

**PERANCANGAN WORKSTATION SORTIR BERDASARKAN WARNA  
MENGUNAKAN DETEKSI WARNA HSV  
PADA SIMULASI AUTOMATED STORAGE AND RETRIEVAL SYSTEM  
DI KEPROFESIAN OTOMASI FAKULTAS REKAYASA INDUSTRI  
UNIVERSITAS TELKOM**

**Rizaldi Nurilhuda<sup>1</sup>, Haris Rachmat<sup>2</sup>, Tatang Mulyana<sup>3</sup>**

<sup>1,2,3</sup>Program Studi S1 Teknik Industri, Fakultas Rekayasa Industri, Universitas Telkom

<sup>1</sup>rizaldi.nurilhuda@gmail.com, <sup>2</sup>haris.bdg23@gmail.com, <sup>3</sup>tatang21april@gmail.com

**Abstract**

*In the era of globalization, the development of technology continues to evolve at a rapid pace, one of which is the development of automation. Automatic Storage And Retrieval System (AS/RS) is one application of automation systems in warehousing. Automation profession that is in the shade of the Faculty of Industrial Engineering has a function to improve the skills of students in the field of industrial automation, especially with regard to the design automation of manufacturing systems that stand alone or integrated. by function, it can be said about the learning model AS / RS according applied in one of the studies contained in this profession. For that needed a learning simulation that can be used as a simulator AS / RS in this profession. Simulation design AS / RS begins with the process of identifying goods using image processing to identify product based on colour.*

*This study began with the identification phase, initialization phase, creative phase, implementation and testing phases and stages of the conclusions and suggestions. System design process begins with design scenarios, and design a system that consists of designing simulation workstation sorting by color, design of image processing programs using MATLAB®, the design of PLC program with TIA (totally Integrated Automation) Portal V12, and the design of human machine interface (HMI) with Wonderware Intouch and the design of the communication between the HMI image processing program by using a comma separated Value(CSV) database.*

*Designed system will then be tested using a simulator based on test scenarios that have been designed for the system that is made to run in accordance with the purpose of research, the results of experiments performed successfully system designed sorting items by color.*

**Keywords :** Automation, AS/RS, Image Processing

---

**Abstrak**

Pada era globalisasi seperti saat ini perkembangan teknologi terus berkembang dengan pesat, salah satunya yaitu perkembangan otomasi. *Automated Storage And Retrieval System (AS/RS)* merupakan salah satu pengaplikasian sistem otomatisasi pada pergudangan. Keprofesian otomasi yang berada di bawah naungan Fakultas Rekayasa Industri memiliki fungsi untuk meningkatkan keahlian mahasiswa dalam bidang otomasi industri khususnya yang berkaitan dengan perancangan otomatisasi sistem manufaktur yang berdiri sendiri maupun terintegrasi. berdasarkan fungsi tersebut maka dapat dikatakan model pembelajaran mengenai AS/RS sesuai diaplikasikan dalam salah satu kajian yang terdapat dalam keprofesian ini. Untuk menunjang hal tersebut diperlukan suatu simulasi yang dapat digunakan sebagai simulator AS/RS pada keprofesian ini. Perancangan simulasi AS/RS ini dimulai dengan proses identifikasi barang berdasarkan metode pengolahan citra untuk mendeteksi barang berdasarkan warna.

Penelitian ini dimulai dengan tahap identifikasi, tahap inialisasi, tahap kreatif, tahap implementasi dan pengujian dan tahap kesimpulan dan saran. Perancangan sistem dimulai dengan perancangan skenario proses, dan perancangan sistem yang terdiri dari perancangan simulasi *workstation* penyortiran berdasarkan warna, perancangan program pengolahan citra dengan menggunakan MATLAB®, perancangan program PLC dengan TIA(*Totally Integrated Automation*) Portal V12, dan perancangan *human machine interface (HMI)* dengan *Wonderware Intouch* dan perancangan komunikasi antara program *image processing* dengan HMI dengan menggunakan *comma separated value (CSV)* database.

Sistem yang dirancang selanjutnya akan diuji coba menggunakan simulator berdasarkan skenario pengujian yang telah dirancang agar sistem yang dibuat dapat berjalan sesuai dengan tujuan penelitian, dari hasil uji coba yang dilakukan sistem yang dirancang berhasil melakukan penyortiran barang berdasarkan warna.

**Kata kunci :** otomasi, AS/RS, pengolahan citra

---

**I. Pendahuluan**

Keprofesian otomasi yang berada dibawah naungan Fakultas Rekaya Industri Telkom university memiliki fungsi untuk meningkatkan keahlian mahasiswa dalam bidang otomasi khususnya yang berkaitan dengan perancangan otomatisasi sistem manufaktur yang bisa berdiri sendiri maupun terintegrasi. berdasarkan fungsinya tersebut model pembelajaran AS/RS sebagai salah satu pengaplikasian sistem otomasi pergudangan sesuai diaplikasikan dalam kajian keprofesian ini. Simulasi AS/RS ayang akan dirancang terdiri dari proses identifikasi berdasarkan warna, proses handling barang menuju batch penyimpanan, dan peoses penyimpanan barang pada rak penyimpanannya.

Pada penelitian ini, peneliti akan berfokus pada perancangan proses identifikasi barang dengan menerapkan sistem machine vision dimana sistem ini melakukan identifikasi berdasarkan pada warna produk yang kemudian akan menyortir produk kedalam linanya masing-masing untuk dibawa oleh robotino® ke dalam rak.

Metode yang digunakan dalam proses identifikasi ini adalah deteksi warna HSV dengan menggunakan pendekatan batas (*boundary aproach*) dengan metode *adaptive treshold* untuk menentukan segmentasi warna dari masing-masing barang. Segmentasi warna ini berfungsi untuk menentukan nilai dari masing-masing warna dalam proses identifikasi warna.

**II. Tinjauan Pustaka**

**II.1.Machine Vision**

*Machine vision (computer vision)* merupakan sebuah teknologi yang digunakan sebagai alternatif indra penglihatan manusia dimana teknologi ini menggunakan kamera dan sistem komputerisasi untuk melakukan pekerjaannya[9].

Untuk membangun sebuah *vision system* secara garis besar terdapat beberapa hal yang harus di ketahui, hal tersebut yaitu sistem pencahayaan, Lensa Kamera, Jumlah kamera yang digunakan, Perangkat yang menghubungkan gambar dengan prosesor atau PC, *Image processor* atau PC, Tampilan yang akan menunjukkan proses kepada operator.

**II.2.Pengolahan Citra Digital**

Pengolahan citra digital (*Digital Image Processing*) adalah sebuah ilmu yang mempelajari tentang teknik-teknik dalam mengolah citra. Citra yang dimaksud adalah sebuah gambar diam/*photo* maupun gambar bergerak/*video*. Sedangkan digital mempunyai maksud bahwa pengolahan citra dilakukan secara digital menggunakan bantuan komputer[4].

Sebuah citra digital dapat diwakili oleh sebuah matriks dua dimensi  $f(x,y)$  yang terdiri dari M kolom dan N baris, dimana perpotongan antara kolom dan baris disebut piksel (pixel = picture element) atau elemen terkecil dari sebuah citra diperlihatkan pada persamaan II.1.

$$f(x,y) \approx \begin{bmatrix} f(0,0) & f(0,1) & \dots \\ f(1,0) & f(1,1) & \dots \\ \vdots & \vdots & \dots \end{bmatrix} \quad (1)$$

Suatu citra  $f(x,y)$  dalam fungsi matematis dapat dituliskan seperti persamaan II.2:

$$\begin{aligned} 0 \leq x \leq M-1 \\ 0 \leq y \leq N-1 \\ 0 \leq f(x,y) \leq G-1 \end{aligned} \quad (2)$$

Dimana :  
 M = jumlah piksel baris (*row*) pada array citra  
 N = jumlah piksel kolom (*column*) pada array citra  
 G = nilai skala keabuan (*graylevel*)

Besarnya nilai M, N dan G pada umumnya merupakan perpangkatan dari dua seperti terdapat pada persamaan II.3.

$$2^m, 2^n, 2^k = 2^i \dots \dots \dots (II.3)$$

dimana nilai m, n dan k adalah bilangan bulat positif. Interval (0,G) disebut skala keabuan (*grayscale*). Besar nilai G tergantung pada proses digitalisasinya. Biasanya keabuan 0 (nol) menyatakan intensitas hitam dan 1 (satu) menyatakan intensitas putih. Untuk citra 8 bit, nilai G sama dengan  $2^8 = 256$  warna (derajat keabuan).

