Jurnal J-Innovation Vol.14, No.2, Desember 2025
P-ISSN 2338-2082
E-ISSN 2808-5620

Implementation of the K-Nearest Neighbor Algorithm on a 4-DoF Manipulator Robot for Color-Based Object Retrieval

Rendyansyah[1], Irmawan[2], Hera Hikmarika[3], and Caroline[4]
[1234] Department of Electrical Engineering, Faculty of Engineering, Universitas Sriwijaya
Jl. Palembang-Prabumulih, Km. 32, Indralaya, Ogan Ilir, Sumatera Selatan
[1]rendyansyah.unsri@gmail.com

ABSTRACT

This study discusses implementing the K-Nearest Neighbor (K-NN) algorithm on a 4-DoF manipulator robot to pick up objects based on color. The main objective of this study is to design and test a robot control system capable of automatically recognizing, picking up, and moving objects according to color classification. The color detection process uses a camera to extract color component values in RGB space, which are then classified using the K-NN algorithm with k = 5. The classification results serve as the basis for the manipulator robot's motion, which is controlled via cubic trajectory-based planning, ensuring smooth, coordinated motion of each servo motor joint. We conducted 30 experimental trials involving red, green, and blue objects, and the system demonstrated a 96% success rate in both color classification and object retrieval. Minor failures occurred due to lighting variations that affected color detection results. Overall, our results demonstrate that integrating the K-NN algorithm with cubic trajectory planning enhances the performance of a manipulator robot in color-based object recognition and retrieval tasks and supports its potential deployment in computer-vision-driven industrial automation systems.

Keywords: *Color recognition, K-Nearest Neighbor, Manipulator robot*

.

ABSTRAK

Penelitian ini membahas implementasi algoritma K-Nearest Neighbor (K-NN) pada robot manipulator 4-DoF untuk melakukan pengambilan objek berdasarkan warna. Tujuan utama penelitian ini adalah merancang dan menguji sistem pengendalian robot yang mampu mengenali, mengambil, serta memindahkan objek sesuai dengan klasifikasi warna secara otomatis. Proses deteksi warna dilakukan menggunakan kamera yang mengekstraksi nilai komponen warna dalam ruang RGB, kemudian diklasifikasikan menggunakan algoritma K-NN dengan parameter k = 5. Hasil klasifikasi tersebut menjadi dasar pergerakan robot manipulator yang dikendalikan melalui trajectory planning berbasis cubic trajectory, sehingga gerakan setiap sendi motor servo berlangsung halus dan terkoordinasi. Pengujian dilakukan sebanyak 30 kali percobaan dengan tiga kategori warna objek, yaitu merah, hijau, dan biru. Hasil eksperimen menunjukkan bahwa sistem mampu mengklasifikasikan warna dan melakukan pengambilan objek dengan tingkat keberhasilan mencapai 96%. Kegagalan minor terjadi akibat variasi pencahayaan yang memengaruhi hasil deteksi warna. Secara keseluruhan, hasil penelitian ini menunjukkan bahwa integrasi antara algoritma K-NN dan cubic trajectory efektif dalam meningkatkan kinerja robot manipulator untuk tugas pengenalan dan pengambilan objek berbasis warna, serta berpotensi diterapkan dalam sistem otomasi industri berbasis visi komputer.

Kata kunci: K-Nearest Neighbor, Pengenalan warna, Robot manipulator

I. INTRODUCTION

The development of robotics technology has significantly impacted various sectors of modern industry. Robots replace human labor in repetitive production processes, improving efficiency, accuracy, and safety. In the era of Industry 4.0, robotics plays a vital role in realizing automated production systems integrated with artificial intelligence and the Internet of Things [1],[2]. Robots' ability to perform sensing, data processing, and autonomous decision-making makes this technology a significant pillar in the digital transformation of the manufacturing, logistics, and service industries.

One type of robot widely used in industry is the manipulator robot [3]. A manipulator robot is a mechanical system with several degrees of freedom that can perform movements similar to those of a human arm [4]. This type of robot is generally used in tasks such as welding [5], assembly, material transfer, and product packaging [6]. The manipulator's ability to achieve precise positions and orientations makes it a vital component in automated production lines. However, for manipulator robots to operate more adaptively in their environment, an intelligent control system is needed to recognize objects and determine appropriate actions [7],[8].

