Color Based Object Separation in Conveyor Belt Using PLC

Öznur Şengül[1], Sıtkı Öztürk[2], Melih Kuncan[3*]

[1]Department of Electronics and Communication Engineering, Kocaeli University, Kocaeli, Turkey (ORCID: 0000-0001-6250-8828)
[2]Department of Electronics and Communication Engineering, Kocaeli University, Kocaeli, Turkey (ORCID: 0000-0003-3804-5581)
[3†]Department of Electrical and Electronics Engineering, Siirt University, Siirt, Turkey (ORCID: 0000-0002-9749-0418)

ATIF/REFERENCE: Şengül, Ö., Öztürk, S., & Kuncan. M. (2020). Color Based Object Separation in Conveyor Belt Using PLC. *Avrupa Bilim ve Teknoloji Dergisi*, (18), 401-412.

Abstract

In today's production systems, industrial automation systems are preferred in order to provide high-efficiency and high-quality production, and therefore it is continuously developing. PLC is the basis of programmable industrial automation systems. In the study, in the PLC controlled conveyor belt system, the process of separating the objects according to their color by image processing was realized in real-time. In the image processing application, recognition is recognized by separating objects according to their colors. The process of sending objects separated by classes to different containers in the system is also done in real-time. The conveyor belt system used in the study was controlled by S7-1200 PLC. Image processing was performed in Matlab. Communication between Matlab and PLC has been used with the OPC interface application, which is widely used in the industry.

This study, experimental prototype of an automation system that is widely used in industrial applications, has been successfully made. Matlab program was used in the image processing part of the conveyor belt object separation system, PLC program was integrated into communication and control part, and it was seen in experimental studies that the system was working efficiently in real-time. It has been observed that the image processing algorithm for this study has been successfully performed to sort objects according to their colors, PLC-OPC system communication, and separation of objects according to specified positions. As a result of the studies on the experimental prototype, it is foreseen that this system can be applied to a commercial or industrial system.

Keywords: Image processing, image processing in Conveyor belt, Matlab and PLC controlled system, object separation on the conveyor belt, OPC.

PLC ve Operatör Panel Kullanarak Konveyör Bantta Renk Temelli Nesne Ayrıştırma

Öz

Günümüz üretim sistemlerinde yüksek verimli ve kaliteli üretimin sağlanabilmesi için endüstriyel otomasyon sistemleri tercih edilmekte bundan dolayı da sürekli gelişmektedir. Programlanabilir endüstriyel otomasyon sistemlerinin temelini ise PLC (Programlanabilir Mantıksal Denetleyici) oluşturmaktadır. PLC'ler sağladıkları işlem hızı, güvenilirlik, maliyet, programlama esnekliği, fiziksel büyüklük ve bakım gibi avantajlarla endüstride her geçen gün daha fazla tercih edilmektedir.

Bu çalışmada, endüstriyel sistemlerde gerçekleştirilen görsel denetleme sistemi uygulamalarına bir örnek gerçekleştirilmiştir. Çalışmada PLC kontrollü konveyör bant sisteminde, nesneleri görüntü işleyerek renklerine göre ayırma işlemi, gerçek zamanlı olarak gerçekleştirilmiştir. Görüntü işleme uygulamasında, nesneleri renklerine göre ayrıştırma yaparak tanıma gerçekleştirilmiştir. Sınıflara

---

[3] Sorumlu Yazar: Siirt Üniversitesi, Elektrik-Elektronik Mühendisliği, Siirt, Türkiye (ORCID: 0000-0002-9749-0418), melihkuncan@siirt.edu.tr

göre ayrılan nesneler sistemde farklı haznelere gönderilmesi işlemi de gerçek zamanlı olarak yapılmıştır. Çalışmada kullanılan konveyör bant sistemi S7-1200 PLC ile kontrol edilmiştir. Görüntü işleme ise Matlab ortamında gerçekleştirilmiştir. Matlab ile PLC'nin haberleşmesi ise endüstride yaygın olarak kullanılan OPC (OLE for Process Control) ara yüz uygulaması kullanılmıştır.

Yapılan bu çalışma sonucunda sanayi uygulamalarında yaygın olarak kullanılan bir otomasyon sisteminin başarılı şekilde deneysel prototipi yapılmıştır. Konveyör bant nesne ayrıştırma sisteminin görüntü işleme kısmında Matlab programı kullanılmış, haberleşme ve kontrol kısmında PLC programı entegre edilerek gerçek zamanlı olarak verimli bir şekilde sistemin çalıştığı yapılan deneysel çalışmalarda görülmüştür. Bu çalışma için yapılan görüntü işleme algoritmasının nesnelerin renklerine göre ayrıştırma işlemi, PLC-OPC sistem iletişimi ve nesnelerin belirlenen konumlara göre ayrıştırılması işlemleri başarıyla gerçekleştirildiği görülmüştür. Deneysel prototip üzerinde yapılan çalışmalar neticesinde, bu sistemin ticari veya endüstriyel bir sisteme uygulanabileceği öngörülmektedir.