**II.2.1. Jenis Citra Digital**

1. *Color Image* atau RGB (*Red, Green, Blue*)  
 Pada jenis *color image* masing-masing image memiliki warna tertentu, warna tersebut adalah merah (*red*), hijau (*Green*), biru (*Blue*). Jika masing masing warna memiliki range antara 0 - 255 maka pada *color image* memiliki  $255^3 = 16.581.375$  variasi warna dalam satu gambar.
2. *Black and White*  
 Citra *digital black and white (grayscale)* setiap pikselnya mempunyai warna gradasi mulai dari putih sampai hitam. Rentang tersebut berarti bahwa setiap piksel dapat diwakili oleh 8 bit, atau 1 byte. Rentang warna pada black and white sangat cocok digunakan untuk pengolahan file gambar. Salah satu bentuk fungsinya digunakan dalam kedokteran (X-ray). Gambar II.2 memperlihatkan contoh gambar yang memiliki format warna *black and white*.
3. *Binary Image*  
 Pada binary image Setiap piksel hanya terdiri dari warna hitam atau putih, karena hanya ada dua warna untuk setiap piksel, maka hanya perlu 1 bit per piksel (0 dan 1) atau apabila dalam 8 bit (0 dan 255), sehingga sangat efisien dalam hal penyimpanan. Gambar yang direpresentasikan dengan biner sangat cocok untuk teks (dicetak atau tulisan tangan), sidik jari (*finger print*), atau gambar arsitektur.

**II.2.2. Segmentasi Citra**

Segmentasi citra adalah suatu proses memecah suatu citra digital menjadi banyak segmen/bagian daerah yang tidak saling bertabrakan[5]. dalam konteks citra digital daerah hasil segmentasi tersebut merupakan kelompok piksel

yang bertetangga atau berhubungan. Segmentasi citra dapat dilakukan melalui beberapa pendekatan, pendekatan yaitu: Pendekatan batas (*Boundary approach*), Pendekatan tepi (*edge approach*), Pendekatan daerah (*region approach*).

Terdapat berbagai macam metode dalam melakukan segmentasi. Untuk melakukan segmentasi warna pendekatan yang dapat digunakan adalah pendekatan batas (*Boundary approach*). Metode-metode yang dapat digunakan dalam melakukan segmentasi berdasarkan warna yaitu *Static Threshold*, *Distance Threshold* dan *Dynamic Threshold*.

### II.2.3. Segmentasi Warna Dengan Menggunakan Deteksi Warna HSV

Untuk menggunakan metode segmentasi dengan deteksi warna HSV sebelumnya perlu dilakukan pemilihan sampel piksel acuan warna untuk menentukan *threshold* yang diinginkan[1]. Citra digital yang umumnya menggunakan model warna RGB perlu di konversi kedalam model warna HSV. Untuk mendapatkan segmen yang sesuai dengan warna yang diinginkan maka perlu ditentukan nilai toleransi untuk dimensi warna HSV. Kemudian nilai toleransi tersebut digunakan dalam perhitungan *adaptive threshold*. Hasil dari proses *threshold* tersebut akan menghasilkan segmen area dengan warna sesuai dengan apa yang diinginkan.

### II.3. Otomasi

Otomasi adalah suatu teknologi yang terkait dengan masalah penerapan sistem mekanik, elektronika dan sistem berbasis komputer yang bertujuan untuk pengoperasian dan pengendalian suatu sistem produksi[2]. Untuk menjalankan prosesnya sistem otomasi memiliki tiga elemen dasar yang harus dipenuhi. Tiga elemen dasar itu yaitu : sumber tenaga, program instruksi dan sistem pengendali.

### II.4. Sistem Kontrol Otomasi

Sistem kontrol menyebabkan suatu proses mencapai suatu fungsi yang telah didefinisikan dan bertujuan untuk melaksanakan beberapa operasi manufaktur[2]. Sistem kontrol otomasi dapat berupa *closed loop* atau *open loop*. Sistem *closed loop* disebut juga sebagai sistem kontrol umpan balik dan di sistem ini variabel *output* dibandingkan dengan parameter *input* dan setiap perbedaan antara keduanya digunakan untuk mendorong suatu kesepakatan lainnya, sedangkan sistem kontrol *open loop* beroperasi tanpa umpan balik, sistem kontrol *open loop* beroperasi tanpa mengukur variabel *output*. Dengan sistem *open loop*, selalu ada risiko bahwa aktuator tidak akan memiliki efek yang diinginkan pada proses. Akan tetapi keuntungan dari sistem *open loop* adalah umumnya rangkaian lebih sederhana dan lebih murah dibandingkan dengan sistem *closed loop*.

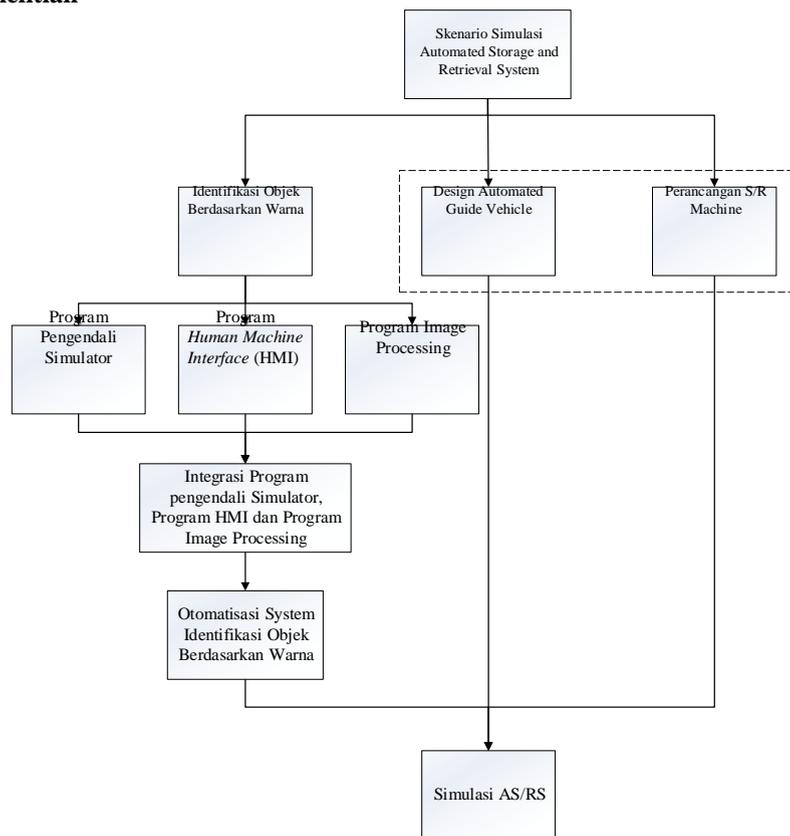
### II.5. Human Machine Interface (HMI)

*Human Machine Interface* (HMI) adalah sistem yang berfungsi untuk menghubungkan antara manusia dengan mesin. Hubungan antara manusia dan mesin ini adalah perubahan bentuk dari bahasa mesin kedalam bahasa manusia agar lebih mudah dipahami. Tugas dari HMI adalah membuat visualisasi sistem secara nyata untuk memudahkan operator dalam melakukan pengawasan proses yang sedang berlangsung. HMI dalam industri manufaktur berupa suatu tampilan layar komputer yang akan digunakan oleh operator mesin maupun pengguna yang membutuhkan data kerja mesin.

