INTEK Jurnal Penelitian
Vol. 11, No. 2, pp. 8-13, October 2024

Development of a Color-Based Image Recognition System for Robotic Sorting and Picking

Sitti Wetenriajeng Sidehabi [1, *,a] Muhammad Fadli Azis [2, b] Muhammad Asbar [3, c]
[1,2,3] Automated Mechanical System, Politeknik ATI Makassar, 220 Sunu St, Makassar, 90211, Indonesia
[*, a] tenri@atim.ac.id (Corresponding Author)
[a] fadli@kemenperin.go.id
[c] 21osp585@atim.ac.id

***Abstract*—This research presents the development of an automated object sorting and picking system using a robotic arm controlled by colour-based image recognition. The system is designed to enhance efficiency and accuracy in manufacturing processes by eliminating the need for manual sorting. A Dobot Magician robotic arm, an Arduino microcontroller, a conveyor belt, a photoelectric sensor, and a camera are integrated to achieve this goal. Colour segmentation is implemented using the HSV colour space, enabling the system to accurately classify objects based on colour. Experimental results demonstrate the system's ability to successfully sort objects of three colours in a random sequence with 100% accuracy over ten trials.**

Keywords—Image Recognition, Robotic handling, HSV (Hue Saturation Value), Automated Sorting, Arduino

I. Introduction

The rapid expansion of the manufacturing industry has driven the development of more efficient and accurate production systems. One of the common challenges in manufacturing is manual sorting, which is often time-consuming and prone to human errors [1], [2], [3]. Automation has become a critical solution, especially with advancements in robotics and computer vision technologies.

Previous studies have explored various methods to enhance the efficiency of automated sorting. A. Luha's research [4] discusses the integration of an Arduino-based robotic arm with a PLC-based conveyor system for automated material handling. This system successfully improved the efficiency of material transfer in various industries, although it was limited to location-based sorting without considering visual parameters such as color.

Another study by Tugino et al. [5] developed a sorting and packaging system using the Dobot Magician robot and an Outseal PLC. The system was able to sort and package objects based on their color, but the focus was mainly on packaging and did not involve more complex image processing.

Pratama Dimas S [6] designed an automated nut-sorting system using a Raspberry Pi and contour area analysis. The system enabled sorting based on nut size by analyzing the contour area of binary images captured by a camera. While accurate, this approach was restricted to size analysis and did not account for objects with varying characteristics, such as color.

Slamet Riyadi et al. [7] implemented a weight-based sorting system using the Dobot Magician robotic arm controlled by blockly programming. The system could separate objects based on weight, supported by a load cell sensor. However, visual parameters like color were not incorporated, limiting the system to objects distinguishable only by weight.

The objective of this research is to develop an automated sorting system capable of sorting objects by color using the Dobot Magician robotic arm. The system integrates an Arduino microcontroller, conveyor belt, photoelectric sensor, and camera, utilizing color segmentation in the HSV color space to accurately recognize and classify objects. This research is expected to contribute significantly to the development of more flexible, efficient, and accurate sorting automation systems in the manufacturing industry.

II. Research Methodology

A. Data Collection Techniques

A two-pronged data collection approach was employed to comprehensively understand automated sorting systems and the specific requirements of this project. Firstly, a thorough literature review was conducted to gather relevant research on computer vision

DOI: https://doi.org/10.31963/intek.v11i2.5009

INTEK Jurnal Penelitian
Vol. 11, No. 2, pp. 8-13, October 2024

[8], robotic arm technology [9], [10], automated sorting systems [11], [12], Dobot Magician [13], Arduino [14], [15], Python programming [8], [9], OpenCV [18], and photoelectric sensors [19]. Secondly, a detailed observation of the equipment to be integrated into the system was carried out. This hands-on approach provided valuable insights into the sensors and actuators' characteristics, capabilities, and limitations, which are crucial for the subsequent design and implementation phases.

