# Computer Vision & IOT Based Smart Conveyor Belt Sorting System

Shreerang Jitendra More
More Engineering Classes and Robotics Training Centre
Dhule, Maharashtra, India.

Ajinkya Jitendra More
More Engineering Classes and Robotics Training Centre
Dhule, Maharashtra, India.

*Abstract*—**This paper proposes a Smart Conveyor Belt Sorting System that utilizes Computer Vision and IoT technologies. The main objective of this system is to sort objects on the conveyor belt based on their color. The system uses an image sensor to capture video frames, which are then processed by the Jetson Nano processor from NVIDIA.**

**The processor applies a color filter to the input video after filtration to extract the appropriate colors. The program requires specific color values to detect the object's exact color, which is calibrated using track bar sliders.**

**The system allows the user to draw bounding boxes and contours around the detected objects. Once an object is detected, Jetson Nano sends a trigger signal to the driving circuit, which is based on node mcu, to control the electronic actuator and change the object's sorting direction. Node mcu acts as a link between the user and the conveyor system.**

*Keywords :* **Jetson Nano, Node mcu, Image sensor, IoT, Color filter.**

## I. INTRODUCTION

With the advancement in technology, our products are becoming more sophisticated day by day, and the manufacturing industries require a technological upgrade. One of the vital components in the manufacturing industry is the conveyor belt. It plays a crucial role in transporting goods from one place to another. As the industry is moving towards its fourth revolution, the conveyor belt is becoming even more important, and the industry is gradually shifting towards conveyor belt-based systems.

The concept of a smart conveyor belt system can be applied to various industries for several applications such as warehouse management systems, where products are placed in different locations, and airport luggage handling, where luggage needs to be sorted. Smart sorting systems can be programmed for different applications and are more efficient than manual sorting, which is time-consuming and requires significant human effort. Automation reduces human effort and saves time, making the process more efficient.

## II. LITERATURE REVIEW

[1] This paper, researcher is presenting an idea for material sorting on a conveyor belt using computer vision. For this purpose, they are applying three algorithms named as Background subtraction, canny edge detection and morphological operators, particle analysis.

By observing this three methods, particle analysis method performed better among all in terms of processing speed and reliability. A micro-controller with camera is used for this application.

[2]This paper presents issues related to conveyor belt systems. As industries are proceeding towards future that is industry 4.0, the work area of industry 4.0 is increasing. It is necessary to study and analyze different areas for the successful accomplishment of industry 4.0, and conveyor belt is one of the important area that should be explored and upgrade.

[3]This paper presenting an idea for fruit sorting along with grading system. This system uses machine vision and it uses a material with minimum cost for its mechanical system. For sorting mechanism, servo motors are used and for machine vision, logitech camera is used.

In the algorithm, first they are accepting images in RGB colour space and it is then converted to HSV color space for image processing and colour segmentation. The final output will gives the output on display screen showing the quality of fruit and this system can produce the output within 500 ms.

[4] As industry is evolving and going towards 4th industrial revolution, PLC plays important role as it execute ladder logic for automation in the industry. Relays are used to control the industrial drives like induction motor for conveyor belt and the relays are operated by programmable logic control, and it kips the weight on the belt within permissible range.

The renewable energy sources like solar energy are also considered in this paper. For the automation of the industry, Big data comes in to a picture. The IoT technology can be utilized, so that the data form sensors, network can be stored on the secured cloud server. The behavior of this PLC based system and efficiency is shown using Matlab/Simulink for realistic visualization, which saves the time.

[5]This paper proposes an idea for candy packing as an intelligent factory design. Smart factory concept is implemented with Physical devices, cloud sever, client and communication network are the layers described with layer to layer interaction. The system supports multiple product on same production line. Proposed system results, efficiency, system diagram are given in this paper.

## III. PROPOSED ARCHITECTURE

The computer vision system is a crucial part of the smart conveyor belt sorting system as it enables the automatic detection and sorting of objects based on their color.

This reduces human effort and improves the efficiency of the sorting process. The computer vision system uses an image sensor to capture images of the objects on the conveyor belt.

These images are processed by the Jetson Nano image processor to filter out the colors of the objects.

To detect the specific color of an object, the color values are fed into the program. Calibrating these color values is done using track bar sliders, which allow the user to adjust the color values to their desired level of sensitivity. Once the objects are detected, the system can draw bounding boxes and contours around them to highlight their position on the conveyor belt.