### II.6. CSV (*Comma Separated Values*)

CSV atau *comma separated value* merupakan sebuah *file* yang berisi nilai-nilai yang berbeda yang dipisahkan oleh pembatas (koma), yang bertindak sebagai tabel database atau bentuk peralihan dari tabel *database*. Dengan kata lain, *file CSV* adalah seperangkat *database* berupa baris dan kolom yang disimpan dalam *file* teks sehingga baris dipisahkan oleh baris baru sementara kolom dipisahkan oleh titik koma atau koma. Sebuah *file CSV* digunakan untuk mengangkut data antara dua *database* dari format yang berbeda melalui program komputer.

### III. Metodologi Penelitian



Gambar III. 1 Metodologi Penelitian

Perancangan simulasi AS/RS untuk keprofesian otomasi dimulai dengan pembuatan skenario proses pada simulasi *automated storage and retrieval system*. rancangan skenario AS/RS ini akan menjadi dasar untuk tahapan selanjutnya yaitu skenario proses indentifikasi objek berdasarkan warna, desain AGV, dan perancangan S/R machine. Dalam penelitian ini tahap yang menjadi fokus penelitian adalah perancangan skenario proses identifikasi objek berdasarkan warna.

perancangan proses identifikasi warna dibagi menjadi tiga tahap yaitu perancangan program pengendali simulator, perancangan program *human machine interface* (HMI) dan perancangan program *image processing*. Perancangan program pengendali simulator dibuat dengan menggunakan TIA Portal V12 dengan menggunakan bahasa pemrograman *ladder diagram* dimana program yang dirancang terdiri dari perancangan *material handling*, pendeteksi kehadiran objek, dan aktuator untuk memisahkan objek hasil identifikasi. Perancangan HMI dibuat dengan program *wonderware intouch* dimana program yang dirancang terdiri dari perancangan sistem *monitoring* sistem, perancangan sistem kontrol sistem, perancangan komunikasi antara HMI dengan program pengendali simulator dan program *image processing*. Perancangan program *image processing* dibuat dengan menggunakan aplikasi MATLAB® yang terdiri dari perancangan *graphic user interface*(GUI), perancangan program *image processing*, dan komunikasi antara program *image processing* dengan program HMI.

Setelah seluruh proses selesai maka akan diintegrasikan sehingga tercipta otomatisasi sistem identifikasi objek berdasarkan warna. Selanjutnya otomatisasi sistem identifikasi objek berdasarkan warna akan diintegrasikan dengan perancangan AGV serta perancangan S/R mechine sehingga akan dihasilkan sebuah model pembelajaran AS/RS.

## IV. Pembahasan

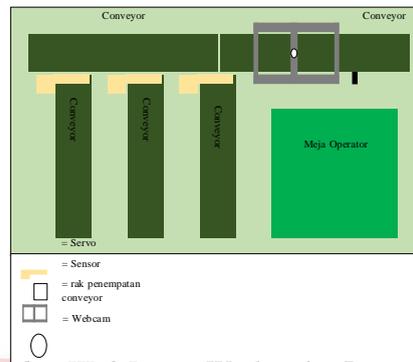
### IV.1. Skenario Proses

Dalam penelitian ini tahap perancangan skenario proses ini akan dijelaskan mengenai alur dari proses kerja dari workstation penyortiran berdasarkan warna dengan menggunakan deteksi warna HSV. Dalam skenario proses ini produk akan dikategorikan menjadi 3 jenis warna yaitu merah, kuning dan biru seperti yang dapat dilihat pada Gambar III.1 dengan ukuran 14 cm x 9 cm x 2 cm.



Gambar III. 2 Produk yang akan disortir

Proses yang dirancang selanjutnya akan disimulasikan pada sebuah workstation yang telah dirancang dimana, simulasi yang dibuat terdiri dari conveyor untuk membawa produk, *webcam* sebagai alat untuk mengambil gambar barang yang akan dibawa conveyor, sensor untuk mendeteksi benda yang melewatinya dan motor servo untuk memisahkan barang ke dalam lininya masing-masing. Layout dari simulasi proses yang dirancang dapat dilihat pada Gambar III.2.



Gambar III. 3 Layout Workstation Penyortiran

#### IV.1.1. Skenario Proses Penyortiran Berdasarkan Warna Secara Otomatis

Skenario proses dalam workstation penyortiran secara otomatis berdasarkan warna dapat dilihat pada Lampiran A. Dimana proses akan dimulai ketika sistem dijalankan sehingga monitoring system dan *conveyor* mulai berjalan. Monitoring system ini terdiri dari HMI yang memiliki akses pemantauan kamera yang akan digunakan untuk mengambil gambar produk yang berada di atas *conveyor*. Produk yang akan dibawa oleh conveyor kemudian dipisahkan berdasarkan warna.

#### IV.1.2. Skenario Proses Penyortiran Berdasarkan Warna Secara Manual

Skenario proses dalam workstation penyortiran secara otomatis berdasarkan warna ini dapat dilihat pada Lampiran B. proses secara manual dapat dilakukan ketika operator menekan tombol manual sehingga menonaktifkan seluruh sensor namun tidak menonaktifkan kamera sehingga proses pemisahan hanya dapat dilakukan dengan menekan tombol masing-masing servo yang terdapat pada HMI.

### IV.2. Penrancangan Sistem

#### IV.2.1. Perancangan Simulasi Workstation Penyortiran Berdasarkan Warna

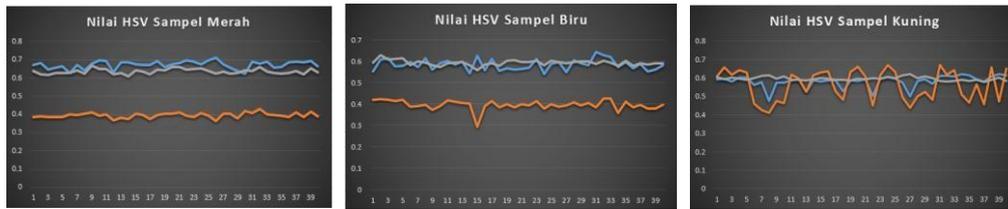
Simulator yang dirancang dalam penelitian ini terdiri dari beberapa komponen untuk menjalankan skenario proses yang dibuat. Adapun perangkat keras yang terdapat pada simulator ini dapat dilihat pada Lampiran C Selain kebutuhan perangkat keras terdapat juga kebutuhan perangkat lunak yang dibutuhkan dalam penelitian ini yaitu:

1. Sistem Operasi Windows XP Service Pack 3 Sebagai *operating system* dalam menjalankan dan merancang sistem dalam bentuk program PLC dan program HMI.
2. *Totally Integrated Automation (TIA PORTAL) V12* Sebagai *software* untuk membuat program PLC dan menjalankan berbagai eksekusi pada PLC.
3. *Wonderware System Management Console* Sebagai *server* berbasis *software* yang menghubungkan komunikasi data antara perangkat HMI dan perangkat PLC dengan menggunakan DASSIdirect.
4. *Wonderware InTouch 10.1* Sebagai *software* untuk merancang HMI (*Human Machine Interface*)
5. *MATLAB® R2013a* sebagai *software* untuk merancang program pengolahan citra.

#### IV.2.2. Perancangan Program Image Processing

Program identifikasi warna yang dirancang dengan menggunakan program *MATLAB®*. Perancangan dilakukan dengan membuat komponen image capturing, identifikasi nilai HSV dengan metode *adaptive threshold* Berikut adalah langkah-langkah dalam membuat program identifikasi barang berdasarkan warna.

