

**LİNEER ROBOTİK BİR SİSTEMİN GÖRÜNTÜ  
TABANLI KONTROLÜ**

**Oğuzhan KURNAZ**

**Yüksek Lisans Tezi  
Elektrik Elektronik Mühendisliği  
Danışman: Dr. Öğr. Üyesi Kağan Koray AYTEN**

**2018  
Her hakkı saklıdır.**



**FEN BİLİMLERİ ENSTİTÜSÜ**  
**YÜKSEK LİSANS TEZİ**

**LİNEER ROBOTİK BİR SİSTEMİN GÖRÜNTÜ TABANLI KONTROLÜ**

**OĞUZHAN KURNAZ**

**Tez Danışmanı: Dr. Öğr. Ü. Kağan Koray AYTEN**

**Anabilim Dalı: Elektrik Elektronik Mühendisliği**

**Erzurum**

**2018**

**Her Hakkı Saklıdır**

**T.C.**  
**ERZURUM TEKNİK ÜNİVERSİTESİ**  
**FEN BİLİMLERİ ENSTİTÜSÜ**  
**TEZ ONAY FORMU**

---

**LİNEER ROBOTİK BİR SİSTEMİN GÖRÜNTÜ TABANLI KONTROLÜ**

Dr. Öğr. Üyesi Kağan Koray AYTEN danışmanlığında, Oğuzhan KURNAZ tarafından hazırlanan bu çalışma 11 / 12 / 2018 tarihinde aşağıdaki jüri tarafından Elektrik Elektronik Mühendisliği Ana Bilim Dalı'nda Yüksek Lisans tezi olarak **Oy birliği ile (3/0)** ile kabul edilmiştir.

Başkan : Dr. Öğr. Üyesi K. Koray AYTEN *İmza* :

Üye : Dr. Öğr. Üyesi Ahmet DURLU *İmza* :

Üye : Dr. Öğr. Üyesi M. Alptekin ENGİN *İmza* :

Üye : ..... *İmza* :

Üye : ..... *İmza* :

Yukarıdaki sonucu onaylıyorum

**Doç. Dr. Arzu GÖRMEZ**  
**Enstitü Müdürü**

Bu tez çalışması ..... tarafından ..... nolu proje ile desteklenmiştir.

## **ETİK KURALLARA UYGUNLUK BEYANI**

Erzurum Teknik Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına uygun olarak hazırladığım bu tez içindeki tüm bilgilerin doğru ve tam olduğunu, bilgilerin üretilmesi aşamasında bilimsel etiğe uygun davrandığımı, yararlandığım bütün kaynakları atf yaparak belirttiğimi beyan ederim.

11 / 12 / 2018

İmzası

Oğuzhan KURNAZ

**ÖZET**  
**YÜKSEK LİSANS TEZİ**  
**LİNEER ROBOTİK BİR SİSTEMİN GÖRÜNTÜ TABANLI KONTROLÜ**

Oğuzhan KURNAZ

Erzurum Teknik Üniversitesi

Fen Bilimleri Enstitüsü

Elektrik Elektronik Mühendisliği Ana Bilim Dalı

Danışman: Dr. Öğr. Üyesi Kağan Koray AYTEN

Günümüzde teknolojinin gelişmesine paralel olarak artan ihtiyaçlar, insanları farklı yeniliklerin arayışına sürüklemektedir. Buharlı makinelerin sanayide kullanımıyla başlayan endüstrileşme süreci günümüze kadar artarak süregelmiştir. Bu süreçte endüstri, belirli aşamalardan geçmiş ve Endüstri 4.0 adıyla günümüze ulaşmıştır. Endüstri 4.0 üretim, tarım, savunma sanayi gibi sektörlerde hızlı, hatasız ve verimli üretim gibi avantajlar sağlamaktadır. Tarım alanında da insanların güçlük çektiği problemlere Endüstri 4.0 tabanlı otonom sistemlerle rahatlıkla çözüm önerileri sunulabilmektedir. Günümüzde tarımda karşılaşılan ve üretimi ekstra özen gerektiren ürünlerden biri de hiç kuşkusuz kültür mantarıdır. Mantar üretimi oldukça hassas işlemler gerektiren ve işlemlerin her aşamasında insan kaynağına oldukça ihtiyaç duyan bir yapıdadır. Bu sebeplerden ötürü, üretim veriminin düşük kalması ve istenilen kalitede ürünlerin elde edilememesi gibi problemler ortaya çıkmaktadır. Mantar üretiminde karşılaşılan bu problemlere Endüstri 4.0 teknolojisiyle çözümler getirilebilir.

Bu tez çalışmasında anlatılan sistem, görüntü tabanlı lineer robotik bir platformun endüstriyel otomasyon kontrol sistemiyle (PLC) kontrolüdür. PLC, kamera ve motorlardan oluşan bir sistemle mantarın yer tespiti yapılacaktır. Bu tezdeki sistemle, daha gelişmiş olan “Robotik Mantar Toplama” sistemine temel olması sağlanacaktır. “Robotik Mantar Toplama” sistemi hareketli bir konveyör sisteminin üzerinde bulunan ve XYZ düzleminde hareket edebilen manipülatör, konveyör bandın üzerinde bulunan kasalar ve kasalarda mantar üretiminin gerçekleştirildiği otonom bir sistemdir.

Bu tez KOSGEB’in Ar-Ge İnovasyon Projesi kapsamında 11/01/2017 tarih ve 2016/13 sayılı proje kapsamında desteklenmiştir. Ayrıca bu tez Erzurum Teknik Üniversitesi Bilimsel Araştırma Projesi 2017/18 No’lu proje tarafından desteklenmiştir.

**2018, 78 sayfa**

**Anahtar Kelimeler:** Görüntü işleme, Nesne Takibi, Kontrol, Haar-Cascade Sınıflandırıcı, Kamera Kalibrasyonu, PLC

## **ABSTRACT**

**MS. Thesis**

### **IMAGE BASED CONTROL OF A LINEAR ROBOTIC SYSTEM**

Oğuzhan KURNAZ

Erzurum Technical University

Graduate School of Natural and Applied Sciences

Department of Electrical Electronic Engineering

Supervisor: Asst. Prof. Kağan Koray AYTEN

Today, in parallel with the development of technology, the increasing needs lead people to seek different innovations. The industrialization process that started with the use of steam machines in the industry has been continuing until the present and is still continuing. In this process, the industry has gone through certain stages and have reached today with the name of Industry 4.0. Industry 4.0 provides advantages such as fast, error-free and efficient production in sectors such as production, agriculture and defense industry. With the Industry 4.0 based autonomous systems can easily offer solutions to many problems in the field of agriculture. One of the products that are encountered in agriculture and require extra care is undoubtedly cultural mushroom. Mushroom production is quite sensitive and requires a lot of human resources at every stage of these processes. For these reasons, problems such as low production efficiency and inability to produce products of the desired quality arise. Solutions to these problems encountered in the production of mushrooms can be brought up with Industry 4.0 technology.

The system described in this thesis is control of an image based linear robotic platform with the industrial automation control system (PLC). In this thesis, a system consisting of PLC, camera and motors will be used to locate the mushroom with the camera. The system of this thesis will be the basis for the more advanced Robotic Mushroom Collection system. “Robotic Mushroom Collection” system is an autonomous system, which is contained a manipulator which is movable on XYZ plane on a moving conveyor system, the cases which are moving on conveyor belts and the mushroom production in the cases.

This thesis was supported within the scope of KOSGEB's R & D Innovation Program within the scope of the project dated 11/01/2017 and numbered 2016/13. In addition, this thesis was supported by the Erzurum Technical University Scientific Research Projects 2017/18.

**2018, 78 pages**

**Keywords:** Image Processing, Object Tracking, Control, Haar-Cascade Classifier, Camera Calibration, PLC

## TEŐEKKÜR

Tez alıőmam boyunca yardımlarıyla alıőmalarımnda yol gsterici konumunda olan Dr. gr. Üyesi Kaęan Koray AYTEN'e,  
Bugünlere gelmemde en büyük emeęi olan ve koőulsuz yanımda olduklarını hissettiren annem, babam ve kardeőime,  
Desteklerini esirgemeyen arkadaşlarıma teőekkür ederim.

**Oęuzhan KURNAZ**

**Aralık 2018**

## İÇİNDEKİLER

<b>ÖZET</b> .....	<b>i</b>
<b>ABSTRACT</b> .....	<b>ii</b>
<b>TEŞEKKÜR</b> .....	<b>iii</b>
<b>İÇİNDEKİLER</b> .....	<b>iv</b>
<b>SİMGELER VE KISALTMALAR DİZİNİ</b> .....	<b>vi</b>
<b>ŞEKİLLER DİZİNİ</b> .....	<b>ix</b>
<b>ÇİZELGELER DİZİNİ</b> .....	<b>xi</b>
<b>2. KAYNAK ÖZETLERİ</b> .....	<b>5</b>
<b>3. MATERYAL ve YÖNTEM</b> .....	<b>11</b>
3.1. Sistemin Genel Çalışma Mantığı .....	11
3.2. Görüntü İşleme.....	14
3.2.1. Görüntü İşlemenin Kullanım Alanları .....	14
3.2.2. Görüntü İşlemenin Yazılımsal Adımları .....	15
3.2.2.1. Visual Studio C#.NET Yazılımı .....	16
3.2.2.2. Emgu.CV Kütüphanesi .....	17
3.3. Görüntü İşleme İle Mantarın Tespit Edilmesi.....	18
3.3.1. Sınıflandırma Algoritması.....	18
3.3.2. Haar-Cascade Sınıflandırma Yöntemi .....	21
3.3.3. Kamera Kalibrasyonu .....	25
3.4. PLC İle Kontrol Kısmı.....	32
3.4.1. Kullanılan Yazılım ve Donanımların Tanıtılması.....	32
3.4.1.1. SIEMENS S7-1200 PLC Cihazı .....	32
3.4.1.2. TIA PORTAL Yazılımı .....	36
3.4.1.3. Step Motor .....	38
3.4.1.4. Step Motor Sürücüsü.....	40
3.4.2. PLC Cihazı İle Bilgisayarın Haberleşmesi .....	44
3.4.3. Visual Studio C# Yazılımı İle TIA PORTAL'ın Haberleşmesi.....	47
3.4.4. PLC Cihazından Step Motor Sürücülerine Konum Bilgilerinin Gönderilmesi.....	48
<b>4. ARAŞTIRMA BULGULARI ve TARTIŞMA</b> .....	<b>52</b>
<b>5. SONUÇ ve ÖNERİLER</b> .....	<b>56</b>
<b>KAYNAKLAR</b> .....	<b>58</b>



<b>EKLER.....</b>	<b>61</b>
EK-1 .....	61
<b>ÖZGEÇMİŞ.....</b>	<b>63</b>

## SİMGELER VE KISALTMALAR DİZİNİ

<u>Simgeler</u>	<u>Açıklama</u>
#	Sharp
%	Yüzde
(x, y)	Resmin X ve Y eksenindeki konumu
$C_1, C_2, \dots, C_N$	Güçlü Sınıflandırıcı
$C_x, C_y$	Resim düzleminin merkezi noktasının X ve Y eksenlerindeki konumu
$d_x, d_y, d_z$	X, Y ve Z eksenlerindeki konumlar
$f_n(x, y)$	Zayıf Sınıflandırıcı
$f_x, f_y$	Kameranın X ve Y eksenlerindeki odak uzaklığı
$k_1, k_2$	Radyal Distorsiyon Katsayıları
$kg_x, kg_y$	Kameradan alınan görüntünün X, Y eksenlerindeki değerleri
$mgb_x, mgb_y$	Mantarın yatay ve dikey eksenlerdeki gerçek boyutları
$mrbn_x, mrbn_y$	Mantarın resim düzlemindeki başlangıç noktasının X, Y değerleri
$mr b_x, mr b_y$	Mantarın resim düzleminde yatay ve dikey eksenlerdeki boyutları
$p_1, p_2$	Teğetsel Distorsiyon Katsayıları
$u_0, v_0$	Resmin Merkezi
$x', y'$	X ve Y'nin 1.dereceden türevi
$x_q, y_q$	Genişletilmiş Fiziksel Koordinatlar
$\Delta$	Komşu iki piksel arası mesafe
$\Sigma$	Toplam
A	İç Yönelme Parametreleri
Amp	Amper
C	Celsius
f	Kamera Odak Uzaklığı
F(x)	Güçlü Sınıflandırıcı
mH	Mili Henry
N	Newton

Nm	Newton Metre
°	Derece
$p$	Resmin piksel noktasının homojen koordinatları
$P(X,Y,Z)$	Cismin global koordinat düzlemindeki konumu
$P1,P2,P3,P4$	$P(x,y)$ 'yi oluşturan noktalar
R	Rotasyon Matrisi
s	Ölçek Faktörü
t	Dönüşüm Vektörü
V	Voltaj
$\Omega$	Ohm
$P(x,y)$	İntegral Resmin (x,y) noktasındaki değeri
$f(x,y)$	X ve Y'ye bağlı f fonksiyonu
$ii(x,y)$	İntegral Resim

<b><u>Kısaltmalar</u></b>	<b><u>Açıklama</u></b>
---------------------------	------------------------

AC	Alternatif Akım
ACW	Saat Yönü Ters
AR	Artırılmış Gerçeklik
CPU	Merkezi İşlem Birimi
CW	Saat Yönü
DC	Doğru Akım
DLT	Doğrudan Lineer Dönüşüm
EOG	Elektrookülografi
HD	Yüksek Çözünürlük
HMI	İnsan-Makine Arayüzü
HOG	Gradyanlar Histogramı
I/O	Giriş ve Çıkışlar
ICA	Bağımsız Bileşen Analizi
IP	İnternet Protokolü
iChair	Akıllı Sandalye
KNN	En Yakın K Komşu
linSVM	Lineer Destek Vektör Makinesi

NN	Sinir Ađı
Pfinder	Birey Bulucu
PLC	Programlanabilir Lojik Kontrolör
ROI	Hedef Bölgesi
RPM	Dakikadaki Dönüş Hızı
SCADA	Merkezi Denetleme Kontrol ve Veri Toplama
SIFT	Deđişmez Ölçek Öznitelik Çıkarımı
SURF	Hızlandırılmış Güçlü Öznitelikler
SVM	Destek Vektör Makinesi
TIA	Tamamen Entegre Otomasyon
TLD	Takip-Öğrenme-Bulma
XML	Genişletilebilir İşaretleme Dili