The Internet of Things (IoT) is another essential component of the system. It provides a platform for remote monitoring and control of the conveyor belt sorting system. This means that the user can access the system from any location using an internet-enabled device.
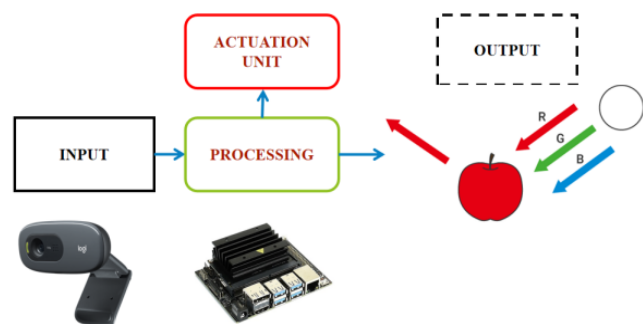


Fig 1. system diagram

The camera continuously monitors the conveyor belt, and as soon as an object is detected, the system counts the objects and sends the count to the Blynk IoT platform. The driver circuit then receives signals from the Jetson Nano processor and actuates the motor to change the path of the object, sorting it to its appropriate destination.

Overall, the implementation of computer vision and IoT technologies in the smart conveyor belt sorting system makes the sorting process more efficient and less labor-intensive, enabling industries to automate their sorting process and reduce costs.
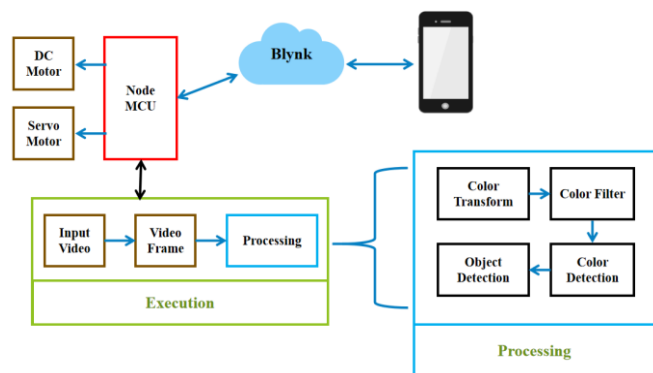


Fig 2. Block diagram

## IV.    SYSTEM DESIGN AND IMPLEMENTATION

*A.  Jetson nano*

The NVIDIA Jetson Nano Developer Kit is a powerful, small computer that is designed to run complex applications such as image classification, object detection, segmentation, and speech processing. It is specifically designed for use in edge computing devices and is particularly useful for applications that require high processing power, such as those that involve computer vision or machine learning.

The Jetson Nano Developer Kit requires a power supply of 5 volts and a current of up to 4 amps. It also requires a micro SD card that is burned with the appropriate operating system. The Jetson Nano Developer Kit uses Ubuntu as its base operating system, which is a popular open-source operating system widely used in the development community.

One of the key features of the Jetson Nano Developer Kit is its 40-pin header, which allows it to interface with a variety of different peripherals. This header is similar to the one found on the Raspberry Pi, which is another popular development board in the maker community. With this header, developers can easily connect different sensors, cameras, and other devices to the Jetson Nano, making it a versatile and powerful platform for building a wide range of applications.



Fig 3. Jetson Nano

*B.  JetPack SDK*

NVIDIA JetPack SDK is a software development kit (SDK) for building AI applications on the NVIDIA Jetson platform. It provides a comprehensive solution to developers by including everything they need to get started with building AI applications.

JetPack SDK includes a software stack that contains the latest OS image, developer tools, libraries, APIs, samples, and documentation. This allows developers to easily set up their development environment and start building applications quickly.

The SDK supports multiple programming languages such as Python, C++, CUDA, and TensorRT. It also includes support for deep learning frameworks such as TensorFlow, PyTorch, and Caffe.

JetPack SDK helps developers take advantage of the powerful capabilities of the NVIDIA Jetson platform, which includes hardware acceleration for AI workloads, high-performance computing, and low-power consumption. It enables developers to build and deploy AI applications at the edge, which is useful for applications such as autonomous vehicles, robotics, and industrial automation.

Overall, JetPack SDK simplifies the process of developing AI applications on the NVIDIA Jetson platform and makes it more accessible to a wider range of developers.