1. Menghubungkan program dengan perangkat *webcam*  
Hardware yang digunakan dalam proses pengolahan citra ini adalah *webcam* Logitech C930E. *Webcam* digunakan untuk mengambil citra produk yang dibawa oleh *conveyor* secara dua dimensi saat proses identifikasi produk berdasarkan warna. Dengan menggunakan *webcam*, citra produk akan diambil lalu diolah untuk menentukan nilai HSV untuk menentukan kategori barang.
2. Pembuatan algoritma pemograman deteksi warna.  
Proses penyortiran berdasarkan deteksi warna HSV dilakukan dengan mengambil nilai HSV dari gambar yang di-*capture webcam* yang kemudian dibandingkan dengan nilai *adaptive threshold* yang didapatkan dari hasil *capture* produk sampel untuk menentukan nilai warna dari produk. Nilai HSV yang didapatkan dari hasil capture produk sampel dapat dilihat pada Gambar III.3



Gambar III. 4 Grafik Nilai HSV Sample

Gambar III.1 menampilkan grafik dari nilai HSV sampel dimana line berwarna biru menampilkan nilai *hue* dari sampel, warna orange menampilkan nilai *saturation* dan warna abu-abu menampilkan *value*. Nilai *adaptive threshold* yang digunakan didapatkan dari nilai terbesar dan nilai terkecil dari nilai HSV sampel. Nilai *adaptive threshold* yang digunakan untuk masing-masing warna adalah sebagai berikut :

Merah =  $0.62338 \leq H \leq 0.71335$ ;  $0.36555 \leq S \leq 0.43109$ ;  $0.60963 \leq V \leq 0.66702$

Biru =  $0.54132 \leq H \leq 0.62436$ ;  $0.29188 \leq S \leq 0.4238$ ;  $0.55891 \leq V \leq 0.62732$

Kuning =  $0.4739 \leq H \leq 0.62171$ ;  $0.40955 \leq S \leq 0.67009$ ;  $0.57769 \leq V \leq 0.62067$

Nilai tersebut selanjutnya akan dimasukkan sebagai pembandingan nilai HSV yang didapatkan dari hasil capture produk yang dibawa *conveyor* untuk menentukan kategori produk. Program MATLAB® yang dibuat dalam penelitian ini dapat dilihat pada Lampiran D

#### IV.2.3. PLC

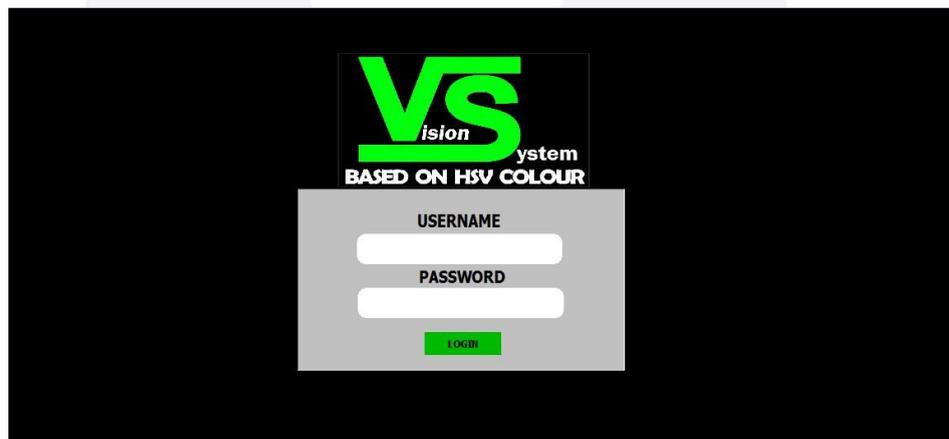
Dalam penelitian ini controller yang digunakan adalah PLC Siemens S7-1200. Bahasa pemrograman yang digunakan dalam PLC adalah *ladder diagram* dengan menggunakan TIA PORTAL V12. Program *ladder diagram* yang dibuat dalam penelitian ini dapat dilihat pada Lampiran E.

#### IV.2.4. Human Machine Interface

HMI yang telah dibuat menggunakan *software Wonderware Intouch* diharapkan sesuai dengan skenario yang telah dibuat pada skenario perancangan HMI. kelemahan dari HMI yang dibuat akan dianalisis untuk dilakukan perbaikan pada penelitian selanjutnya. Pengujian yang dilakukan terhadap HMI yang dibuat adalah sebagai berikut:

##### 1. Interface Login

*Interface login* yang dibuat dalam perancangan HMI pada penelitian ini dapat dilihat pada Gambar III.4

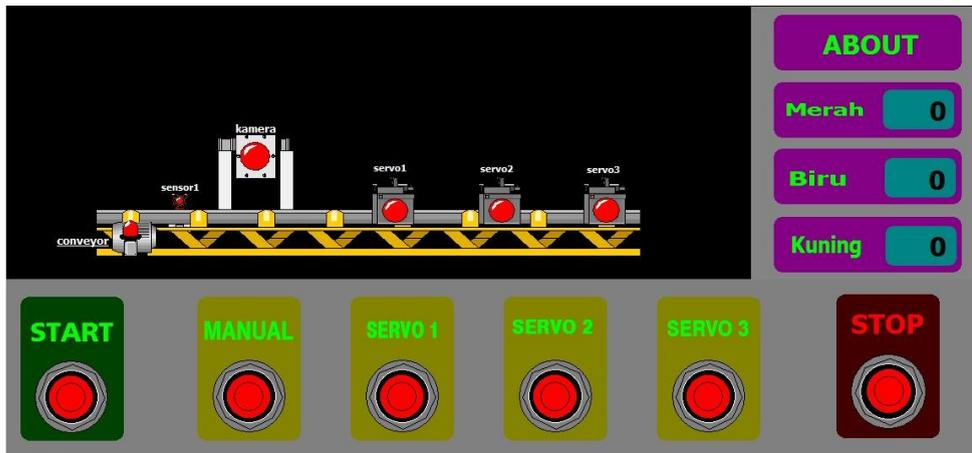


Gambar III. 5 interface log in

Untuk masuk kedalam *interface monitoring* dan *control user* terlebih dahulu harus melakukan *log in*. untuk melakukan *log in* user dapat memasukkan *username* serta *password* pada kolom *username* dan *password*. Jika user memasukkan *username* dan *password* yang salah maka *interface monitoring* dan *control* tidak dapat dibuka. Apabila user memasukkan *username* dan *password* dengan benar sesuai dengan *database username* dan *password* yang terdapat pada *user account Wonderware Intouch* maka user dapat menekan tombol “LOGIN” dan *interface monitoring* dan *control* akan muncul.

##### 2. Interface Monitoring dan Control

Keluaran yang dihipkan dari perancangan HMI ini adalah user dapat melakukan kontrol dan monitoring sistem ppenyortiran berdasarkan deteksi warna HSV sesuai dengan skenario yang dibuat. *Interface monitoring* dan *control* hasil rancangan dapat dilihat pada Gamber V.9.



Gambar III. 6 interface monitoring dan kontrol

3. Interface About

Interface About menampilkan tentang perancang sistem pada workstation penyortiran berdasarkan warna. interface ini dapat diakses dengan cara menekan tombol “ABOUT” pada interface monitoring dan control. interface about yang dirancang pada penelitian ini dapat dilihat pada Gambar V.10.



Gambar III. 7 Interface About

IV.3. Pengujian Sistem Workstation Penyortiran Berdasarkan Warna

Berdasarkan hasil rancangan, dapat ditentukan apakah sesuai dengan skenario yang dibuat atau belum. Pengujian sistem yang dilakukan dengan melihat keakuratan dari sistem yang dirancang untuk menyortir produk berdasarkan warnanya.

Tabel III. 1 Hasil Pengujian Sistem Hasil Rancangan

| Warna  | AKURASI SENSOR | AKURASI KATEGORI | AKURASI AKTUATOR | AKURASI SORTIR | AKURASI TOTAL |
|--------|----------------|------------------|------------------|----------------|---------------|
| Merah  | 100%           | 97%              | 100%             | 87%            | 96%           |
| Biru   | 100%           | 97%              | 100%             | 100%           | 99%           |
| Kuning | 100%           | 100%             | 100%             | 100%           | 100%          |

Pengujian dilakukan dengan melihat keakuratan dari proses yang dibuat berdasarkan skenario untuk masing-masing warna. Berdasarkan hasil pengujian terhadap sistem hasil rancangan yang diperlihatkan pada Tabel III.1 terlihat bahwa sistem workstation penyortiran berdasarkan warna yang dirancang sudah berhasil dirancang.

Dalam mengidentifikasi produk keakuratan sensor dalam mendeteksi objek berwarna merah, biru dan kuning sebesar 100% hal ini menunjukkan bahwa selama proses pengujian barang yang dibawa oleh conveyor dapat terdeteksi dan membuat conveyor berhenti bergerak untuk melakukan capture gambar produk. Gambar yang telah diambil kemudian diidentifikasi kategori warnanya berdasarkan nilai adaptive threshold yang dirancang.