Anahtar Kelimeler: Görüntü işleme, konveyör bantta görüntü işleme, Matlab ve PLC kontrollü sistem, konveyör bantta nesne ayrıştırma, OPC

1. Introduction

Automation systems take the place of manpower in production. The control of the working systems is carried out in a high quality and efficient manner without the need of manpower with automation systems. Programmable Logical Controller (PLC) is the basis of industrial automation systems. While PLCs offer a cost-effective solution for the control of complex systems, they offer advantages such as applicability to other systems quickly and easily, flexibility in system integration, resistance to electrical noise, vibration, and impact [1]. With the widespread use of automation systems, the workload demanded by PLCs has become too heavy to be performed by a single processor. In most cases, the meaning integrity and requirements of the process are overcome by using multiple PLCs. In the communication of these PLCs, it is mainly realized with the interface that serves high data rates such as Profinet, Modbus, and Profibus. With the additional modules added to the PLC, communication with RS232, RS485 protocols, and GPRS (General Packet Radio Service) networks can also be realized. In literature studies, object separation with PLC, conveyor belt system applications with PLC, image processing with conveyor belt, quality control in image processing system with conveyor belt, and so on. both academic and industrial studies are available in many different fields [2-7]. In the thesis prepared by Turgay Erdoğan in 2012, real-time object recognition and separation of objects were performed with the camera image taken from the band system using artificial neural networks, and studies were performed on various learning methods and performance [8]. In the article published by Sedat Metlek and Bayram Cetişli in 2012, the control of the band system created by using Festo experiment sets was carried out with the digital camera and image processing [9]. In the article published by Filiz Harmankuyu et al., The separation of three different objects in the tape system was controlled by using PLC and Arduino by separating the objects in Matlab environment according to their color and shape using a webcam [10].

In this study, as shown in Figure 1, in the PLC controlled conveyor belt system, the process of separating the objects according to their color was carried out in real-time. Another part of the study, image processing was carried out in Matlab environment. In the image processing stage, the object separation process is performed by taking the image of the objects on the conveyor belt in real-time and determining them according to the colors of the objects with the image processing algorithm. The purposes defined in Matlab are transferred to the PLC by OLE for Process Control (OPC) and carried out for object separation on the conveyor belt in accordance with the object information.
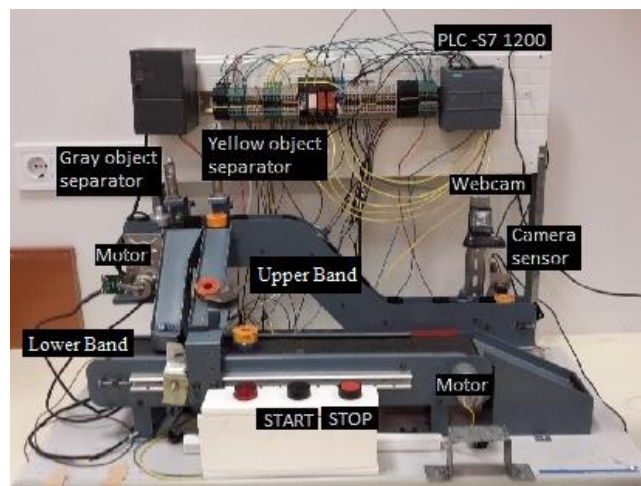

Figure 1. Test setup

2. Operator Panel and OPC

OPC (Ole for Process Control) is a standard for the transmission of real-time data. It is used for communication of SCADA () modules such as PLC, actuator, and sensor used in industrial automation systems. Prior to OPC, each additional module manufacturer defined a unique communication structure. In this case, a software company developing an HMI had to customize the software is developed for each different PLC brand and model. By standardizing OPC communication, for example, HMI software can be developed as the standard for all PLCs. The OPC server is a protocol converter that converts communication of modules that do not conform to the OPC standard to the OPC standard. The first OPC standards were built on Microsoft COM. After 2010, the OPC Foundation began publishing OPC UA (Unified Architecture) features. With OPC UA server,

- Web services and SOA (Service Oriented Architecture) architecture instead of COM,
- Collection of OPC data models (DA, A&E, HDA) under a single service model,
- Can be applied on platforms that contain embedded modules other than Microsoft,
- Reliable and efficient access to highly structured data,

are intended. The OPC server is, in short, a software application. Provides access to a set of data sources (read / write) or communication path. If the OPC client is, OPC is software used by an application to communicate with the server. The client acts as a data receiver, initiates communication, and provides control of the communication. Figure 2 shows the connection between the OPC server and the client.
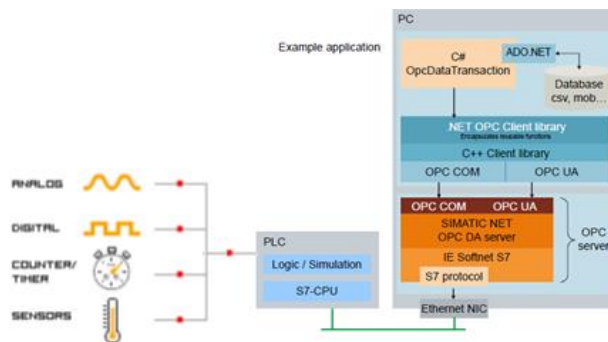

Figure 2. OPC server-client connection

The OPC DA (Data Access) standard provides communication between the client and server and performs real-time data read / write functionality. Matlab – OPC Toolbox is an OPC DA client and can be connected to compatible servers. This enables the OPC server to be accessed in Matlab and read and write OPC data directly.