Fig4. Jetpack SDK environment

*C.* Logitech C270

As an image sensor in this project, a Logitech c270 webcam is used. It can capture 720p at 30fps and 55-degree field of view. We are using the resolution of 630*480 for this project.



Fig 5. Logitech c270

*D.* Iot Architecture

The Internet of Things (IoT) architecture consists of multiple layers, and each layer has its functionalities. The first layer is the physical layer, which is responsible for the actual sensing and measuring of the data. In the case of the smart conveyor belt sorting system, the physical layer includes the sensors that detect the objects on the conveyor belt, as well as the image sensor that reads the video frames for color detection.
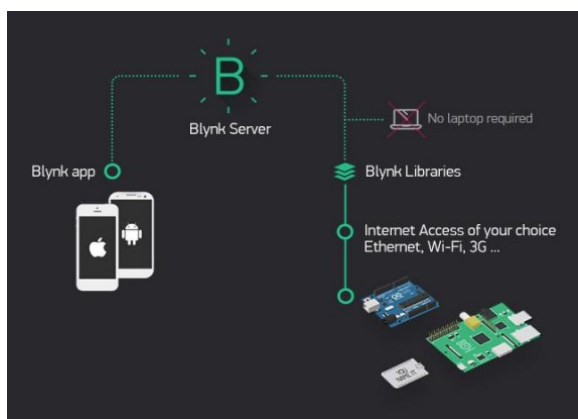


Fig 6. Blynk Architecture

The node MCU uses Wi-Fi for communication and network routing. It collects the data related to the sorting of objects on the conveyor belt and sends it to the cloud-based system (Blynk in this case).

The second layer of the IoT architecture is the data link layer. In this layer, the data collected from the physical layer is transmitted and received by the node MCU, which is a microcontroller that handles various protocols required for reliable communication throughout the network.

The third layer of the IoT architecture is the network layer. In this layer, the data received by the cloud-based system is processed and analyzed to generate insights that can be used to optimize the conveyor belt sorting system. The insights generated by this layer are sent back to the data link layer for further processing and to control the sorting of objects on the conveyor belt.

The fourth layer of the IoT architecture is the application layer. In this layer, the insights generated by the network layer are used to develop applications that can be used to optimize the smart conveyor belt sorting system. These applications can be accessed on smartphones, tablets, or any other device with an internet connection, providing remote access to the sorting system data. Overall, the IoT architecture provides a scalable and flexible framework for building and managing smart conveyor belt sorting systems that can help to reduce human effort, save time, and increase efficiency in industries.

*E.* Node MCU

The NodeMCU is an inexpensive microcontroller board that uses the ESP8266 Wi-Fi module. It has open-source firmware and a development kit that enables programming using either the Lua scripting language or the Arduino IDE. The board features a USB-to-serial converter that facilitates programming with the Arduino IDE.

Because of its built-in Wi-Fi connectivity, NodeMCU is frequently utilized in IoT projects that involve communication between devices and cloud servers. For example, in a patient health monitoring system, NodeMCU gathers data from various sensors attached to the patient, including body temperature and heart rate, and sends the data to the cloud-based system for further processing and analysis.

NodeMCU can integrate with various sensors, making it an excellent choice for data collection from various sources in IoT applications. It's small size and low cost also make it popular among DIY enthusiasts and hobbyists interested in developing their own IoT projects.



Fig 7. Node mcu

*F.* Blynk

The implementation of a software solution that is efficient, user-friendly, and compatible is crucial for the success of our system. We have chosen the Blynk app, an open-source IoT platform that is widely recognized for its fast performance and cross-platform compatibility with most Android and iOS systems.

The Blynk app offers a comprehensive and easy-to-use dashboard that enables users to add various components such as gauges, graphs, GPS locations, and more. In our system, we have decided to incorporate LCD, LED, and value display components to provide a comprehensive overview of the data collected.
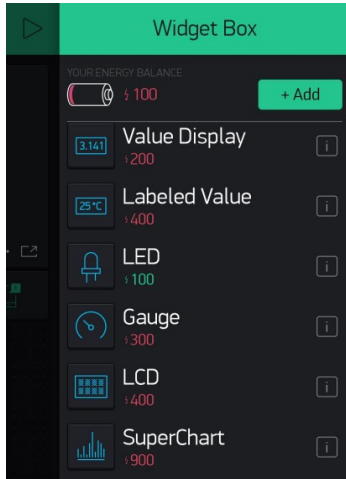


Fig 8. Blynk widget box

The three core elements of our system are the Node MCU, Blynk library, and smart-phone application. The smart-phone application serves as a user-friendly interface for data visualization, while the Blynk library enables the Node MCU to transfer data to the Blynk server efficiently. With these components working together seamlessly, our system is poised to provide accurate and real-time data analysis.

