ISSN:
Print - 2277 - 0593
Online - 2315 - 7461

Journal of Natural Science, Engineering and Technology

DEVELOPMENT OF A COLOUR-OBJECT SORTING ROBOTIC ARM SYSTEM USING A PIXY2 CAMERA

S.O. OWOEYE[1]*, F.O. DURODOLA[1], B.U. ANYANWU[2], A.A. ISHOLA[1], A.A. ADENUGA[1], J. O. ODULATE[1], B.M. BISIRIYU[1]

[1]Department of Mechatronics Engineering, Federal University of Agriculture, Abeokuta, Nigeria.

[2]Department of Mechanical Engineering, Federal University of Agriculture, Abeokuta, Nigeria

[*]**Correspondence Author:** owoeyeso@funaab.edu.ng

ABSTRACT

There are growing needs for detection and sorting of objects owing to the large application of artificial intelligence in industrial and agricultural engineering applications, among other fields. Object detection is extensively used in various areas of the knowledge society to offer assistance whenever necessary. There is need to develop sorting systems that are completely automated to reduce these challenges. This paper describes the implementation and testing of an object-sorting system based on colour using a robotic arm, arduino nano microcontroller, pixy2 camera and a conveyor system. The system was designed to sort objects into different categories based on their dominant colour, as identified by the camera and processed by the microcontroller. Using a predetermined colour identification algorithm, this model assessed the ability of a robotic arm to sort various objects. The robotic arm acted as the sorting mechanism, picking up objects from the conveyor and placing them into designated containers. The system was tested with a variety of objects with different colours and shapes. It was found to have an accuracy rate of 85% in colour detection and sorting, a low latency value of 30%, object orientation of 45 % and robustness against different lighting conditions. The obtained values demonstrated the effectiveness, accuracy, and cost-effectiveness of employing computer vision in conjunction with a robotic arm for sorting objects, based on colour and shape. The study therefore provides a system that represents a promising solution for automating colour-based object sorting tasks.

Keywords: Object-Sorting System, Arduino Nano Microcontroller, Pixy2 Camera, Robotic Arm, Conveyor System, Colour Identification, Automation

INTRODUCTION

In the realm of computer vision, the process of identifying, recognizing, and tracking objects through a sequence of frames is a prominent area of research and development known as object detection and sorting (Abdullah-Al-Noman *et al.*, 2022). Object detection and sorting refers to the process of identifying and categorizing objects within an image or video stream. Object detection algorithms are designed to locate objects and draw bounding boxes around them, while object sorting algorithms categorize the objects into different classes (Owoeye *et al.,* 2022a). There are growing needs for detection and sorting of objects owing to the large application of artificial intelligence in industrial and agricultural engineering applications, among other fields. This is why experts keep looking for means to develop several technologies that can be useful for such fields.

Several researchers have executed different works in the area of object detection and sorting. Kale and Kulkarni (2013) presented a smart approach for real-time inspection and selection of objects in continuous flow. The authors improved the existing methods used in object identification by developing a system that integrated four stations of identification, processing, selection and sorting with a new image-processing feature. The project uses sensors to detect an object's colour and size, and sends this information to a microcontroller. The microcontroller then sends a signal to a circuit that controls the robotic arm's motors, allowing it to grip the object and move it to a specific location. Once there, the arm releases the object and returns to its original position. In another study, Pereira *et al.,* (2017) developed a system controlled by Programmable Logic Circuit to sort objects according to their colours. The project comprised two main parts: the first part involved software, specifically ladder logic programming that is utilized to program PLC, which controls the entire project's process systematically based on the input data sequence. The second part is the hardware, integrated with the conveyor system to transport objects, colour sensors to detect the colour of the objects, an electronic system that sorts the objects, and motors that drive the conveyor belt to ensure that the objects are sorted. Also, Taniksha *et al.,* (2016) designed a conveyor belt system using stepper and servo motors, along with mechanical structures and computer vision methods. This system is capable of identifying and sorting different objects, which greatly reduces the amount of manual labour required and shortens the time it takes to bring products to market.

