

Article

Intelligent Lighting System Using Color-Based Image Processing for Object Detection in Robotic Handling Applications

Uğur Akış¹ and Serkan Dişlitaş^{2,*} ¹ Department of Energy Systems Engineering, Hitit University, 19030 Çorum, Türkiye² Department of Computer Engineering, Hitit University, 19030 Çorum, Türkiye

* Correspondence: serkandislitas@hitit.edu.tr

Abstract: In applications reliant on image processing, the management of lighting holds significance for both precise object detection and efficient energy utilization. Conventionally, lighting control involves manual switching, timed activation or automated adjustment based on illuminance sensor readings. This research introduces an embedded system employing image processing methodologies for intelligent ambient lighting, focusing specifically on reference-color-based illumination for object detection and positioning within robotic handling scenarios. Evaluating the system's efficacy entails analyzing the illuminance levels and power consumption through a tailored experimental setup. To minimize illuminance, the LED-based lighting system, controlled via pulse-width modulation (PWM), is calibrated using predetermined red, green, blue and yellow (RGBY) reference objects, obviating the need for external sensors. Experimental findings underscore the significance of color choice in detection accuracy, highlighting yellow as the optimal color requiring minimal illumination. Successful object detection based on color is demonstrated at an illuminance level of approximately 50 lx, accompanied by energy savings contingent upon ambient lighting conditions.

Keywords: intelligent lighting; image processing; color-based object detection; robotics



Citation: Akış, U.; Dişlitaş, S. Intelligent Lighting System Using Color-Based Image Processing for Object Detection in Robotic Handling Applications. *Appl. Sci.* **2024**, *14*, 3002. <https://doi.org/10.3390/app14073002>

Academic Editors: Daynier Rolando Delgado Sobrino and Rafał Gołębski

Received: 28 February 2024

Revised: 24 March 2024

Accepted: 1 April 2024

Published: 3 April 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The efficient use and saving of electrical energy are as important as its generation [1]. Natural and artificial light sources are used in lighting, one of the most important usage areas of electrical energy. Daylighting is a quality light source and has the best color rendering ability. In order to save energy, it is necessary to make maximum use of daylight and to use intelligent lighting systems in this context [2]. There are intelligent lighting systems with commercial, energy-efficient and advanced features for sectoral needs [3]. Commercial intelligent lighting systems are systems with functional features, such as on/off, dimming, monitoring and programmability. Energy-saving intelligent lighting systems are systems, which consume less electrical energy, usually by working with the help of some sensors within an algorithm and software. In advanced intelligent lighting systems, in addition to energy saving, light quality control can be achieved by adjusting features such as light intensity, direction, angle and distance with various algorithms and artificial-intelligence-based applications [4–9].

Image-processing-based robotic systems are widely used in industrial areas for fast and accurate detection, tracking and classification of objects for handling operations. With a general definition, image processing is a technology, which enables some operations, such as improvement, simplification, analysis and inference, with the help of various algorithms by transferring a real image to digital media [10]. The classification of objects based on image processing can be performed using shape-, motion-, color- and texture-based methods [11]. Color-based classification methods are widely used because of the high processing speed and success rate [12]. Determining the position of an object detected in image processing algorithms is based on the principle of finding the centroid [13].

Change in lighting conditions is an important issue in color-based image processing applications [14]. Depending on poor ambient lighting, there are also factors, which negatively affect the perception of objects, such as shadows and glare. Weak light causes shadows, and strong light causes glare and discoloration of the object as a result of too much reflection. In addition, application of light from a single point will increase the shadow effect; therefore, it is preferable to apply it from different points. When an object receives a sufficient amount of light in illumination, it will appear in its original color because it will reflect sufficiently [15]. In this respect, ambient lighting, the direction and amount of light are important issues in terms of image quality, accurate detection of objects and energy use [16,17]. In order to obtain successful results and save energy in image processing, a minimum ambient light level must be provided. Light emitting diode (LED) lamps are widely used in lighting applications due to their advantages, such as long life, energy efficiency and fast switching [18]. In addition, LEDs are an ideal light source because controls such as light intensity can be easily performed in image-processing-based applications [19,20].

In traditional methods, ambient lighting is achieved by continuously turning on the light source, manually turning it on and off as needed, using a timer, dimmer or automatically measuring the illuminance level with the help of various sensors [16,21–25]. According to the literature research, it is observed that industrial applications based on image processing for lighting and robotic systems are rapidly becoming widespread. In Ref. [26], a design method was proposed for the positioning of light sources in order to obtain good illumination in the scene viewed with an external light source, and a study was carried out to support this suggestion with simulation results. In the study carried out in Ref. [5], the authors designed an intelligent lighting system and enabled the visualization of the change in the illuminance level with the help of real-time data. In Ref. [27], the authors discussed the optimization of lighting conditions, camera height and colors by using parameters such as light type, light intensity, light source height and camera height in an adjustable way in citrus image processing. In Ref. [24], a study was conducted on the design and implementation of a low-cost fluorescent lamp lighting system for the autonomous robot workspace area. In Ref. [16], a study was conducted on automatic-lighting-method-based images' brightness quality analysis to achieve automatic control of illumination in the motion of a robot under dark conditions. In Ref. [28], a mobile robot was designed for real-time tracking of objects of different colors by using image processing techniques based on the principle of finding the center of gravity, and its properties were examined. In Ref. [29], in a Raspberry-Pi-based robot system designed for object tracking, the process of tracking the object—whose real position is determined by finding the centroid with color-based image processing using the open-source computer vision library (OpenCV)—was carried out. In a study on color-based object tracking based on moment calculation with image processing [14], simulation was performed with MATLAB/Simulink R2016a software, and tracking of the red ball was achieved with a six-axis industrial robot. In Ref. [30], a study was conducted to follow a 2D hexagonal object with a fuzzy logic controller robot using image processing.

Within this investigation, an embedded system is devised, leveraging image processing methodologies to facilitate intelligent ambient lighting anchored in reference-color principles. This system is tailored for precise object detection and positioning within the realm of robotic handling applications. The evaluation of system performance revolves around scrutinizing the illuminance levels and power consumption parameters within a meticulously designed experimental framework. To achieve optimal illuminance levels without using any sensors, the LED-based lighting system's power through PWM is adjusted against red, green, blue and yellow (RGBY) objects with known positioning—an approach divergent from traditional methodologies.

2. The Designed System

2.1. Overview of the System

The general view of the designed system is shown in Figure 1. In this designed three-axis cartesian robot system, reference-color-based intelligent lighting is performed in order to detect and position the objects on the platform with image processing. The designed system is portable and can be used in a way, which does not directly expose an external light source, which may cause reflection and glare. The general specifications of the system are given in Table 1.