## V.    SYSTEM FLOWCHART
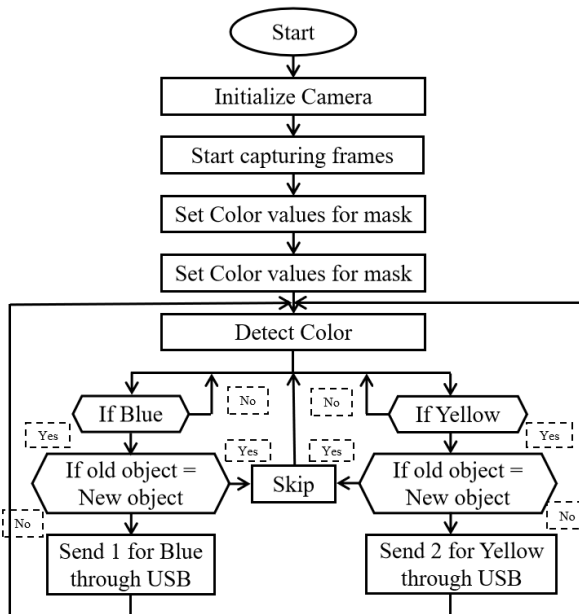


Fig9. Flowchart 1

Flowchart 1 illustrates the process flow of our system for object detection using computer vision. The input video from the camera is continuously monitored by the computer vision algorithm. The algorithm detects the presence of objects on the

conveyor belt and identifies their color. Once an object is detected, the information about the object is sent to the NodeMCU through a serial connection.

In Flowchart 2, we have illustrated the working of NodeMCU and IoT links. NodeMCU receives the data from the computer vision algorithm and sends it to the Blynk IoT platform through Wi-Fi connectivity. The Blynk IoT platform then processes the data and displays it on the user interface. Users can access the data from the cloud-based system from their smart-phones or tablets.
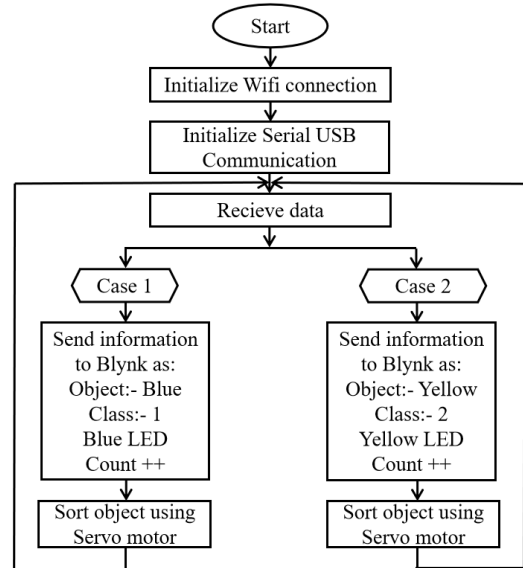


Fig 10. Flowchart 2

Overall, these flowcharts provide a clear understanding of the various components involved in our system and the flow of data between them. This helps in identifying any potential issues or bottlenecks in the system and improving its overall efficiency.

## VI.    EXPERIMENTATION AND WORKING

The Smart Conveyor Belt Sorting system utilizes computer vision to detect objects on the conveyor belt. This detection is based on the color of the objects, which is achieved through color masking. To calibrate a specific color, the track bar feature is utilized in Python coding.

Once a color is calibrated, a mask is created that only allows that particular color to pass through and be detected. There are several color spaces available, such as RGB, HSV, HLS, Grayscale, etc. In this project, HSV color space is used for masking. The program is calibrated for two colors - Blue and Yellow.