Sanjaya, *et al.*, (2018) focused on the construction of a 5 Degree of Freedom (DoF) Robot Arm using an Arduino microcontroller for real-time sorting of coloured objects. The sorting task involves picking and placing the objects accurately. To achieve this, an algorithm was developed that interfaces with the system and utilizes image processing techniques. The image processing was carried out using Python 2.7 and the OpenCV library, enabling the detection of coloured objects based on the Red-Green-Blue (RGB) identifier. The detected objects were assigned coordinates through the image processing system. These coordinate data are then utilized to build an inverse kinematic model, which is trained using the Artificial Neural Network (ANN) method to control the Robot Arm. The implemented algorithm enabled the Robot Arm to successfully pick and place coloured objects with a high level of accuracy. Notably, this Coloured Object Sorting system offers the advantages of being open source and low cost. Also, Garad (2018), presented an object-sorting robot based on shape. The author implemented a robotic arm using ARM7 and the shape detection was done using MATLAB. The image is captured using the camera and then image processing is performed for shape identification. The robotic arm is controlled by a microcontroller with a DC motor. The project was aimed at dealing with a fully automatic industrial material handling system.

Al Hinai *et al.,* (2019) focused on the development of a colour sensor system utilizing image processing techniques to accurately identify coloured objects. The system incorporates a programmable logic controller

(PLC) for efficient and automated sorting processes. A compact conveyor belt is designed to transport four different coloured objects, while a Raspberry Pi, equipped with a camera, serves as the colour sensor. The input signals captured by the colour sensor are transmitted to the PLC, which then generates output signals to control the CD drives responsible for ejecting the objects into their respective colour boxes. Additionally, a human-machine interface (HMI) is employed for monitoring and control purposes, displaying both the system-assigned number of each object and its corresponding colour.

Al Fadhil *et al.,* (2020) introduced a mechatronic sorting system that utilizes image processing techniques with a web camera. The system operates in real-time by sensing existing parts, capturing them, and analyzing their colours and geometric data. The information obtained from this analysis is then used to control a 5 DOF robotic arm, enabling the sorting of objects with different geometries (circle, rectangular, and triangle) and colours (red, green, blue, brown, orange, and cyan). The project focused on automating the material handling process by classifying objects based on their colour, size, and geometry as they arrive on a conveyor belt, and subsequently selecting and placing them in their designated locations. Image processing is performed using OpenCV and Visual Studio software, where images are treated as matrices of pixels. Various image processing techniques are employed to digitize and enhance the images. Shape identification is accomplished using the canny edge detection algorithm. The sorting process is automated using a microcontroller equipped with computer vision capabilities.

In their research, Mohammad *et al.* (2021) presented a project of a four-degree-of-freedom robotic arm designed and implemented to sort objects using colour recognition techniques. The arm is constructed using a lightweight composite material called plastic wood, with the aid of SolidWorks software. The colour recognition module accurately detects objects without any errors. To control the robotic arm, inverse kinematics algorithms are employed to calculate the required control inputs. These inputs are then sent to an Arduino Mega microcontroller board for execution. The Arduino board controls the motors integrated into the arm, allowing for the sorting and placement of objects based on their colours. The robotic arm successfully achieves the task of sorting different objects. Due to its low-cost construction and effective performance, this type of robotic arm holds the potential for utilization in industrial applications.

In all these studies, recognizing objects, which people perceive as a straightforward process, poses a significant challenge for computer systems. To accomplish this task, object identification algorithms utilize appearance-based or feature-based approaches, relying on techniques such as matching, learning, and pattern recognition. The objective of this paper is to create a robotic system for industrial applications that utilizes robotic automation to reduce the physical presence of the operator on the site while performing their assigned duties. The system's focus is to detect and sort objects according to their specific colours through the use of an RGB technique, which involves conducting a colour detection process based on the object's appearance and colour.