## ŞEKİLLER DİZİNİ

Şekil 3.1. Mantarların yetiştirildiği odalardaki raf sistemi .....	12
Şekil 3.2. Mantar Toplama Sisteminin Genel Görüntüsü .....	12
Şekil 3.3. Sistemin Akış Şeması .....	14
Şekil 3.4. Microsoft Visual Studio C#.NET çalışma ekranı .....	16
Şekil 3.5. Visual Studio yazılımının gerçekleştirdiği görevler .....	17
Şekil 3.6. Solda İntegral Resim ve sağda Orijinal Resim .....	22
Şekil 3.7. Haar Benzeri Öznitelikler(Viola and Jones, 2001) .....	23
Şekil 3.8. AdaBoost ile belirlenen uygun öznitelik (solda) ve uygun olmayan öznitelik (sağda) .....	23
Şekil 3.9. Cascade Sınıflandırıcısı .....	24
Şekil 3.10. İğne Deliği Kamera Modeli .....	26
Şekil 3.11. Mantarın tespit aşamaları .....	29
Şekil 3.12. Kamera Kalibrasyonu için satranç tahtası köşelerinin tespit edilmesi.....	29
Şekil 3.13. PLC'nin çalışma mantığı ve sistem elemanları .....	33
Şekil 3.14. S7-1200 PLC Cihazı .....	35
Şekil 3.15. S7-1200 modeline ait PLC çeşitleri ve teknik özellikleri .....	35
Şekil 3.16. TIA PORTAL programı ara yüzü .....	37
Şekil 3.17. Projeye kullanılan cihazın eklenmesi.....	38
Şekil 3.18. Program bloklarının oluşturulması .....	38
Şekil 3.19. Step Motor ve bağlantı kısımları.....	39
Şekil 3.20. Sistemde kullanılan motor sürücüsü .....	41
Şekil 3.21. Sürücü ön görünüşü ve bağlantı soketleri .....	42
Şekil 3.22. Sürücü üst görünüşü ve bağlantı soketleri .....	42
Şekil 3.23. Sürücü alt görünüşü ve bağlantı portları.....	43
Şekil 3.24. Motor sürücü ile motor arasındaki bağlantı .....	43
Şekil 3.25. Motor sürücü ile PLC cihazı arasındaki bağlantı.....	44
Şekil 3.26. PLC cihazının projeye eklenmesi .....	45
Şekil 3.27. PLC cihazına IP adresi atanması.....	46
Şekil 3.28. Çalıştırılan PLC kodunun gerçek zamanlı takibi .....	46
Şekil 3.29. C# kodları .....	48
Şekil 3.30. Mantar Tespiti ve Konum Bilgilerinin PLC'ye Gönderimi .....	49

Şekil 3.31. X ve Y eksenlerine ait konum ve hız bilgilerinin grafiksel gösterimi .....	50
Şekil 4.1. Visual Studio C#.NET platformundaki Form uygulaması ve Mantar Tespiti.....	52
Şekil 4.2. TIA Portal yazılımında hazırlanmış olan Program Blokları ile tespit edilen mantarın konum bilgilerinin alınması .....	53
Şekil 4.3. X eksenindeki gerçek konum ve hesaplanan konumların grafiksel gösterimi.....	54
Şekil 4.4. Y eksenindeki gerçek konum ve hesaplanan konumların grafiksel gösterimi.....	55

## ÇİZELGELER DİZİNİ

Çizelge 3.1. Bilgi Olarak Verilen Veri Seti .....	19
Çizelge 3.2. Cinsiyeti Erkek Olan Bireylere Ait Bilgiler.....	20
Çizelge 3.3. Cinsiyeti Kız Olan Bireylere Ait Bilgiler .....	20
Çizelge 3.4. (10), (11) ve (12) numaralı denklemlerde kullanılan parametrelerin açıklamaları .....	31
Çizelge 3.5. 57 mm flanş uzunluğuna sahip S tipi step motorun teknik özellikleri...	40
Çizelge 3.6. Motorlardaki dahili enkoderin özellikleri .....	40
Çizelge 4.1. X ve Y eksenlerindeki gerçek ve hesaplanan konum bilgileri ve hata oranları .....	54

### 1. GİRİŞ

Teknolojinin gösterdiği gelişimlerin sonucu olarak ortaya çıkan otonom sistemler, günümüzde üretim, tarım, inşaat, sağlık gibi yaşama yön veren birçok alanda kullanılabilecek uygulamalar haline gelmiştir. Buharlı makinelerin icadıyla sanayiye yeni bir vizyon kazandırılmış ve Endüstri 1.0 kavramı literatürdeki yerini almıştır. Buhar gücüyle çalışan makinelerin artışı sonrası makineleşme ve endüstrileşme büyük bir hız kazanmıştır. Daha sonra üretimde elektriğin kullanılmaya başlanmasıyla Endüstri 2.0 kavramı ortaya çıkmıştır. Üretim bantları ve seri üretim kavramları, fabrikalarda kullanılır hale gelmiştir. Teknolojideki gelişimin devamına bağlı olarak değişen ve gelişen çağ artık dijital çağa dönüşmüştür. Elektronik cihazlar ile yazılım teknolojilerinin bir arada kullanılmasıyla beraber üretim sistemleri artık otonom hale gelmiştir. Üretim sistemlerinin otonom hale gelmesi, insan faktörünün üretimdeki etkisinin biraz daha azalmasına neden olmuştur. Yani makineler, üretim sektöründe yavaş yavaş insanın yerini almaya başlamıştır. Bu gelişmeler sonucunda ise Endüstri 3.0 çağına geçilmiştir. Bulut teknolojileri, nesnelerin interneti, siber-fiziksel sistemler kavramları daha çok konuşulmaya ve üretimde daha çok söz sahibi olmaya başlamıştır. Üretim aşamasında neredeyse insandan bağımsız gerçekleştirilen, kendi içinde birbirleriyle haberleşerek çalışabilen akıllı fabrikalar ortaya çıkmaya başlamıştır. Bunun bir sonucu olarak insanın üretim alanında üstlendiği rol giderek azalmaya başlamış ve insanlardan bağımsız üretimler hali hazırda gerçekleşmiştir. Bu insansız üretim aşamalarının bir özeti niteliğinde olan Endüstri 4.0 kavramı böylece karşımıza çıkmıştır. Endüstri 4.0 teknolojisi, üretimin optimize edilmesinde önemli bir rol üstlenmeye başlamıştır. Endüstri 4.0 teknolojisinin gelişimine paralel olarak gıda, otomotiv, havacılık, enerji gibi sektörlerde gerçekleştirilen üretim kalitesi ve hızı artmaya devam etmektedir. Özellikle üretim sektöründe insansız üretim aşamalarında elde edilen ürün kalitesi azımsanmayacak ölçütlerde gelişim göstermiştir. Üretim aşamalarında insanın fark edemediği birçok detay, insansız sistemler tarafından kolaylıkla fark edilebilmekte ve gerekli önlemler alınabilmektedir. Üretim aşamalarında ürünlerin yetiştirme şartlarının farklılıklarına bağlı olarak üretim kalitesi de değişkenlik göstermektedir. Yetiştirme şartları göz



## 1. GİRİŞ

---

önüne alındığında üretimi zorlu bir süreç gerektiren ürünlerden biri de kültür mantarıdır.

Kültür mantarı, sıcaklık ve nem miktarı kontrol altında tutulabilen, havalandırılması kolay, güneş ışığı almayan yerlerde yetiştirilebilir. Mağaralar, soğuk hava depoları, ışığa karşı yalıtılmış seralar, tünel ve galeriler, kümesler, ambarlar, bodrumlar ve modern mantar işletmeleri mantar üretimi yapılabilen yerlerdir(Anonim, 2017a). Mantar üretiminde dikkat edilmesi gereken bazı hususlar ise şunlardır:

- Mantarların, beyaz renge sahip olana kadar hiçbir şekilde gün yüzü görmemesi gerekir.
- Mantarın yetiştiği ortam nemi %70-75 arası bir değerde olmalıdır.
- Ortam sıcaklığı 25 °C olması gerekir.
- Ortamda istenilen nem sağlanamadıysa toprak 2 günde bir sprey ile sulanmalıdır.
- Mantar beyaz renge ulaştığı zaman bile yine doğrudan güneş ışığına maruz bırakılmamalıdır. Bunun yerine 10 saat beyaz ışık altında 25 °C ortam sıcaklığında bekletilmelidir.
- Tüm bu işlemler sonucunda ortalama 15 gün içerisinde ürün alınması gerekmektedir(Anonim 2017b).

Tüm bu şartlar göz önünde bulundurulduğunda kültür mantarı üretimi özen gerektiren ve meşakkatli bir süreçtir. Mantarın ilk ekiminden yetişme sürecinin tamamlandığı döneme kadar çalışan işçi sayısı ve bu işçilerin çalışma süreleri önemli bir etkidir. İşçilerin çalışma sürelerine bağlı olarak fabrikaların çalışır durumda bulunduğu zaman dilimi üretim verimini doğrudan etkilemektedir. İşçilerin vardiya saatlerinin ortalama 8-10 saat olduğu düşünülürse ve otonom bir sistemle kıyasla insanların hata yapma olasılıklarının daha yüksek olması mantar üretimi için farklı çözüm arayışlarına sebep olmuştur.

Tez çalışması bu tür problemlere etkin çözümler sunabilecek niteliklere sahip bir sistemdir. Sistemimiz, mantarların ekil durumda bulunduğu kasa, harici kameralar, 2 adet Step Motor sürücü modülü, 2 adet Step Motor, 1 adet Siemens

## 1. GİRİŞ

---

marka S7-1200 tipi PLC donanımlarının yanı sıra, görüntü işleme yöntemlerinin uygulandığı Microsoft Visual Studio C#.NET platformu ile motor kontrolü ve PLC ile haberleşme için kullanılan PLC ara yüzü Siemens TIA Portal yazılımlarını barındırmaktadır.

Robotik Mantar Toplama sistemine temel oluşturacak tezdeki deneysel çalışma aşamamız ise şu şekilde sıralanabilir:

- Konveyör bant üzerinde olduğu düşünülen ve mantarların ekil halde bulunduğu kasanın hareketi,
- Kasaların taşınmaya başlaması ile birlikte kamera sisteminin devreye girmesi,
- Kamera sistemi ve gerekli yazılımların desteği altında mantarların lokasyon bilgilerinin tespit edilmesi,
- Lokasyon bilgileri tespit edilen mantarların bulunduğu koordinat verilerinin gerekli haberleşme protokolleri kullanılarak PLC'ye veri olarak gönderilmesi,
- PLC'ye gönderilen koordinat verilerinin değerlendirilmesi ve bu verilerin Step Motor sürücülerine gönderilmesi,
- Step Motor sürücülerine gönderilen bu koordinat verileri ışığında X ve Y düzlemlerinde hareketini gerçekleştiren Step Motorların, mantarın bulunduğu lokasyona hareket etmelerinin sağlanması,
- Bulduğu lokasyon zamana bağlı olarak değişen mantarın anlık olarak takibinin gerçekleştirilmesidir.

Sisteme dair bahsi geçen bu aşamalar ile kasa içinde ekil halde bulunan mantar, otonom bir sistem ile tespit edilecek ve toplayıcı robotun da mantarın bulunduğu lokasyona varması sağlanacaktır. Böylece fabrikalarda mantar üretim hatlarında karşılaşılan zaman kaybı ve üretimde verim düşüklüğü gibi problemlere çözümler sunulmuş olacaktır. Bu tez çalışmasında elde edilecek tecrübe ve bilgi birikimi ile daha sonra üretiminin KOSGEB AR-GE İNOVASYON projesi kapsamında tamamlanması planlanan Robotik Mantar Toplama sisteminin kurulacak bir mantar tesisinde 7 gün 24 saat çalışması sayesinde, üretimde insan faktörünü

## 1. GİRİŞ

---

büyük oranda elimine ederek zamandan tasarruf sağlanması ve üretimde verimin artırılması gerçekleşecektir. Ayrıca insanların bulunabileceği olumsuz psiko-fizyolojik şartların, üretim kalitesine olumsuz olarak yansıtılabileceği göz önünde bulundurulacak olursa, buna alternatif olarak sunulan otonom bir sistem olan Robotik Mantar Toplama sistemi kalıcı bir çözüm sağlamak adına atılmış önemli bir adım olarak değerlendirilebilir.

Bu tez çalışmasında gerçekleştirilen en büyük kazanımlardan birisi, sıradan bir USB veya dahili kamera ile, görüntüdeki herhangi bir nesneye ait piksel bazlı konumdan, global koordinat sistemindeki gerçek konumuna ulaşılabilecek bir yöntem olan kalibrasyon tekniğinin kullanılmış olmasıdır. Microsoft Kinect, stereo kamera sistemi vb. sistemlerle elde edilmesi nispeten daha kolay olan bu konum bilgileri, kalibrasyon işlemi sayesinde standart bir kamera ile de gerçekleştirilmiştir.

### 2. KAYNAK ÖZETLERİ

Bu bölümde, görüntü işleme ve kontrol teknikleri yardımıyla tarım ürünlerine yönelik çalışmalar içeren araştırmalardan bahsedilmektedir. Öncelikle görüntü işleme ile yapılmış olan tarıma yönelik uygulamalara dair benzer çalışmalar sunulacaktır. (Sari and Adinugroho, 2017), çalışmalarında domates resimlerinin, olgunluklarına göre sınıflandırılarak birbirinden ayrıştırılması amacıyla ortaya atılmış yeni bir metot anlatmışlardır. Çalışmada kameradan domateslere ait görüntüler alındıktan sonra bu görüntülere gerekli görüntü işleme yöntemleri uygulanmıştır. Bu uygulanan yöntemlerden sonra görüntüler K-means algoritmasından geçirilerek olgunluklarına göre sınıflandırılması yapılmıştır. Burada 6-means olarak ifade edilmesinin sebebi ise domatesin ham halinden olgunluk haline kadar geçirdiği evre sayısının 6 olmasıdır. Başka bir çalışmada (SOLAK ve ALTINIŞIK, 2018), fındık meyvesine ait görüntülerin kameradan elde edilmesi sonrasında gerekli görüntü işleme teknikleri uygulanmıştır. Daha sonra bu teknikler yardımıyla fındıklara ait boyut ve alan verileri hesaplanmış, elde edilen bu veriler değerlendirilerek küçük, orta ve büyük olmak üzere 3 sınıfa ayrılmıştır. Bu işlemler, ortalama tabanlı sınıflandırma ve K-means kümeleme yöntemleri kullanılarak gerçekleştirilmiştir. Daha sonra bu iki yönteme ait sonuçlar karşılaştırılmıştır. (Mahmud et al., 2015), çalışmalarında tarımsal ürünlerin kalite denetiminin sağlanması amacıyla geliştirilmiş olan mobil bir robotun tanıtımını gerçekleştirmişlerdir. Aforge.Net kütüphanesi kullanılan ve C# ara yüzü ile gerçekleştirilen bu projede, robot 2 kameraya sahiptir. Bunlardan ilki ürünün bulunduğu lokasyona doğru gerçekleştirilen hareket sırasında robotun izlediği yolun görüntülenmesine yardımcı olmaktadır. Diğer kamera ise tespit edilen ürüne gerekli filtreleme işlemlerini uygulayarak daha iyi bir görüntüleme ve ekin kalite denetimi imkânı sunmaktadır. (Mehta and Burks, 2016) çalışmasında, hasat için kullanılan robotlarda görsel geri beslemeli kontrol sistemi yönteminin uygulanması anlatılmıştır. Meyvelerin hareketleri, görüntü düzleminde ve kameranın optik eksenine boyunca analiz edilmiştir. Bu analizler aşamasında, asılı olan meyvelerin hareketleri analiz edilirken, meyvenin yatay ve dikey eksenindeki hareketlerinden dolayı meyvenin hareketi 2 serbestlik dereceli kütle-yay sistemine benzetilmiştir. Daha sonra hareket kontrolü kısmında ise Lyapunov kontrol yöntemi kullanılarak sonuca