Once an object is detected on the conveyor belt, Jetson Nano sends a signal to the Node MCU through USB. The Node MCU is responsible for counting the number of objects on the conveyor belt and sending the information to the Blynk server. The Blynk application displays the current object color using the LED widget, the object count, and the class of the object (i.e. Yellow or Blue).
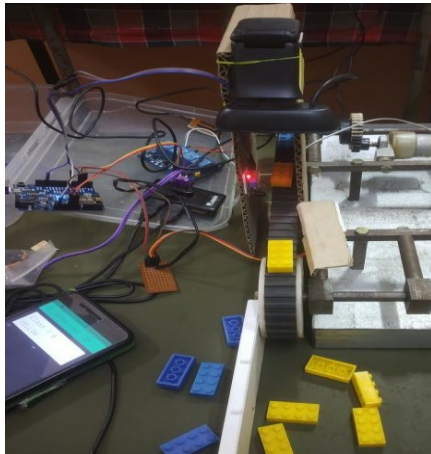
Fig 11. Experimental setup

## VII. RESULTS

In the Smart Conveyor Belt Sorting system, the color masking algorithm is used to detect the objects on the conveyor belt. The applied color mask passes only the selected values and shows the final output to the main output frame.

Additionally, a square rectangle can be drawn to indicate the detected objects. This allows for fine-tuning of the program using track bars. The program can be calibrated and customized for detecting multiple colors.
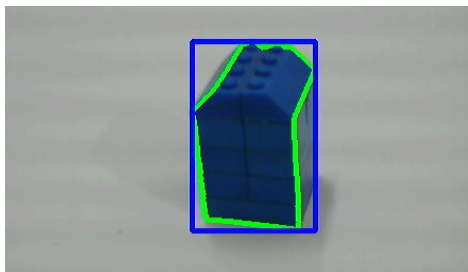


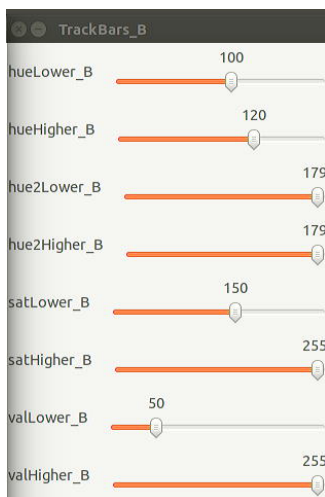Fig 12. Blue object detected



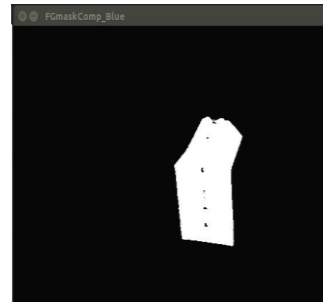Fig 13. Track bars settings for Blue color



Fig14. Blue color mask

To detect multiple colors, the user can simply calibrate each color using track bars and provide all the color values in the form of a tuple. This simplifies the process and eliminates the need for a separate set of track bars for each color.
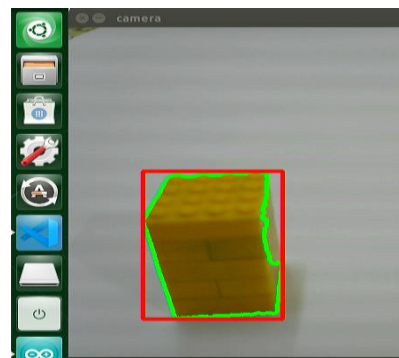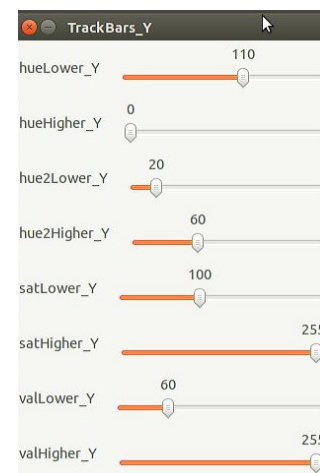


Fig 15. Yellow object detected



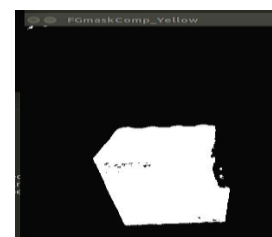Fig 16. Track bars settings for Yellow color