In recent years, the application of intelligent systems such as machine learning and computer vision has become a trend in robotics development, particularly in robot manipulators. The integration of these methods has resulted in robots that can recognize objects based on visual characteristics such as shape, size, and color [9],[10]. Robots operate mechanically with this pattern recognition capability and make decisions independently based on object classification or detection results. Researchers have used various machine learning algorithms for this purpose, including Artificial Neural Networks (ANNs), Support Vector Machines (SVMs), and K-Nearest Neighbors (K-NNs).

The K-Nearest Neighbor (K-NN) method is a simple yet effective supervised learning algorithm for classification tasks, including color and visual pattern recognition [11]. K-NN works by measuring the proximity of a test data point to several training data points (neighbors) within the feature space. In robotics, researchers use this method to classify objects based on sensor data or camera images. K-NN offers advantages such as easy implementation, no complex training, and the ability to deliver accurate results when the training dataset is sufficient and the K parameter is appropriately selected.

Based on this concept, this study implements the K-NN algorithm on a 4-DoF manipulator robot to identify and pick up objects based on color. This system enables the robot to detect object colors using a camera, classify them using K-NN, and move the manipulator to pick up and relocate the objects based on their color categories. Thus, this system is an object-picking tool and a model for applying a machine-learning-based visual recognition system in robot manipulator control.

This research aims to design, implement, and test a control system for a 4-DoF manipulator robot capable of recognizing and picking up objects by color using the K-NN method. The benefits of this research include contributing to the development of intelligent robotics systems in the object classification process and supporting the implementation of artificial intelligence-based technology in industrial automation systems. The theoretical basis used in this research includes robot manipulator kinematics, pattern recognition using the K-NN algorithm, and digital image processing for color feature extraction.

II. RESEARCH METHOD

A manipulator robot has several degrees of freedom (DoF) that enable movement on various axes to achieve specific positions and orientations in the workspace [12]. In this study, we designed and implemented a four-degree-of-freedom (4-DoF) manipulator robot for object-picking tasks. This manipulator system uses four servo motors to actuate each joint and equips the end effector with a gripper that clamps and moves objects. A computer programmed with the K-NN algorithm and an integrated trajectory planning system controls the robot [13],[14], ensuring that its movements coordinate with the results of the object color classification. An Arduino microcontroller serves as a slave controller, executing commands from the computer by adjusting the rotation angle of each servo motor according to the K-NN-derived instructions. Figure 1 shows the physical form of the 4-DoF manipulator robot used to pick up objects based on color.

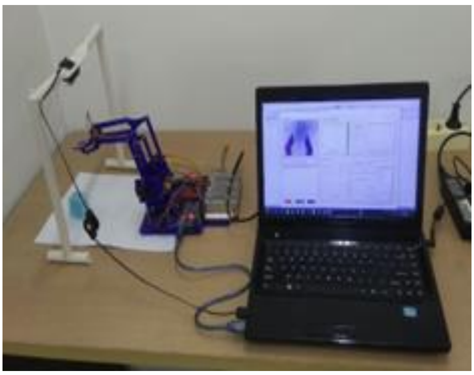


Figure 1. 4-DoF robot manipulator.

The instrument shown in Figure 1 consists of several main components: a computer, an Arduino microcontroller, a camera mounted on a support pole, and a 4-DoF manipulator robot. The computer is a control center that runs a K-NN algorithm-based program and a trajectory planning system to coordinate the manipulator's movements. The camera and Arduino microcontroller connect to the computer via USB for serial communication. The camera is a visual sensor that detects objects in the work area and sends the captured images to the computer for further processing. The image processing stage extracts color components in the RGB color space and uses them as input parameters for the K-NN classification system. The test objects used are small cubes with three color variations, red, green, and blue, each representing a different class in the classification process. The results of this color recognition become the basis for the computer to determine the manipulator's movement commands in picking up objects according to their color categories.