gidilmiştir. (Zhao et al., 2016), gerçekleştirdikleri çalışmada ürün tespiti ve toplanması için tasarlanmış görüntü tabanlı bir mimariye sahip olan hasat robotu anlatmışlardır. Çalışmada bahsi geçen mobil robot 2 adet kamera ve manipülatör donanımlarına sahiptir. Mobil robotun üstünde bulunan ve manipülatör ile fiziki bir bağlantısı bulunmayan sabit kamera ürün tespitinin gerçekleştirilmesini sağlar. Manipülatör, kendi uç kısmında bulunan end-efektörün ürüne yönelimini gerçekleştirir. End-efektörün üst kısmında bulunan kamera ise çeşitli sınıflandırma algoritmaları kullanılarak ürün tespiti yapılması amacıyla kullanılmıştır. Algoritmaların tespit doğruluğu değerlerle karşılaştırılmıştır. (Jun et al., 2012) çalışmasında, domates hasat robotları için tasarım ve eş zamanlı simülasyon problemleri ele alınmıştır. Robot manipülatörü 4 serbestlik derecelidir ve 2 doğrultuda hareketi sağlayan paralel bir eklemdedir çalışmaktadır. Kişisel bir bilgisayar, manipülatör, end-efektör ve kesici kısımların kontrolünü sağlamaktadır. (Li et al., 2010), çalışmalarında gerçek zamanlı ve doğruluk oranı yüksek olan bir ürün eşleştirme uygulaması gerçekleştirmişlerdir. Projede ürün olarak ananas seçilmiştir ve görüntü sistemi olarak ise çift gözlü görüntü sistemi kullanılmıştır. Kameralardan gerçek zamanlı görüntü alımı gerçekleştirilir. Daha sonra hızlı segmentasyon yapılarak ve Speeded Up Robust Features(SURF) algoritması kullanılarak sol ve sağ taraftaki 2 ayrı resmin eşleşme eksenlerinin belirlenmesi sağlanır. (Feng et al., 2015) çalışmasında, taze domates hasat robotunun tasarım ve test çalışmaları anlatılmıştır. Mobil hareket eden bir platform ve bunun üzerinde bulunan robot manipülatörü, renk uzayındaki belirlenen değerlere göre görüntülenen ürünün yetişip yetişmediğine dair değerlendirme yapıp yetiştiği belirlenen domateslerin uç kısmında bulunan end-efektör vasıtasıyla toplanmasını sağlar.(Shao et al., 2014), çalışmalarında ağaçların gövdelerinin BP Neural-Network metodu kullanılarak tanımlanması anlatmışlardır. Öncelikle renk işaretlemeleri ve Neural-Network verilerinin eğitimi için örnek toplama işlemleri tamamlanır. Daha sonra Neural-Network ile model eğitimi gerçekleştirilir. Son kısımda ise Hough Transform metodu ile gövde kısmı tanımlanır. (Issac et al., 2016)'nın gerçekleştirdiği çalışmada, bir balığın örnek resim üzerinde görüntü işleme metotları kullanılarak solungaçlarının segmentasyonunun otomatik ve verimli bir şekilde gerçekleştirilmesi sağlanmıştır. Bu sayede test aşamasında balığın tazelik onayı veya herhangi bir böcek zehrinin bulunup bulunmadığı konularının araştırılmasında kolaylık sağlar. (Masoudian, 2013) çalışmasında, ürünlerin otonom bir şekilde hasadının

gerçekleştirilmesi için görüntü işleme yöntemlerine yönelik kullanılan metotlardan bahsedilmiştir. Çalışmada örnek ürün olarak ise mantar kullanılmıştır. Görüntü sistemlerinden alınan mantar resimlerinin tanımlanması işlemi Support Vector Machine (SVM) ile yapılmaktadır. Mantar resmi üzerindeki gerekli morfolojik işlemler yapıldıktan sonra mantarın yetişip yetişmediğine dair özellik çıkarımı ve değerlendirme problemi ise Scale Invariant Features Transform(SIFT) ile gerçekleştirilir. Son aşamada ise değerlendirilmesi yapılan mantar yine SVM metodu ile sınıflandırılır.

Tarıma yönelik uygulamalar dışında görüntü işleme tekniklerinin uygulandığı başka alanlar da bulunmaktadır. (Shaikh et al., 2013), çalışmalarında manipülatör vb. robot kolların hareket kontrolünün gerçek zamanlı görüntü işleme yöntemleri ile insan kol hareketine benzetimi anlatmışlardır. Bu metot, hareketi taklit eden manipülatörün Slave, insan kolunun ise Master olarak atfedilebileceği bir Master-Slave metodu olarak düşünülebilir. Manipülatör ile insan kolu arasındaki haberleşme ise kablosuz olarak gerçekleştirilir. (Enzweiler and Gavril, 2009)'nın hazırlamış oldukları çalışmada, akıllı sistemler, gözetim uygulamaları ve ileri seviye robotların kullanıldığı çalışmalarda ihtiyaç duyulabilecek insan tespiti probleminin çözümüne yönelik öneriler sunulmuş ve sonuçlar sayısal verilerle değerlendirilmiştir. Dalgacık temelli AdaBoost Cascade, HOG/linSVM, NN/LRF ve birleştirilmiş şekil-doku tespiti yöntemleri kullanılmıştır. Sonuçlar, yüksek resim çözünürlüğü ve düşük işleme hızında HOG/linSVM metodunun açık bir şekilde avantaj sağladığını, düşük resim çözünürlüğü ve yüksek işleme hızında ise Dalgacık temelli AdaBoost Cascade yönteminin başarılı olduğunu göstermiştir. (Wren et al., 1997) ise hazırlamış oldukları çalışmada, Pfänder sisteminde bahsetmişlerdir. Pfänder (person finder), insanların tespiti ve takibini gerçekleştirerek onların davranışlarını yorumlayan gerçek zamanlı bir sistemdir. Bir kişinin sabit bir kameradan alınan 2 boyutlu görüntüsü bilgisayar ortamına aktarılarak bunun 2 boyutlu benzetimi şekilsel olarak gösterilir ve hareketler anlık olarak bu benzetime aktarılır. (Babenko et al., 2011)'nin çalışmasında, ilk görüntü karesinde lokasyonu belirtilen ve daha sonra herhangi bir bilgisi bulunmayan objenin takibinin gerçek zamanlı olarak yapılması konusu anlatılmaktadır. Fark alıcı bir sınıflandırma anlık davranışları arka plandan ayırarak objenin tespitinin yapılmasını sağlar. Takipçideki küçük yanlışlıklar objenin doğru tespit edilmesini engelleyebilir hatta sınıflandırıcının işlevselliğinin daha da

düşmesine sebep olabilir. Bunun için Multiple Instance Learning metodu kullanılarak bu problemlerin çözümlerine ulaşılmıştır ve diğer sınıflandırıcı örnekleri ile kıyaslanmıştır. (Kalal et al., 2011) çalışmasında ise, videolarda bilinmeyen objelerin uzun süreli takip konusunu anlatmaktadır. Objeye, ilk görüntü karesinde tespit edilir ve her karede bu objenin takibi sağlanır. Çalışmada önerilen yöntem ise TLD (Tracking-Learning-Detection) yöntemidir. Burada objenin izlenmesi, objeye dair doğrulamayı gerçekleştiren yapay öğrenme algoritmalarının işlendiği kısım ve objenin tespiti ayrı ayrı işlevsel süreçler olarak ele alınmıştır. Yapay öğrenme yöntemi olarak ise P-N Learning yöntemi kullanılmıştır. (Yadav et al., 2014) çalışmasında, gerçek zamanlı video izleme sistemlerinde hareket halindeki objelerin tespit edilmesi için arka plan çıkarma yöntemi anlatılmaktadır. Bu işlemler 3 aşamada gerçekleştirilir. Başlangıçta ilk görüntü karesindeki arka plan modellenir. Daha sonra hareketli obje arka plandan çıkarılır ve arka plan tekrardan yeniden güncellenir. Son aşamada ise obje tespit kalitesini artırıcı morfolojik yöntemler uygulanır. (Manolescu, 1998)'nin çalışmasında, Kalman Filtresi ve Gait Özellik çıkarımı metotları kullanılarak obje takibinin gerçekleştirilmesi konusu anlatılmıştır. Objeye takip uygulaması 3 kısımda incelenmiştir. Bunlar objenin bulunması, takibi ve tanımlanması konularıdır. Objenin tespiti için Bayesian Kalman Filtresi kullanılmıştır. Objenin tanımlanması için ise Gait metodu kullanılarak güçlü bir obje tespit sistemi dizayn edilmiştir. (Diaz et al., 2017) çalışmasında ise, görüntü işleme yöntemleri kullanılarak pas tespitinin yapılması amaçlanmıştır. Görüntü işleme, resmin matris formunda nicelenmesini sağlayan bir yöntem olarak bilinir. Bu nicelme işlemi yapılırken resimde pas vb. herhangi bir nesnenin bulunmasını da sağlar. Gerçekleştirilen prosesler, pas bulunan resimler üzerinde pasın tespit edilmesi konusunda %90'lık bir başarı göstermiştir. (Kanayama and Tabusa, 2010)'nin hazırladığı çalışmada ise, arka plan çıkarımı ve hareket tespit teknikleri ile otomatik yüz yakalama ve resminin elde edilmesini sağlayan sistemin geliştirilmesi konusu anlatılmıştır. Bu sistem, kamera karşısında herhangi bir kişinin algılanması durumunda otomatik olarak resminin çekilmesini sağlar. Bunun için USB kamera ile MATLAB Simulink ara yüzünde geliştirilmiş olan model sistemi kullanılmıştır. (Soundariya and Renuga, 2018) çalışmasında, Elektrookülografi (EOG) kullanılarak kişilerin göz hareketlerinden duygusal tanımlama sisteminin anlatılması gerçekleştirilmiştir. EOG sinyalleri, Multi-class SVM yöntemi kullanılarak mutlu, üzgün, kızgın, korku ve hoş gibi sınıflara ayrılmıştır. Özellik çıkarımı için ICA

(Independent Component Analysis) yöntemi kullanılmıştır. Bu yöntemler kullanılarak insanların duygusal durumlarının Augmented Reality(AR) sisteminde tanımlanması sağlanmıştır. (Cyganek and Gruszczynski, 2013), çalışmalarında yakın kızılötesi resimlerden sürücülerin göz hareketlerinin tanımlanması sistemi anlatmıştır. Sistem iki ayrı sınıfta incelenebilir. Bunlar göz tespiti ve daha sonra da göz durumunun onaylanması aşamalarıdır. Göz tespiti yapılır ve daha sonra gözde meydana gelen deformasyon başlangıçtaki göz durumu ile karşılaştırılır. (Utamingrum et al., 2017) çalışmasının amacı akıllı tekerlekli sandalye için bir engelli mesafe tahmini ve gösterimini sunmaktır. Akıllı tekerlekli sandalye, engeli önlemek ve gezinmek için kamera ve çizgi lazeriyle donatılmış elektrikli tekerlekli bir sandalyedir. Kamera, yol koşulunu hissetmek için ortamdan görüntü yakalamak için kullanıldı. Lazer ise, belirli açıda lazer görüntüsünün şekline dayanan yolda bir engeli tanımak için kamera ile birlikte kullanıldı. Belirlenen engellerin modelini tanımak için lazer görüntüsünde Blob yöntemi tespiti uygulanmıştır. (Leaman et al., 2017) çalışmasında bahsedilen akıllı elektrikli tekerlekli sandalye (iChair), daha hareketli, duysal ve bilişsel bozukluklara sahip insanlara daha kaliteli, daha bağımsız bir yaşam tarzı sunmalarına yardımcı olmak için tasarlanmıştır. iChair, çeşitli giriş yöntemleri, ses, dokunmatik, yaklaşım sensörü ve kafa izleme kamerası ile çeşitli engellilik türlerinin araştırılması ve tespitinde kullanılacaktır. High Definition(HD) kamera ve 3 Boyutlu(3B) tarayıcı, engelleri tespit etmek, 3B harita oluşturmak, kılavuzları takip etmek, olayları öngörmek ve seyir yardımı sağlamak için gerekli hassasiyete sahip güvenilir veriler sunacak şekilde monte edilmiştir.

Sistemde çalışan aşamalardan birisi de görüntü işleme algoritmalarından elde edilen bilgilerin PLC (Programmable Logic Controller)'ye gönderilmesi aşamasıdır. Bu kısımda ise PLC cihazının kullanımına yönelik gerçekleştirilmiş olan çalışmalardan bahsedilecektir. (Tzokev and Topalova, 2010), çalışmalarında görüntü işleme, veri ön işleme ve PC ve PLC ile Neural Network(NN) uygulaması arasındaki haberleşme için hazırlanmış olan modelden bahsetmiştir. Model, gerçek zamanlı olarak birçok kameradan veri alıp, önceden tanımlanmış olan algoritmaları kullanarak paralel görüntü ön işleme işlemlerini gerçekleştirip, NN için giriş bilgi vektörünü hesaplar ve PLC NN uygulamasına veri olarak gönderir. (Peilin and Hong, 2017) çalışmasında, silindirik metal parçalarının üzerlerindeki pürüzlerin tespit edilmesi ve sınıflandırılması işlemlerini gerçekleştiren PLC ve buna entegresi



sağlanmış olan makine görüntü sistemi üzerine araştırmadan bahsedilmiştir. İlk olarak kameralardan görüntü alınması sağlanmıştır ve daha sonra bir görüntü işleme algoritması kenar pürüzlerinin tespiti ve tanımlanması işlemini gerçekleştirir. Son aşamada PLC Data Register kısmında tasarlanmış olan bir kontrol yöntemi ile sınıflandırma işleminin kontrolü sağlanır. (Sun et al., 2013) çalışmasında, güçlü bir şerit tespit metodu anlatılmıştır. Adaptif Eşikleme, Hough Dönüşümü yöntemleri uygulanarak şerit kenarlarının çıkarımı gerçekleştirilmiştir. İki farklı kamera kullanılarak gerçekleştirilen bu işlemlerin sonunda şerit tespiti yapılmıştır. (Mehta and Burks, 2014) çalışmasında ise narenciye ürünlerinin yetişip yetişmeme durumlarına göre değerlendirilip daha sonra bu ürünlerden yetişenlerinin toplasını sağlayan sistemin tasarımı anlatılmıştır. Sistemde sabit bir kamera ürünlerin yer tespitinin yapılmasını sağlamaktadır. Robot manipülatörünün son eklemesinde bulunan kamera ise ürünün yetişme durumunu değerlendiren kamera olarak görev yapmaktadır. Yetişen ürün ise robotun uç işlevcisi vasıtasıyla toplanır.

### 3. MATERYAL ve YÖNTEM

Bu bölümde, sistemde kullanılması planlanan yazılım-donanım araç-gereçleri ile algoritma ve metotlarından bahsedilecektir. Bu kısmı genel olarak 3 başlıkta incelenecektir.