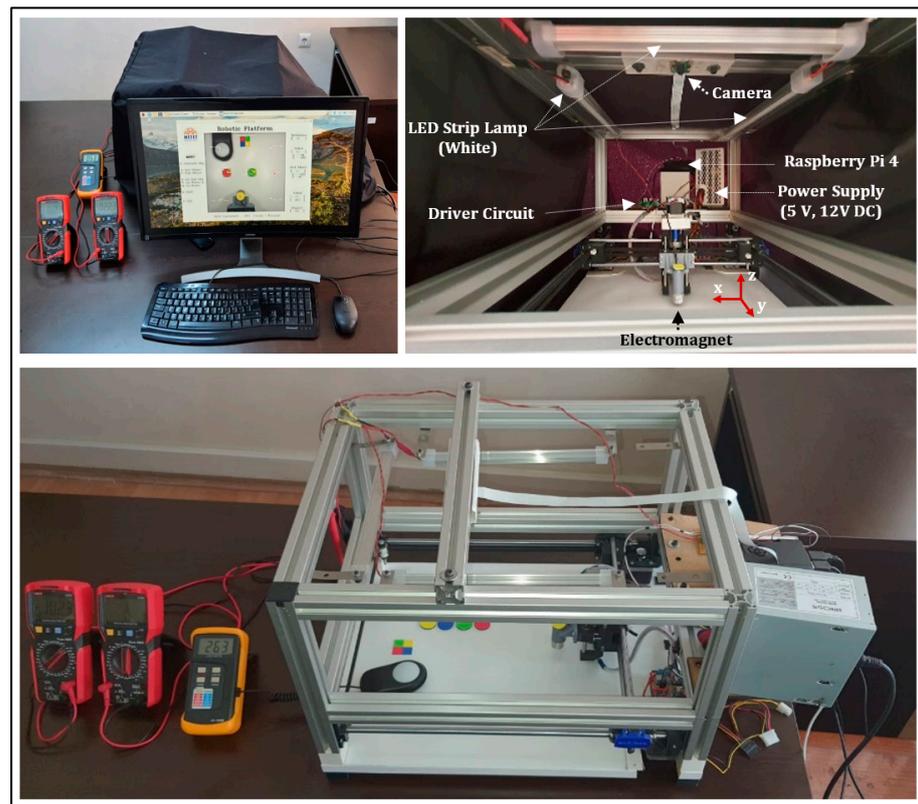


Figure 1. General view of the system.

Table 1. General characteristics of the designed system.

Characteristics	Properties
Application area	Object detection, positioning and classification
Working method	Color-based image processing with intelligent lighting
User interface	GUI (Raspberry Pi/Python)
Color definitions	Reference : Red, Green, Blue, Yellow (RGBY)/square (1.5 cm × 1.5 cm)
	Robot : Yellow
	Objects : Green (Good), Red (Bad)
Robot	Three-axis cartesian robot
Robot controller	Arduino Uno (GRBL firmware/G-Code) and CNC Shield
Platform lighting	12 V DC 780 mW white LED strip lamp (adjustable with PWM)
Gripper design	On/off controlled electromagnet (12 V DC, 3 kg)
Platform dimensions	50 cm × 40 cm × 40 cm

2.2. System Hardware

The general block diagram of the designed system is shown in Figure 2. The system basically consists of a data acquisition and control unit, robotic platform, driver unit, user interface and power supplies.

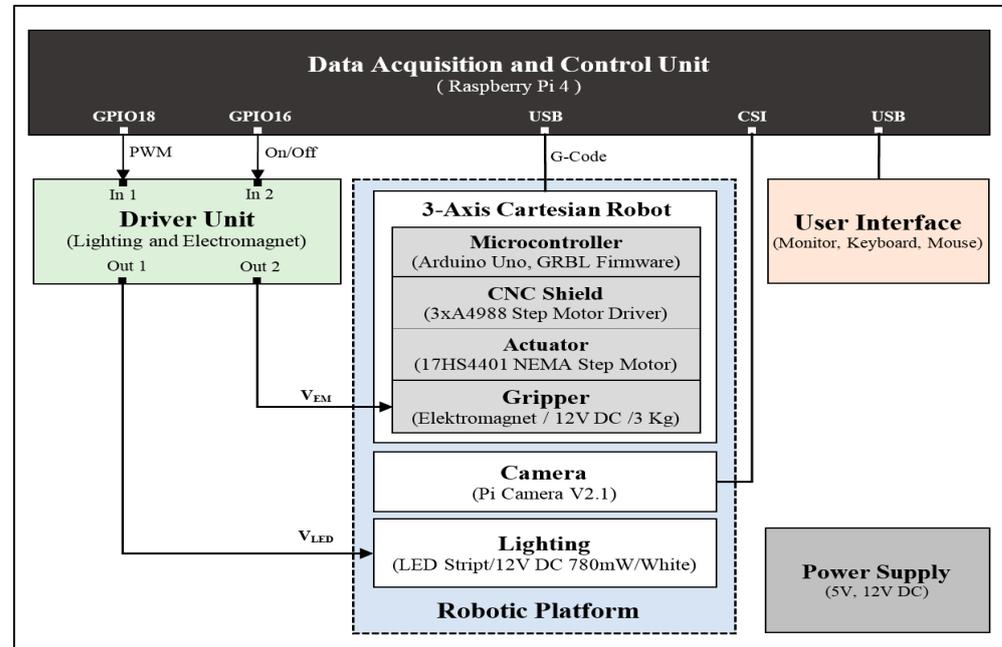


Figure 2. General block diagram of the designed system.

A Raspberry Pi 4 mini-computer development board is used as a data acquisition and control unit in the designed robot cell. In this unit, in addition to color-based image processing, three-axis robot handling with G-Codes (RS-274, Geometric Code) and adjustment of the ambient light level using PWM signal are provided. The Raspberry Pi development board has a 1.5 GHz processor frequency, a 2 GB RAM memory, 40 general-purpose pins, hardware PWM outputs and a camera serial interface (CSI) connection. Step motor, motor driver and belt set are used for the three-axis movement of the robot in the robotic platform. In the system, G-Code programming language instructions are generated depending on the position information for the three-axis movement of the computer numerical control (CNC) robot. In order to compile G-Code instructions, the free open-source GRBL firmware installed on the Arduino Uno board is used. With the help of a GRBL-compatible CNC Shield driver card, the motion direction and speed of the stepper motors are determined, and the robot is positioned on the three axes [31–33].

The robot's motion area is determined by creating a platform with an area of 30 cm × 30 cm at the base of the robot cell. In addition, an electromagnet placed on the Z-axis of the robot is used for handling operations in the system. The cylindrical ($r = 2$ cm, $h = 2$ cm) electromagnet, which has the feature of attracting metals with the magnetic field created due to the electric current passing through the coil wound on the core, works at 12 V DC and has a load-carrying capacity of 3 kg [34].

In order to take platform images in the system, an 8 Mp Pi Camera module connected to a Raspberry Pi 4 development board via a 15-pin CSI connector is used. The camera module is placed in the ceiling center of the robot cell at a height of 32 cm from the platform surface and with a bird's eye view from above. For the platform lighting, a white 12V 780 W DC strip LED with a diffuser, driven via adjustment of light intensity with the PWM signal, is used. The U-shaped LED strip lamp is fixed to the robot cell at a height of 28 cm from the platform surface, lighting the platform from the front, under the camera. For the base platform of the robot cell, a non-reflective white matte plastic background used in photography is used. In addition, for experimental studies, the robot cell is covered

with a black fabric, so that ambient lighting is not affected by environmental natural and artificial light sources.

In Figure 3, the block diagram of the closed-loop control system for adjusting the platform illuminance level is shown in the designed system. With the feedback, the electrical power level of the system is adjusted by making the necessary corrections depending on the error, and in this way, the illuminance level is obtained at the desired value. In the designed system, control is performed depending on the error rate between the RGBY reference object coordinates and the coordinates obtained as a result of image processing in determining the minimum illuminance level required for color-based image processing. The power of the lighting system is adjusted by reference to RGBY objects, whose actual centroid positions are known before, without the need for an external sensor or device in the system.