MATERIALS AND METHODS

The system is designed to recognize different colours of objects and to sort them into their respective categories. The robotic arm is used to pick and place them into locations based on their colour. The hardware requires a Pixy2 Camera module, Arduino Nano Microcontroller, IR proximity sensor, Robot Arm, Conveyor system and Power Supply. The following tests were carried out on the robotic system: Colour identification, Robotic Arm Movement, Sensitivity of the IR sensor, Object carriage on the Conveying system, Speed, Endurance, Interference and Error Handling.

Pixy2 Camera Module

The Pixy2 camera module is a compact, fast, and easy-to-use vision sensor for robotic projects and beyond. It is designed to interface with microcontrollers, such as Arduino, to provide object detection and tracking capabilities. The Pixy2, (Figure 1), uses an image sensor and powerful processor to quickly analyze video frames and identify objects and then provides the data to the microcontroller via an I2C or SPI interface. The pixy2 camera module allows for object identification based on their RGB colour values. To test its efficacy, the camera was used to photograph three randomly selected colours, which were then analyzed 16 times using the colour identification program. This analysis was conducted under different lighting conditions to ensure accuracy.

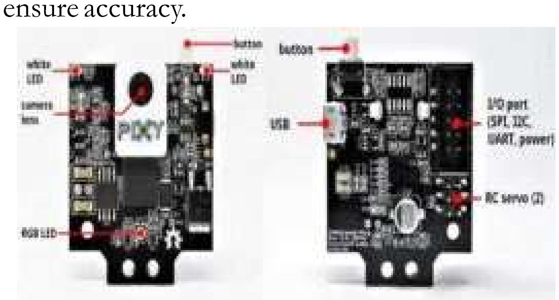

Figure 1: Pixy2 camera module

Arduino Nano Board

The Arduino Nano (Figure 2) is a compact and versatile microcontroller board based on the ATmega328P (for version 3. x and 5. x) or ATmega168 chipset (Owoeye, *et al.,* 2022b). It is designed for use with an open-source hardware platform and software development environment, making it an ideal choice for beginners and hobbyists. Some of the features of the Arduino Nano include:

- 14 digital input/output pins, 6 of which can be used as PWM outputs
- 8 analog inputs
- 32 kB of program memory (2 KB of which is used by the bootloader)
- 1 KB of SRAM
- 16 MHz clock frequency
- USB connection for programming and power
- A small form factor (0.73" x 1.70")

Figure 2: Arduino Nano board

IR Proximity Sensor

An IR (infrared) proximity sensor is a type of sensor that uses infrared light to detect the presence or absence of objects (Figure 3). By emitting a beam of infrared light and measuring the amount of reflected light, this device operates. Based on the time it takes for the emitted light to return, the sensor determines the distance to the object. These sensors are often used in applications where the precise distance to an object is important, such as in industrial automation, robotics, and security systems. They are also used in consumer electronics, such as mobile phones, to detect when the device is being held up to the ear.

Model:	HiLetgo TCRT5000 IR Proximity Sensor
Detection Range:	10 cm to 80 cm
Supply Voltage:	5V DC ± 10%
Current Consumption:	< 20 mA
Output Type:	NPN Open Collector
Response Time:	< 1 ms
Operating Temperature:	-20°C to 60°C
Storage Temperature:	-40°C to 85°C
Resolution:	1 mm
Repeat Accuracy:	±2%
Light Source:	Infrared LED
Material:	ABS Plastic Housing.

Figure 3: IR Proximity Sensor

The Robotic Arm

This component performs the picking and placing task. It's majorly actuated by servomotors and other rotating parts. It does this task using a gripper for material handling. It is responsible for picking up objects from a conveyor belt and placing them into the appropriate bin based on their specific attributes, such as size, shape, colour, and material. The arm, (Figure 4) is equipped with sensors and cameras that allow it to identify and analyze each object. It then uses its mechanical gripper to pick up the object and move it to the designated bin, where it is deposited. The robotic arm is programmed to work quickly and accurately, allowing for high-speed and efficient object sorting. The arm is capable of adapting to new objects and changing sorting requirements, making it a flexible and efficient solution for a wide range of sorting tasks. Several experiments were conducted to observe the movement cycle of the arm, which comprises picking up the object using the end-effector from the standby position, transporting it, dropping it into a designated bin, and returning it to the standby or home position.