The success of the proposed automated sorting system relies heavily on accurate image processing techniques. Digital images captured by a camera are numerical representations of light intensity at various points in a scene. The HSV colour model, an alternative to RGB, was chosen for this study due to its intuitive representation of the colour. Hue, saturation, and value components provide a more human-perceptible way to describe colours, making them suitable for object segmentation and classification [20]. Table 1 presents this research's specific HSV ranges used for colour detection.

Table 1. HSV color range

Colour	HSV Lower Limit (Lower range)	HSV Upper Limit (Upper Range)
Red	(136, 87, 111)	(180, 255, 255)
Green	(25, 52, 72)	(102, 255, 255)
Blue	(94, 80, 2)	(120, 255,255)
Brown	(5, 164, 61)	(25, 184, 141)
Yellow	(25, 70, 120)	(30, 255, 255)
Orange	(10, 100,20)	(25, 255, 255)
Cyan	(85, 100, 20)	(95, 255, 255)
Black	(0, 0, 0)	(50, 50 ,50)

B. Block Diagram

Based on the system framework in Figure 1, there are two primary inputs: a camera processed by a computer using Python programming and a photoelectric sensor connected to an Arduino to control a relay. An Arduino Mega 2560, powered by a USB port, is the controller. The system's outputs are controlled by a relay, which acts as a switch for the speed-controlled conveyor via a motor controller, and a Dobot Magician equipped with a suction cup tool.

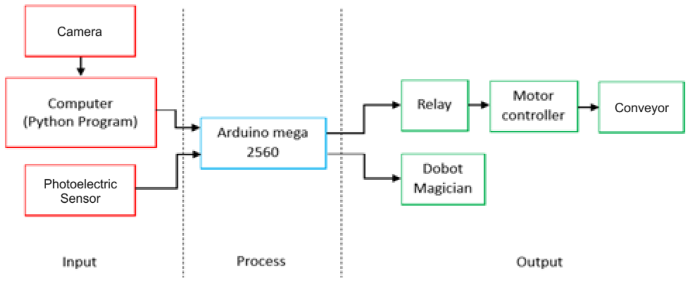

Figure 1. Block diagram

C. Flowchart

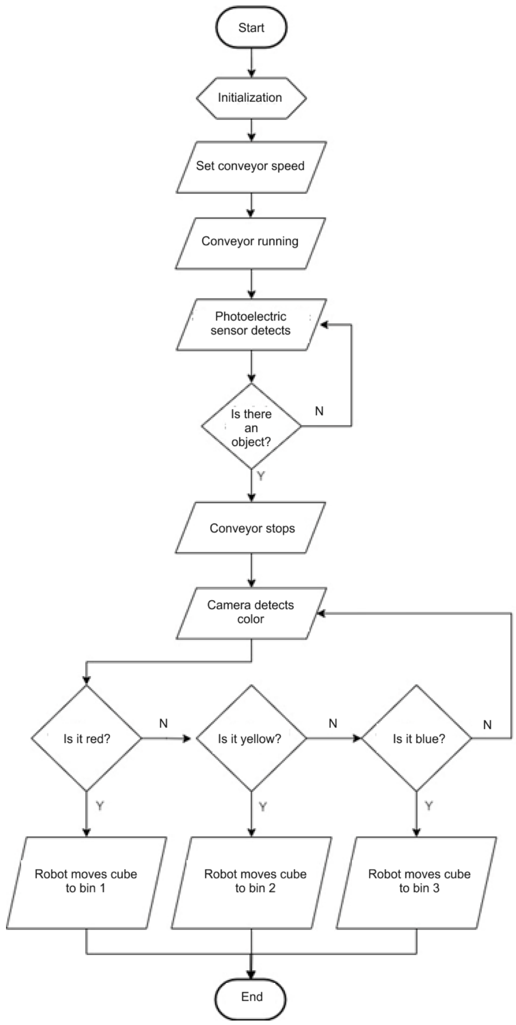

Figure 2. Flowchart

DOI: https://doi.org/10.31963/intek.v11i2.5009

Figure 2 illustrates the overall workflow of the automated sorting system. Once the photoelectric sensor detects a cube, the conveyor belt pauses, allowing the camera to capture an image of the cube. A computer vision algorithm then processes the captured image to determine the cube's colour. Based on the identified colour, the robotic arm is instructed to move the cube to the corresponding bin.