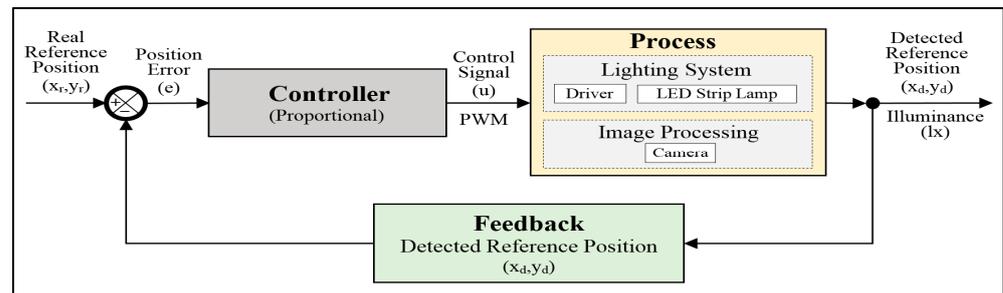


Figure 3. Closed-loop lighting control system.

In the designed robotic system, a MOSFET-based driver circuit controlled by Raspberry Pi is designed (Figure 4). With this driver circuit, the power required for the LED strip lamp and the electromagnet is provided. An electromagnet working at 12 V DC is used for the handling process in the system. With the on/off control for the electromagnet, objects can be held and released. In industrial applications, high-efficiency switching mode converters are preferred instead of linear regulators for LED drivers [35]. With the MOSFET switching in the driver circuit, the 5 V DC PWM signal is converted to the 12 V DC PWM signal required for the LED strip lamp.

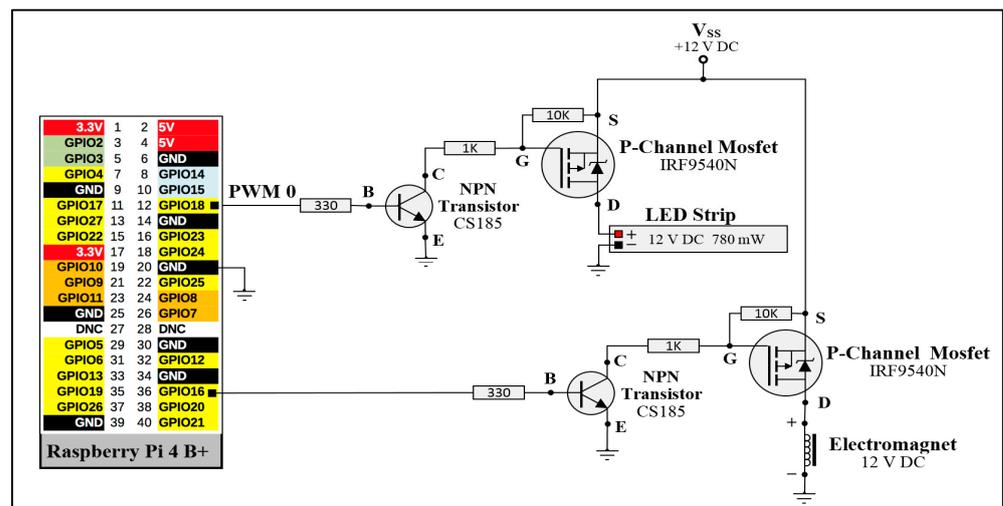


Figure 4. Raspberry-Pi-based LED strip lamp and electromagnet driver circuit.

In a lighting system where the PWM signal is used, the desired light intensity is adjusted by quickly switching the power provided for the LED lamp on and off. The basis of PWM is based on the principle of applying a square wave digital signal—whose pulse width is adjusted at a certain frequency—to the load. In this respect, signal frequency and

duty cycle are important parameters in the use of PWM. The duty cycle (D) is expressed as in Equation (1) by considering the high time of the pulse (T_{on}) and the low time of the pulse (T_{off}) in a digital signal [36,37]. By changing the current between 0 and I_{max} in proportion to the value, which D will take between 0 and 100 in the system, the electrical power applied to the LED strip lamp, and thus the lighting intensity, can be adjusted. Thus, the electrical power (P_{LED}) of the LED strip lamp depending on the D value of the PWM signal can be expressed in W as in Equation (2).

$$D = \frac{T_{on}}{T_{on} + T_{off}} 100\% \quad (1)$$

$$P_{LED} = V_{LED} \cdot I_{max} \frac{T_{on}}{T_{on} + T_{off}} \quad (2)$$

2.3. System Software

The designed system software basically consists of an interface and embedded system software, which provides image processing, lighting and three-axis cartesian motion. A Debian-based Raspberry Pi OS (Raspbian) operating system is installed on the Raspberry Pi 4 mini-computer, which is used as a data collection and control unit in the system. The interface software is coded in Python programming language, and OpenCV library is used for image processing, object detection and coordinate determination [13,38]. In order to process the G-Code instructions and position the three-axis robot in the system, an Arduino Uno micro-controller board—which works as a CNC controller and is loaded with GRBL firmware—is used [32,33].

A screenshot of the developed software for the robotic platform is shown in Figure 5. Here, the (X, Y) coordinate information is placed on the image taken with the camera with a resolution of 1024×768 . The platform coordinates are determined as follows: Origin X_0, Y_0 (0, 0), top left $-X, +Y$ ($-200, +200$), top right $+X, +Y$ ($+200, +200$), bottom left $-X, -Y$ ($-200, -200$) and lower right $+X, -Y$ ($+200, -200$). The Park (Home) and Waste area coordinates are also available on the platform. As seen in Table 2, RGBY reference objects, whose actual centroid positions on the platform are known beforehand, are used to adjust the illuminance level. Depending on the color-based processing of the obtained image, the detected objects are expressed as follows: Robot (Yellow), Good (Green) and Bad (Red). In addition, the electromagnet placed on the robot head in order to be able to handle objects has the ability to move in the vertical axis (Z).

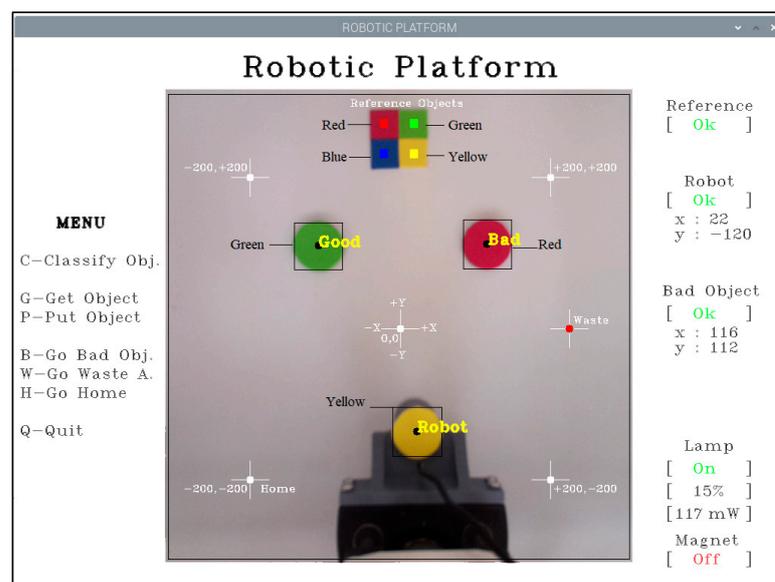


Figure 5. Robotic platform and X-Y-Z coordinates.