2.1 OPC Configuration Between PLC-Matlab

To enable OPC communication between PLC-MATLAB, the TIA Portal, and Matlab programs must be pre-set. The necessary actions are given below. Since the PLC project will be created in TIA Portal, OPC configuration is performed in the TIA Portal project. After a new project is created in TIA, PLC and PC station are added to the project via the "add new device ında tab and 'controller' and 'PC station 'interfaces used in the system. In the beginning, the empty PC station is added to the "IE General" communication module and "OPC server from the Catalog Hardware Catalog" tab. As a result of this process, Figure 3 is given the PC station, which forms the OPC structure created in the project.
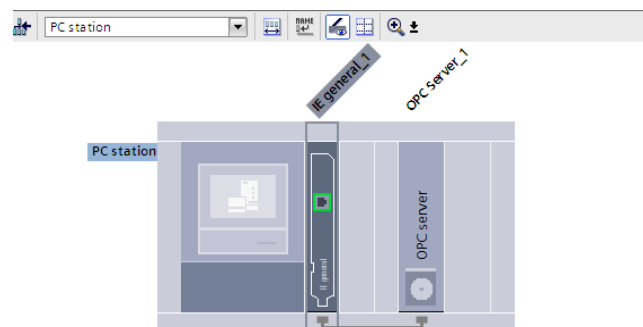

Figure 3. PC station with module added

In the project tree for the IP configuration of the PLC, the IP assignment is performed by following the "Properties >> General >> Profinet interface >> Ethernet address" tabs of the PLC's properties and clicking the "add new subnet" button to assign a subnet to the PROFINET interface. The "Properties >> General >> Profinet interface >> Protection" tabs are followed, and the Allow PLC, HMI, OPC access via PUT / GET communication tab is activated. Figure 4 also shows the interface for the IP assignment.
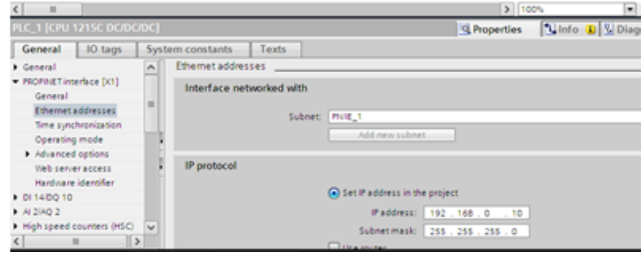

Figure 4. PLC IP assignment

For the PC's network connection configuration, TCP / IPv4 from the Ethernet features of the PC's network and internet settings is first turned on, and an IP assignment is performed on the same network as the PLC. Figure 5 shows the interface where the IP assignment of the PC added to the network is presented.
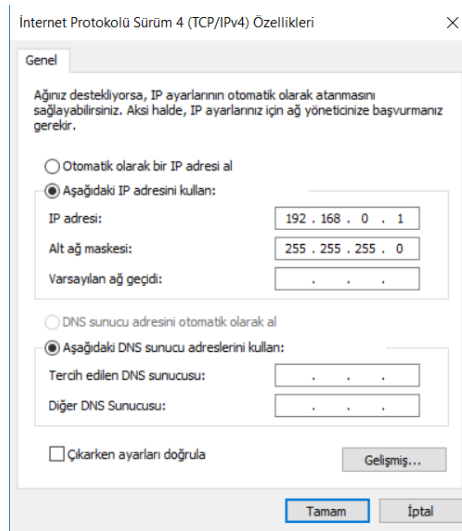

Figure 5. IP assignment of the PC for network connection

The IP assignment of the IE General communication module added to the PC station for OPC configuration is made from Properties >> General >> Profinet interface >> Ethernet address IE from the IE properties. Figure 6 shows the assignment of the IP address of the PC to the IE General module for OPC communication.
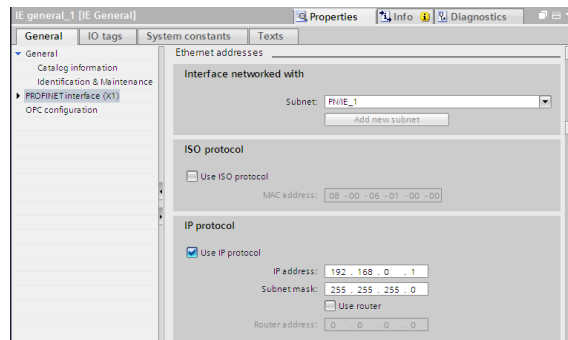

Figure 6. IP assignment of the IE General communication module

The IP address of the PC is also the same as the IE address, whereas the PLC shows the same network IP address, and the PN / IE_1

network of this connection is shown in Figure 7. PN / IE_1 network connection is established between PLC and PC.
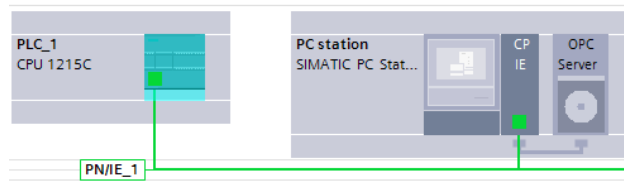