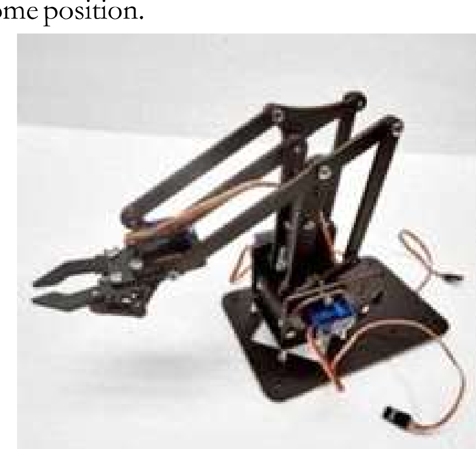

Figure 4: The Robotic Arm

Objects to Be Sorted

The test objects to be identified and sorted with cross-section were fabricated from lightweight plastics for weight considerations (Figure 5) and the three containers were provided as bins here the objects will be delivered according to their specified location. The objects were of different shapes and colours, preferably, red, green and blue.

Figure 5: The fabricated objects

The Conveying System

A conveying system (Figure 6) is typically used in an object-sorting system to transport objects to the location where they will be sorted by the robotic arm. Several different approaches can be used to design a conveying system for this purpose. One approach is to use a belt conveyor, which consists of a continuous loop of material (e.g. rubber, plastic, or canvas) that is driven by a motor. The objects to be sorted are placed on the conveyor belt, which transports them to the sorting location. Another approach is to use a roller conveyor, which consists of a series of rollers that are powered by a motor. The sorting location receives the objects that are placed on the rollers, which serve as a means of transportation. A third approach is to use a vibrating conveyor, which consists of a trough or tube that is vibrated by a motor. The objects to be sorted are placed in the trough or tube, and the vibration causes them to move along the conveyor to the sorting location.

Figure 6: The conveying system

The Software requirement for the object-sorting system includes the following: C++, Arduino IDE, PixyMon software, and Graphical assignment of objects. In terms of software, the Pixy2 camera uses machine learning algorithms to "train" itself to recognize specific objects based on their colour and shape. This training process typically involves feeding the camera a large number of images of the objects that it should be able to recognize, along with labels indicating what the objects are. The camera uses this training data, usually by graphical representation or assignment, to learn the characteristics that define different objects and to develop algorithms that can be used to identify these objects in new images. Once the Pixy2 camera has been trained to recognize specific objects, it can use this knowledge to identify these objects in real-time as they are being sorted. It does this by analyzing the images it captures and looking for the characteristics that it has learned to associate with specific objects. If it finds a match, it can then send a signal to the robotic arm indicating which object has been detected and where it should be placed.

The implementation of this object-sorting system based on colour using a robotic arm involves the use of a microcontroller, specifically Arduino Nano Microcontroller, which acts as the brain of the system and to which other hardware components are connected or attached. The system block diagram (Figure 7) shows the basic connection paths of the components with the Arduino Nano being powered by the DC power supply, reading data from the Pixy2 camera module and IR sensor and actuating the conveyor and the robotic arm. The flowchart, (Figure 8) depicts the flow of operation of the system, and the circuit diagram (Figure 9) shows the schematic connections of the hardware parts.