Table 2. RGBY reference objects and their actual centroid positions in the designed system.

Reference Object	Actual Centroid Position	
	X_r	Y_r
R—Red Object	−20	250
G—Green Object	20	250
B—Blue Object	−20	230
Y—Yellow Object	20	230

An interface software was developed to ensure that bad objects detected on the robotic platform are taken to the waste area. Figure 6 shows the flowchart for the operation of the interface software algorithm. In the developed algorithm, first, the required illuminance level is provided by creating controls for the RGBY reference objects, whose actual centroid positions on the robotic platform are known. For this process, the minimum illuminance level for image processing is adjusted by increasing the power of the LED strip lamp with the PWM signal. The control of lighting power is performed depending on the error between the actual centroid values of RGBY reference objects predefined in the software and the centroid values instantly detected and calculated based on image processing. With the correct finding of the reference centroids, robot and object detection is performed on the platform. When an object is found on the platform, the centroid is determined based on image processing, and the resulting G-Code instructions are sent to the CNC controller according to the working status. In this way, it is ensured that the robot moves to the centroid of the detected bad object. Within the scope of the handling process, the bad object is picked up with the help of an electromagnet and taken to the relevant place and left. If there is no bad object, the robot is allowed to go to the predetermined parking area and wait.

Figure 7 shows the flowchart of object detection and position determination based on reference-color-based image processing. In the algorithm, first of all, color selection, hue–saturation–value (HSV) color space lower and upper limit values and minimum area size settings for object detection are performed [39,40]. The HSV color space model lower and upper limit ranges for RGBY reference colors used in the designed system are shown in Table 3. After the initial settings, an image of the robotic platform is taken with the help of the camera connected to the system. Then, Gaussian blur filtering with the size of 11×11 is applied to remove details and noise from the image. In OpenCV, instead of the RGB color format, the blue–green–red (BGR) color format—in which the colors are arranged differently—is used [13]. However, since better results are obtained in the detection of colored objects in image processing applications, the image is converted from BGR color space to HSV color space [41]. Depending on the defined color and its range, masking is performed. In the morphological transform stage, the image is rearranged by performing edge erosion and dilation to remove noise from the image. In this stage, a 3×3 rectangular structuring element is used by applying the erosion and dilation operation twice, consecutively. A contour is created to increase the distinguishability of the detected colored regions. By entering into a loop, a frame is drawn for each contour, which provides the minimum area size for object detection, and then, the centroid is found with the help of moment calculations. In the last stage, the (X, Y) coordinate for the two axes is determined for the valid contours, and the reporting process is performed. Mathematically, the position determination for each contour is performed by calculating the zeroth and first moments of the coordinates (X, Y) in Equation (3) and the centroid in Equation (4) [42].

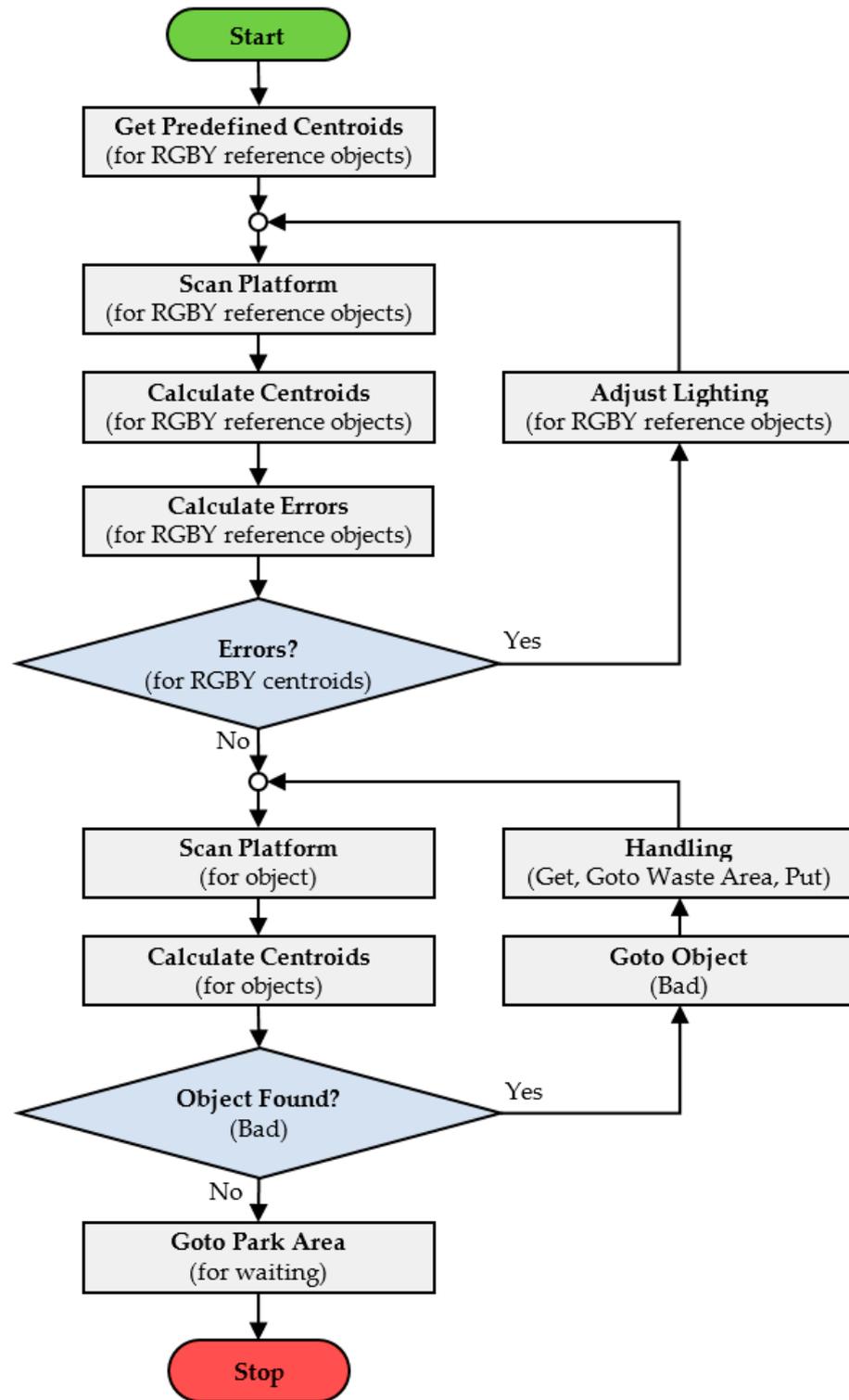


Figure 6. Flowchart for the robotic system process.