Keakuratan nilai adaptive threshold untuk kategori produk berwarna merah yang dirancang memiliki tingkat keakuratan sebesar 97%, untuk produk berwarna biru sebesar 97% dan untuk produk berwarna kuning sebesar 100%. Nilai itu didapatkan berdasarkan perbandingan tingkat kegagalan dengan jumlah pengujian pada masing-masing warna. tingkat keakuratan <100% didapatkan karena nilai HSV yang didapatkan berada diluar range adaptive threshold yang dirancang hal ini dapat terjadi karena perbedaan fokus dari gambar hasil capture webcam yang digunakan.

Keakuratan aktuator berdasarkan hasil pengkategorian barang didapatkan sebesar 100% untuk seluruh kategori produk. nilai ini didapatkan karena selama proses pengujian aktuator dapat bergerak sesuai dengan hasil kategori barang baik ketika hasil kategori salah maupun benar. Dalam melakukan penyortiran produk yang dibawa *conveyor* kedalam *line*-nya masing masing nilai keakuratan sistem yang dirancang sebesar 87% untuk produk berwarna merah nilai ini didapatkan karena pada saat melakukan pengujian produk yang dibawa gagal untuk dipindahkan hal ini dikarenakan posisi penyortiran produk berwarna merah berada pada sambungan antar *conveyor* sehingga mempengaruhi laju dari produk yang dibawa *conveyor*. Sedangkan untuk produk berwarna biru dan kuning tingkat akurasi yang didapatkan sebesar 100% hal ini membuktikan bahwa produk berwarna biru dan kuning yang dibawa *conveyor* berhasil disortir kedalam *line*-nya masing-masing.

Akan tetapi masih terdapat kekurangan terhadap sistem hasil rancangan yaitu sebagai berikut :

1. Waktu yang diperlukan untuk melakukan identifikasi produk berdasarkan deteksi warna HSV yang dimulai ketika *conveyor* berhenti untuk *webcam* meng-*capture* gambar produk yang dibawa *conveyor* sampai *conveyor* mulai bergerak kembali adalah 2 detik hal ini dikarenakan dibutuhkan waktu untuk *webcam* mengatur fokus lensa sebelum meng-*capture* gambar.
2. Sistem yang dirancang belum bisa menghitung jumlah produk yang berhasil disortir dengan tepat. Hal ini terjadi karena trigger yang digunakan sebagai *counter* memiliki jeda waktu yang berbeda-beda.
3. Trigger yang digunakan untuk servo bergerak CW sebesar 90° (menutup dan membawa produk ke *line*-nya) adalah timer sehingga perubahan kecepatan yang dilakukan pada *conveyor* dapat mempengaruhi keakuratan sistem hasil rancangan.
4. Posisi jatuh produk yang telah disortir tidak dapat terprediksi sehingga produk tidak tepat jatuh kedalam *slider* yang digunakan untuk menampung produk yang akan dibawa AGV.
5. Sistem penyortiran berdasarkan warna yang dirancang hanya dapat melakukan penyortiran satu persatu, dimana produk yang masuk untuk disortir hanya dapat dilanjutkan ketika produk sebelumnya telah disortir.

## V. Kesimpulan

Berdasarkan hasil pengujian sistem yang telah dirancang maka ditarik kesimpulan sebagai berikut:

1. Sistem otomasi penyortiran produk berdasarkan warna dengan menggunakan deteksi warna HSV berhasil dilakukan. Hal ini dapat diketahui berdasarkan hasil percobaan yang dilakukan sebanyak tiga puluh kali pada masing-masing warna. berdasarkan hasil pengujian tersebut didapatkan nilai keakuratan sistem dalam melakukan penyortiran produk berwarna merah sebesar 96%, produk berwarna biru sebesar 99%, dan produk berwarna kuning sebesar 100%.
2. Pada penelitian ini komunikasi antara program PLC dengan HMI dilakukan dengan menggunakan DASIIDIRECT dan komunikasi antara HMI dengan Program *image processing* menggunakan file csv. Komunikasi yang dirancang dapat berjalan dengan lancar sehingga program penyortiran berdasarkan warna dapat berjalan sesuai dengan skenario yang dibuat.

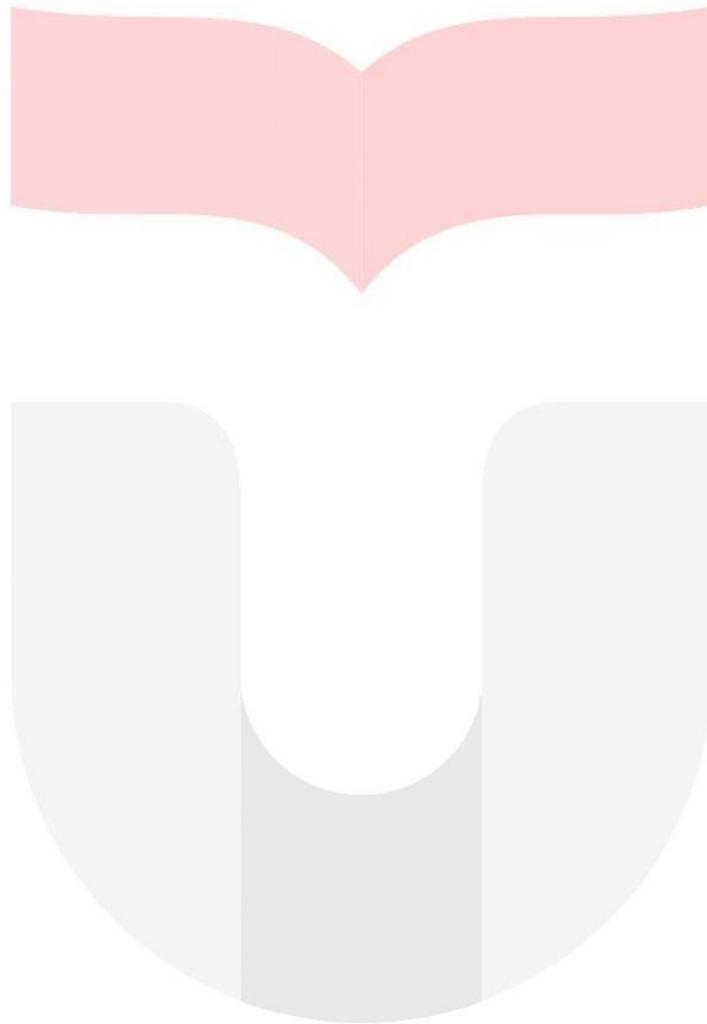
terdapat saran untuk penelitian selanjutnya yang akan dilakukan :

1. Sebaiknya digunakan sensor sebagai trigger untuk servo bergerak CW 90° sehingga dalam proses penyortiran nilai keakuratan dapat meningkat ketika terdapat perubahan kecepatan *conveyor*.
2. HMI yang dirancang terdapat sistem *counter* yang digunakan untuk mengetahui jumlah produk yang telah disortir.
3. *Layout conveyor* yang dirancang pada penelitian ini berdasarkan pada skenario simulasi AS/RS. Untuk meningkatkan akurasi proses sortir, *layout conveyor* dapat dirubah sehingga akurasi penyortiran dapat meningkat pada penelitian selanjutnya.
4. Sebaiknya terdapat *database* yang dapat digunakan sebagai *report* barang hasil sortir dan sebagai trigger untuk AGV mulai beroperasi sehingga sistem simulasi AS/RS dapat terintegrasi.
5. Sebaiknya perancangan sistem penyortiran dilakukan secara *continue*, sehingga sistem penyortiran dapat lebih optimal dari segi waktu penyortiran serta terdapat analisis kecepatan yang digunakan dalam perancangan workstation penyortiran berdasarkan warna.
6. Terdapat alat bantu untuk merubah posisi produk hasil sortir agar dapat masuk kedalam slider penampungan produk yang akan dibawa leh AGV.