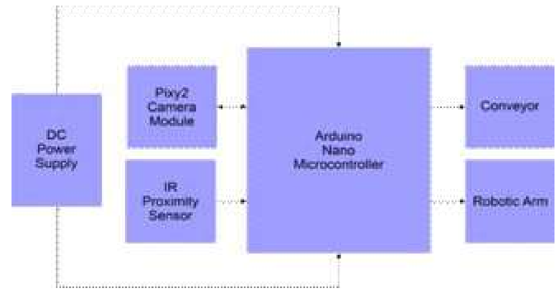

Figure 7: The system block diagram

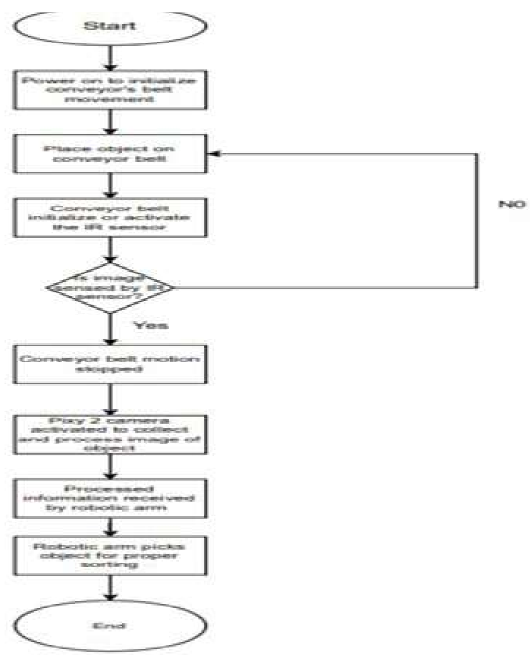

Figure 8: The system flow chart

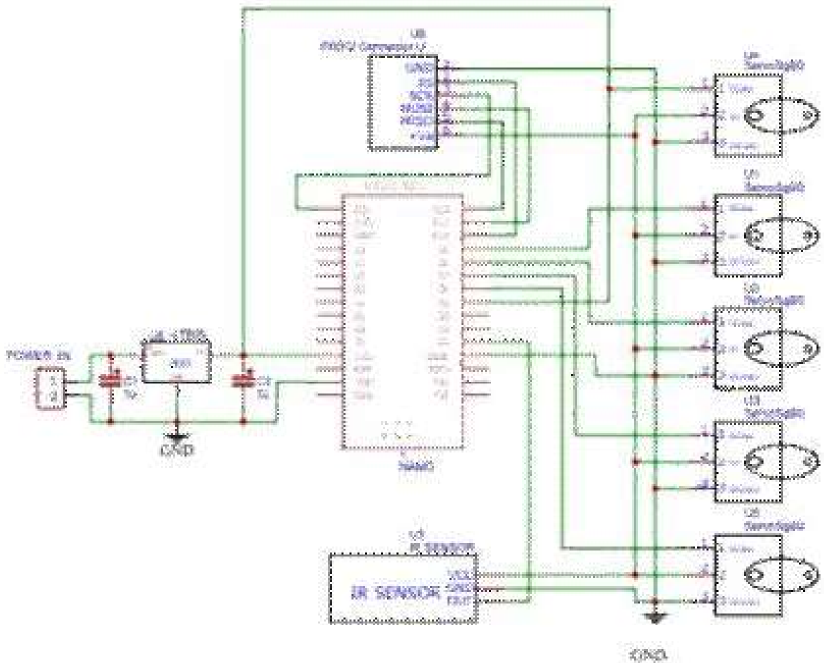

Figure 9: The system circuit diagram

Description of the Process

Here is a detailed description of the process of how these components work together in designing an object-sorting system based on colour:

Power Supply: The power supply unit provides the necessary voltage and current to all the components in the system. It is responsible for converting the main AC power supply to a stable DC voltage that the microcontroller, camera module, and robotic arm can use.

Arduino Nano Microcontroller: The Arduino Nano microcontroller is the brain of the system and is responsible for controlling all the other components. It receives data from the Pixy2 camera module and IR proximity sensor, processes this data, and subsequently transmits commands to the mechanical arm to select and position the items. The microcontroller also communicates with the pixy2 camera module to retrieve the colour information of the objects and uses this information to make decisions about which bin to place the object in.