$$M_{00} = \sum_x \sum_y I(x, y) \quad M_{10} = \sum_x \sum_y x.I(x, y) \quad M_{01} = \sum_x \sum_y y.I(x, y) \quad (3)$$

$$x_c = \frac{M_{10}}{M_{00}} \quad y_c = \frac{M_{01}}{M_{00}} \quad (4)$$

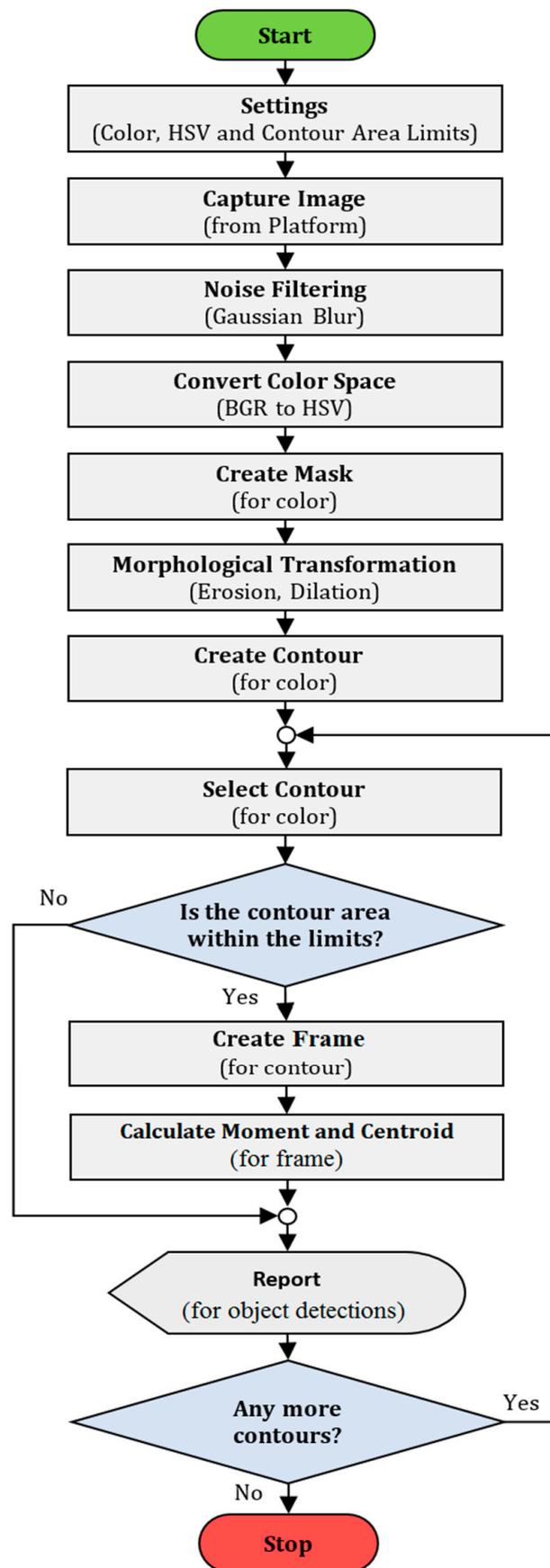


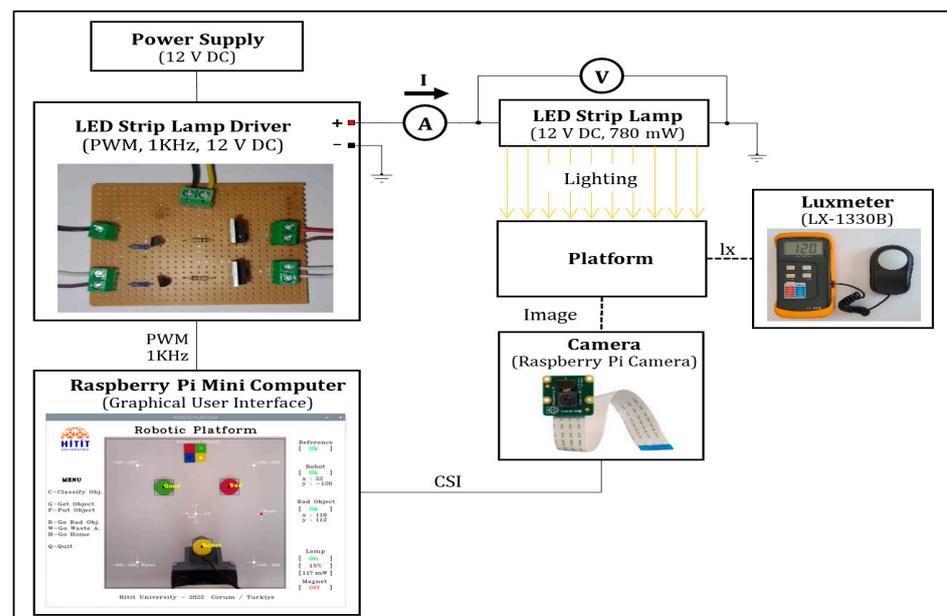
Figure 7. Flowchart of object detection and position determination based on reference-color-based image processing.

Table 3. Color ranges of the HSV color space model for RGBY reference colors in the designed system.

Reference Color	Lower Limit (H, S, V)	Upper Limit (H, S, V)
R—Red	(160, 100, 100)	(179, 255, 255)
G—Green	(38, 100, 100)	(75, 255, 255)
B—Blue	(75, 100, 100)	(130, 255, 255)
Y—Yellow	(22, 100, 100)	(38, 255, 255)

3. Results and Discussion

In order to determine the performance of the designed system, an experimental setup is created, in which current, voltage and illuminance measurements can be performed in the system (Figure 8). Here, the digital luxmeter is used to externally measure the illuminance level for the robotic platform. In addition, the power consumed by the LED strip lamp is determined by obtaining current and voltage measurements with a True RMS digital multimeter. In the experimental study, in order to obtain the colors properly, the illumination is performed with white light, and a low-reflection white background is used on the platform floor. Within the scope of the study, the system is covered with a black fabric cover in order to eliminate the effects of external environment, such as reflection and glare.

**Figure 8.** Experimental measurement setup.

In the experimental study, first, the switching performance of the driver circuit used to adjust the platform illuminance level is examined. For this purpose, the power change in the LED strip lamp driven by the MOSFET-based circuit depending on frequency and duty cycle change in the PWM signal is investigated. In the experimental setup, the power changes applied to the LED strip lamp depending on the duty cycle value for the PWM frequencies 100 Hz, 1 KHz, 5 KHz and 10 KHz are obtained (Figure 9). In the lighting system, in order to minimize flicker, the LED strip lamp is driven using the hardware PWM output on the Raspberry Pi GPIO18 output. In a PWM application, the flicker effect on LED lighting decreases as the frequency increases and cannot be perceived visually, as it exceeds 100 Hz. In this respect, the PWM frequency of the driver circuit in the designed system is determined as 1 KHz because it is closest to the ideal. In order to perform the switching properly depending on the duty cycle at higher frequencies in the system, it is necessary to drive a double MOSFET separately for the pulse, on and off.