This study uses the K-NN algorithm, implemented in a computer program as the central control system, to perform color classification. The K-NN algorithm uses supervised learning, classifying test data based on their proximity to labeled training data. In this study, the training data consisted of color values in the Red, Green, and Blue (RGB) format, obtained from red, green, and blue objects. When the camera captured a new object image, the system calculated the distance between the test data and each training data point to determine the closest color class. The degree of closeness between data is measured using the Euclidean distance as shown in Equation (1), which serves as the basis for determining the shortest distance between the color feature vector of the test object and the vector in the training dataset.

$$d(X,Y)=\sqrt{\begin{array}{l}(R_X-R_Y)^2\\+(G_X-G_Y)^2\\+(B_X-B_Y)^2\end{array}} \tag{1}$$

Where: d(X, Y) is the Euclidean distance between the test data and the training data, Rx, Gx, Bx are the color component values of the test data, and Ry, Gy, By are the color component values of the training data.

The results of this color classification are then used as a reference in decision-making to move the manipulator robot to the appropriate object position.

The stages of the K-NN algorithm in this study can be explained as follows [15],[16],[17]:

1. **Determining the k parameter**.
   The first step is to determine the k parameter, the number of nearest neighbors used in the classification process. This k value affects the classification accuracy.
2. **Calculating the Euclidean distance.**
   After setting the k parameter, the system calculates the distance between the test data and all training data using the Euclidean distance equation. A smaller distance indicates a higher degree of similarity between two data points.
3. **Sorting distances in ascending order.**
   The calculated distance values are sorted in ascending order (from smallest to largest) to determine the k training data with the shortest distance to the test data.
4. **Selection of nearest neighbors.**
   The sorting results show that k training data with the shortest distance are the nearest neighbors. Each training data point has a specific label or category used in the classification process.
5. **Majority voting.**
   The system performs the final classification using the majority voting principle, assigning the test data to the class that appears most frequently among the k nearest neighbors.
6. **Prediction of classification results.**
   The voting results determine the final class assigned to the test data. In this study, the classification results identify the object's color (red, green, or blue), and the system then uses this information to control the manipulator robot's movements during the object retrieval process.

After the K-NN algorithm completes the color classification process, the system uses a trajectory planning algorithm to control the manipulator robot's movement path for each joint. The trajectory planning method used in this study is a third-order polynomial, also known as a cubic trajectory. The mathematical model of the cubic trajectory is shown in Equation (2), which represents the relationship between the position, velocity, and acceleration of the robot joint over time, from the initial position q_s to the final position q_f in time t_f.

$$q(t) = q_s + 3\left(\frac{q_f - q_s}{t_f^2}\right)t^2 - 2\left(\frac{q_f - q_s}{t_f^3}\right)t^3 \qquad (2)$$

We chose this approach because it produces smooth, continuous trajectories in position and velocity, making it suitable for motion control applications in manipulator robots. The system applies this trajectory throughout the entire robot work process, from picking up the object to moving it to the target position and placing it according to the predetermined color classification results. Thus, integrating K-NN and cubic trajectory enables the manipulator robot to operate well in color-based object pickup tasks.

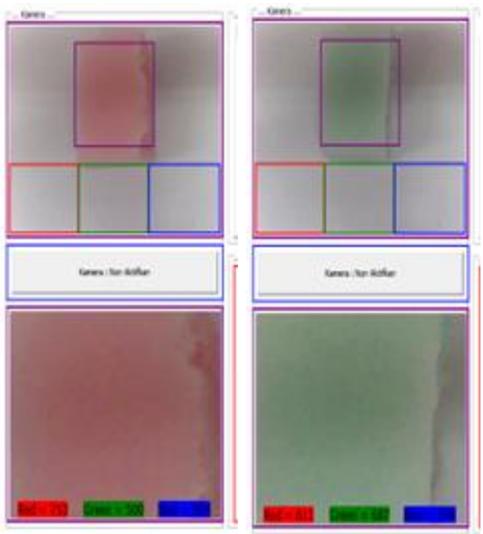
III. RESULTS AND DISCUSSION

Experiments on the 4-DoF manipulator robot were conducted at the Control Engineering and Robotics Laboratory, Sriwijaya University, using the experimental apparatus shown in Figure 1. In this experiment, the camera on the support pole functioned as a vision sensor used to detect objects based on color differences. Each detected object had its Red, Green, and Blue (RGB) color component values extracted, which were then normalized according to Equation (3) to ensure consistency of color intensity values against changes in lighting. The test objects consisted of three primary colors, red, green, and blue, as shown in Figure 2.