D. Software Development

An application has been developed to simplify the implementation process. This application can access internal and external computer cameras for colour identification and establish serial USB communication as input for the Arduino.

E. Device Fabrication

Based on the desired system operation, the device is fabricated according to the previously determined specifications and components.

F. Testing

The object sorting and picking system using a colour-based robotic handler will be tested. The system testing will include:

1. Testing camera identification with three colours: red, yellow, and blue, through the application.
2. Conveyor testing.
3. Testing the Arduino connection with the Dobot Magician using X, Y, and Z coordinate positions.
4. Overall system testing.

III. Results and Discussion

A. Wiring Diagram

A wiring diagram was created to illustrate the electrical connections between all components and devices used. In Figure 3, the adapter is connected to a 220 VAC power source and converted to a 12 VDC voltage to power the E3X-NA11 photoelectric sensor and the conveyor motor, whose speed is controlled using a motor speed driver. The E3X-NA11 photoelectric sensor detects objects on the conveyor, and when an object is detected, the relay interrupts the current to the motor driver, causing the conveyor to stop. The camera captures images of the objects, which are then processed by a computer for colour identification. The identification results are sent to Arduino for decision-making. The Arduino Mega 2560 processes inputs from the sensor and controls the conveyor and robot.

In addition, Table 2 shows the list of I/O pin connections between the components and the Arduino Mega 2560. The photoelectric sensor is the input, while the relay and Dobot Magician are the outputs.

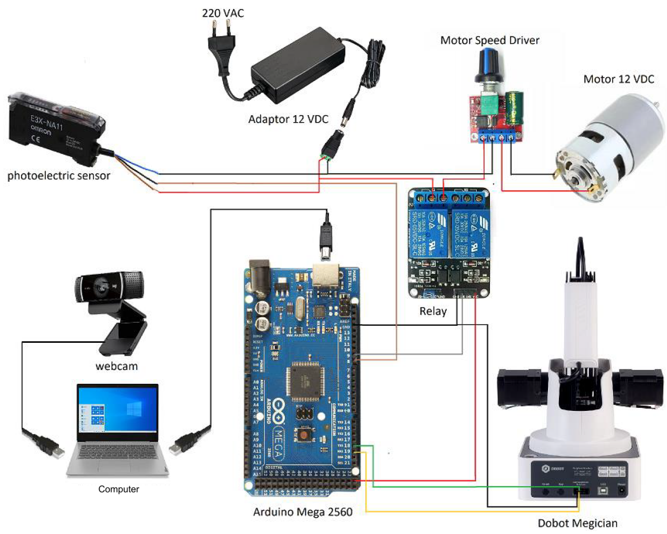

Figure 3. Wiring diagram

Table 2. I/O List

Component	Pin Component	Pin Arduino
Photoelectric sensor E3X-NA11	Output 0V	D8 GND
Optocoupler relay	In1 VCC GND	D9 5V GND
Dobot Magician	RX TX GND	TX1 RX1 GND

B. Software

The application was developed using the Python programming language and the Visual Studio Code IDE. To create a standalone executable file, the code was packaged using pip install pyinstaller, resulting in a .exe file named color_detection.exe. As depicted in Figure 4, the user interface features a camera frame for real-time video capture and separate frames for displaying the

DOI: https://doi.org/10.31963/intek.v11i2.5009

INTEK Jurnal Penelitian
Vol. 11, No. 2, pp. 8-13, October 2024

detected red, yellow, and blue colours. The application leverages the OpenCV library for image-processing tasks.