Figure 7. Connecting PLC and PC network Network connection between PLC and OPC

To connect the OPC server to the network, open the "Devices and networks" tab in the project tree and open the "Connections >> S7 connection" interface to make the connection from the toolbar.
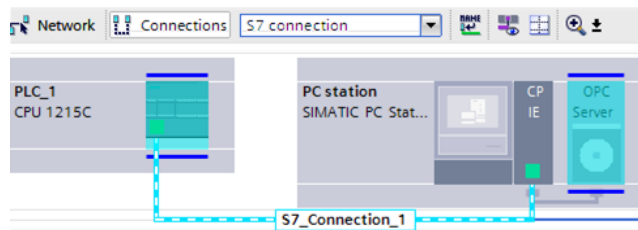

Figure 8. The network connection between PLC and OPC

In order to configure the variables between PLC and Matlab in OPC, the 'Properties >> General >> S7 >> OPC tags un tabs of the OPC server are followed in Figure 9, and the configured option is activated. The XDB file is then created, as shown in Figure 10, to create the variables. The XDB file is created by selecting >> Properties >> General >> XDB Configuration PC from the PC station properties.
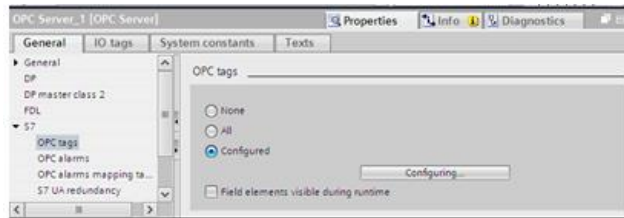

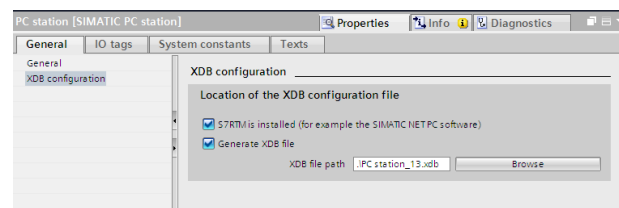Figure 9. OPC server variable configuration

Figure 10. XDB file configuration for OPC server variable configuration

In addition to the TIA Portal program, the Station configurator editor is opened from Simatic Net software for data access with the OPC server. With the interface shown in Figure 10, the XDB file containing IE General and OPC Server configurations is loaded by selecting 'import Station.' After these configurations are performed, and the PLC program is written, the PLC program is loaded to the PLC, and the OPC configuration is loaded to the OPC server. When loading into OPC, the OPC Station interface is created as in Figure 11. This indicates that the configuration of the OPC is correct.
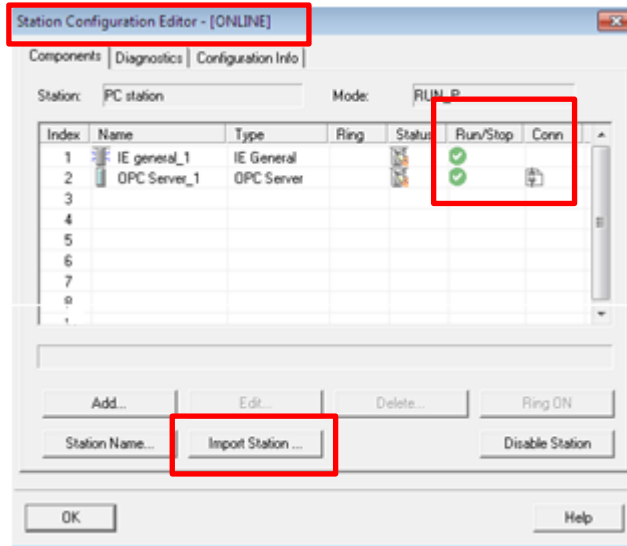
Figure 11. A structured view of the OPC Station interface

OPC Scout software is supported by Simatic Net and used as an OPC client. Using the OPC Scout, the data of the S7-1200 CPU is accessed via the OPC server. The OPC server consists of three objects: OPC Server, OPC Group, and OPC Item. The OPC Item object is the smallest unit used for reading and writing data and is associated with the client's variables. The OPC Item object is accessed using group objects and is not available directly. Figure 12 Access to item objects using OPC Scout is right in 'DA view 1'; the connection of the variables through OPC is correct, and the 'Quality' value should be good as shown in Figure 13. Thus, it shows that the necessary configuration for PLC communication with Matlab over OPC is performed correctly [11-12].
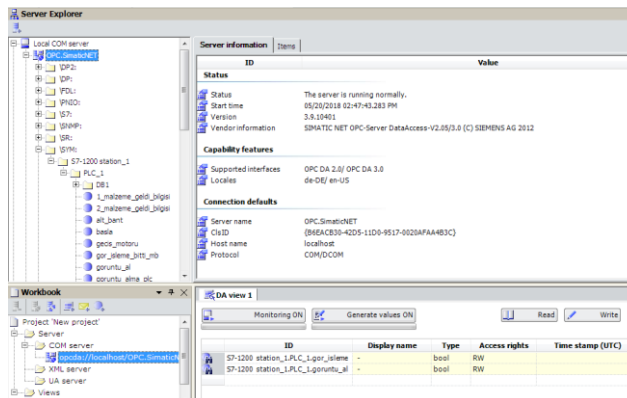

