toolbox model for ResNet-18 network.
classification, providing output results to predict the type of workpiece block depicted in the input image.

Fig. 22. Design of the ResNet-18 model architecture.

C. The Workflow for Machine Learning

The system is trained on two groups of workpiece block datasets, enabling it to learn and distinguish between various types of workpiece blocks using a feature extraction and modeling system. The resulting model can then be utilized to recognize and predict the type of block from images captured by the Logitech HD webcam C170 with accuracy. Deep learning is used as the machine learning process, as depicted in Fig. 24.

Fig. 24. The learning procedure of the classification system.

In this research, the detection learning process was developed using image recognition as a tool to teach models to analyze images. The model was trained using two groups of work blocks, which were separated into two different trainings: color training and shape training.

In the color training group from Fig. 25, there were 43 blue images, 53 green images, 11 orange images, 42 red images, 11 verdant images, and 54 yellow images. To ensure an equal number of images for each class in the training data, oversampling was done, resulting in 38 images per group as shown in Fig. 26. The GUI data training in color training was displayed in Fig. 27 during the ResNet-18 method training process. Once the training was completed, the mean training and validation accuracy in color training were computed and displayed in Fig. 28, indicating the accuracy in each category.

Fig. 25. Examples of image data employed in classification.

Fig. 26. Outcomes of the oversampling technique in color training.
Phichitphon Chotikunnan, Evaluation of Single and Dual Image Object Detection through Image Segmentation Using ResNet18 in Robotic Vision Applications
A. RGB to Grayscale Conversion to Image Processing

The results from the first test in the experimental study on RGB to grayscale conversion for image processing, which aimed to prepare for single image processing using the RGB to grayscale conversion technique and utilized images from Fig. 34, Fig. 35, Fig. 36, and Fig. 37, indicated that the basic method used in the test had limited ability to distinguish between the workpiece tray and the parts. Specifically, with the images from Fig. 32 and Fig. 33, this technique was able to successfully isolate only 1 out of 5 and 1 out of 10 specimens, respectively.

Technically, by transforming the original RGB image to grayscale and then to black and white, the program can efficiently identify the color and object shapes using ResNet-18, and accurately determine the edges of the object and CG values within the image. However, the program has limitations due to the difference in color levels between the tested part and the workpiece tray, with grayscale values in the range of 140-180, while the grayscale value of the part image is either 0-160 or 180-255. Thus, this technique may not be suitable when there is a diverse range of colors in the workpiece.
B. Single Image Processing

The second test of experiments focuses on the preparation process of single image processing using images from Fig. 32 and Fig. 33, as depicted in Fig. 38, Fig. 39, Fig. 40, and Fig. 41. The results demonstrate the program's ability to extract the color range between the background and the workpiece and convert the original image into black and white image, accurately separating the edges of the object and CG values within the image with a 100% success rate for objects located in the workpiece tray. The ResNet-18 algorithm efficiently identifies the color and object shapes, and the results are accurate in all tested photographs.

Although some images have devices in the background, such as the handle head or reflected light, that can affect the completeness of the image cutting, the program can be edited by setting the maximum and minimum values of the space within the object and adding more rules to cut off the RGB color group.

C. Dual Image Processing

The third test in the experimental study focused on the preparation process of dual image processing using images from Fig. 32 and Fig. 33, and the results are presented in Fig. 42, Fig. 43, Fig. 44, and Fig. 45. Similar to the single image processing results, image processing techniques were utilized, but with additional capabilities. The program was able to extract the color range between the background and the color of the workpiece, resulting in cropping the original image into a black and white image that accurately separated the edges of the object and CG values within the image, with a 100% success rate for objects located in the workpiece tray.

Furthermore, the program was able to use image position information to determine the location of the object within the
image and specify the CG value. By using ResNet-18, the program accurately identified the color and shape of the object in all photographic tests, even with reflections that were not completely removed from the original image. Moreover, the program was capable of cutting the background image and the head of the robot gripper, expanding its capabilities beyond that of single image processing. Overall, dual image processing has more advanced capabilities than single image processing.

![Original image](image1)

**Fig. 42.** The results of test I in dual image processing experimental.

![Binary image](image2)

**Fig. 43.** Final results of Test I in the dual image processing experiment.

![Crop the object in image](image3)

![Type and CG of object in image](image4)

**Fig. 44.** The results of test II in dual image processing experimental.

![Final result of image](image5)

**Fig. 45.** Final results of Test II in the dual image processing experiment.

D. Summary of the results

The experimental results demonstrate the superiority of the dual image processing technique over both RGB to grayscale conversion and single image processing. The program achieved a 100% success rate in accurately separating the edges of the object and CG values within the image, while also identifying the color and shape of the object using ResNet-18. The program was able to use image position information to determine the location of the object within the image and specify the CG value. Additionally, the program was capable of cutting the background image and the head of the robot gripper, expanding its capabilities beyond that of single image processing. As a result, the dual image processing technique provides advanced capabilities for object recognition and can be a valuable tool in industrial automation applications.

However, both single and dual image processing methods have limitations and disadvantages. Single image processing requires checking the light and color tone within the image to cut off other objects that need to be removed, which may require many conditions. On the other hand, dual image processing can extract background images and objects if there is a picture in the reference, making the conditions for cutting objects within the image less, but may become problematic if the light is shifted.

To solve the aberration image problem, the background clipping condition may need to be set as the nature of the index data or may use NN to solve the problem, depending on the amount of information or conditions that arise to create a boundary of conditions.

VII. Conclusion

In summary, this study examined the use of RGB to grayscale conversion, single image processing, and dual image processing methods for object recognition in industrial automation applications. The findings suggest that the dual image processing technique offers superior capabilities in accurately identifying object edges, determining CG values, and cutting background images and gripper heads. However, single image processing may still have advantages in certain scenarios with sufficient image information and favorable lighting conditions. Overall, this study provides valuable insights into the strengths and limitations of different object recognition techniques for industrial automation applications. Future research could focus on further improvements and
optimization of these methods, including the possibility of separating objects into boxes of each type and converting the image's coordinate data into the coordinates of the robot's working area to design a point-to-point robot path. These areas may require further investigation in future research.

ACKNOWLEDGMENT

The researcher would like to thank the Research Institute, Academic Services Center, and College of Biomedical Engineering, Rangsit University for the grant of research funding to the research team.

REFERENCES