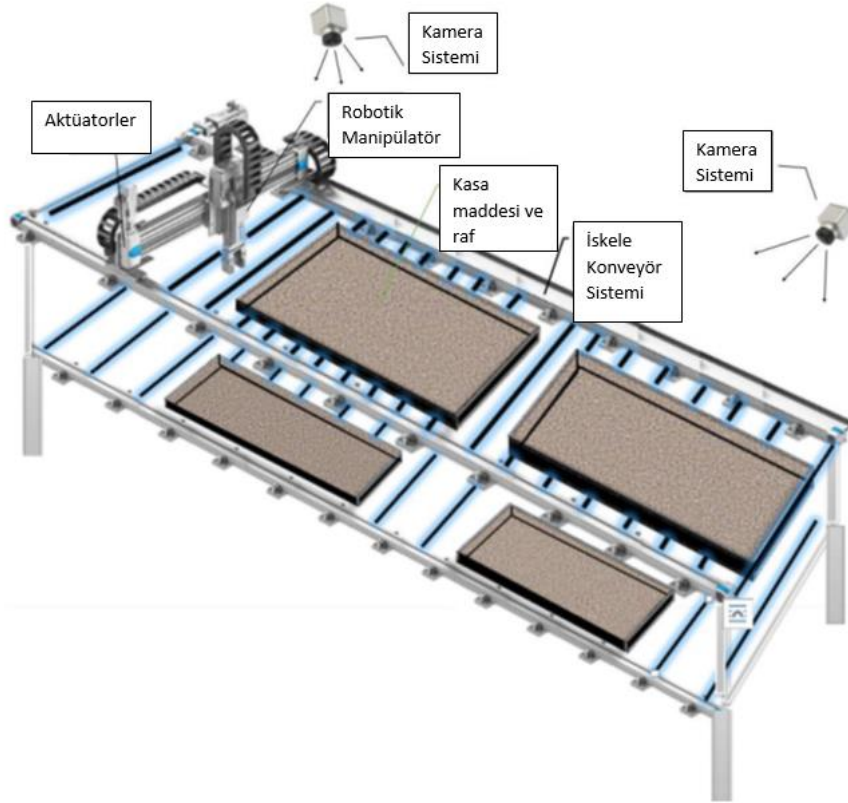
İlk olarak sistemin genel olarak çalışma mantığı anlatılacaktır. Daha sonra sistemin görüntü işleme kısmında, görüntü işleme ile ilgili kullanılan yazılım araçlarının ve kullanılan metotların tanıtımı yapılacak ve bu işlemi takiben bilgisayardaki PLC yazılımı olan TIA PORTAL yazılımı ile haberleşme konusu anlatılacaktır. En son aşamada ise PLC ile kontrol kısmına girilecektir. PLC'nin bilgisayar ile haberleşmesi, bilgisayarın C# yazılımı ile haberleşmesi ve nihayetinde PLC ile C# yazılımının haberleşmesi problemleri çözüme kavuşturularak sistemde hedeflenen çalışmalar gerçekleştirilecektir.

#### 3.1. Sistemin Genel Çalışma Mantığı

Sistem, kasada ekili halde bulunan mantarın yerinin tespitinin yapılmasını ve daha sonra X ve Y koordinatlarında lineer hareket eden Kartezyen robotun bu konuma yönlendirilmesini sağlamaktadır. Aslında sıradan bir mantar yetiştiren fabrikada mantarlar şekil 3.1'deki rafların üzerinde yetişmektedir. Bizim bu tez çıktısı sonucunda elde edeceğimiz tecrübe ve bilgi sayesinde ve KOSGEB'in desteklediği bu tez kapsamında daha sonra oluşturulacak olan Robotik Mantar Toplama sisteminde ise Şekil 3.2'deki gibi insan faktörünün minimum olduğu bir konveyör bant sistemi ve bu sistem üzerinde hareket eden mantarın bulunduğu kasalar ve bu kasaların altından geçtiği bir XY lineer robotik platform bulunmaktadır.



Şekil 3.1. Mantarların yetiştirildiği odalardaki raf sistemi

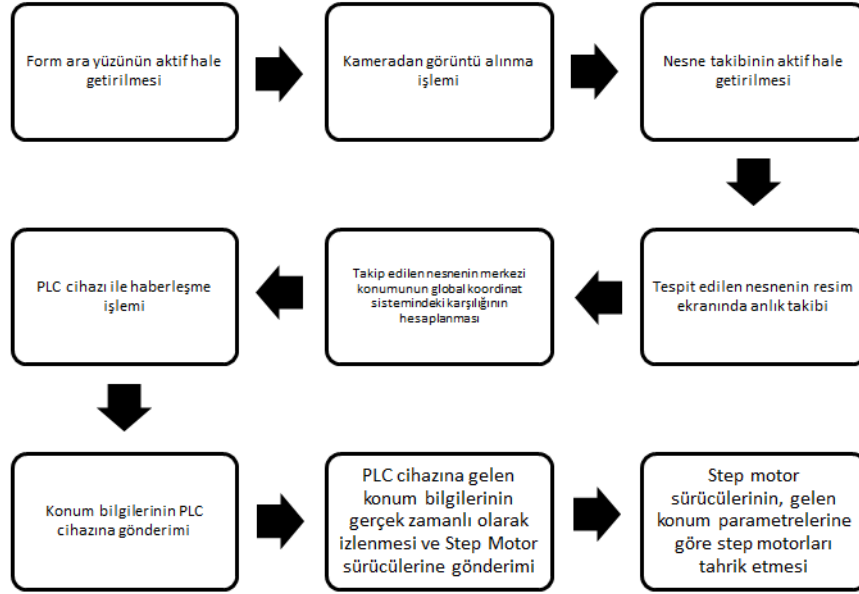


Şekil 3.2. Mantar Toplama Sisteminin Genel Görüntüsü

Kasalar içinde ekil halde bulunan mantarların tespiti için sistem girişinde bir kamera bulunmaktadır. Bu kamera, yazılım kısmında bulunan Haar-Cascade

sınıflandırıcısının mantar ürünü için eğitilmesi sonucu mantar ürününün tespitini sağlamaktadır. Önce nesnenin bulunduğu pozitif resim olarak nitelendirilen resimlerde nesnenin bulunduğu alanlar seçilir. Bu kısımda Region of Interest(ROI) kavramından bahsedilebilir. ROI, belirli bir amaç için tanımlanan bir görüntünün veya veri kümesinin bir alt kümesidir ve bir nesnenin veya bir çizimin görüntüsünde verilen sınırlarla tanımlanır. Daha sonra bu alanlardaki yapılan tarama ile nesneye ait bir ortalama piksel değeri elde edilir. Nesnenin bulunmadığı negatif resimler de sisteme tanıtılır fakat bu resimlerde herhangi bir tarama işlemi gerçekleştirilmez. Bu işlemlerin akabinde sistem çıktısı olarak bir çalışma dosyası elde edilir. Bu dosyanın yazılıma eklenmesi ile mantar tespiti işlemi gerçekleştirilir. Tespit edilen nesne öncelikle bir dörtgen içine alınır. Daha sonra bu dörtgenin merkezi konumu piksel cinsinden elde edilir. Elde edilen bu piksel değerleri gerekli kamera kalibrasyon yöntemi ve gerekli formülasyonlar kullanılarak global koordinat sisteminde XY eksenlerindeki konum bilgilerine dönüştürülür. Kameradan tespit edilen mantarın konumu Visual Studio ve TIA Portal yazılımlarının haberleşmesi ile PLC'ye lokasyon bilgisi olarak gönderilir. Bu bilgilerin işleme kısmının gerçekleştiği PLC yardımıyla Step Motor sürücülerine bu lokasyon bilgisi X ve Y eksenlerinde koordinat bilgisi olarak gönderilmektedir. Sürücülere gelen konum bilgileri ışığında step motorların bu konuma yönlendirilme aşaması gerçekleştirilir.

Sistemde gerçekleştirilen görevlerin aşamalarının bulunduğu akış şeması ise şekil 3.3'de gösterilmiştir.



Şekil 3.3. Sistemin Akış Şeması

## 3.2. Görüntü İşleme

Bu kısımda, sistemde kullanılan görüntü işleme metotları ve adımlarından bahsedilmiştir. Bu sistemde görüntü işleme metotları, nesnenin tespiti, tespit edilen nesnenin takibi ve takip edilen nesnenin global koordinat sistemindeki konumunun belirlenmesi için kullanılmıştır.

### 3.2.1. Görüntü İşlemenin Kullanım Alanları

Bir resim,  $x$  ve  $y$  uzaysal koordinatlarına bağlı 2 boyutlu bir fonksiyon ( $f(x, y)$ ) olarak ifade edilebilir. Bu  $f$  fonksiyonunun  $(x, y)$  noktalarındaki genliğine resmin o noktadaki parlaklığı ya da gri seviyesi denir. Bu noktanın  $x, y$  değerleri ile fonksiyonun bu noktadaki genliği ayrık ve sonlu değerlere sahip ise bu görüntü sayısal görüntü olarak ifade edilir. Bu görüntülerin, bilgisayarın anlayabileceği formata dönüştürülüp üzerinde gerekli işlemler yapılmasına ise sayısal görüntü işleme denir. Sayısal görüntüde, herhangi bir  $(x, y)$  noktasındaki elemana ise genel olarak piksel denir (Gonzalez and Woods, 2002).

Görüntü işleme konusu günümüzde birçok alanda kullanılmaktadır ve sağladığı avantajlar sayesinde hem bilimsel literatürde hem de endüstriyel

uygulamalarda popülerliğini sürdürmekte olan bir alandır. Kullanılan alanlar ve uygulamalar şöyle sıralanabilir (Gonzalez and Woods, 2002):

#### **Mühendislik ve Teknoloji**

- Elektronik
- Kontrol
- Yapay Zeka ve Makine Öğrenmesi Uygulamaları

#### **Tıp ve Biyoloji**

- Biyomedikal Görüntülerin Kullanımı
- MR ve Ultrasonik taramalardan elde edilen görüntülerin işlenmesi

#### **Coğrafi Bilimler**

- Jeodezi, hava ve uydu görüntülerinden hava tahmini
- Doğal kaynakların incelenmesi
- Coğrafi haritaların çıkarımı
- Ziraî ürünlerin tahmini
- Ormanların gelişiminin takip edilmesi
- Hava durumu

#### **Uzay Bilimleri**

- Uydu, mikrodalga radar görüntüleri
- Uzay derinliklerinden elde edilen görüntülerin içerdiği cisimlerin anlaşılması ve analizi

#### **Savunma Sanayi**

- Gece görüş, akıllı robot sistemleri, radar ve sonar sistemleri
- Hedef algılaması yapılarak mermi ve hava taşıtlarına kılavuzluk ve manevra yaptırılması

#### **Güvenlik Sistemleri**

- İris-parmak izi tanıma, güvenlik kamera sistemleri
- Görüntü iletimi ve depolama uygulamaları
- Güvenlik gözleme sistemleri için kapalı devre TV

#### **3.2.2. Görüntü İşlemenin Yazılımsal Adımları**

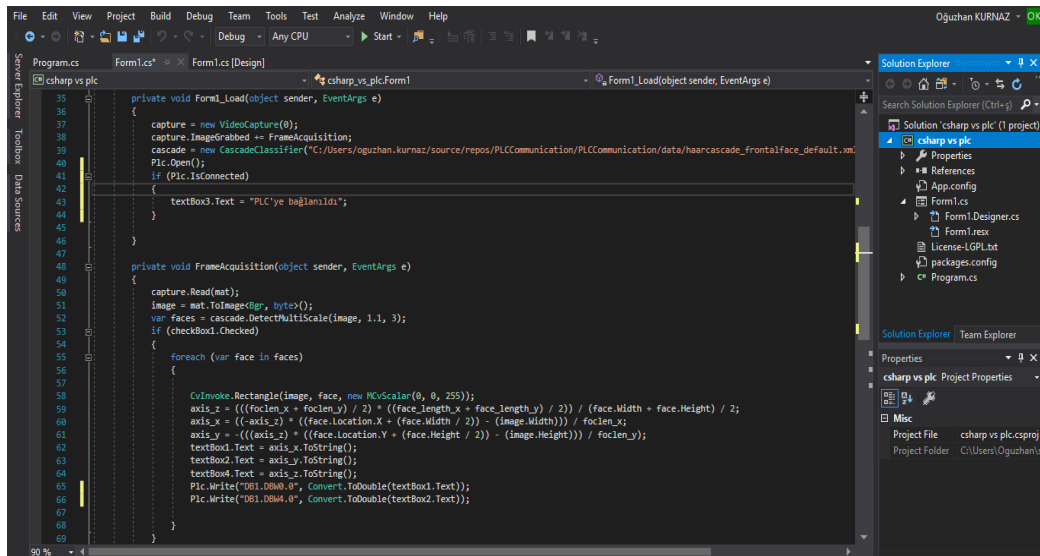
Görüntü işleme, günümüz teknolojisinde kullanılmaktadır ve birçok probleme çözüm getirmektedir. Ortaya çıkan problemin gerektirdiği ihtiyaçlara bağlı

### 3. MATERYAL ve YÖNTEM

olarak farklı görüntü işleme yöntemleri kullanılmaktadır. Bu bölümde, tez kapsamında nesne tespiti ve takibi için kullanmayı tercih ettiğimiz yazılımsal yöntemler ve yazılım araçlarından bahsedilmiştir.

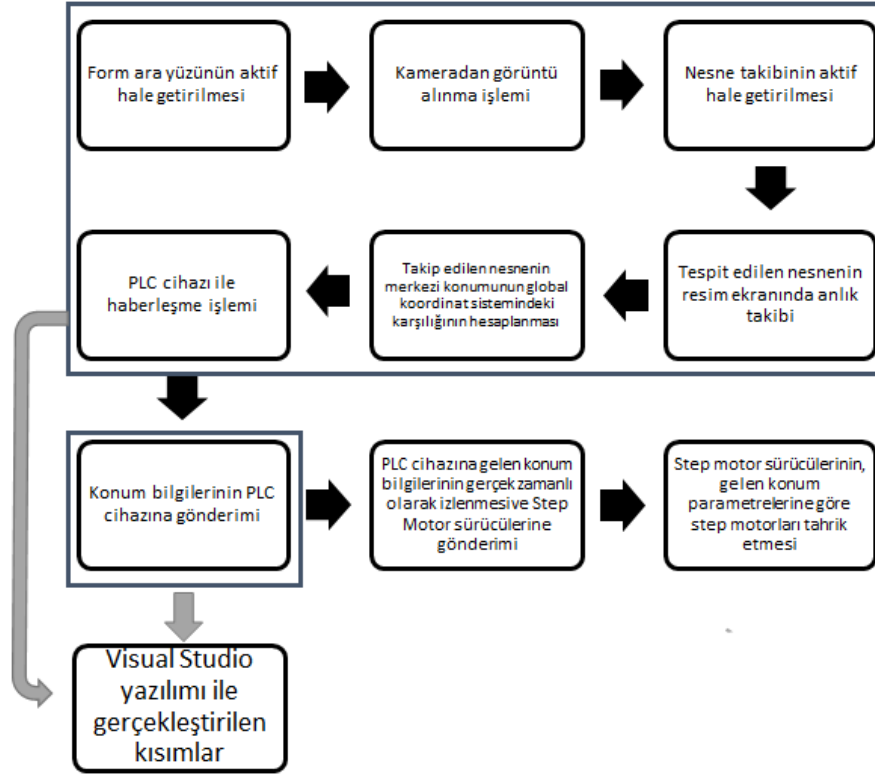
#### 3.2.2.1. Visual Studio C#.NET Yazılımı

C#, C/C++ dillerinin devamı niteliğinde olan ve Microsoft firması tarafından geliştirilmiş olan nesne yönelimli bir programlama dilidir. C# dili, C/C++ dillerinin güçlü fonksiyon yapısı ve esnekliği ile Visual Basic dilinin basit dil yapısı ve etkili geliştirme yeteneğini bir arada toplayan kullanışlı bir dildir (Y. Wang, 2011). Microsoft Visual C#.NET platformu ise, C# dili kullanılarak, Windows Form Applications vb. görsel programlamaya yönelik uygulamaların geliştirilmesini sağlamaktadır. Visual Studio C#.NET platformu, sunduğu kolay programlama ve programlamada görselliğin artırılması gibi faydaların yanı sıra, kendi bünyesinde barındırdığı ve eklenebilir birçok kütüphane, sınıf ve metotlar sayesinde görüntü işleme alanındaki çalışmalara büyük kolaylıklar sağlamaktadır. Gerçekleştirilen tez çalışmasında bu yazılımın kullanılmasının en önemli sebebi sağlamış olduğu görsel zenginliktir. Yazılımda bir form ara yüzü oluşturulmuştur. Oluşturulan bu form ara yüzü sayesinde sistemimizde tespit edilmesi istenen mantar nesnesinin tespit ve takip edilmesi görevleri gerçek zamanlı olarak gerçekleştirilmiştir. Yazılıma dair çalışmaların gerçekleştirildiği çalışma penceresi Şekil 3.4’de belirtilmiştir.