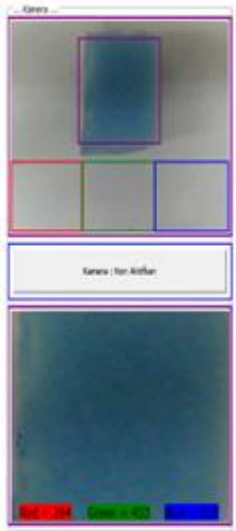
$$\begin{aligned} Red &= \frac{R}{R+G+B} \\ Green &= \frac{G}{R+G+B} \\ Blue &= \frac{B}{R+G+B} \end{aligned} \qquad (3)$$

Figure 2. Colored objects.

The image acquisition process in this study used a capture size of 120 × 120 pixels to detect objects based on color differences. Figure 3 shows how the camera captures and processes the scene, producing color-detection results that identify three object types: red, green, and blue. The system collected five datasets for each object color, capturing RGB value variations influenced by lighting changes and object positioning as part of the supervised learning process in the K-NN algorithm. Table 1 presents the extracted RGB values and the dataset used to train and test the color classification system for the 4-DoF manipulator robot.



(a) (b)



(c)

Figure 3. Object detection, (a) red, (b) green, and (c) blue.

Table 1. Object color classification dataset.

No	Value			Label	
	R	G	B		
1	0.43	0.3	0.28	1	Red
2	0.43	0.28	0.29	1	
3	0.43	0.29	0.27	1	
4	0.44	0.28	0.27	1	
5	0.43	0.29	0.28	1	
6	0.33	0.35	0.32	2	Green
7	0.34	0.34	0.32	2	
8	0.31	0.39	0.3	2	
9	0.32	0.37	0.31	2	
10	0.32	0.36	0.32	2	
11	0.27	0.34	0.39	3	Blue
12	0.29	0.32	0.39	3	
13	0.24	0.35	0.42	3	
14	0.28	0.33	0.39	3	
15	0.3	0.32	0.38	3	

Jurnal J-Innovation Vol.14, No.2, Desember 2025
P-ISSN 2338-2082
E-ISSN 2808-5620

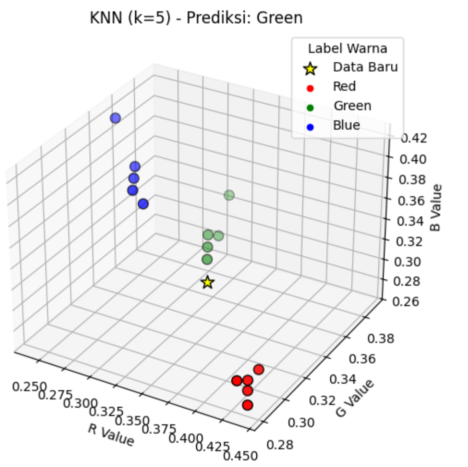


Figure 4. Green object detection experiment.

This study uses the K-NN algorithm with k = 5, assigning each test sample to its 5 nearest training points. The object color detection system was tested ten times for each color, namely red, green, and blue, with random data collection to test the consistency and reliability of the classification method. The results showed that the K-NN algorithm could identify color classes with a reasonable accuracy rate of 96%, in accordance with the actual color categories based on the proximity of the data in the RGB feature space. As an illustration, Figure 4 shows an example of the classification results where the system accurately recognized green objects. The system also employs 3D graphical visualization to illustrate the spatial relationship between test samples and training data, enabling more precise identification of the five nearest neighbors (k = 5) for each test point.

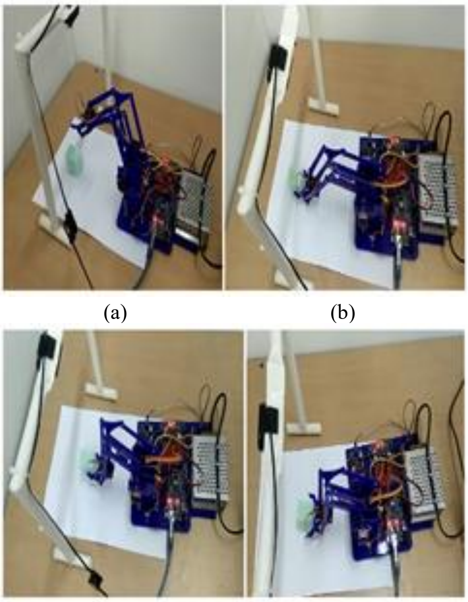
Tabel 2. Robot trajectory from the initial position to the final position.