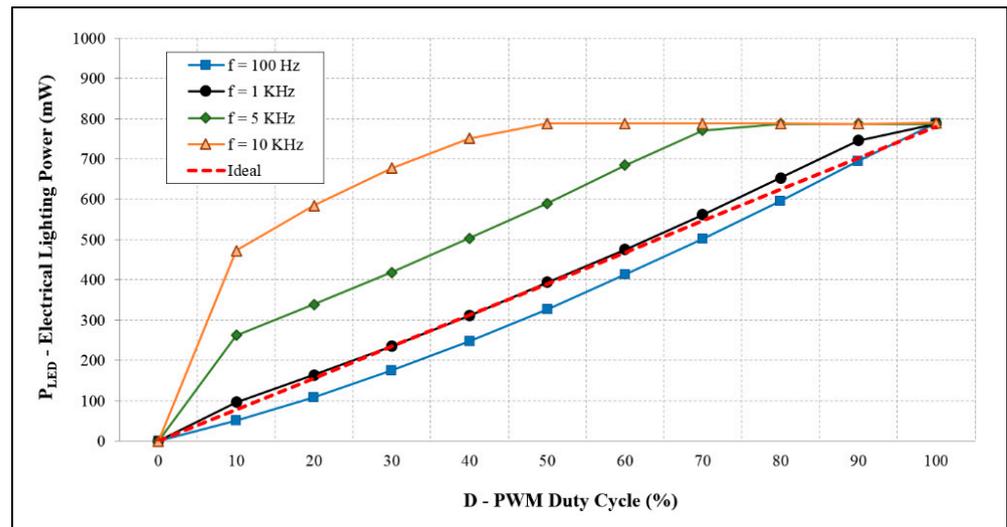


Figure 9. Output power values depending on the switching performance of the LED strip lamp driver for different PWM frequencies and duty cycles.

In the second stage of the experimental study, the changes in platform illuminance level depending on electrical lighting power are investigated for different environmental initial illuminance level (E_i) values. For this purpose, the E_i on the platform is adjusted separately as 0 lx, 50 lx, 100 lx, 150 lx and 200 lx by utilizing the daylight condition and using a black cover. For each E_i value, the electrical lighting power of the LED driver circuit is adjusted to the required value by changing the duty cycle of the 1 KHz PWM signal. The platform illuminance values obtained in this way are measured with a digital luxmeter (Figure 10). In the lighting system, when $E_i = 0$ lx at the beginning, the P_{LED} value is adjusted to between 0 and 780 mW with the help of the PWM signal, and it is observed that the obtained E increases approximately linearly between 0 and 163 lx.

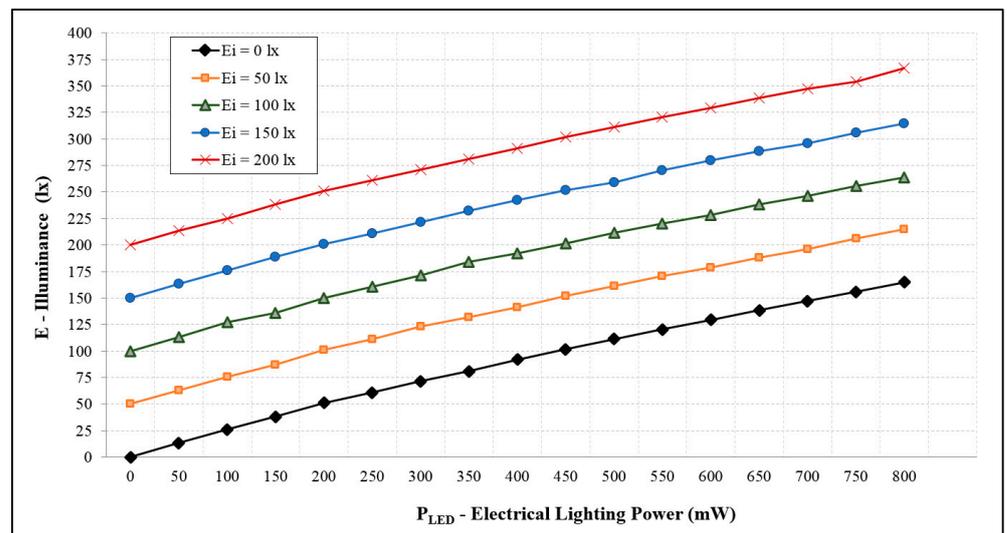


Figure 10. Changes in platform illuminance depending on electrical lighting power.

In the third stage of the experimental study, the minimum platform illuminance levels required for color-based image processing for object detection processes are investigated according to different colors. For this purpose, while the initial platform illuminance level is 0 lx, the minimum illuminance level (E_{min}) values—where successful results are obtained in image processing—are measured with a digital luxmeter by adjusting the electrical illumination power with the PWM signal for RGBY-colored objects with a diameter of

2.5 cm (Figure 11). Among the colors used in the experimental study, it is observed that yellow ($E_{\min} = 21$ lx) is the color with the lowest illuminance level and therefore requiring the least energy consumption, followed by green ($E_{\min} = 32$ lx), red ($E_{\min} = 40$ lx) and blue ($E_{\min} = 50$ lx) colors, respectively. In addition, based on Figures 10 and 11, it is concluded that working with the Yellow object in the system when $E_i = 0$ lx provides 34%, 48% and 58% electricity savings compared to the Green, Red and Blue objects, respectively. Additionally, Table 4 shows the images obtained for different colored objects at various illuminance levels and their detection status.

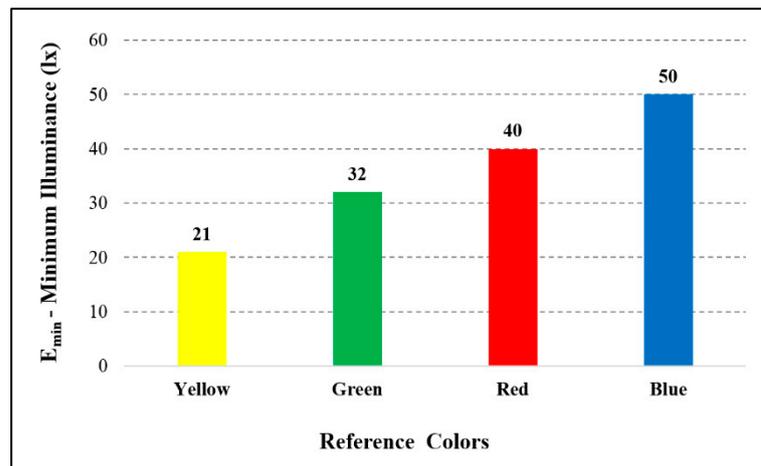


Figure 11. Minimum illuminance levels needed for detecting objects with different colors by image processing.

Table 4. Images obtained at various illuminance levels for objects and their detection status.

Object	Detection Status						
	$E = 0$ lx	$E = 10$ lx	$E = 21$ lx	$E = 32$ lx	$E = 40$ lx	$E = 50$ lx	$E = 100$ lx
RGBY Reference Colored Objects							
Yellow Object							
Blue Object							
Green Object							
Red Object							

In the fourth stage of the study, the system illumination speed for the detection of RGBY reference colors is examined. When $E_i = 0$ lx on the platform, the times taken to achieve the minimum illuminance level are 1.33 s, 2.17 s, 2.67 s and 3.26 s for the yellow, green, red and blue colors, respectively (Figure 12). It can be said that lighting control in the developed RGBY reference-color-based system is slower than traditional sensor-based methods. This is because the illumination speed, which is controlled by a software in the system, is directly related to the performance of the computer used in the design, which suggests that using a more powerful computer would increase this speed.