Şekil 3.4. Microsoft Visual Studio C#.NET çalışma ekranı

Daha önce belirtilen akış diyagramında oluşturulan form ara yüzü, kameradan görüntü alınması, nesne takibi, nesnenin konum bilgilerinin alınması ve konum bilgilerinin PLC cihazına gönderimi kısımları Visual Studio yazılımı sayesinde gerçekleştirilmektedir. Visual Studio yazılımının kullanıldığı aşamalara dair bir görsel şekil 3.5'te sunulmuştur.



Şekil 3.5. Visual Studio yazılımının gerçekleştirdiği görevler

#### 3.2.2.2. Emgu.CV Kütüphanesi

Görüntü işleme uygulamalarında sıkça kullanılan OpenCV kütüphanesi, C# dili ile doğrudan uyumlu bir kütüphane değildir. Bunun yerine geliştirilmiş olan Emgu.CV kütüphanesi, OpenCV kütüphanesini ve yöntemlerini içinde barındıran ve aynı zamanda C# dili ile uyumlu bir kütüphanedir. Emgu.CV, nesne tespiti ve sınıflandırılması konusunda oldukça kullanışlı bir kütüphanedir (Von Söhsten and Murilo, 2014). Emgu.CV kütüphanesinin bu bahsedilenler dışında birçok avantajı bulunmaktadır. Bunlardan bazıları şöyledir(Satari et al. 2015):



- Genel renk ve derinlik ile görüntü sınıfı,
- Gereksiz verilerin otomatik olarak tespiti ve çıkarımı,
- Resim sınıfının kullanımı veya OpenCV fonksiyonlarının çağırılması,
- Resim pikselleri üzerinde genel operasyonların uygulanmasıdır.

Gerçekleştirilen bu tez çalışmasında Emgu.CV kütüphanesi, kamera ile görüntü alınması, nesne takibi ve konum bilgilerinin elde edilmesi kısımlarında kullanılan bir kütüphanedir. Kameradan görüntü almak ve Form uygulamasında görüntülemek için gerekli olan fonksiyonlar, hedeflenen nesnenin takip edilebilmesi için gerekli fonksiyonlar ve takip edilen nesnenin konum bilgilerinin elde edilebilmesi için gerekli olan kalibrasyon işlemlerini gerçekleştiren yazılımsal fonksiyonların kullanılabilmesi için Visual Studio C# projemizde Emgu.CV kütüphanesinin kullanımı bize oldukça avantaj sağlamaktadır.

#### **3.3. Görüntü İşleme İle Mantarın Tespit Edilmesi**

Görüntü tabanlı lineer robotik platform, mantar tespiti ve konum bilgilerinin alınmasını sağlayan otonom bir sistemdir. Bu sistemde mantarın tespit edilmesi ve konum bilgilerinin belirlenmesi görüntü işleme teknikleri sayesinde gerçekleştirilir. Mantarın tespit edilmesi kısmında Haar-Cascade sınıflandırıcısı kullanılmıştır. Akabinde sınıflandırıcının eğitilmesi ve yazılım kodları içinde kullanılmak üzere XML dosyası oluşturulacaktır. Oluşturulan XML dosyası kod sistemine eklenerek kodların çalıştırılması sağlanacak ve kamera kalibrasyonunun yapılmasının ardından mantar tespit işlemi gerçekleştirilecektir. Tespit edilen mantarların konumlarına dair bilgiler hem yazılım ekranında görülecek hem de PLC cihazına konum bilgisi olarak gönderilecektir.

##### **3.3.1. Sınıflandırma Algoritması**

Sınıflandırma, elde bulunan veri kümesinin üzerinde çeşitli özelliklere göre ayrılmış olan sınıflara dağıtılması işlemidir. Sınıflandırmanın temel mantığında, elde olan ve sınıflandırılması gerçekleştirilmiş olan verilerin örnek olarak kullanılıp, sınıflandırılması istenen verilerin bu örnekler kullanılarak sınıflandırılmasını sağlamaktır. Sınıflandırma, hasta semptomlarına dayanarak bir hastalığın teşhisi için hasta veri tabanından bilgi edinme, sahtekârlık işlemlerini tanımlamak için kredi

### 3. MATERYAL ve YÖNTEM

---

kartı işlemlerini analiz etme, el yazısı örneklerine dayanan harf veya rakamların otomatik olarak tanınması ve yüksek oranda ayırt edilmesi, yüz tanıma, nesne tanıma, adli suçlu tespiti, hava tahmini gibi çeşitli uygulamalarda kullanılmaktadır (An, 2008). Ayrıca hava tahminleri, işbirlikçi filtreleme, tavsiye sistemleri, örüntü tanıma, iş başarısızlığı tahmini, portföy yönetimi ve borç riski değerlendirmesi gibi alanlarda da yine sınıflandırma yöntemleri kullanılmaktadır (Kiang, 2003).

Örnek olarak çizelge 3.1'te veri seti verilmiştir. Bu verileri yaş, boy, kilo ve cinsiyet özelliklerini kullanarak sınıflandırma işlemlerini gerçekleştirebiliriz.

<b>Yaş</b>	<b>Boy</b>	<b>Kilo</b>	<b>Cinsiyet</b>
20	175	70	Erkek
21	179	80	Erkek
19	162	50	Kız
22	169	55	Kız
20	183	90	Erkek
19	181	75	Erkek
21	171	57	Kız

Çizelge 3.1. Bilgi Olarak Verilen Veri Seti

Verilen bu örnek çalışmada verilerin cinsiyete göre sınıflandırılması gerçekleştirilmiştir. Cinsiyeti Erkek olan bireylerin Yaş, Boy ve Kilo bilgileri çizelge 3.2'de verilmiştir.

### 3. MATERYAL ve YÖNTEM

Yaş	Boy	Kilo	Cinsiyet
20	175	70	Erkek
21	179	80	Erkek
20	183	90	Erkek
19	181	75	Erkek
20	179,5	78,75	Ortalama

Çizelge 3.2. Cinsiyeti Erkek Olan Bireylere Ait Bilgiler

Cinsiyeti Bayan olan bireylere ait Yaş, Boy ve Kilo bilgileri ise çizelge 3.3'de verilmiştir.

Yaş	Boy	Kilo	Cinsiyet
21	171	57	Kız
19	162	50	Kız
22	169	55	Kız
20,6667	167,33	54	Ortalama

Çizelge 3.3. Cinsiyeti Kız Olan Bireylere Ait Bilgiler

Bu gibi sınıflandırma işlemlerinin gerçekleştirilmesi amacıyla kullanılan sınıflandırma algoritmaları bulunmaktadır. Bazı sınıflandırma algoritmaları ise şöyledir (Anonim, 2013):

- Destekçi Vektör Makinesi(Support Vector Machine)
- Doğrusal Olmayan SVM(Non-Linear SVM)
- Veri Akış Madenciliği(Data Stream Mining)
- Naif Bayes Sınıflandırıcısı(Naive Bayes Classifier)
- Karar Ağacı Öğrenmesi(Decision Tree Learning)
- K-En Yakın Komşu(KNN, K-Nearest Neighborhood)

Bu algoritmaların dışında, 2001 yılında Paul Viola ve Michael Jones tarafından geliştirilen ve obje tespiti uygulamalarında sıkça kullanılan bir sınıflandırma yöntemi olan Haar-Cascade Sınıflandırma (Viola and Jones, 2001) yöntemi, bu çalışmada mantar tespiti için seçilmiş olan sınıflandırma metodudur.

#### 3.3.2. Haar-Cascade Sınıflandırma Yöntemi

Haar-Cascade sınıflandırıcısı, kullanımı geniş çaplı olan bir sınıflandırma algoritmasıdır. Sınıflandırma algoritmalarının kullanım alanlarından biri de nesne tespittir. (Viola and Jones, 2001) tarafından gerçekleştirilen çalışmada insanların yüz tespiti için bir algoritma geliştirilmiştir. Zayıf sınıflandırıcı adı verilen Haar Benzeri özniteliklerin üst üste uygulanarak kullanılmasıyla güçlü sınıflandırıcı olarak kabul edilen Haar-Cascade sınıflandırıcısı ortaya çıkmıştır. Bu sınıflandırıcı ilk olarak yüz tanıma için ortaya çıkmış bir çalışma olmasına karşın zamanla diğer nesnelerin tespiti ve takibi için de kullanılmaya başlanmıştır. Algoritmanın eğitilebilir olması nesne tespiti için büyük bir avantaj sağlamaktadır. Sağladığı yüksek doğruluk oranında nesne tespiti ve hızlı işlem gerçekleştirme kapasitesinin yanında bu algoritma aynı zamanda istenen herhangi bir nesne için eğitilebilir (Hiromoto et al., 2009). Bu tez çalışmasında da sağladığı bu avantajların yanında mantar nesnesinin tespiti için daha işlevsel bir algoritma olan Haar-Cascade sınıflandırıcı kullanılmıştır.

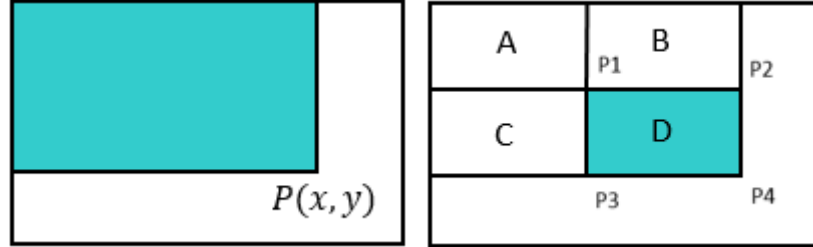
Bu algoritma; Haar öznitelikleri, İntegral Resim, AdaBoost öğrenme algoritması ve Cascade sınıflandırıcısı olarak dört temel bölümden oluşmaktadır. Algoritma verimliliğini artırmak için ilk olarak İntegral Resim üretilmektedir. İntegral Resim  $ii(x, y)$  ile gösterilir. Burada  $(x, y)$  resmin konum noktasını ifade etmektedir. Bu resme ait denklem ise denklem 1'de verilmiştir.

$$ii(x, y) = \sum i(x', y') \quad x' \leq x, y' \leq y \quad (1)$$

Burada  $ii(x, y)$  İntegral Resim ve  $i(x, y)$  ise orijinal resimdir.

İntegral Resim ise şöyle bulunur; herhangi bir  $(x, y)$  noktasındaki pikselin değerine, o noktanın solu ve üst kısmında kalan tüm piksellerin toplam değeri yazılır. Bu şekilde resimde bulunan tüm piksel değerleri yerine, kendi bulunduğu

konumun solunda ve üstünde kalan piksel değerlerinin toplamının yazılması ile İntegral Resim elde edilmiş olur. Şekil 3.6'de sol tarafta İntegral Resmin  $(x, y)$  noktasındaki değeri  $P(x, y)$  olarak gösterilmiştir. Sağda ise Orijinal Resim vardır ve P1, P2, P3 ve P4 noktaları gösterilmiştir.



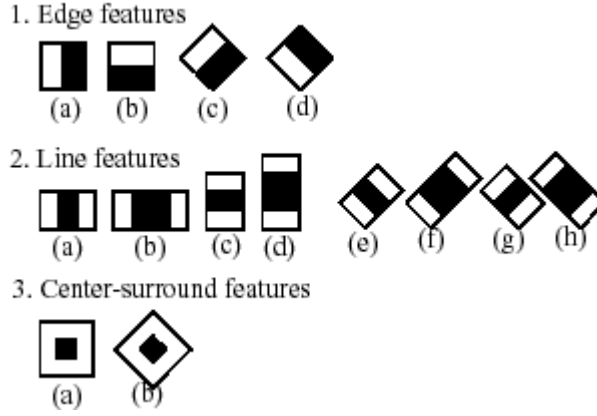
Şekil 3.6. Solda İntegral Resim ve sağda Orijinal Resim

Soldaki İntegral Resim'de  $P(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y')$  şeklinde ifade edilir.

Orijinal Resimde gösterilen bu 4 noktanın piksel değerleri toplamı, solda gösterilen  $P(x, y)$  değerine eşittir. İntegral Resim tanımına geri dönecek olursak, İntegral Resim'deki herhangi bir noktanın piksel değeri, o noktanın solunda ve üstündeki tüm piksel değerlerinin toplamına eşittir. Yani;

$$P1 = A, P2 = A + B, P3 = A + C, P4 = A + B + C + D \quad (2)$$

şeklinde ifade edilebilir. Nesne tespiti için, öncelikle algoritmanın eğitilmesi gerekmektedir. Bu eğitime işlemi için, tespit edilecek olan nesnenin içinde bulunduğu pozitif resimler ile nesnenin bulunmadığı negatif resimlerin oluşturduğu veri setine ihtiyaç vardır. Pozitif resimlerde hedef nesnenin tespit edilmesi için, öncelikle eğitim için kullanılacak olan verilerde hedef nesnenin bulunduğu kısımlar seçilmelidir. Nesne tespitinin doğruluk oranının yüksek olması için hedef nesne sınırlarının nizami bir şekilde seçilmesi gerekmektedir. Hedef nesne seçimi yapıldıktan sonra, şekil 3.7'de gösterilen Haar benzeri öznitelikler kullanılarak öznitelikteki siyah alanların piksel değerleri toplamı ile beyaz alanların piksel değerleri toplamının farkı sonucunda özniteliğin yoğunluğu, denklem 3'teki gibi hesaplanmış olur.



Şekil 3.7. Haar Benzeri Öznitelikler(Viola and Jones, 2001)

$$f(x) = \sum_{SiyahAlan}(Yoğunluk) - \sum_{BeyazAlan}(Yoğunluk) \quad (3)$$

Bu öznitelikler kullanılarak eğitim aşaması gerçekleştirilirken, 24x24 boyutlarında oluşturulan bir alt-pencere, resmin üzerinde kaydırılarak pozitif ve negatif bölgelerin tespiti yapılır. Pozitif ve negatif resimlerin tespiti sırasında piksellerin toplamı ve değer elde edilmesi çok zaman alıcıdır. Normal resim kullanılarak nesne tespiti yapılacak olsaydı, eğitim aşamasında 160000'den fazla öznitelik çıkarılması ve bunlardan piksel değerlerinin hesaplanması gerekirdi. Bu uzun süreli işlemleri kısaltması amacıyla (Viola and Jones, 2001), İntegral Resim yöntemini öne sürmüştür. Bu sayede hesaplama süresi minimize edilerek algoritmanın çalışması daha da hızlandırılmış olur. Şekil 3.8'de gösterilen resimde ise algoritmanın AdaBoost kısmında tüm bu özniteliklerden, hedef nesne için uygun olanlarının seçilip diğer özniteliklerin elimine edilmesi işlemi gerçekleşir. Öznitelikler sırayla denir ve hedef nesne için uygulandığı bölgeye uygun olmayan öznitelikler elimine edilir. Böylece öznitelik sayısı düşürülerek hız veriminin artırılması sağlanmıştır.