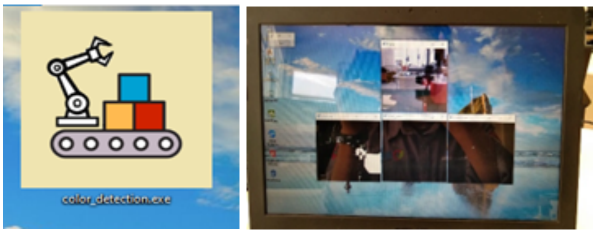

Figure 4. Software interface

C. *Fabrication Results*

The designed colour-based object sorting and picking system with a robotic handler and image recognition has been successfully fabricated and integrated into a 23.5 x 11.5 x 7 cm trainer box, as shown in Figure 5. An Arduino Mega 2560 microcontroller serves as the system's brain, processing image data from the computer and controlling the robot's movements and conveyor operation.

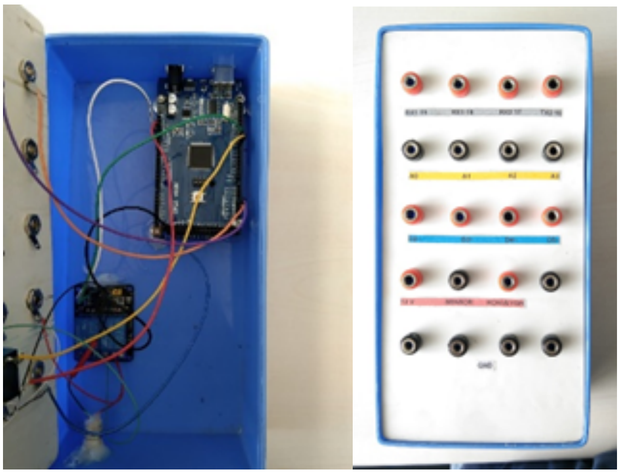

Figure 5. Fabrication Results

D. *System Integration*

Figure 6 illustrates the complete integration of the system, showcasing the interconnectedness of the sensor, conveyor, computer, Arduino controller, and Dobot. The photoelectric sensor detects objects on the conveyor and sends a signal to the Arduino. The Arduino then controls the conveyor motor and communicates with the computer. The computer, running the colour detection application, processes the image captured by the camera and sends control signals to the Arduino. The Arduino, in turn, controls the Dobot to pick up the object based on its colour and position. All components are connected via USB and power cables.

Figure 6. System Integration

E. *Color Identification Application Testing*

The colour reading application was tested to determine the accuracy of detecting three different colours and to verify the serial connection between the application and the Arduino Mega. Table 3 presents the results of three trials, showing the detected colour ranges in HSV format, which corresponds to the colour masks displayed on the computer monitor. Each detected colour generates a serial output character, 'B' for blue, 'Y' for yellow, and 'R' for red.

Table 3. Colour identification from the camera

Range HSV (hue saturation value)	**Detected Color**			**Serial Output**	**Notes**
	1	**2**	**3**		
Blue Lower = (160, 75, 75) Upper = (180, 255, 255)	Blue	Blue	Blue	B	Matches
Yellow Lower = (100, 100, 100) upper = (140, 255, 255)	Yellow	Yellow	Yellow	Y	Matches
Red lower = (20, 100, 100) upper = (30, 255, 255)	Red	Red	Red	R	Matches

F. *Conveyor Testing*

Conveyor testing was conducted to evaluate its performance based on the photoelectric sensor's output. When the sensor detects an object, it generates a logic high signal, causing the relay to switch off the conveyor. Conversely, when no object is detected, the sensor produces a logic low signal, activating the relay and turning on the conveyor. Table 4 presents the test results, demonstrating the conveyor's response to the sensor's input.