## Daftar Pustaka

- [1]Giannakopoulos, T. (2008). <http://cgi.di.uoa.gr/>. Retrieved from Theodoros Giannakopoulos: Personal Web Page: <http://cgi.di.uoa.gr/~tyiannak>
- [2]Groover, M. P. (2005). *Otomasi, Sistem Produksi, dan Computer Integrated Manufacturing*. Surabaya: Guna Widya.
- [3]Jain, R., Kasturi, R., & Schunck, B. G. (1995). *Machine Vision*. McGraw-Hill, Inc.
- [4]Kusumanto, R., & Tompunu, A. N. (2011). Pengolahan Citra Digital Untuk Mendeteksi Obyek Menggunakan Pengolahan Warna Model Normalisasi RGB. *Seminar Nasional Teknologi Informasi & Komunikasi Terapan 2011 (Semantik 2011)*.
- [5]Putranto, B. Y., Hapsari, W., & Wijana, K. (2010). Segmentasi Warna Citra Dengan Deteksi Warna HSV Untuk Mendeteksi Objek. *JURNAL INFORMATIKA, Volume 6, Nomor 2*.

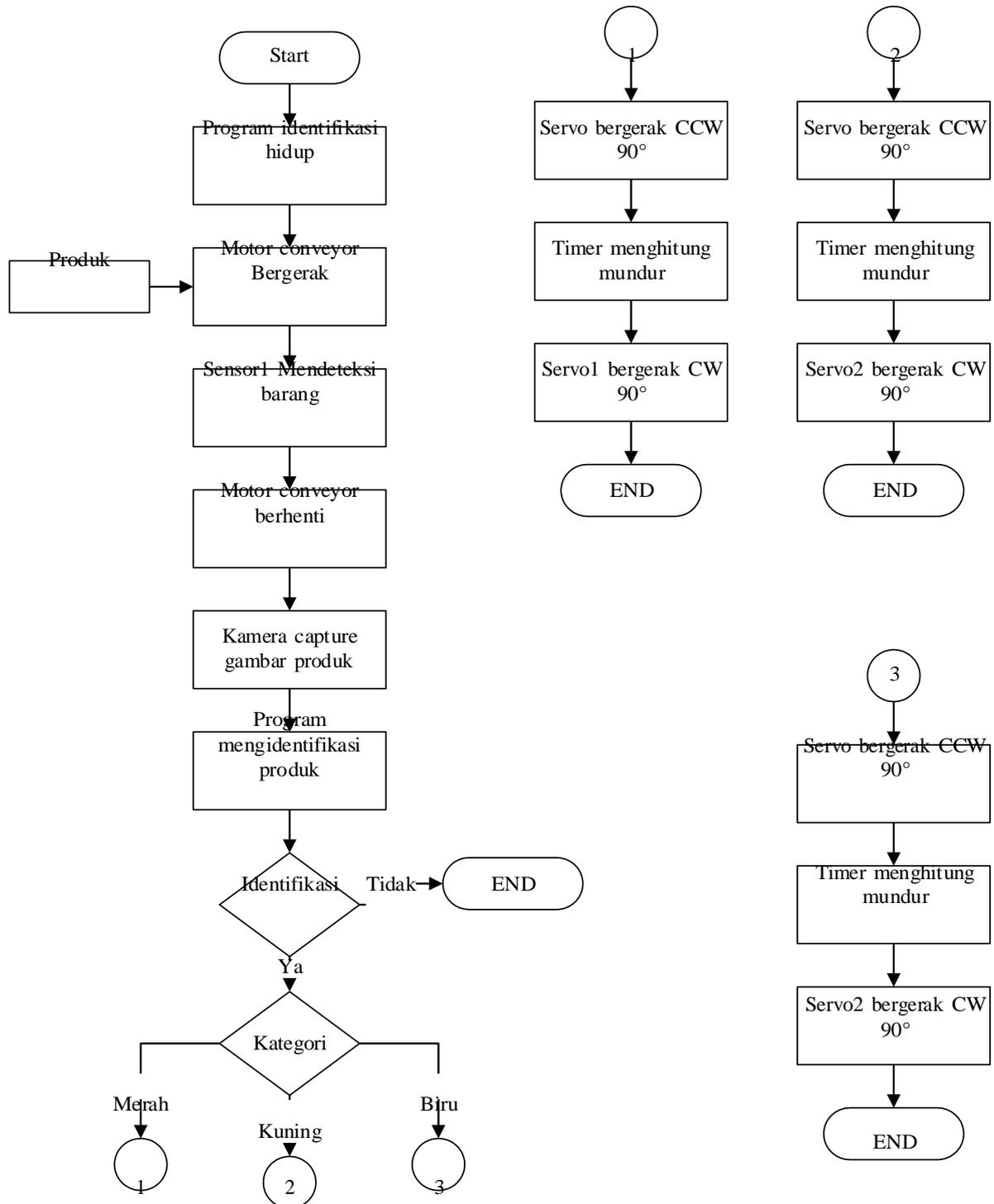
- [6]Roodbergen, K. J., & Vis, I. F. (2008). *A Survey of Literature On Automated Storage and Retrieval Sytem*. European Journal Of Operational Research.
- [7]Soemana, A. P. (2012). *Manager Product and Solution PT. Siemens Indonesia*.
- [8]Sugiyono. (2010). *Metode Penelitian Pendidikan Pendekatan Kuantitatif, Kualitatif, dan R&D*. Bandung: CV.Alfabeta.
- [9]Vision Machine. (2007). *U.I Association, Vision System Handbook*.
- [10]Wicaksono, H. (2012). *SCADA Software dengan Wonderware InTouch*. Yogyakarta: Graha Ilmu.