40 mm Wide Conveyor: The conveyor belt is used to transport the objects to be sorted. The belt moves at a constant speed, allowing the objects to be transported to the robotic arm.

Pixy2 Camera Module: The Pixy2 camera module is used to detect the colour of the objects on the conveyor. The module captures an image of the object and sends the colour information to the microcontroller, which uses this information to decide which bin to place the object in.

IR Proximity Sensor: The IR proximity sensor is used to detect the presence of an object on the conveyor. Upon detection, the sensor dispatches a signal to the microcontroller, prompting the activation of the robotic arm for object retrieval. To ensure appropriate timing, the IR proximity sensor is strategically positioned to detect objects before they reach the robotic arm.

Robotic Arm: The task of collecting and depositing objects into their designated bin is assigned to the robotic arm. Equipped with a gripper or suction cup, the robotic arm is under the control of the microcontroller. To collect the objects and move them to the appropriate bin, the robotic follows a pre-defined path, also directed by the microcontroller.

The Software Implementation

Programming the Pixy2 camera for colour recognition and identification involves several steps:

- Installing the PixyMon software: PixyMon is a graphical interface that you can use to configure and program your Pixy2 camera. It can be downloaded from the official Pixy website.
- Connecting the Pixy2 camera: Connect the Pixy2 camera to your computer via a USB cable. The PixyMon software should automatically detect the camera.
- Calibrating the camera: In PixyMon, select the "Calibrate" tab to set the colour recognition parameters. Use the sliders to adjust the hue, saturation, and brightness of the colours you want to detect. You can also use the live camera feed to help you set the parameters.
- Training the camera: Once you have set the colour recognition parameters, you can train the camera to recognize specific colours. To do this, select the "Color Signature" tab and clickthe "Teach" button. Hold the object you want to recognize in front of the camera and click the "Learn" button.
- Creating blocks: Blocks are how Pixy2 organizes and categorizes the objects it detects. You can create blocks for specific colours or colour combinations by selecting the "Color Signature" tab and clicking the "Add" button.
- Writing code: Once you have trained the camera and created blocks, you can write code to control the Pixy2 camera. Pixy2 supports several programming languages, including C++, Python, and Arduino. You can use the PixyMon software to generate code examples to get you started.
- Testing: Test your code by holding objects in front of the camera and verifying that the Pixy2 camera correctly recognizes and categorizes the objects.

Software specifications: C++, Arduino IDE, PixyMon software, Graphical assignment of objects.

RESULTS AND DISCUSSION

Colour Identification Test Results

The RGB values obtained range between 57 and 235 pixels from the camera's output and colour identification. Also the input from the test sheet recorded values ranging from 36 to 176 pixels (Figure 10). The level of confidence of identified colours input from the test sheet was higher than the level of the camera's output colour.

These values are represented on the multiple bar chart (Figure 10). The test presents an accuracy rate of 85% in colour detection and sorting, as well as a low latency value of 30% and object orientation of 45 %.

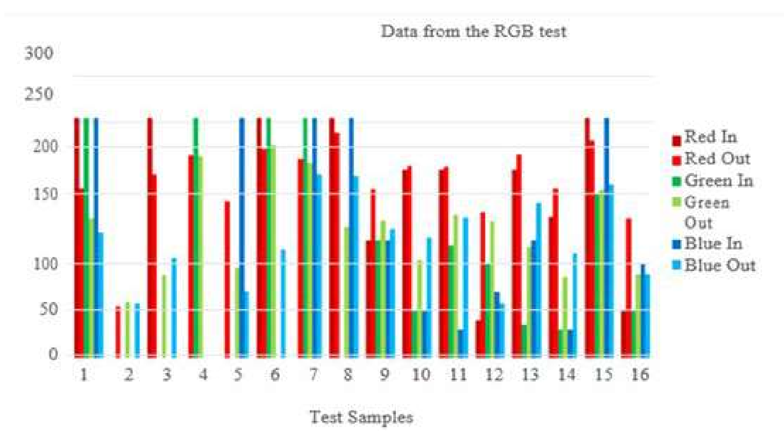

Figure 10: Differences in RGB values captured by the Pixy2 camera module

Robotic Arm Movement Result

The average time required for each robotic arm movement shows that as the cycle time progresses or increases, the average time of the robot arm increases (Table 1). The average time recorded by the robot arm was 14.33 seconds. This value is within the range specified for most articulated robotic arms. This shows the reliability and efficient performance of the robot with respect to movement from one location to another (Garad 2018).