DOI: https://doi.org/10.31963/intek.v11i2.5009

Table 4. Conveyor testing

Sensor Condition	Relay Logic	Conveyor Condition
Detecting	*High*	OFF
Not Detecting	*Low*	ON

G. Robot Control Testing

Table 5. Robot control testing

Arduino Control Coordinates			Dobot Studio App Coordinates			Coordinate Difference	Accuracy (%)
X	Y	Z	X	Y	Z		
229	-72	94	229	-72	94	0	100
22	247	9	-22	247	9	0	100
77	235	10	77	235	10	0	100
149	198	9	149	198	9	0	100
236	-73	15	236	-73	15	0	100

To verify the accuracy of the robot's movement based on Arduino commands, the robot's coordinates were compared to those set in Dobot Studio. Five different coordinate points were tested. The Arduino sent position commands to the robot, which moved to the specified coordinates. The coordinates displayed on the Dobot Studio dashboard matched the values sent by the Arduino. The results in Table 5 show a coordinate difference of 0, indicating 100% accuracy in robot control using the Arduino.

H. Overall System Testing

This test evaluated the overall performance of the object sorting system, integrating electrical, mechanical, and software components. Table 6 presents the results of ten trials using three different coloured cubes. In all trials, the cubes were successfully sorted into their respective bins: red in bin 1, yellow in bin 2, and blue in bin 3, resulting in 100% accuracy. The colour readings from the camera, conveyor operation, robot coordinate movements, and sorting process functioned correctly. Compared to previous research that primarily focused on either mechanical handling or size-based sorting [4], [6], [7], this system offers a significant advancement by incorporating colour-based image recognition with HSV segmentation, enabling more flexible sorting capabilities.

Table 6. Overall system testing

Cube Colour	Identified Colour	Conveyor Status	Starting Cube Coordinates			Ending Cube Coordinates			Bin	Notes
			X	Y	Z	X	Y	Z		
Red	Red	Good	236	-73	15	149	198	9	1	Matches
Yellow	Yellow	Good	236	-73	15	77	235	10	2	Matches
Blue	Blue	Good	236	-73	15	-22	247	9	3	Matches
Blue	Blue	Good	236	-73	15	-22	247	9	3	Matches
Yellow	Yellow	Good	236	-73	15	77	235	10	2	Matches
Red	Red	Good	236	-73	15	149	198	9	1	Matches
Blue	Blue	Good	236	-73	15	-22	247	9	3	Matches
Yellow	Yellow	Good	236	-73	15	77	235	10	2	Matches
Blue	Blue	Good	236	-73	15	-22	247	9	3	Matches
Red	Red	Good	236	-73	15	149	198	9	1	Matches

DOI: https://doi.org/10.31963/intek.v11i2.5009

IV. Conclusion

This study successfully implements a colour-based object sorting and picking system using image recognition and robotics. The system accurately identifies and sorts objects of different colours with a 100% success rate. While the system demonstrates promising results, future work can focus on expanding its capabilities to include more complex object recognition tasks and higher throughput.

Acknowledgment

The researcher would like to express their gratitude to the Institute for Unit Transformasi Digital 4.0 (UTD 4.0) of Politeknik ATI Makassar for their support in this research.

References

[1] H. Putami, "Sistem Pensortiran Barang Berdasarkan Warna Menggunakan Antarmuka Visual Studio," *MSI Trans. Educ.*, vol. 02, 2021.

[2] M. I. Sari, R. Handayani, S. Siregar, and B. Isnu, "Pemilah Benda Berdasarkan Warna Menggunakan Sensor Warna TCS3200," *TELKA - Telekomun. Elektron. Komputasi dan Kontrol*, vol. 4, no. 2, pp. 85–90, 2018, doi: 10.15575/telka.v4n2.85-90.

[3] Kartiria, Siti Amalia, Rafika Andari, and Ari Saputra, "Perancangan Sistim Penyortiran Barang Berdasarkan Berat Berbasis Mikrokontroller ATmega328," *J. Tek. Elektro Inst. Teknol. Padang*, vol. 11, no. 1, pp. 64–70, 2022.

[4] A. F. Luha, "Integrasi Lengan Robot Berbasis Arduino Dan Sistem Convenyor Berbasis PLC Dalam Sistem Pemindah Barang Otomatis," *Semnasti 2023*, vol. 3, no. 2, pp. 212–218, 2023.