Figure 12. Access to item objects via OPC Scout

3. Object Separation with Image Processing

Image processing is a process other than "signal processing", a study for converting data into a readable format on another device after capture, measurement and evaluation, or transferring it from one electronic medium to another. Image processing is mainly used to process existing images that have been recorded, i.e. to modify, alienate or improve existing images and graphics. In recent years, image processing applications are widely used in many fields. Some of these are industry, health, military, security, quality control, object separation, etc. there are many different fields. In some applications, defining colors has a very important place. In order to define colors, mathematical models which are expressed as color space have been developed. Color spaces are created to represent all colors. Color spaces are designed in 3D. Because according to Grassmann's first law, which forms the basis of the science of colorimetry, three independent variables are needed to determine a color [13-16].
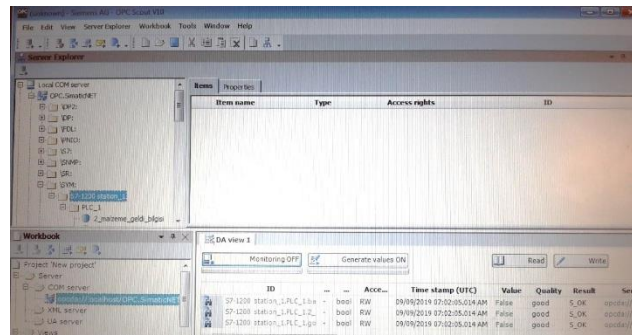
Figure 13. Access to item objects via OPC Scout

Extracting information about the object contained in an image from the image is one of the primary purposes of image processing. The most basic approach for separating an object from an image is the extraction of visual attributes (such as shape, color, texture). Color spaces are mathematical models used to define colors. Color spaces are designed in three dimensions to represent all colors. Because according to Grassmann's first law, three independent variables are needed to determine a color. The place of colors in the color space is determined according to these variables [17]. Each color space has its own standards for color rendering. While creating color spaces, it can be transformed into another color space by linear or non-linear methods [18]. Commonly used color spaces, RGB (Red, Green, Blue), HSV (Hue, Saturation, Value) and NTSC (National Television System Committee) are the color spaces [19-22].

In this study, it is preferred to use HSV color space because it is closer to the human vision mechanism and provides color determination by looking at color essence and saturation levels without being affected by brightness changes. The HSV color space consists of three bands: H (color extract), S (saturation), and V (brightness). It is the color extract that determines the color and is defined by angular values and takes values between 0°-360°. Saturation refers to the vividness of color. A low saturation value brings the color closer to gray tones, while a high saturation value makes the color more vivid. Brightness determines the ratio of white in color, i.e., the intensity of the color.
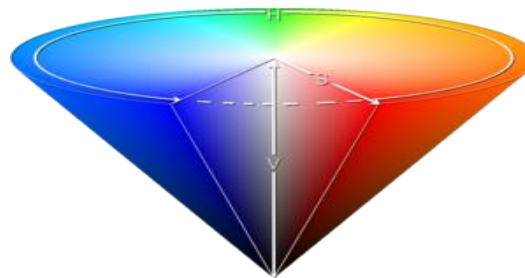

Figure 14. HSV color space

Equations 1, 2, and 3 are used to convert an image of the RGB color space to the HSV color space. In this study, two different materials, yellow and gray, were used, and other materials were identified as foreign materials. A web camera is installed in the apparatus comprising the tape system so as to see the beginning of the system from the top. The transfer of the received image to the computer was realized via the USB connection. The image processing algorithm was executed via Matlab. Since the received image is in RGB format, the HSV color space is provided first. After the threshold values were determined for the objects used in the application, the acquired images were scanned pixel by pixel, and the pixels of the corresponding color were identified and masked. In the obtained image, the noise effect is suppressed by filtering. The detection of the object was carried out by removing unconnected small pieces.

$$H = \begin{cases} \left(\dfrac{G-B}{V-\min(R,G,B)}\right)*60, & \text{if } V=R \text{ and } G \geq B; \\[3mm] \left(\dfrac{B-R}{V-\min(R,G,B)}+2\right)*60 & \text{if } V=R; \\[3mm] \left(\dfrac{R-G}{V-\min(R,G,B)}+4\right)*60 & \text{if } B=V; \\[3mm] \left(\dfrac{R-B}{V-\min(R,G,B)}+5\right)*60, & \text{if } V=R \text{ and } G < B; \end{cases} \qquad H \in [0,360] \tag{1}$$

$$S = \frac{V - \min(R,G,B)}{V} \quad S \in [0,1] \tag{2}$$

$$V = \min(R,G,B) \quad V \in [0,255] \tag{3}$$

The outputs of the program developed in Matlab are shown in Figure 15, Figure 16, and Figure 17.
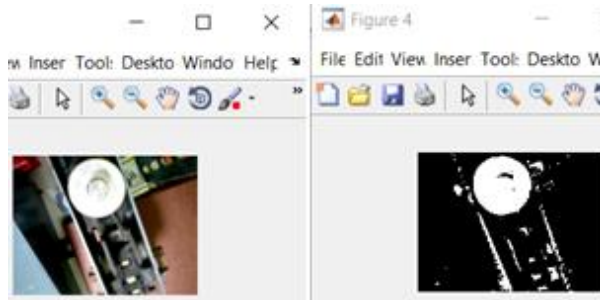