Table 1: Time required for each robotic arm movement

Cycle no	Time (s)
1	14.25
2	14.18
3	14.25
4	14.27
5	14.33
6	14.44
7	14.48
8	14.45
Average	14.33

IR Proximity Sensor Result

The performance evaluation of the IR Proximity sensor (Table 2) showed non-linearity characteristics or features of the sensor applied in the study. At a distance of 6 cm, a sensitivity voltage of 1.2 was recorded. This falls within the acceptable sensitivity value for a typical IR Proximity sensor (Mohammad *et al.* 2021).

The farther away the sensor from the object under detection, the more the sensitivity deteriorates, as seen from the voltage readings taken across the sensor terminals and vice versa (Table 2). The drop in voltage must have been as a result of decrease in the magnetic flux. This trend is in line with the findings of Fadhil et al, 2020.

Table 2: Sensor Sensitivity test results

Tests	Distance Between IR Sensor and Object(cm)	Voltage(v)
1	1	5.00
2	2.5	4.80
3	4	3.40
4	6	1.20
5	8	0.45

The line graph (Figure 11) of the Voltage-Distance relationship confirmed the non-linearity nature of the sensor (Owoeye et al, 2022b). It also confirmed the optimal placement of the object in front of the sensor as 6 cm, being the point of intersection between the linear voltage and the voltage output of the IR sensor.

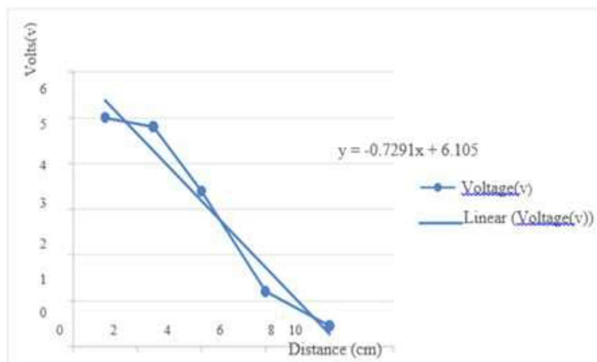

Figure 11: The sensor sensitivity response

Conveyor Test Results

The weight of the object had a great effect on the speed of rotation and acceleration of the conveyor (Table 3). This determines extensively the rate at which the conveyor belt travels within the system. As the weight of the objects increases, the speed of rotation and acceleration decreases, and this is in line with what Pereira et al, 2014 reported in their study. The conveyor speed of 150 RPM and acceleration of 1.5 m/s^2 was achieved by the test object weight of 50g at an inter distance of 5 cm, this value falls within the ranges obtainable for a typical conveyor belt (Sanjaya, 2018). The conveyor test can thus, be used as a reference to determine the speed of rotation and acceleration for light objects placed given their weight and inter-distance.

Table 3: Weight, Inter-distance and speed relationships of the conveying system

Test	Weight(g),	Inter Distance (cm),	Speed of Rotation (RPM)	Acceleration (m/s^2)
1	20	20	300	3.0
2	30	15	250	2.5
3	40	10	200	2.0
4	50	5	150	1.5

CONCLUSION

The successful development and testing of an object-sorting system have been accomplished. The system relies on colour-based sorting, a robotic arm, a Pixy2 camera, an IR proximity sensor, and a conveyor system. The system has demonstrated the ability to accurately identify and sort objects based on colour, making use of the Pixy2 camera for object detection and the IR proximity sensor for object sensing. The robotic arm and conveyor system work in conjunction to effectively transport and sort objects, making the system highly efficient.