[5] Tugino, Ahmad Dwi Indra Buana, Mohammad Arsyad, and Hasta Kuntara, "Sistem Picking Dan Packaging Menggunakan PLC Outseal dan Robot Magician," *Tek. Elektron.*, vol. 2022, no. November, pp. 160–166, 2022.

[6] Pratama Dimas S, "Rancang Bangun Conveyor Penyortir Mur Berbasis Raspberry Pi Menggunakan Metode ContourArea," *J. Tek. Elektro*, vol. 11, no. 02, pp. 246–254, 2022.

[7] S. Riyadi, M. Suhanda, and E. Agung Nugroho, "Kinematika Penyortiran Material Berbasis Berat Dengan Blockly Program," *Ramatekno*, vol. 1, no. 2, pp. 9–16, 2021, doi: 10.61713/jrt.v1i2.16.

[8] Arnita, F. Marpaung, F. Aulia, and R. C. Nabila, "COMPUTER VISION DAN PENGOLAHAN CITRA DIGITAL," 2022.

[9] A. Trayektori, S. Pick, P. Dobot, N. Rosmawati, S. Riyadi, and E. A. Nugroho, "ANALISIS TRAYEKTORI SISTEM PICK AND PLACE DOBOT MAGICIAN," *Ramatekno*, vol. 4, no. 1, pp. 47–51, Apr. 2024, doi: 10.61713/JRT.V4I1.161.

[10] W. A. Utomo, A. R. Pamungkas, and A. P. Y. Krisnanda, "Rancang Bangun Lengan Robot Otomatis Pengawas Konveyor Pada Pabrik Teh Berbasis Arduino Uno," *Go Infotech J. Ilm. STMIK AUB*, vol. 24, no. 1, p. 64, 2018, doi: 10.36309/goi.v24i1.93.

[11] M. F. Amin, S. R. Akbar, and E. R. Widasari, "Rancang Bangun Sistem Sortir Buah Apel Menggunakan Sensor Warna Dan Sensor Suhu," *J. Pengemb. Teknol. Inf. dan Ilmu Komput. Univ. Brawijaya*, vol. 1, no. 3, pp. 236–240, 2017.

[12] F. S. Hadisantoso, S. Indriyani, and Janizal, "Kinematics Analysis Pick And Place Materials By Color With Blockly Program," *Ramatekno*, vol. 1, no. 2, pp. 0–1, 2021, doi: 10.61713/jrt.v1i2.17.

[13] L. Shenzhen Yuejiang Technology Co., "Dobot Magician User Manual," vol. 1.2.5, pp. 1–135, 2018, [Online]. Available: www.dobot.cc

[14] "Mega 2560 Rev3 | Arduino Documentation." Accessed: Oct. 24, 2024. [Online]. Available: https://docs.arduino.cc/hardware/mega-2560/

[15] I. Habib Satria, "APLIKASI ARDUINO DAN SENSOR," *Insight Mediat.*, 2023, Accessed: Oct. 24, 2024. [Online]. Available: https://repository.insightmediatama.co.id/books/article/view/41

[16] "The Python Tutorial — Python 3.13.0 documentation." Accessed: Oct. 24, 2024. [Online]. Available: https://docs.python.org/3/tutorial/index.html

[17] A. Suharto, "Fundamental Bahasa Pemrograman Python," 2023, Accessed: Oct. 24, 2024. [Online]. Available: https://repository.penerbiteureka.com/id/publications/560015/fundamental-bahasa-pemrograman-python

[18] "OpenCV: Writing documentation for OpenCV." Accessed: Oct. 24, 2024. [Online]. Available: https://docs.opencv.org/4.x/d4/db1/tutorial_documentation.html

[19] S. Manual and F. Amplifier, "E3X-Na," pp. 1–24.

[20] A. D. Goenawan, M. B. A. Rachman, and M. P. Pulungan, "Identifikasi Warna Pada Objek Citra Digital Secara Real Time Menggunakan Pengolahan Model Warna HSV," no. April, pp. 279–289, 2022.