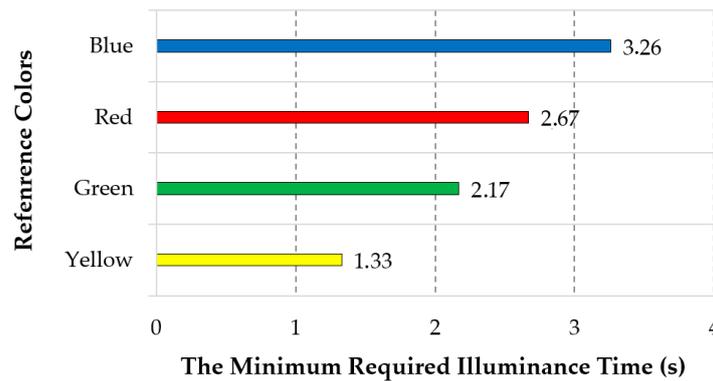


Figure 12. The minimum illuminance times required to detect RGBY reference colors ($E_i = 0$ lx).

In the last stage of the study, the electrical lighting power values consumed by the system depending on the environmental illuminance level are investigated for object detection and position determination. For this purpose, the required electrical power values consumed (P_{LED}) by the LED strip lamp are obtained by operating the system for different E_i values between 0 and 100 lx (Figure 13). While the E_i value in the system is 0 lx, the power consumed by lighting is approximately 225 mW. This consumed power decreases inversely with the E_i value and becomes 0 W by turning off the electrical lighting power above 50 lx.

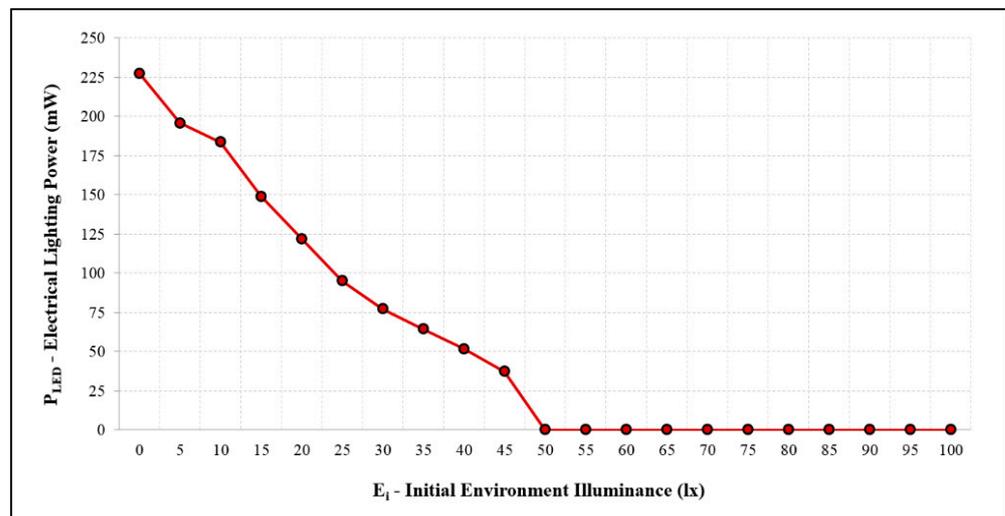


Figure 13. The required electrical lighting power values depending on the initial environmental illuminance level for the designed system.

4. Conclusions

In this study, an embedded system with reference-color-based intelligent ambient lighting is designed for image-processing-based object detection and position determination in robotic handling applications. In this system, it should be noted that an adaptive lighting process based on the detection of RGBY reference objects with color-based image processing is performed without the need for any hardware, such as an external sensor/device. Thanks to image-processing-based object detection via the provision of adaptive lighting according to the color of the workpiece in the system, energy savings in ambient lighting are maintained, and the negative effects of over-lighting are eliminated.

Based on the experimental findings, it became evident that color choice played a pivotal role in the detection process, with yellow requiring the lowest illuminance level among the tested colors. The following yellow, green, red and blue colors were observed to require progressively higher illuminance levels. It was deduced that the detection of

blue-colored objects under white lighting against a white background required higher illuminance levels. Thus, employing a blue-colored object as a reference enabled the determination of the optimal illuminance level for object detection. Within the system, successful color-based object detection was achieved at an illuminance level of approximately 50 lx, with power requirements ranging from 0 to 225 mW, contingent upon ambient lighting conditions. This underscored how an intelligent lighting system adjusts the illuminance level based on environmental factors, thereby curbing energy consumption in industrial settings. Consequently, it was evident that an intelligent lighting system ensures a minimum level of illumination tailored to environmental conditions, thus leading to reduced energy consumption in industrial settings.

Considering some issues in the implementation of the work, some improvements could be made in the system in the future. On the platform, detection was carried out in the XY plane only with the top view and illumination, without taking into account the height of the objects. In future studies, 3D objects could be detected via illumination and image acquisition from different directions and angles. If desired, it would be possible to use other non-standard colors similar to RGBY reference colors by defining HSV color space ranges in the software. Object detection and localization operations could be performed on the images obtained via instant lighting in the form of flash in order to save more energy on platforms where only robots are used and where there is no human-eye tracking.

Author Contributions: Conceptualization, S.D.; Methodology, U.A. and S.D.; Software, U.A. and S.D.; Investigation, U.A.; Writing—Original Draft, U.A.; Writing—Review and Editing, S.D.; Supervision, S.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Skaria, S.; John, M.; Paul, B. Automatic lighting controller. *Int. J. Eng. Res. Dev.* **2014**, *10*, 29–36.
2. Singh, M.; Garg, S. Illuminance estimation and daylighting energy savings for Indian regions. *Renew. Energy* **2010**, *35*, 703–711. [[CrossRef](#)]
3. Chew, I.; Karunatilaka, D.; Tan, C.P.; Kalavally, V. Smart lighting: The way forward? Reviewing the past to shape the future. *Energy Build.* **2017**, *149*, 180–191. [[CrossRef](#)]
4. Neida, B.V.; Manicria, D.; Tweed, A. An analysis of the energy and cost savings potential of occupancy sensors for commercial lighting systems. *J. Illum. Eng. Soc.* **2001**, *30*, 111–125. [[CrossRef](#)]
5. Miki, M.; Kasahara, Y.; Hiroyasu, T.; Yoshimi, M. Construction of Illuminance Distribution Measurement System and Evaluation of Illuminance Convergence in Intelligent Lighting System. In Proceedings of the IEEE SENSORS, Kona, HI, USA, 1–4 November 2010; pp. 2431–2434.
6. Oh, J.H.; Yang, S.J.; Do, Y.R. Healthy, natural, efficient and tunable lighting: Four package white leds for optimizing the circadian effect, color quality and vision performance. *Light Sci. Appl.* **2014**, *3*, 141. [[CrossRef](#)]
7. Wang, T.; Chen, T.; Hu, Y.; Zhou, X.; Song, N. Design of intelligent LED lighting systems based on STC89C52 microcomputer. *Optik* **2018**, *158*, 1095–1102. [[CrossRef](#)]
8. Li, L.; Wang, J.; Yang, S.; Gong, H. Binocular stereo vision based illuminance measurement used for intelligent lighting with LED. *Optik* **2021**, *237*, 166651. [[CrossRef](#)]
9. Fang, P.; Wang, M.; Li, J.; Zhao, Q.; Zheng, X.; Gao, H. A Distributed Intelligent Lighting Control System Based on Deep Reinforcement Learning. *Appl. Sci.* **2023**, *13*, 9057. [[CrossRef](#)]
10. Gonzales, R.C.; Woods, R.E. *Digital Image Processing*, 3rd ed.; Prentice Hall: Hoboken, NJ, USA, 2008.
11. Tiwari, M.; Singhai, R. A Review of Detection and Tracking of Object from Image and Video Sequences. *Int. J. Comput. Intell. Res.* **2017**, *13*, 745–765.
12. Fan, L.; Wang, Z.; Cail, B.; Tao, C. A Survey on Multiple Object Tracking Algorithm. In Proceedings of the IEEE International Conference on Information and Automation, Ningbo, China, 1–3 August 2016; pp. 1855–1862.