# **Lampiran A**

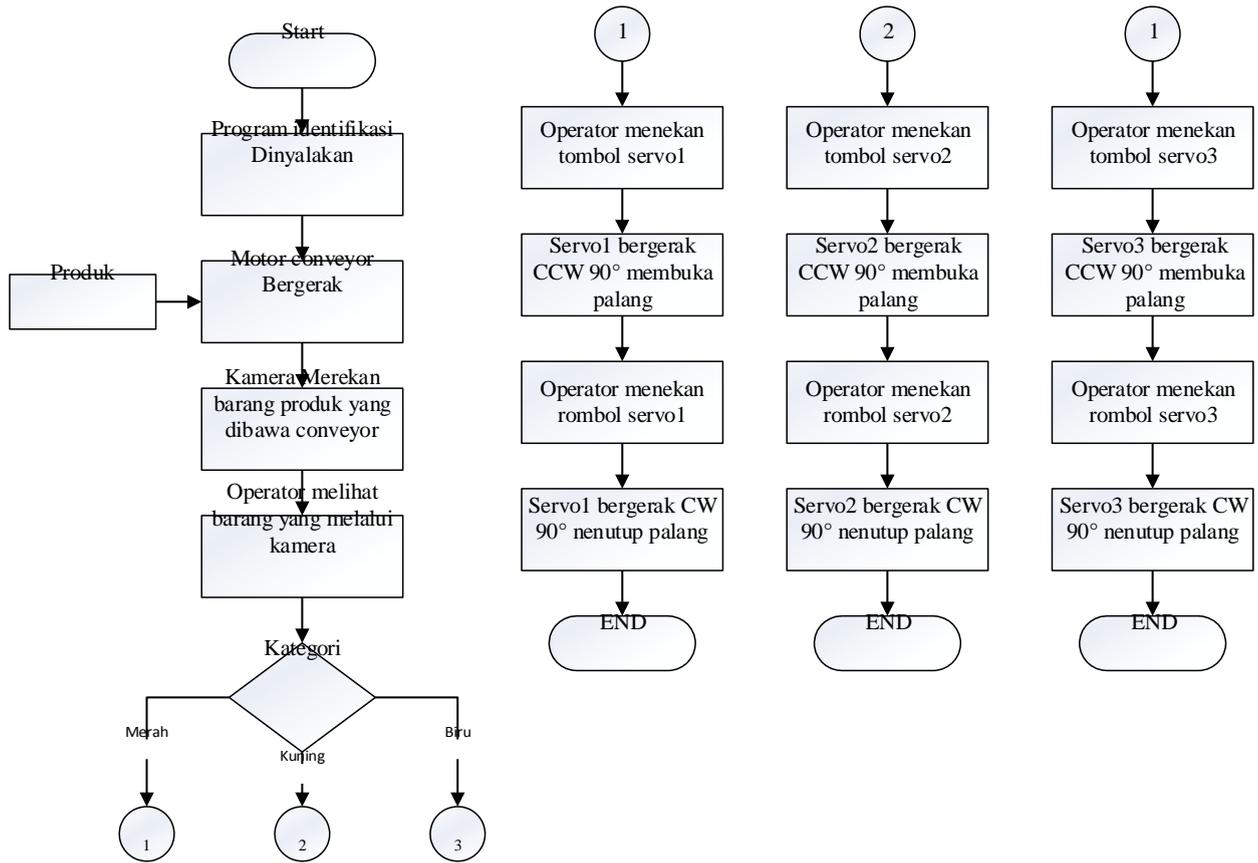
---

Skenario Proses Penyortiran Otomatis



# **Lampiran B**

**Skenario Proses Penyortiran Manual**



# **Lampiran C**

**Kebutuhan Perangkat Keras**

| Komponen                              | Spesifikasi  | Keterangan   |
|---------------------------------------|--|--|
| Motor Conveyor YUEMA type Y3A56B      | <ol style="list-style-type: none"> <li>1. Arus 3 phase</li> <li>2. 220-420V</li> <li>3. Max 50Hz</li> <li>4. Max 0.11 KW</li> <li>5. Max 14 HP</li> </ol>                                    | Conveyor berfungsi untuk membawa barang yang nantinya akan di sortir   |
| Photo Sensor Autonics BMS2M-MDT-P     | <ol style="list-style-type: none"> <li>1. Power 12-24 v</li> <li>2. Retroreflective</li> <li>3. Detecting distance 0.1 -2m</li> <li>4. Control output PNP</li> </ol>                         | Berfungsi sebagai pendeteksi keberadaan barang dan digunakan sebagai input untuk <i>webcam</i> mengambil gambar.       |
| <i>Webcam</i> Logitech C930E          | <ol style="list-style-type: none"> <li>1. Video resolution HD 1080p</li> <li>2. Photo Quality 15 MP</li> <li>3. Autofocus</li> <li>4. Hi-Speed USB 2.0</li> <li>5. Carl Seiz Lens</li> </ol> | Digunakan sebagai pengganti fungsi mata manusia, yaitu menangkap citra yang nantinya di proses dengan aplikasi MATLAB. |
| Motor Servo                           | <ol style="list-style-type: none"> <li>1. 39.5x20x35.6mm</li> <li>2. Weight 41g</li> <li>3. Speed 0.230 sec/60°</li> <li>4. Torque 3.4 kg-cm</li> </ol>                                      | Digunakan untuk menggerakkan palang pemisah untuk memindahkan barang ke dalam masing-masing <i>line</i> .              |
| Inverter Industrial System SV008iC5-1 | <ol style="list-style-type: none"> <li>1. Input 200-230v 1 phase</li> <li>2. Output 0-input V 3 phase</li> <li>3. 5 A</li> <li>4. 1 HP / 0.75 Kwh</li> </ol>                                 | Digunakan untuk merubah tegangan 3 phase menjadi 1 phase   |
| Push Button Shemesco SC16             | <ol style="list-style-type: none"> <li>1. Push button lock</li> <li>2. Res. AC 250V 3A</li> <li>3. Res. DC 30V 5A</li> </ol>   | Digunakan sebagai tombol pada simulator.   |

# **Lampiran D**

**Program Image Processing MATLAB®**

```

function varargout = VisionSystem(varargin)
% Last Modified by GUIDE v2.5 07-Jun-2015 23:19:33

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',   gui_Singleton, ...
                  'gui_OpeningFcn',  @VisionSystem_OpeningFcn, ...
                  'gui_OutputFcn',   @VisionSystem_OutputFcn, ...
                  'gui_LayoutFcn',   [] , ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before VisionSystem is made visible.
function VisionSystem_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to VisionSystem (see VARARGIN)

% Choose default command line output for VisionSystem
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

figure(handles.figure1); axes(handles.axes1);
vid=videoinput('winvideo',1,'YUY2_352x288');
hImage=image(zeros(288,352,1),'parent',handles.axes1);
preview(vid,hImage);

tic;
while toc <=99999999999;
    pause (1);
b=csvread('E:\TUGAS AKHIR\capture.csv');

if b==999;
    a=getsnapshot(vid);
    hsv=rgb2hsv(a);
    H=hsv(:, :, 1);
    H=mean(mean(H), 2);
    S=hsv(:, :, 2);
    S=mean(mean(S), 2);
    V=hsv(:, :, 3);
    V=mean(mean(V), 2);

    if H >= 0.62338 && H <= 0.71335 && S >= 0.36555 && S <= 0.43109 && V >=
0.60963 && V <= 0.66702;
        obj = 1;
        csvwrite('E:\TUGAS AKHIR\kategori.csv',obj);
    end
end
end

```

```
elseif H >= 0.54132 && H <= 0.62436 && S >= 0.29188 && S <= 0.4238 && V >=
0.55891 && V <= 0.62732
    obj = 2;
    csvwrite('E:\TUGAS AKHIR\kategori.csv',obj);
elseif H >= 0.4739 && H <= 0.62171 && S >= 0.40955 && S <= 0.67009 && V >=
0.57769 && V <= 0.62067
    obj = 3;
    csvwrite('E:\TUGAS AKHIR\kategori.csv',obj);
else
    obj = 0;
    csvwrite('E:\TUGAS AKHIR\kategori.csv',obj);
end
nilai=[H,S,V]
csvwrite('E:\TUGAS AKHIR\nilai.csv',nilai);
end
end
```

# **Lampiran E**

**Program PLC**

Totally Integrated Automation Portal

TA\_RZD(man auto) / PLC\_2 [CPU 1212C AC/DC/Rly] / Program blocks

Main [O81]

Main Properties

**General**

|      |      |        |      |                 |
|------|------|--------|------|-----------------|
| Name | Main | Number | Type | OB.ProgramCycle |
|------|------|--------|------|-----------------|