Figure 15. a) Original image taken from the camera for a gray object, b) Image resulting from thresholding and masking
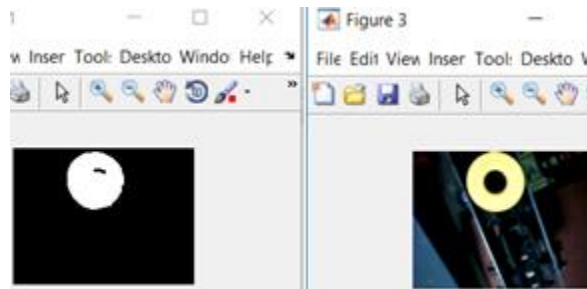

Figure 16. a) Detected gray object, b) Original image from camera for yellow object

4. Operating Principle of the System

Conveyor belts are transfer systems used to transport loads or equipment from the ground or at a certain height, at spaced distances. The conveyor belt system realized in this study is given in Figure 18. There are two conveyor belts in the system.
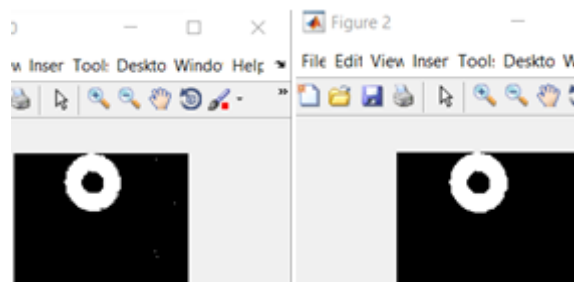

Figure 17. a) The image resulting from thresholding and masking, b) Yellow object detected
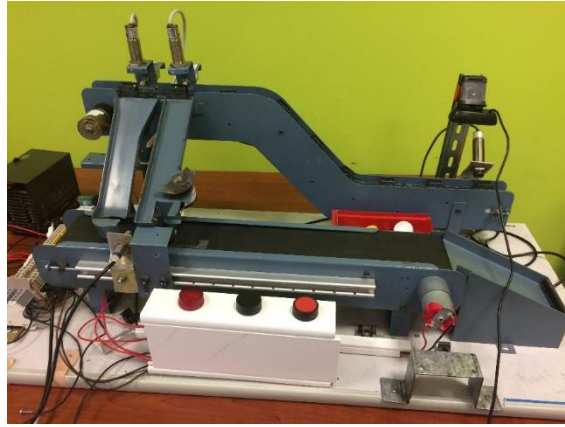
Figure 18. The first stage of the realized conveyor belt system

In work done, the start command given to the system by pressing the start button ensures that the operations take place sequentially. With the start command, the upper band starts to work, and the camera sensor at the entrance of the system informs whether the object has arrived. Upon receipt of the object arrival information, the upper band moves for a while and then stops. This ensures that the object is entirely in front of the camera and, thus, possible errors that can result from misalignment. When the object is completed in front of the camera, the command 'Take Image' is sent from PLC to Matlab via the OPC server. The color of the object in the received image is determined by the developed image processing algorithm, and the object information is sent to the PLC through the OPC server. The information obtained from the PLC is written to an array, such as first in first out (FIFO). The reason for writing to the array is that there is a possibility of a new object coming in front of the camera until the object is imaged. When the first sensor (yellow object separation sensor) sees the object, the yellow object is decomposed from the upper band to the lower band by activating the corresponding yellow object output of the yellow object while the object information in the recorded sequence. If the object information in the recorded array is not the yellow object information, the object continues to move through the tape. The second sensor (gray object separation sensor) sees the object this time. If the object information in the recorded sequence is the gray object, the corresponding gray object output of the PLC is activated, and the gray object is decomposed from the upper band to the lower band. If the object information in the recorded array is not the information of the gray object, the object continues to move through the tape and is parsed as faulty material. As the gray object coming to the bottom belt moves through the band, it collects the yellow object at the end of the band. Thus, the separation of objects occurs. These operations continue as long as the stop button is not pressed. The system implemented in Figure 19, and the operating algorithm of the system is given in Figure 20.
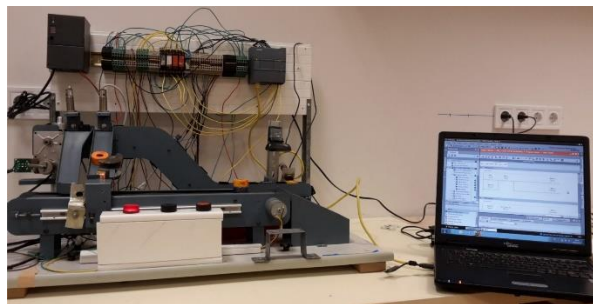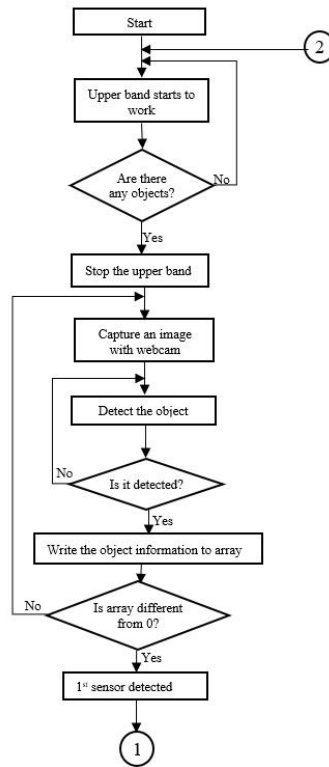