Track	Trajectory
Track 1 (Red Object)	1. Robot moves toward the object 2. Robot picks up the object 3. Robot moves the object 4. Robot places object in red box
Track 2 (Green Object)	1. Robot moves toward the object 2. Robot picks up the object 3. Robot moves the object 4. Robot places object in green box
Track 1 (Blue Object)	1. Robot moves toward the object 2. Robot picks up the object 3. Robot moves the object 4. Robot places object in blue box

The next stage in this research is to test the object retrieval system using a 4-DoF manipulator robot based on the color classification results obtained from the K-NN algorithm. Once the system identifies the object's color, the robot autonomously approaches the object, grasps it, transfers it, and places it using a cubic-trajectory planning method, with each movement governed by stored joint-angle data for each servo motor. The test was conducted 10 times for each object color (red, green, and blue) for 30 trials. Table 2 presents the trajectory details for each object color, describing how the robot moves from the initial position to the final position.

Each trajectory, whether for the red, green, or blue object, follows the same sequence of movements: the robot moves toward the object, grasps it, transfers it, and places it in the box corresponding to its color. The Cartesian coordinates for each track have been predetermined and stored in the program memory. At the same time, the system applies the cubic trajectory formula to control the end-effector position along the x, y, and z axes. The system uses the cubic trajectory method to calculate the position, velocity, and acceleration of each joint throughout each stage of the movement, ensuring that the initial and final positions of every joint are achieved smoothly and accurately. For instance, when the robot moves from its initial position toward the red object (Track 1), the trajectory equations ensure the gripper moves smoothly, maintains object stability, and allows a smooth transition to the grasping stage. The same approach applies to Track 2 (green object) and Track 3 (blue object), allowing the entire manipulator motion sequence to be executed consistently and synchronously across all joints.

Figure 5 shows the robot experiment in picking up and moving objects based on color classification. Based on the test results, the robot system accurately picked up and moved objects using K-NN color classification, achieving a 96% success rate. A minor failure occurred in only one experiment, due to lighting interference that affected the camera's color readings, thereby increasing classification error.



(a) (b)

(c) (d)

Figure 5. Movement of the manipulator robot in picking up and moving objects, (a) moving toward the object, (b) picking up, (c) moving, and (d) placing the object on the colored box.

IV. CONCLUSIONS

The experimental results confirm that the K-NN algorithm operates effectively on the 4-DoF manipulator robot for performing color-based object pickup tasks. The developed system can detect, classify, and control robot movements in an integrated manner between the computer, camera, and Arduino microcontroller. The color classification results using the K-NN method with a parameter value of k = 5 showed an accuracy rate of 96%, indicating that this method can recognize color differences in objects in the RGB feature space. Integrating the K-NN algorithm with cubic trajectory-based planning enables the robot to move smoothly during object-picking and moving tasks guided by color classification results. The system's overall performance indicates that this approach can serve as an initial model for a machine-learning-based visual recognition system in industrial manipulator robot control.

Future work can enhance the system by implementing a more stable color space, such as HSV, to reduce lighting variability and by adding more datasets to improve classification accuracy. In addition, using an optimal trajectory planning method and advanced learning algorithms, such as Deep Learning, can be an alternative to enhance robot manipulators' performance and adaptive capabilities.

REFERENCES

[1] M. B. Palungan and Y. S. Kendek, “Pengaruh Implementasi Sistem Robotik Berbasis Kecerdasan Buatan terhadap Kualitas Produk Pemesinan CNC,” *J. Rekayasa Mater. Manufaktur dan Energi*, vol. 8, no. 2, pp. 411–422, 2025.

[2] A. Chairany, R. Buaton, and R. Puspadini, “Penerapan Internet of Thingspada Mekanik Prototipe Robot Pengaduk Gabah,” *Repeater Publ. Tek. Inform. dan Jar.*, vol. 3, no. 3, pp. 89–101, 2025.

[3] M. Bartoš, V. Bulej, M. Bohušík, J. Stanček, V. Ivanov, and P. Macek, “An overview of robot applications in automotive industry,” in *14th International scientific conference on sustainable, modern and safe transport*, 2021, pp. 837–844.