Fig 17. Yellow color mask

To create a user-friendly interface, the Blynk IoT platform provides various types of widgets that can be chosen according to the user's requirements. This enables the user to customize the display and view the required information in an easily understandable manner.
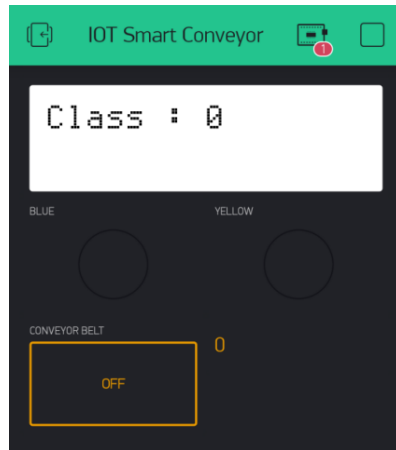


Fig 18. Blynk environment

## VIII. CONCLUSION

The successful implementation of the computer vision based conveyor belt sorting system combined with IoT marks a significant advancement in the field of industrial automation. By utilizing machine learning algorithms, the system can be further enhanced to become more robust and flexible, catering to specific applications and requirements.

The concept of a smart conveyor belt system is an integral part of Industry 4.0, which emphasizes the integration of advanced technologies such as IoT, artificial intelligence, and machine learning in the manufacturing process. With the smart conveyor belt system, industries can optimize their manufacturing process, increase productivity, reduce waste, and minimize errors, thereby increasing their overall efficiency and profitability. In conclusion, the successful implementation of this technology represents a significant leap toward the future of industrial automation and holds immense potential for revolutionizing the manufacturing industry.

## REFERENCES

[1] Zhang, M., Chauhan, V., & Zhou, M. (2021, January). A machine vision based smart conveyor system. In Thirteenth International Conference on Machine Vision (Vol. 11605, p. 116050C). International Society for Optics and Photonics.

[2] Fedorko, G. (2019). Implementation of Industry 4.0 in the belt conveyor transport. In MATEC Web of Conferences (Vol. 263, p. 01001). EDP Sciences.

[3] Afrisal, H., Faris, M., Grezelda, L., & Soesanti, I. (2013, November). Portable smart sorting and grading machine for fruits using computer vision. In 2013 International Conference on Computer, Control, Informatics and Its Applications (IC3INA) (pp. 71-75). IEEE.

[4] Chakraborty, S., Arvind, P., Poddar, S., Acharya, A. K., & Kumar, S. D. (2021). Integration of IoT Based PLC for Smart Relaying of a PV-Fed Induction Motor Driven Conveyor Belt. In Proceeding of Fifth International Conference on Microelectronics, Computing and Communication Systems (pp. 155-165). Springer, Singapore.

[5] Wang, S., Wan, J., Imran, M., Li, D., & Zhang, C. (2018). Cloud-based smart manufacturing for personalized candy packing application. The Journal of Supercomputing, 74(9), 4339-4357.

[6] Dana H. Ballard; Christopher M. Brown (1982). Computer Vision. Prentice Hall. ISBN 978-0-13-165316-0.

[7] Huang, T. (1996-11-19). Vandoni, Carlo, E (ed.). Computer Vision : Evolution And Promise (PDF). 19th CERN School of Computing. Geneva: CERN. pp. 21–25. doi:10.5170/CERN-1996-008.21. ISBN 978-9290830955.

[8] Milan Sonka; Vaclav Hlavac; Roger Boyle (2008). Image Processing, Analysis, and Machine Vision. Thomson. ISBN 978-0-495-08252-1.

[9] Reinhard Klette (2014). Concise Computer Vision. Springer. ISBN 978-1-4471-6320-6.

[10] Linda G. Shapiro; George C. Stockman (2001). Computer Vision. Prentice Hall. ISBN 978-0-13-030796-5.

[11] Tim Morris (2004). Computer Vision and Image Processing. Palgrave Macmillan. ISBN 978-0-333-99451-1.

[12] Bernd Jähne; Horst Haußecker (2000). Computer Vision and Applications, A Guide for Students and Practitioners. Academic Press. ISBN 978-0-13-085198-7.

[13] Lasi, H., Fettke, P., Kemper, H. G., Feld, T., & Hoffmann, M. (2014). Industry 4.0. Business & information systems engineering, 6(4), 239-242.

[14] Frank, A. G., Dalenogare, L. S., & Ayala, N. F. (2019). Industry 4.0 technologies: Implementation patterns in manufacturing companies. International Journal of Production Economics, 210, 15-26.

[15] Tripathi, S., Shukla, S., Attrey, S., Agrawal, A., & Bhadoria, V. S. (2020). Smart industrial packaging and sorting system. In Strategic System Assurance and Business Analytics (pp. 245-254). Springer, Singapor.