Figure 19. Operational status of the implemented system

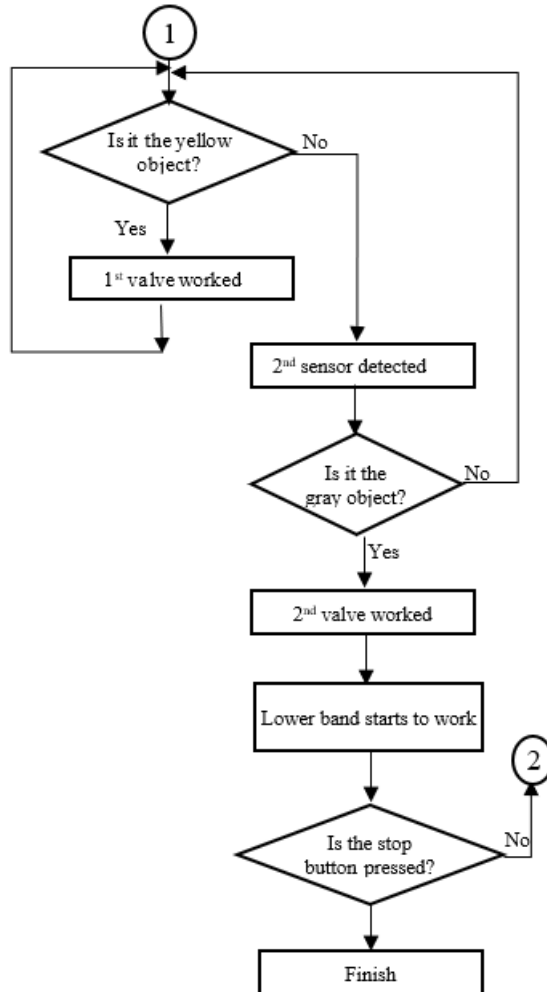Figure 20.a. Flow diagram of the system

Figure 20.b. Flow diagram of the system (continuation)

5. Conclusion and Suggestion

In this study, the data obtained as a result of the image processing algorithm developed in the Matlab environment is sent to Siemens S7-1200 PLC via the OPC server, and the control of the experimental setup is synchronized in real-time. The control of the designed band system, image acquisition, and separation are performed in real-time. The web camera used in the system was used to differentiate colored objects, and it was found to be sufficient to recognize the object successfully. It was observed that the developed image processing algorithm was distinguished according to the colors of the objects in 5, 6 seconds. It is thought that using a high-quality industrial camera to reduce this time will give more successful results. For this reason, it has been determined that using high-resolution cameras and different lenses for industrial applications, more prosperous and faster responses can be obtained than applications. It was determined that the image processing application developed in different color spaces could be compared to success rates. In addition, synchronous operation of the image processing application with PLC, which is a must in many industrial applications, has been realized.

As a result of these studies, PLCs, which are among the indispensable products in the industrial automation sector, were processed in real-time in a conveyor system, image processing, and OPC was operated synchronously on a prototype test apparatus. It has been seen that the image processing algorithm for this study has been successfully performed to sort objects according to their colors, PLC-OPC system communication, and separation of objects according to specified positions. As a result of the studies on the experimental prototype, it is foreseen that this system can be applied to a commercial or industrial system.

6. Acknowledgments

7. References

[1] Mrosko, M., & Miklovičová, E. (2012). Real-time implementation of predictive control using programmable logic controllers. International Journal of Systems Applications, Engineering & Development, 6(1), 106-113.

[2] Bakır, A., Güney, Ö.F., Kuncan, M., ve Ertunç., H.M., (2012). 3 Eksenli Robot Mekanizmasına Monte Edilmiş Bir Kamera Vasıtasıyla Farklı Rotasyon ve Boyutlardaki Geometrik Cisimlerin Tanımlanarak Vakum Tutucu İle Ayrılması. Otomatik Kontrol Ulusal Toplantısı, TOK 2012, October 11-13, 2012, Niğde, Turkey.

[3] Çubukçu, A., Kuncan, M., İmren, M., Erol, F., Ertunç, H.M., Öztürk, S., ve Kaplan, K., (2015). Görüntü İşleme İle 3 Eksenli Robot Mekanizması Üzerinde Nesne Ayırt Edilmesi ve Sıralanması, Object Discrimination and Sorting with Image Processing on 3-Axis Robot Mechanism, Otomatik Kontrol Ulusal Toplantısı, TOK 2015, September 10-12, 2015, Denizli, Turkey.