The system has also been designed to be easily expandable, with the capability to add additional sensors and robotic arms if needed. This makes it a versatile solution for a range of object-sorting applications. In summary, the object-sorting system has been designed to be reliable, efficient, and flexible, making it a suitable solution for a variety of industries. The system is ready for implementation, and we anticipate that it will provide significant benefits in terms of time and cost savings.

For further improvement, the following enhancements can be made to the system:

- Improve the speed of the sorting process: The speed of the sorting process can be improved by increasing the speed of the conveyor system and optimizing the movement of the robotic arm. This can be achieved by using high-speed motors and reducing the number of movements required to sort objects.
- Improve the safety of the system: The system should be designed with safety in mind. To achieve this, an IR proximity sensor can be added to detect any obstacles in the path of the robotic arm, and to prevent any damage to objects being sorted.
- User interface and documentation: The system should have an easy-to-use user interface, and documentation should be provided to help users understand how to operate and maintain the system.

REFRENCES

Abdullah-Al-Noman, Anika Nawer Eva, Tabassum Binth Yeahyea, Riasat Khan (2022) "Computer Vision-based Robotic Arm for Object Color, Shape, and Size Detection," *Journal of Robotics and Control (JRC),* 3(2), pp. 180-186.

Al Hinai, R. F., & Farh, A. F. A. (2019). Building colour sensor based on image processing technology for automatic colour object sorting using a PLC system. *In 2019 International Conference on Digitization (ICD)* (pp. 222-226). IEEE.

AT Fadhil, KA Abbar, AM Qusay (2020), "Computer Vision-Based System for Classification and Sorting Color Objects," *IOP Conference Series: Materials Science and Engineering*, 745(1).

Garad, P. V. (2018). Object sorting robot based on the shape. *International Journal of Advanced Research., Ideas and Innovovation. in Technology*, *3*, 129-134.

Mohammad Harun-Or-Rashid, SM Abid Shah, Wasi Shadman, Mohammad Alomgir Hossain (2021) "Design and Development of a robotic arm for sorting coloured object," *International Conference on Automation, Control and Mechatronics for Industry 4.0 (ACMI),* pp. 1-5.

Kale V.R., VA Kulkarni, (2013). "Object sorting system using a robotic arm," (2013), *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, 2(7), pp. 3400-3407.

Owoeye S.O. F.O. Durodola, O.J. Odeyemi (2022a): Object Detection and Tracking: Exploring a Deep Learning Approach and Other Techniques. In Video Data Analytics for Smart City Applications: Methods and Trends, 20pp, Published by Betham Publisher.

Owoeye S.O., F.O Durodola, S.A Akinade, T. Farinde, O. Sulaimon (2022b), Development of a Solar-Powered Smart Iot-Based Egg Incubator for Small-Scale Poultry Farmers, *Ife Journal of Information and Communication Technology*, 6(1): 46- 53.

Pereira, V., Fernandes, V. A., & Sequeira, J. (2014). Low-cost object sorting robotic arm using Raspberry Pi. In *2014 IEEE Global Humanitarian Technology conference-South Asia Satellite (GHTC-SAS)* (pp. 1-6). IEEE

Sanjaya W.S, Dyah Anggraeni, Madinatul Munawwaroh, M Yusuf S Nurasyidiek, Darmawan Setia Rahayu, Ahmad Samsudin, Ikhsan Purnama Santika, Endah Kinarya Palupi (2018), "Colored object sorting using 5 DoF robot arm based artificial neural network (ANN) method," *Journal of Physics: Conference Series,* 1090(1) pp. 42-53.

Tanisha Singh, Dnyanesha Dhaytadak, Pradnya Kadam, R Sapkal (2016), "Object sorting by robotic arm using Image processing'" *International Research Journal of Engineering and Technology (IRJET),* 3(4), 1472-1476.

Manuscript received: 19th June, 2023; accepted: 19th December, 2023