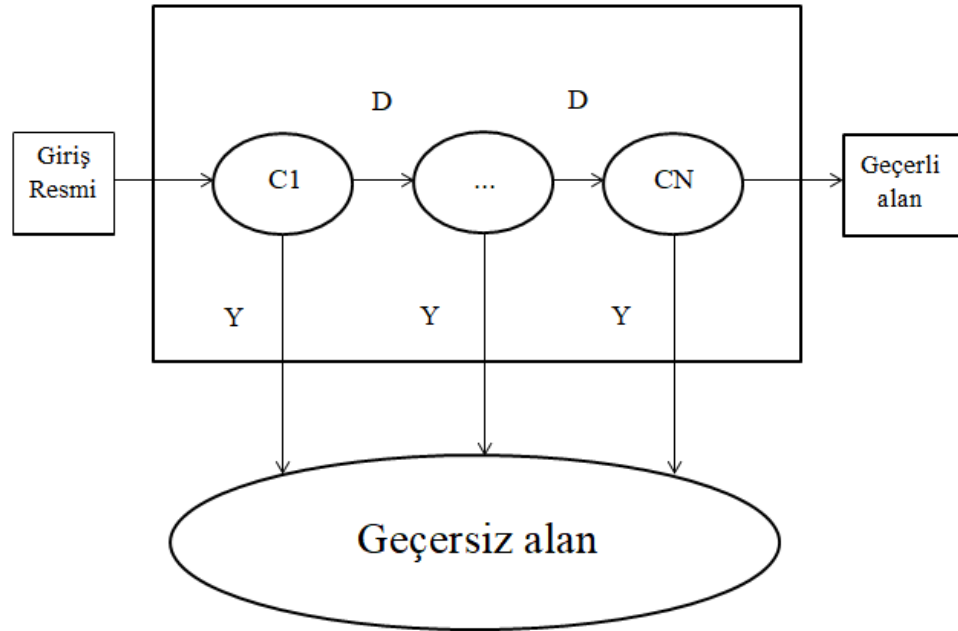


Şekil 3.8. AdaBoost ile belirlenen uygun öznitelik (solda) ve uygun olmayan öznitelik (sağda)(Lazebnik ve Viola, 2016)

Kullanılan bu sınıflandırıcıda her bir öznitelik aynı zamanda zayıf bir sınıflandırıcı olarak nitelendirilir. Denklem 4'te gösterilen denklemde güçlü bir sınıflandırıcı oluşturulması için zayıf sınıflandırıcı olan özniteliklerin kullanımı gösterilmiştir.

$$F(x) = a_1f_1(x) + a_2f_2(x) + a_3f_3(x) + \dots \quad (4)$$

Bu denklemde  $f_n(x)$  zayıf sınıflandırıcı,  $F(x)$  ise güçlü sınıflandırıcı olarak nitelendirilir. Algoritmanın son aşamasında ise bu güçlü sınıflandırıcıların art arda uygulanması ile şekil 3.9'de gösterilen Cascade sınıflandırıcısı elde edilir. Her bir  $C_1, C_2 \dots C_N$  elemanı, güçlü sınıflandırıcıdır.



Şekil 3.9. Cascade Sınıflandırıcısı (Viola and Jones, 2001)

Yukarıdaki şekil 3.9'da Haar-Cascade sınıflandırıcısının nesne tespiti aşamalarının özeti gösterilmiştir. Nesneyi barındıran ve pozitif resim olarak adlandırılan resimlerde nesnenin bulunduğu alanlar bir dörtgen içerisine alınır. Daha sonra Haar Benzeri Öznitelikler yardımıyla nesnenin bulunduğu alanın ortalama piksel hesabı yapılır. Önce Kenar, daha sonra Satır ve son olarak da Merkez-Çevre Öznitelikleri pozitif resim üzerinde nesnenin bulunduğu alana yerleştirilir. Yerleştirme işlemlerinden sonra özniteliklerdeki siyah ve beyaz alanlarda kalan piksellerin toplam değerleri bulunur. En son aşamada bu beyaz ve siyah alandaki

toplam piksel değerlerin arasındaki yoğunluk farkı bulunarak zayıf sınıflandırıcılar ( $f_n(x)$ ) elde edilmiş olur.

Nesnenin bulunduğu dörtgen alanda yapılan tarama işleminde elde edilen her bir zayıf sınıflandırıcı denklem 4'deki gibi gerekli formülasyonlar uygulanarak güçlü sınıflandırıcılar ( $F(x)$  veya  $C_N$ ) elde edilir. Tüm bu işlemlerin son aşamasında elde edilen bu güçlü sınıflandırıcılar, sistemde alınan görüntü üzerinde uygulanarak görüntüde nesnenin var olup olmadığına dair tespit işleminin gerçekleştirilmesini sağlamaktadır.

Tüm bu anlatılan aşamalar Haar-Cascade sınıflandırıcısının nesne tespiti sırasında gerçekleştirilen aşamalarıdır. Bu tez çalışmasında mantar nesnesinin tespiti için kullandığımız Haar-Cascade sınıflandırıcı tüm bu işlemleri mantar nesnesi için gerçekleştirmektedir. Önce mantar nesnesine ait resimler pozitif resim olarak kaydedilmektedir. Sonrasında bu pozitif resimlerde mantar nesnesinin bulunduğu alanlar dörtgen bir bölge içine alınarak sınıflandırıcıya tanıtılmaktadır. Daha sonra sınıflandırıcı bu dörtgensel alana Haar-Benzeri Öznitelikleri yerleştirerek tarama işlemlerini başlatır. Her bir öznitelik yerleştirildikten sonra siyah ve beyaz alanlarda kalan piksellerin değerlerinin toplamı bulunur. Akabinde ise bu siyah bölge ve beyaz bölgeden elde edilen değerler arasındaki fark değeri bulunarak zayıf sınıflandırıcılar elde edilir. Sonraki aşamada ise bu zayıf sınıflandırıcılar belirli bir formülasyona bağlı olarak toplanır ve güçlü sınıflandırıcılar elde edilir. Elde edilen bu güçlü sınıflandırıcı bizim eğitim sonrası elde ettiğimiz ve yazılım kısmında sisteme tanıttığımız sistem dosyasıdır. Bu sistem dosyası ile kameradan alınan görüntü üzerinde tarama yapılır. Güçlü sınıflandırıcı ile elde ettiğimiz değerlere yakın değerlerde olan kısımlarda nesnenin var olduğu kısımlardır ve bu tespit edilen nesne bir dörtgensel alanda gösterilir ve nesnenin takibi bu şekilde gerçekleştirilir.

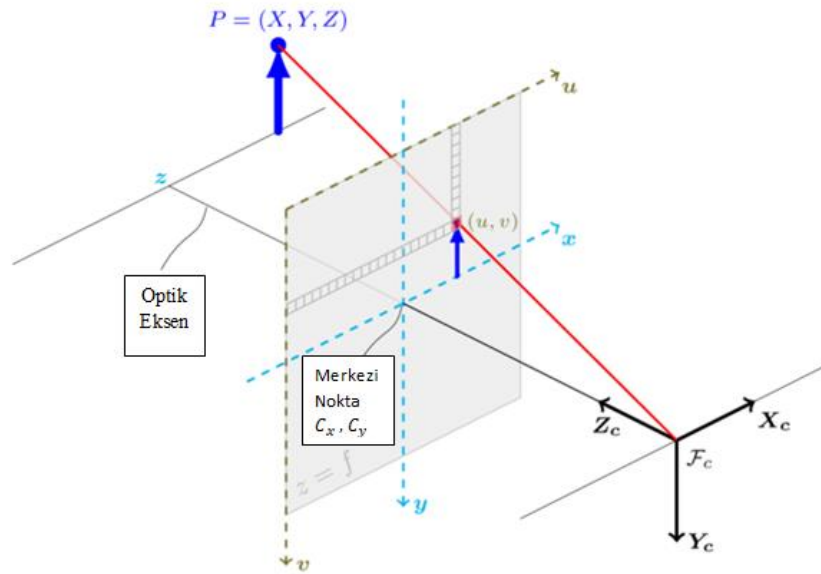
#### **3.3.3. Kamera Kalibrasyonu**

Dijital kameralar, görüntü işleme uygulamalarında sıklıkla kullanılmaktadır. İşlenmek üzere elde edilen görüntüler, dahili veya haricen bağlanmış olan kameralar ile gerçekleştirilir. Dijital kameralar ile hassas ölçümlerin yapılabilmesi için mutlaka kalibre edilmeleri gerekmektedir. Fotogrametrik yöntemler, cismin bir veya daha



fazla resmini kullanarak cismin şeklini, boyutunu ve konumunu belirlemeyi amaç edinmiş bir bilim dalıdır. Fotogrametrik yöntemlerin kullanımında, kameraya ait iç yöneltme parametreleri bilinir ve cisim noktalarının koordinatları bulunur. Kamera kalibrasyonu ise bu yöntemlerin tersi mantığında çalışır. 3 boyutlu düzlemde koordinatları bilinen cisim noktaları ile kameraya ait iç yöneltme parametrelerinin elde edilmesi sağlanır. Literatürde birçok kamera kalibrasyonu metodu mevcuttur. Bunlar lineer, lineer olmayan ve çoklu adım teknikleri olarak sınıflandırılabilir. Lineer yöntemler diğerlerine nazaran daha hızlıdır ve iterasyon gerektirmez. Fakat doğruluğu düşüktür çünkü kamera modeli sadeleştirilmesi gerekmektedir. Bu kategoride en yaygın olarak kullanılan metotlar DLT (Direct Linear Transform) ve Tsai's metotlarıdır (Yakar, 2009). Bu çalışmada kullanılan yöntem ise DLT yöntemidir.

DLT yöntemi, çözümün lineer olması ve yaklaşık değer probleminin olmaması konusunda sağladığı avantajlar ile kullanışlı bir yöntem haline gelmiştir. DLT eşitlikleri sayesinde resim koordinatlarından uzay koordinatlarına ulaşmak mümkündür. Kameraya ait 11 parametrenin çıkarılması ile iç yöneltme elemanları tespit edilmiş olur. Daha sonraki aşamada bulunan bu değerler yardımıyla 2 boyutlu bir resimde piksel değerleri bilinen bir noktanın gerçekteki konumunun tespiti yapılabilir hale gelmektedir.



Şekil 3.10. İğne Deliği Kamera Modeli(Olsson, 2013)

Şekil 3.10’de gösterilen koordinat sistemlerinin dönüşümleri vasıtasıyla kamera kalibrasyonu işlemi gerçekleştirilir. Kalibrasyon için öncelikli olarak gerçek koordinat sisteminin, kamera koordinat sistemine dönüşümü aşağıda verilen denklem 5 ile sağlanır;

$$\begin{pmatrix} X_C \\ Y_C \\ Z_C \end{pmatrix} = R \begin{pmatrix} X_P \\ Y_P \\ Z_P \end{pmatrix} + t \quad (5)$$

Burada R, 3x3 boyutlu rotasyon matrisi ve t ise 3x1 boyutlu dönüşüm matrisidir. P noktası, resim düzleminde gösterilen cismin global koordinat sistemindeki konumunu göstermektedir. C noktası ise, cismin kamera koordinat sistemindeki konumunu ifade eder.

Daha sonra kamera koordinat sistemi, iğne deliği kamera modeli (Olsson, 2013) yöntemi kullanılarak resim düzlemi koordinat sisteminde fiziksel koordinatlar denklem 6’daki gibi elde edilir.

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} X_C/Z_C \\ Y_C/Z_C \end{pmatrix} \quad (6)$$

Burada x ve y noktaları, cismin resim düzlemi koordinat sistemindeki koordinatlarıdır. Radyal ve Teğetsel Lens bozulmalarının da dâhil edilmesi ile resim düzlemi koordinat sistemindeki genişletilmiş fiziksel koordinatlar ( $x_q$  ve  $y_q$ ) aşağıdaki gibi elde edilir.

$$\begin{pmatrix} x_q \\ y_q \end{pmatrix} = (1 + k_1 r^2 + k_2 r^4) \begin{pmatrix} X_C/Z_C \\ Y_C/Z_C \end{pmatrix} + \begin{pmatrix} 2p_1 xy + p_2(r^2 + 2x^2) \\ p_1(r^2 + 2y^2) + 2p_2 xy \end{pmatrix} \quad (7)$$

Burada  $k_1$  ve  $k_2$  radyal distorsiyon katsayıları,  $p_1$  ve  $p_2$  ise teğetsel distorsiyon katsayılarıdır. Ayrıca  $r^2 = x^2 + y^2$ ’dir. Son olarak elde edilen genişletilmiş fiziksel koordinatlar, piksel koordinatlarına aşağıda verilen denklem ile dönüştürülür.

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} f_x x_q + u_0 \\ f_y y_q + v_0 \end{pmatrix}, \quad f_x = f \cdot \frac{s}{\Delta x} \quad \text{ve} \quad f_y = f / \Delta y \quad (8)$$

Burada ise  $(u_0, v_0)$  temel noktadır ve genellikle resmin merkezi olarak seçilir.  $f$ , kameranın odak uzaklığıdır.  $s$ , ölçek faktörüdür.  $\Delta x$ , yatay ekseninde iki komşu piksel arasındaki mesafenin mm cinsinden değeridir.  $\Delta y$ , dikey ekseninde iki komşu piksel arasındaki mesafenin mm cinsinden ifadesidir.

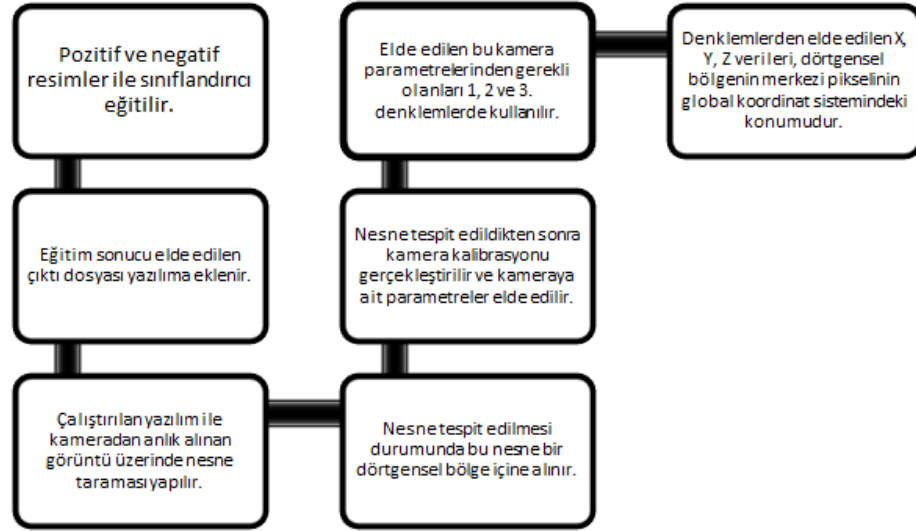
Bu eşitlikler kullanılarak:

$$s \cdot p = A(R|t) \cdot P \quad (9)$$

denklemi elde edilir. Bu denklemde  $p = (u, v, 1)^T$ , resmin piksel noktasının homojen koordinatlarıdır.  $P = (X, Y, Z, 1)^T$ , uzaysal düzlemdeki noktanın homojen koordinatlarıdır.  $A$ , kameranın iç yöneltme parametrelerinin bulunduğu matristir.  $(R|t)$ , harici parametrelerin bulunduğu matristir. Bu denklem ifadeleri sayesinde resimdeki bir nokta ile uzay düzlemindeki bir nokta arasındaki ilişki tanımlanmış olur (Y. M. Wang et al. 2010).