[4] Köse, İ., Öztürk, S., ve Kuncan, M., (2019). PANTOGRAPHY APPLICATION WITH REAL-TIME PLC BASED ON IMAGE PROCESSING IN GANTRY ROBOT SYSTEM. European Journal of Technique, 9(2), 219-229.

[5] Bargal, N., Deshpande, A., Kulkarni, R., & Moghe, R. (2016). PLC based object sorting automation. International Research Journal of Engineering and Technology, IRJET, 3(07).

[6] Selver, M. A., Akay, O., Alim, F., Bardakçı, S., & Ölmez, M. (2011). An automated industrial conveyor belt system using image processing and hierarchical clustering for classifying marble slabs. Robotics and Computer-Integrated Manufacturing, 27(1), 164-176.

[7] Jyothi, H. S., & Harsha, B. K. (2017). Design a Conveyor Based on Size and Color Separation of Product using Arduino UNO Microcontroller and Wireless Monitoring on Labview. International Journal Of Creative Research Thoughts (Ijcrt), 5(4), 2532-2539.

[8] Erdoğan, T. (2012). Hareketli konveyor üzerinde kamera görüntüsü ile nesne tanıma ve nesneleri yerine koyma uygulaması (Doctoral dissertation, DEÜ Fen Bilimleri Enstitüsü).

[9] Meltek, S., & Çetişli, B. (2012). Gerçek Zamanlı Endüstriyel Nesne Tanımada Kameraların Kullanılması. Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü Dergisi, 16(2), 212-217.

[10] Harmankuyu, F.,.Öğretmenoğlu, C., Sevgi, F., Şeker, B., Özer, S.M., Ünal, H.U., (2014). Görüntü İşleme Tabanlı Yürüyen Bant Sistemi ile Nesne Ayırma. 16. Otomatik Kontrol Ulusal Toplantısı, TOK 2014.

[11] Tekinalp, Z., Öztürk, S., & Kuncan, M. (2013). OPC Kullanılarak Gerçek Zamanlı Haberleşen Matlab ve PLC Kontrollü Sistem. Otomatik Kontrol Ulusal Toplantısı, TOK2013, September 26-28, 2013, p. 26-28 Malatya, Turkey.

[12] Erol, F., Emren, M., Öztürk, S., & Kuncan, M. PI Denetleyici İle Sıvı Seviye Kontrolünün Gerçek Zamanlı Olarak PLC İle Gerçeklenmesi Real Time PI Implementation on Liquid Level Control by means of PLC. Otomatik Kontrol Ulusal Toplantısı, TOK 2015, , September 10-12, 2015, Denizli, Turkey.

[13] Karakoç, Y., Öztürk, S., & Kuncan, M. (2012). Hareket Eden Renkli Nesnelerin Takibinin PID ile Gerçeklemesi. Otomatik Konrtol Ulusal Toplantısı, TOK 2012, October 11-13, 2012, Niğde, Turkey.

[14] E. Horoz, H. F. Öten, M. Kuncan, H. M. Ertunç, (2013). Kamera Yardımı ile Ayırt Edilen ve Tanımlanan Cisimlerin 3 Eksenli Robot Mekanizması ile Taşınması. Otomatik Kontrol Ulusal Toplantısı, September 26-28, 2013, p. 1236-1240, Malatya, Turkey.

[15] Yılmaz, İ., (2002). Renk Sistemleri, Renk Uzayları ve Dönüşümler. Selçuk Üniversitesi Jeodezi ve Fotogrametri Mühendisliği 30. Yıl Sempozyumu, Bildiri Kitabı, p. 340-350, 16-18 Ekim, 2002, Konya, Turkey.

[16] Sasidhar, K., Farooqi, S., Moin, M. A., & Sachin, M. (2018). Design and Development of a Colour Sorting Machine using PLC and SCADA.

[17] Agoston, G. A., (1987). Color Theory and Its Application in Art and Design", Second Completely Revised and Updated Edition, Springer-Verlag, Berlin.

[18] Kınık, M., Öztürk, M. S., (2017). Tipografik Tasarımda Rengin Okunurluğa Etkisi Konusunda Öğrenci Görüşlerinin İncelenmesi, Selçuk Ün. Sos. Bil. Ens. Der. 2017; (37): 327-335 - Grafik Tasarım/ Araştırma.

[19] Copîndean, R., Holonec, R., & Drăgan, F. (2018). The PLC Implementation of an Automated Sorting System using Optical Sensors. Acta Electrotehnica, 58(4).

[20] Teimouri, N., Omid, M., Mollazade, K., Mousazadeh, H., Alimardani, R., & Karstoft, H. (2018). On-line separation and sorting of chicken portions using a robust vision-based intelligent modelling approach. Biosystems engineering, 167, 8-20.

[21] Kamboj, D., & Diwan, A. (2019). Development of Automatic Sorting Conveyor Belt Using PLC. International Journal of Mechanical Engineering and Technology, 10(8).

[22] Latad, S. K., Dhole, P., Sonare, R., Kukadkar, A., & Mankar, P. (2019). Automatic Object Sorting Machine.