[4] H. N. Halimah, R. Maulana, and R. Maulana, “Kontrol Robot Manipulator Berdasarkan Pergerakan Lengan Manusia Menggunakan Electromyography,” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 4, no. 6, pp. 1866–1874, 2020.

[5] A. Saepullah, I. N. Budiarsa, and I. M.

Astika, "Penerapan Electrode Tape Mengunakan Sistem Gun Spot Welding Pada End Effector Robot Arm," *J. Ilm. Tek. DESAIN Mek.*, vol. 14, no. 3, pp. 214 – 218, 2025.

[6] A. Imran, Firdaus, and M. R. Pakondo, "Prototype Robot Lengan Pemindah Barang Pada Conveyor Secara Otomatis Berbasis Arduino Uno," *J. MEDIA Elektr.*, vol. 20, no. 3, pp. 112–118, 2023.

[7] A. N. Lakapu, A. H. Ginting, and H. J. Djahi, "Pengontrolan Lengan Robot Berdasarkan Jarak Objek," *JTekEL J. Tek. Elektro*, vol. 2, no. 1, pp. 58–65, 2025.

[8] F. M. T. Huda, Y. A. R. Pratama, F. R. Saputra, R. Hadiazzaka, and A. S. Priambodo, "Penerapan Kinematika Terbalik Pada Robot Lengan Lima Sendi (5 Dof) Dengan Citra Digital," *JITET (Jurnal Inform. dan Tek. Elektro Ter.*, vol. 13, no. 1, pp. 231–242, 2025.

[9] M. R. V. Aditya, N. L. Husni, D. A. Pratama, and A. S. Handayani, "Penerapan Sistem Pengolahan Citra Digital Pendeteksi Warna pada Starbot," *J. Tek.*, vol. 14, no. 2, pp. 185–191, 2020.

[10] I. Sulistiyowati, H. M. Ichsan, and I. Anshory, "Konveyor Penyortir Objek Dengan Deteksi Warna Menggunakan Kamera Esp-32 Berbasis Open-CV Python," in *Seminar Nasional Rekayasa, Sains dan Teknologi*, 2024, pp. 35–41.

[11] J. C. G. Sogen, A. N. Weking, and B. Deta, "Klasifikasi Jenis Mangrove Berdasarkan Bentuk Daun Menggunakan AlgoritmaK-Nearest Neighbor," *J. Artif. Intell. Digit. Bus.*, vol. 4, no. 3, pp. 1375–1380, 2025.

[12] I. H. Ayega, T. A. Tamba, and B. M. Arthaya, "Rancang Bangun Purwarupa Manipulator Lengan Robot Dengan Tiga Derajat Kebebasan," *ELKOMIKA J. Tek. Energi Elektr. Tek. Telekomun. Tek. Elektron.*, vol. 11, no. 3, pp. 796–810, 2023.

[13] A. P. P. Prasetyo, Rendyansyah, and K. Exaudi, "Implementasi Trajectory Planning pada Robot Manipulator 4 DOF untuk Mencari Kebocoran Gas," *J. J-Innovation*, vol. 6, no. 2, pp. 1–8, 2017.

[14] Rendyansyah, A. P. P. Prasetyo, K. Exaudi, and S. Sembiring, "Application of Learning Vector Quantization and Trajectory Planning On a 4-DoF Robotic Arm to Move the Object," *J. Ecotipe (Electronic, Control. Telecommun. Information, Power Eng.*, vol. 10, no. 2, pp. 268–276, 2023.

[15] A. Yudhana, Sunardi, and A. J. S. Hartanta, "Algoritma K-NN Dengan Euclidean Distance Untuk Prediksi Hasil Penggergajian Kayu Sengon," *Transm. J. Ilm. Tek. Elektro*, vol. 22, no. 4, pp. 123–129, 2020.

[16] F. T. Admojo and Ahsanawati, "Klasifikasi Aroma Alkohol Menggunakan Metode KNN," *Indones. J. Data Sci.*, vol. 1, no. 2, pp. 34–38, 2020.

[17] A. N. Hilmi, E. Y. Puspaningrum, and H. E. Wahanani, "Implementasi Algoritma K-Nearest Neighbor (KNN) untuk Identifikasi Penyakit pada Tanaman Jeruk Berdasarkan Citra Daun," *Router J. Tek. Inform. dan Terap.*, vol. 2, no. 2, pp. 107–117, 2024.