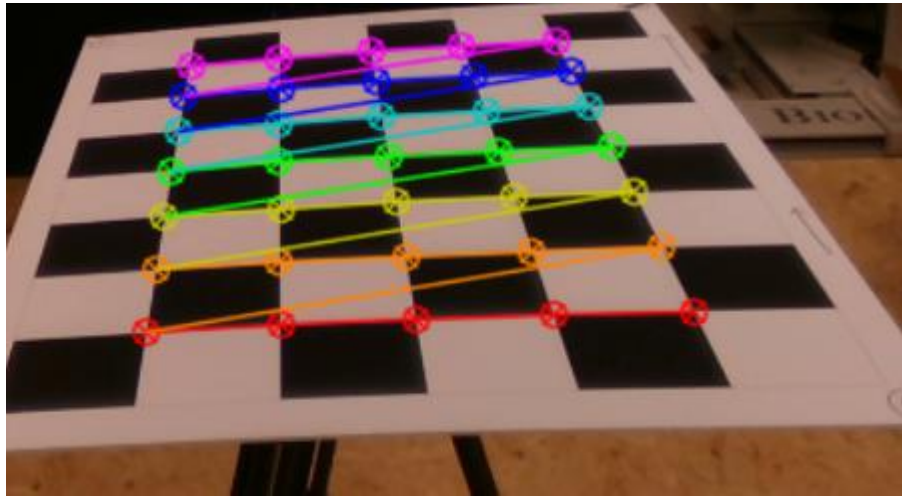
Kalibrasyonun gerçekleştirilmesi aşamasında, Emgu.CV kütüphanesi ile görsel olarak satranç tahtası resimleri kullanılarak kalibrasyon gerçekleştirilecek. Daha sonra kalibrasyon ile elde edilecek olan parametreler, resimde tespit edilen objenin merkezi pikselinin lokasyon bilgisi kullanılarak objenin gerçekte hangi konumda olduğu tespit edilecektir. Kalibrasyon için öncelikle kameraya satranç tahtası resmi farklı açılardan gösterilecek ve resimleri kaydedilecektir. Daha sonra kaydedilen bu resimlerin köşe noktalarının bulunması sağlanacaktır. Köşe noktaları bulunduktan sonra resme ait iç ve dış yöneltme parametreleri bulunacak ve hata analizleri gerçekleştirilecektir. Bulunan bu iç ve dış parametreler ile birlikte tespiti gerçekleştirilen mantarın görüntüsünün merkezindeki pikselin lokasyon bilgisi kullanılarak mantarın uzay düzleminde bulunduğu konumu tespit edilecektir. Tespit edilen konum sayesinde robotun bu konuma yönlendirilmesi gerçekleştirilecektir.

Mantarın tespiti aşamasında gerçekleşen işlemler şekil 3.11'de gösterilmiştir.



Şekil 3.11. Mantarın tespit aşamaları

Şekil 3.12’de ise kalibrasyon işlemi gerçekleştirilirken kullanılan satranç tahtasının köşe noktalarının bulunması aşaması gösterilmiştir.



Şekil 3.12. Kamera Kalibrasyonu için satranç tahtası köşelerinin tespit edilmesi

Kalibrasyon işleminin gerçekleştirilerek konum bilgilerinin doğrudan elde edilebilmesi için 3 boyutlu eksenle iz düşüm alma özelliği kullanılmaktadır. 3 boyutlu eksenle iz düşüm alma özelliğinin kullanılabilmesi için stereo kamera sisteminin kullanılması gerekmektedir. Fakat sistemde ürün tespiti ve konum bilgisi

alınması amacıyla tek bir kamera kullanıldığı için konum hesaplama işlemleri farklı bir yöntemle gerçekleştirilecektir.

Gerçekleştirilen bu kalibrasyonu işlemi ile kullanılan kameraya ait X ve Y eksenlerindeki odak uzaklıkları elde edilir. Bu odak uzaklıkları konum hesaplanmasında kullanılır. Göz bebeği konumunun tespitinin anlatıldığı (Wood and Bulling, 2014) çalışmasında kullanılan formülasyonların bizim sistemimize uyarlanması ile X, Y ve Z eksenlerindeki konumlar elde edilebilir.

$$d_z = ((f_x + f_y)/2) * ((dg_x + dg_y)/2) / ((mrm_x + mrm_y)/2) \quad (10)$$

$$d_x = ((-d_z) * \left( mrb_x + \left( \frac{mrm_x}{2} \right) \right) - (kg_x)) / (f_x) \quad (11)$$

$$d_y = -((d_z) * \left( mrb_y + \left( \frac{mrm_y}{2} \right) \right) - (kg_y)) / (f_y) \quad (12)$$

(10), (11) ve (12) numaralı denklemlerde kullanılan parametrelerin anlamları ise şöyledir:

### 3. MATERYAL ve YÖNTEM

$d_z$	Z eksenindeki konum
$d_x$	X eksenindeki konum
$d_y$	Y eksenindeki konum
$f_x$	Kameranın X eksenindeki Odak Uzaklığı
$f_y$	Kameranın Y eksenindeki Odak Uzaklığı
$mgb_x$	Mantarın gerçekteki yatay boyutu
$mgb_y$	Mantarın gerçekteki dikey boyutu
$mrb_x$	Mantarın resim düzlemindeki yatay boyutu
$mrb_y$	Mantarın resim düzlemindeki dikey boyutu
$mrbn_x$	Mantarın resim düzlemindeki başlangıç noktasının X eksenindeki değeri
$mrbn_y$	Mantarın resim düzlemindeki başlangıç noktasının Y eksenindeki değeri
$kg_x$	Kameradan alınan görüntünün X eksenindeki boyutu
$kg_y$	Kameradan alınan görüntünün Y eksenindeki boyutu

Çizelge 3.4. (10), (11) ve (12) numaralı denklemlerde kullanılan parametrelerin açıklamaları

Tespit edilen mantarın pozisyonu, gerçekleştirilen kalibrasyon işlemleri ile daha hassas bir şekilde tespit edilir. Tespit edilen bu koordinatların PLC cihazına gönderilmesi için S7.Net kütüphanesi kullanılmaktadır. Bu kütüphanenin Visual Studio ortamına eklenmesi vasıtasıyla ve gerekli kod aşamalarının tamamlanmasıyla

PLC cihazı ile Visual Studio programı haberleşmesi gerçekleştirilir. Bu kısma dair teknik detaylar ve kod kısmının bulunduğu resimler daha sonra Visual Studio C# Yazılımı ile PLC Cihazının Haberleşmesi kısmında bahsedilecektir. Sistemde nesne tespiti için hazırlanmış olan yazılım kodları EK-1’de sunulmuştur.

#### **3.4. PLC İle Kontrol Kısmı**

Görüntü tabanlı lineer robotik sisteminin 2. aşaması olan PLC cihazı ile kontrol sisteminde, görüntü işleme teknikleri ile tespit edilen mantarların konum bilgilerinin alınıp PLC cihazına gönderilmesi aşaması ile devam eden kısımlar gerçekleştirilir. Daha sonra PLC cihazına gelen bu koordinat verileri, gerekli değerlendirme aşamalarından geçirilir ve robotun bu koordinatlara gönderimi sağlanır.

##### **3.4.1. Kullanılan Yazılım ve Donanımların Tanıtılması**

Kontrol kısmında PLC cihazı, step motor sürücüler ve step motorlar donanım olarak kullanılacak olan malzemelerdir. PLC cihazının programlanabilmesi için SIEMENS firması tarafından geliştirilen ve oldukça kullanışlı bir ara yüz olan TIA PORTAL programı, program akış şemalarının derlenip denemelerinin gerçekleştirilmesini ve anlık izlenmesini sağlar. Step motor sürücü olarak FESTO marka CMMS-ST-C8-7-G2 tipi sürücüler ve EMMS-ST-57-S-SE-G2 tipi step motorlar kullanılmıştır. Bu cihazların birbirleriyle haberleşmesi için gerekli olan haberleşme protokolleri gerçekleştirilerek robotun istenilen konuma gönderilmesi problemi çözüme kavuşturulmuş olur. Aşağıda belirtilen alt bölümlerde çalışmanın yazılım ve donanım tarafında kullanılan kısımlar belirtilmiştir.

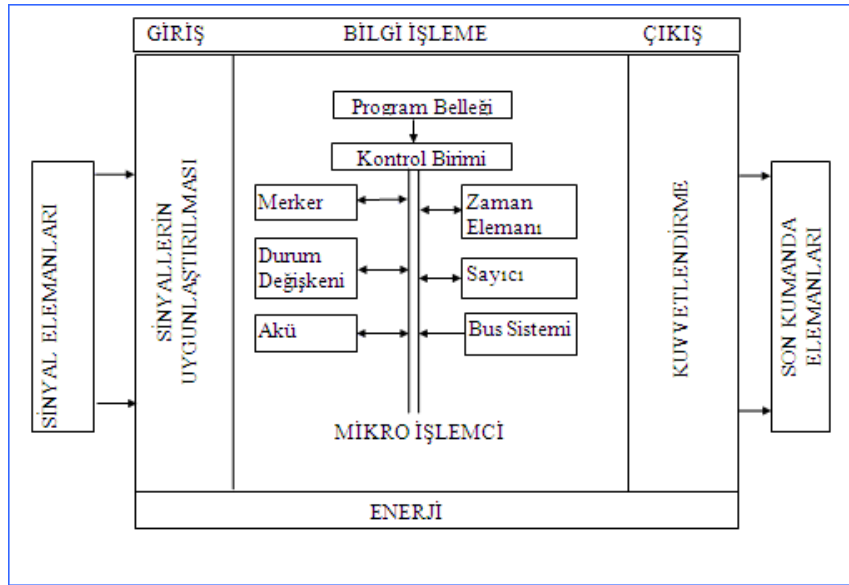
##### **3.4.1.1. SIEMENS S7-1200 PLC Cihazı**

Programlanabilir denetleyiciler (PLC=Programmable Logic Controller), herhangi bir problemten bağımsız olarak geliştirilmiş olan genel kumanda ve kontrol elemanlarıdır. Bütün kumanda sorunlarının çözüme kavuşturulmasında; mantık işlemleri, bellek fonksiyonları, zaman ve sayıcı gibi elemanların kullanılması gerekebilir. Tüm bunların entegre edildiği cihazlar olan PLC’ler, basit bir programlama ile problemlerin çözümüne imkan sunmaktadır (Eminoğlu, 2015). Bir

### 3. MATERYAL ve YÖNTEM

PLC, genel olarak üç bölümden oluşur. Birincisi merkezi işlem bölümü (CPU), ikincisi bütün sinyal elemanlarının bağlandığı giriş bölümü (Giriş Kartı) ve son olarak kumanda edilecek elemanların bağlandığı çıkış bölümüdür (Çıkış Kartı). Genelde her PLC ilk önce kendi besleme gerilimi olan 24V DC'yi veya 220V AC'yi CPU için 5 Volta indirgeyen bir modüle sahiptir. Giriş ve çıkış bölümleri için genelde DC 24 Volt kullanılmaktadır fakat bu değer 220 Volta kadar değişebilmektedir (Eminoğlu, 2015).

Bir PLC'ye ait temel çalışma mantığı ve sistem elemanları Şekil 3.13'de gösterilmiştir.



Şekil 3.13. PLC'nin çalışma mantığı ve sistem elemanları

PLC cihazı, girişten alınan bilgilerin alındıktan sonra değerlendirmesinin yapılarak bu değerlendirme sonucuna göre çıkış üreten gelişmiş bir mikrodenetleyicidir. PLC cihazına sensörler, limit anahtarları, butonlar gibi Sinyal Elemanları'ndan giriş bilgileri gelir. Bu gelen sinyaller analog veya dijital olabilir. Bu sinyallerin PLC cihazının anlayabileceği uygun forma dönüştürülme işlemi, Sinyallerin Uygunlaştırılması kısmında gerçekleştirilir. Sinyaller gerekli forma dönüştürüldükten sonra Bilgi İşleme evresine geçilmiş olur ve Program Belleği'nde tasarlanmış ve kaydedilmiş olan senaryo bu gelen sinyallere göre gerçekleştirilir. Kontrol Birimi ise bu senaryonun gerçekleştirilmesi aşamasında gerekli olabilecek Zaman Elemanı, Sayıcı gibi özelliklerin kullanımında yardımcı olur. Bilgi İşleme

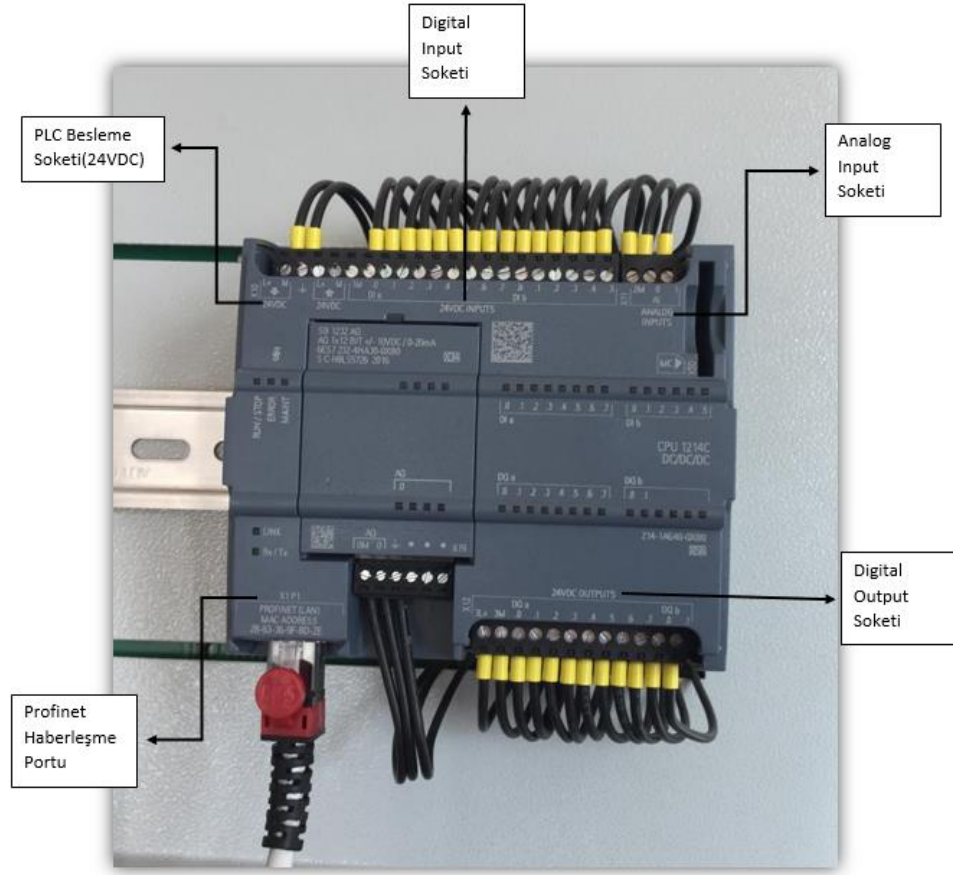


evresinde gerçekleşen değerlendirmeler sonucunda elde edilen çıktı sistem çıktısı olarak kabul edilir. Yine bu çıktılar da analog veya dijital olabilir. Bu çıktının çıkış elemanları tarafından anlaşılabilir formda dönüştürülmesi için çıkış sinyallerinin güçlendirilmesi ve sinyallerin uygunlaştırılması gerekmektedir. Gerekli uygunlaştırma işlemlerinden sonra sinyallerin güçlendirilme işlemi ise Kuvvetlendirme kısmında gerçekleştirilmektedir. Son kısımda ise bu çıkış sinyalleri ile alarmlar, valfler, motorlar, röleler gibi çıkış elemanları olarak nitelendirilen elemanlara çıkış verilmesi sağlanır.