13. Minichino, J.; Howse, J. *Learning OpenCV 3 Computer Vision with Python*, 2nd ed.; Packt Publishing: Birmingham, UK; Mumbai, India, 2015.
14. Deshpande, R.; Nair, A.; Razban, A. Color Based Object Tracking Robot. *Robot. Autom. Eng. J.* **2018**, *2*, 60–64.
15. Zettl, H. *Sight, Sound Motion: Applied Media Aesthetics*; Wadsworth Publication: Belmont, CA, USA, 1999; pp. 54–59.
16. Wang, H.; Wang, J.; Chen, W.; Xu, L. Automatic illumination planning for robot vision inspection system. *Neurocomputing* **2018**, *275*, 19–28. [[CrossRef](#)]
17. Jeon, H.-S.; Park, S.-H.; Im, T.-H. Grid-Based Low Computation Image Processing Algorithm of Maritime Object Detection for Navigation Aids. *Electronics* **2023**, *12*, 2002. [[CrossRef](#)]
18. Barwar, M.K.; Sahu, L.K.; Bhatnagar, P. Reliability analysis of flicker-free LED driver based on five-level rectifier. *Optik* **2022**, *268*, 169762. [[CrossRef](#)]
19. Li-Li, Z.; Yan-Hua, W.; Xue-Feng, Z.; Hong-Yu, L. Implementation of a Novel LED Backlight Device Used for Glass Bottle Detection. In Proceedings of the 2013 Seventh International Conference on Image and Graphics, Qingdao, China, 26–28 July 2013; pp. 766–769.
20. Linnartz, J.P.M.G.; Feri, L.; Yang, H.; Colak, S.B.; Schenk, T.C.W. Communications and Sensing of Illumination Contributions in a Power LED Lighting System. In Proceedings of the ICC'08 IEEE International Conference, Beijing, China, 19–23 May 2008; pp. 5396–5400.
21. Liu, H.; Zhou, Q.; Yang, J.; Jiang, T.; Liu, Z.; Li, J. Intelligent Luminance Control of Lighting Systems Based on Imaging Sensor Feedback. *Sensors* **2017**, *17*, 321. [[CrossRef](#)] [[PubMed](#)]
22. Sung, W.-T.; Lin, J.-S. Design and Implementation of a Smart LED Lighting System Using a Self Adaptive Weighted Data Fusion Algorithm. *Sensors* **2013**, *13*, 16915–16939. [[CrossRef](#)]
23. Sun, F.; Yu, J. Indoor intelligent lighting control method based on distributed multi-agent framework. *Optik* **2020**, *212*, 164816. [[CrossRef](#)]
24. Pratomo, A.H.; Zakaria, M.; Prabuwo, A.S.; Omar, K. Illumination systems for autonomous robot: Implementation and design. *J. Eng. Appl. Sci.* **2009**, *4*, 342–347.
25. LCA Lighting Controls Association. Available online: <https://lightingcontrolsassociation.org/2017/07/21/introduction-to-lighting-controls> (accessed on 1 February 2024).
26. Kopparapu, S.K. Lighting design for machine vision application. *Image Vis. Comput.* **2006**, *24*, 720–726. [[CrossRef](#)]
27. Adelkhani, A.; Beheshti, B.; Minaei, S.; Java, P. Optimization of Lighting Conditions and Camera Height for Citrus Image Processing. *World Appl. Sci. J.* **2012**, *18*, 1435–1442.
28. Uzer, M.; Yilmaz, N.; Bayrak, M. A real-time tracking application of different coloured objects with a vision based mobile robot. *J. Fac. Eng. Arch. Gazi Univ.* **2010**, *25*, 759–766.
29. Kumar, G.H.; Rupa, G.; Sweatha Suresh, B.; Sneha, B.R.; Sushmitha, R. Object Tracking Robot on Raspberry Pi using Opencv. *Int. J. Eng. Trends Technol.* **2016**, *35*, 160–163.
30. Khairudin, M.; Yatmono, S.; Nugraha, A.C.; Ikhsani, M.; Shah, A.; Hakim, M.L. Object Detection Robot Using Fuzzy Logic Controller through Image Processing. *J. Phys. Conf. Ser.* **2021**, *1737*, 012045. [[CrossRef](#)]
31. Gandhi, A.; Sangeetha, M. Development of an Image Processing Algorithm for Smart CNC Machines. *IEIE Trans. Smart Process. Comput.* **2018**, *7*, 232–235. [[CrossRef](#)]
32. Sarguroh, S.S.; Rane, A.B. Using GRBL-Arduino-based controller to run a two-axis computerized numerical control machine. In Proceedings of the International Conference on Smart City and Emerging Technology (ICSCET), Mumbai, India, 5 January 2018; pp. 1–6.
33. Megalingam, R.K.; Raagul, S.; Dileep, S.; Sathi, S.R.; Pula, B.T.; Vishnu, S.; Sasikumar, V.; Gupta, U.S.C. Design, Implementation and Analysis of a Low Cost Drawing Bot for Educational Purpose. *Int. J. Pure Appl. Math.* **2018**, *118*, 213–230.
34. Stump, D.R. Electromagnetism. In *Encyclopedia of Energy Reference Works*; Cleveland, C.J., Ed.; Elsevier Science: Amsterdam, The Netherlands, 2004; pp. 319–328.
35. Bento, F.; Cardoso, A.J.M. Comprehensive survey and critical evaluation of the performance of state-of-the-art LED drivers for lighting systems. *Chin. J. Electr. Eng.* **2021**, *7*, 21–36. [[CrossRef](#)]
36. Liu, Y.N.; Liu, Y.J.; Chen, Y.C.; Ma, H.Y.; Lee, H.Y. Study of pulse width modulated LED for enhancing the power efficiency of dye-sensitized solar cells. *Optik* **2018**, *158*, 1567–1574. [[CrossRef](#)]
37. Hung, M.W.; Chen, C.J.; Chang, C.L.; Hsu, C.W. The impacts of high frequency pulse driving on the performance of LED light. *Phys. Procedia* **2011**, *19*, 336–343. [[CrossRef](#)]
38. OpenCV Reference Guide. Available online: <https://docs.opencv.org/4.7.0/> (accessed on 1 February 2024).
39. Hema, D.; Kannan, S. Interactive Color Image Segmentation using HSV Color Space. *Sci. Technol. J.* **2019**, *7*, 37–41. [[CrossRef](#)]
40. Manipriya, S.; Mala, C.; Mathew, S. Performance Analysis of Spatial Color Information for Object Detection Using Background Subtraction. *IERI Procedia* **2014**, *10*, 63–69. [[CrossRef](#)]
41. Chena, Y.; Xiaoa, X.; Liub, H.; Fenga, P. Dynamic color image resolution compensation under low light. *Optik* **2015**, *126*, 603–608. [[CrossRef](#)]
42. Salhi, A.; Jammoussi, A.Y. Object tracking system using Camshift, Meanshift and Kalman Filter. *World Acad. Sci. Eng. Technol.* **2012**, *6*, 421–426.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.