Language LAD

**Information**

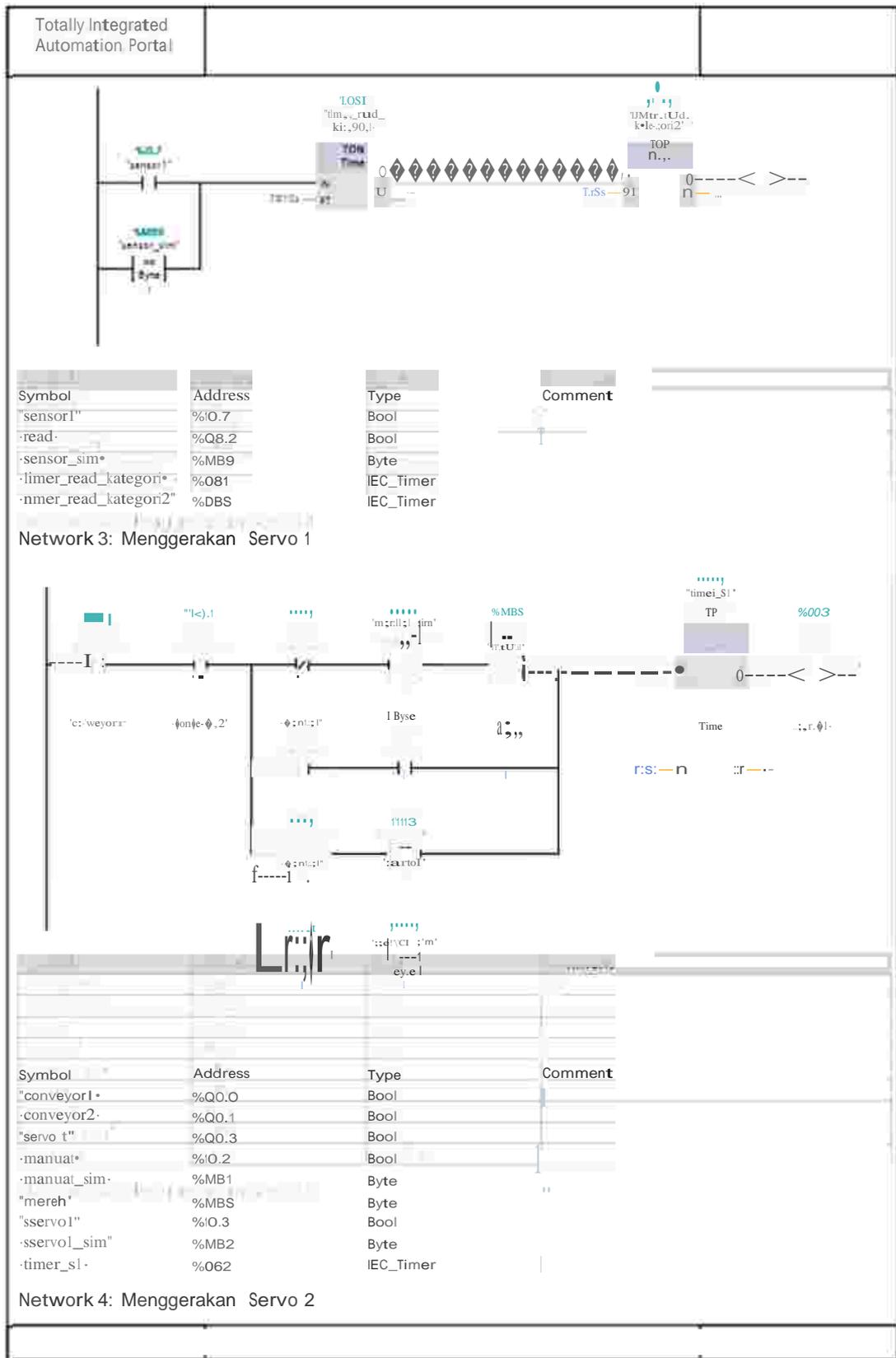
|        |        |         |     |         |                 |
|--------|--------|---------|-----|---------|-----------------|
| Title  | Author | Version | 0.1 | Comment | User-defined ID |
| Family |        |         |     |         |                 |

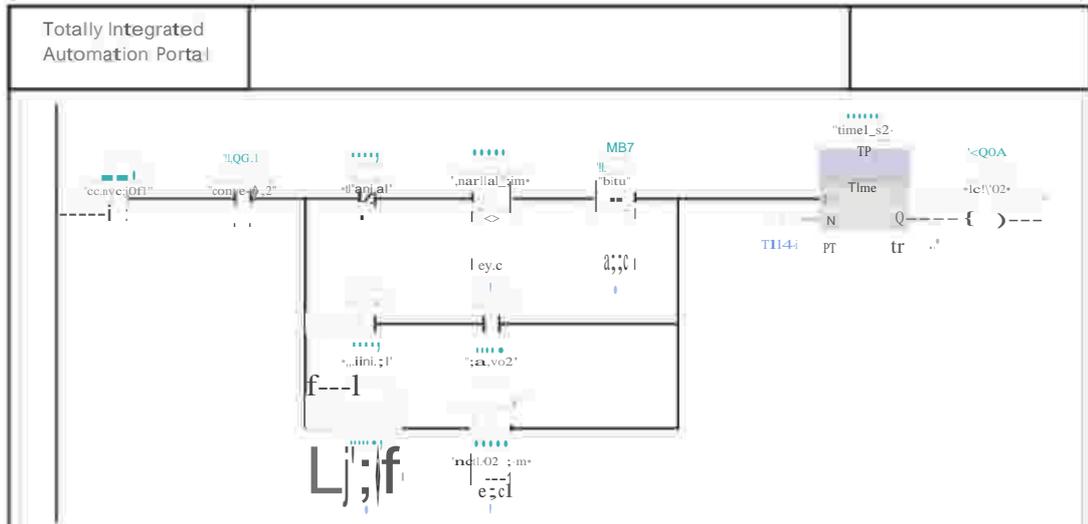
| Name | Data type | Offset | Comment |
|------|-----------|--------|---------|
| Temp |           |        |         |

Network 1: Menjalankan Conveyor

| Symbol     | Address | Type | Comment |
|------------|---------|------|---------|
| start      | %I.6    | Bool |         |
| start_sim  | %MBO    | Byte |         |
| sensor     | %I.7    | Bool |         |
| conveyor1  | %Q0.0   | Bool |         |
| conveyor2  | %Q0.1   | Bool |         |
| conveyor3  | %Q0.2   | Bool |         |
| step_sim   | %MBB    | Byte |         |
| read       | %Q8.2   | Bool |         |
| sensor_sim | %MB9    | Byte |         |

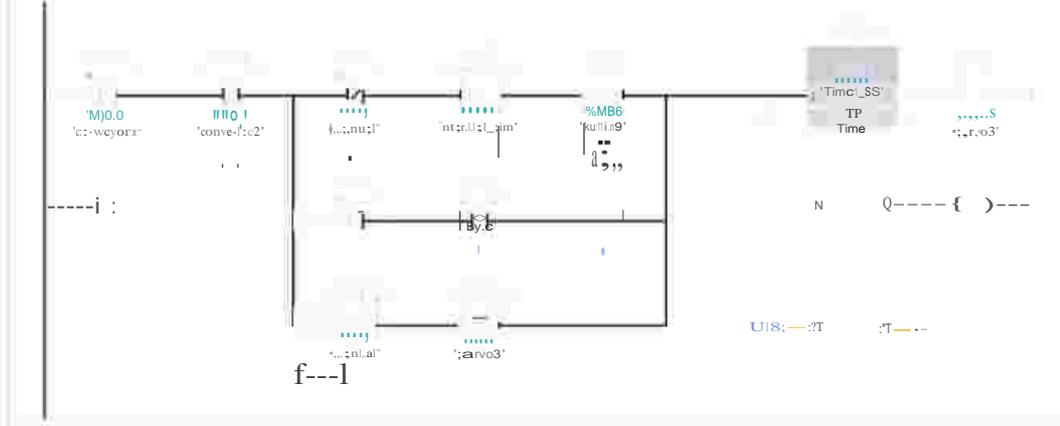
Network 2: Membaca file CSV





| Symbol        | Address | Type      | Comment |
|---------------|---------|-----------|---------|
| "conveyor1"   | %Q0.0   | Bool      |         |
| "conveyor2"   | %Q0.1   | Bool      |         |
| "servo2"      | %Q0.4   | Bool      |         |
| "biru"        | %MB7    | Bool      |         |
| "manual"      | %I.2    | Bool      |         |
| "manual_sim"  | %MB1    | Byte      |         |
| "sservo2"     | %I.4    | Bool      |         |
| "sservo2_sim" | %MB3    | Byte      |         |
| "timer_S2"    | %083    | IEC_Timer |         |

Network 5: Menggerakan servo 3



| Symbol        | Address | Type | Comment |
|---------------|---------|------|---------|
| "conveyor1"   | %Q0.0   | Bool |         |
| "conveyor 2"  | %Q0.1   | Bool |         |
| "kuning"      | %MB6    | Byte |         |
| "manual"      | %I.2    | Bool |         |
| "manual_sim"  | %MB1    | Byte |         |
| "sservo3"     | %I.5    | Bool |         |
| "sservo3_sim" | %MB4    | Byte |         |
| "servo3"      | %Q0.5   | Bool |         |

| Totally Integrated Automation Portal  |         |           |         |
|---------------------------------------|---------|-----------|---------|
| Symbol                                | Address | Type      | Comment |
| timer_s3                              | %084    | IEC_Timer |         |
| <b>Network 6: Mematikan Servo HMI</b> |         |           |         |
|                                       |         |           |         |
| Symbol                                | Address | Type      | Comment |
| "servo1"                              | %Q0.3   | Bool      |         |
| "servo2"                              | %Q0.4   | Bool      |         |
| "servo3"                              | %Q0.5   | Bool      |         |
| "turnoff"                             | %Q8.0   | Bool      |         |
| "hapus3"                              | %068    | IEC_Timer |         |
| "hapus2"                              | %067    | IEC_Timer |         |
| "hapus"                               | %066    | IEC_Timer |         |