SIEMENS firması tarafından birçok PLC tipleri üretilmiş ve piyasaya sürülmüştür. S7-200, S7-300 ve S7-400 tipi PLC'ler bu zamana kadar en sık tercih edilen PLC tipleri olmuştur. Daha sonra firma tarafından S7-1200 ve S7-1500 tipi PLC'ler geliştirilmiştir ve günümüzde yaygın olarak kullanıldığı bilinmektedir. S7-1200 ve S7-1500 modellerinden önce S7-200 modelleri küçük ölçekli projeler kapsamında, S7-300 modeli orta ölçekli projeler kapsamında ve S7-400 modeli ise büyük ölçekli projeler kapsamında kullanılmaktaydı. Ancak gelişen teknoloji ve buna bağımlı olarak artan istek ve ihtiyaç talepleri, daha gelişmiş cihazların üretimini gerektirmekteydi. Firma tarafından üretilen S7-1200 tipi PLC, bunun ilk adımı olarak görülebilir.

Bu çalışmada kullanılacak olan PLC tipi ise S7-1200 tipi modeldir. Modele dair görünüm şekil 3.14'de ve teknik detaylar ise şekil 3.15'de sunulmuştur.

### 3. MATERYAL ve YÖNTEM



Şekil 3.14. S7-1200 PLC Cihazı

ÖZELLİK		CPU 1211C	CPU 1212C	CPU 1214C	CPU 1215C
Fiziksel boyut		90x100x 75	90x100x 75	110x100x 75	130x100x 75
Kullanıcı belleği	Çalışma	25 Kbayt	25 Kbayt	50 Kbayt	100 Kbayt
	Yükleme	1 Mbayt	1 Mbayt	4 Mbayt	4 Mbayt
	Kalıcı	10 Kbayt	10 Kbayt	10 Kbayt	10 Kbayt
Kompakt I/O	Dijital	6 Giriş / 4 Çıkış	8 Giriş / 6 Çıkış	14 Giriş / 10 Çıkış	14 Giriş / 10 Çıkış
	Analog	2 Giriş	2 Giriş	2 Giriş	2 Giriş/2 Çıkış
Proses İmaj alanı	Giriş (I)	1024 bayt	1024 bayt	1024 bayt	1024 bayt
	Çıkış (Q)	1024 bayt	1024 bayt	1024 bayt	1024 bayt
Bit memory (M)		4096 bayt	4096 bayt	8192 bayt	8192 bayt
Genişletilebilir sinyal modülü (SM)		Yok	2	8	8
Sinyal (SB) veya haberleşme (CB) bord'u		1	1	1	1
Haberleşme modülü (CM) sol yan		3	3	3	3
Hızlı sayıcı	Toplam	3	4	6	6
	Tek fazlı	3 adet 100 KHz SB 2 adet 30 KHz	3 adet 100 KHz 1 adet 30 KHz SB 2 ad 30 KHz	3 adet 100 KHz 3 adet 30 KHz	3 adet 100 KHz 30 KHz 3 adet
	Dört fazlı	80 KHz 3 adet SB 2 ad 20 KHz	80 KHz 3 adet 20 KHz 1 adet SB 2 ad 20 KHz	80 KHz 3 adet 20 KHz 3 adet	80 KHz 3 adet 20 KHz 3 adet
Pals çıkış		4	4	4	4
Hafıza kartı		SIMATIC Hafıza kartı (Opsiyonel)			
Gerçek zaman saati kalıcılık		20 gün, 40°C en az 12 gün			
PROFINET		1 Ethernet haberleşme kanalı			2
Matematiksel işlem hızı		2.3 µs/komut			
Binary işlem hızı		0.08 µs/komut			

© Festo Eğitim ve Danışmanlık

Yavuz EMİNOĞLU

Şekil 3.15. S7-1200 modeline ait PLC çeşitleri ve teknik özellikleri

S7-1200 tipi PLC'nin sağladığı bazı avantajlar ise şöyledir:

- Üzerinde bulunan Ethernet portu ile haberleşme işleminin daha kolay hale gelmesine yardımcı olur.
- Eski model PLC'lere nazaran yazılan programların çevrim süresi (cycle time) daha kısadır.
- Boyut olarak oldukça kompakt bir yapıdadır.
- CPU ve I/O terminallerinin kolayca sökülüp takılmasına imkan sağlar. Böylece olası arıza durumlarında doğrudan PLC'nin değiştirilmesi yerine terminallerin değiştirilmesi sayesinde saha az maliyetle arıza giderimi sağlanmış olur.
- S7-1200 tipi PLC'nin kullanıldığı haberleşme protokolü Profinet(Ethernet)'tir. Bu protokol, cihazın üzerinde bulunan Ethernet portu sayesinde gerçekleştirilmektedir. Bunun yanında, ihtiyaç halinde RS485 ve RS232 haberleşme portlarından da yararlanılabilir.

PLC'nin programlanabilmesi için ise TIA PORTAL ara yüzü geliştirilmiştir. Bu ara yüze dair bilgiler bir sonraki bölümde anlatılmıştır.

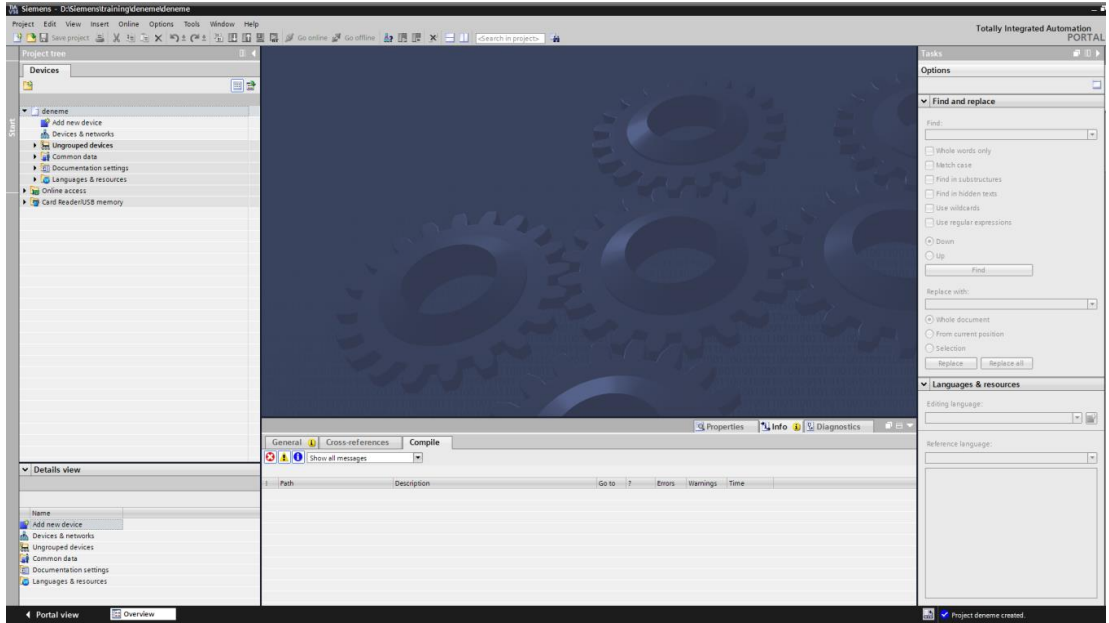
#### 3.4.1.2. TIA PORTAL Yazılımı

TIA (Totally Integrated Automation) PORTAL yazılımı, SIEMENS firması tarafından geliştirilmiş olan ve PLC, HMI, Scada sistemleri ile haberleşme vb. problemlere çözüm getiren oldukça kullanışlı bir arayüzdür. Bir önceki bölümde bahsedilen eski model PLC'lerin programlanabilmesi için hepsinin ayrı ayrı programlama yazılımları mevcuttu. S7-200'ler için MICROWIN, S7-300/400'ler için SIMATIC MANAGER yazılımları kullanılmaktaydı ve bu yazılımların yapıları birbirinden farklıydı. Ancak TIA PORTAL ile eski model PLC'lerin yanı sıra piyasaya sürülen S7-1200 ve S7-1500 tipi PLC'lerin programlanması problemi tek bir çatı altında toplanmış oldu. Kullanılması istenen PLC tipinin TIA PORTAL yazılımında modül olarak eklenmesi vasıtasıyla gerekli programlama işlemleri gerçekleştirilebilir. TIA yazılımı ile, istenilen PLC'lerin yanı sıra sisteme HMI (Human Machine Interface), haberleşme modülleri, Scada sistemi ile haberleşme, motor sürücüler gibi elemanların eklenerek yazılım ortamında istenilen cihazların

### 3. MATERYAL ve YÖNTEM

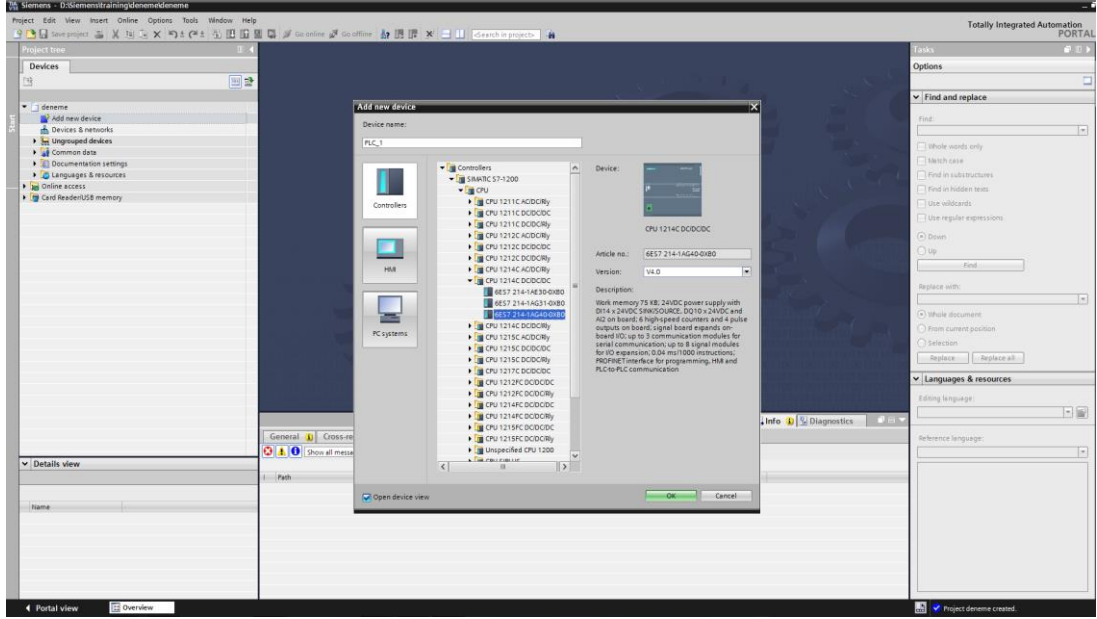
birbirleriyle haberleşmesi problemleri tek çatı altında toplanmıştır. Sisteme ilave herhangi bir yazılım kullanmadan kontrol cihazlarının birbirleriyle haberleşmesi sağlanır.

TIA PORTAL yazılımına ait olan ve genel ara yüz, cihaz ekleme kısmı ve programlama kısmı gibi kısımlar, şekil 3.16, şekil 3.17 ve şekil 3.18’de gösterilmiştir. Öncelikle TIA programı çalıştırılır ve karşımıza şekil 3.16’daki ara yüz çıkar. Projede çalışacağımız PLC’nin modelinin seçimi ve projeye eklenmesi şekil 3.17’de gerçekleştirilir. PLC eklendikten sonra şekil 3.18’de gösterilen boş proje sayfası karşımıza çıkar.

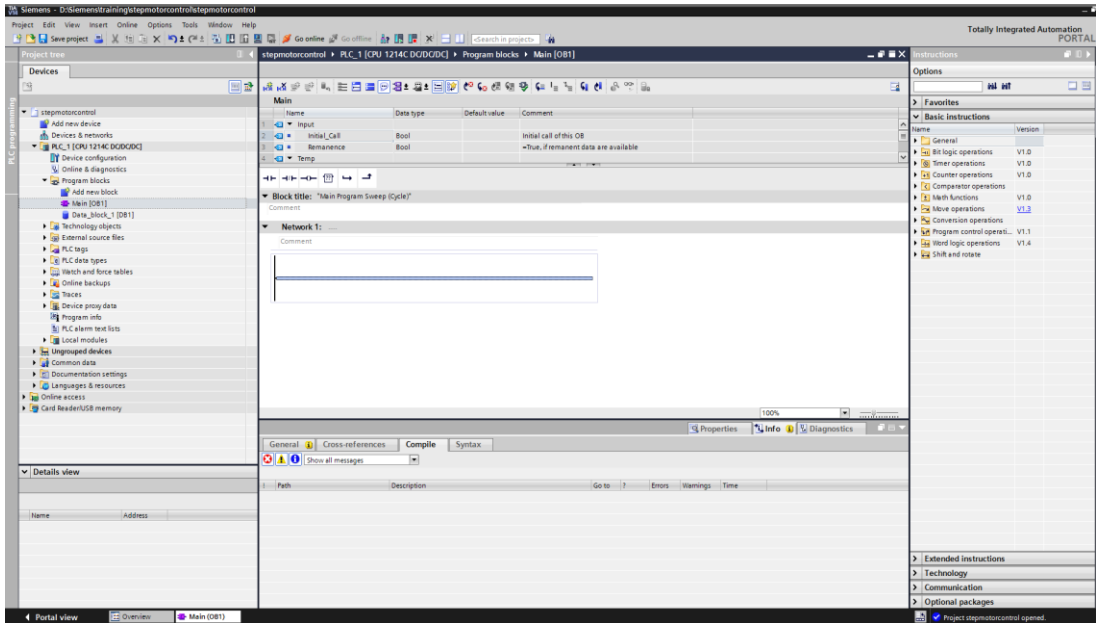


Şekil 3.16. TIA PORTAL programı ara yüzü

### 3. MATERYAL ve YÖNTEM



Şekil 3.17. Projeye kullanılan cihazın eklenmesi



Şekil 3.18. Program bloklarının oluşturulması

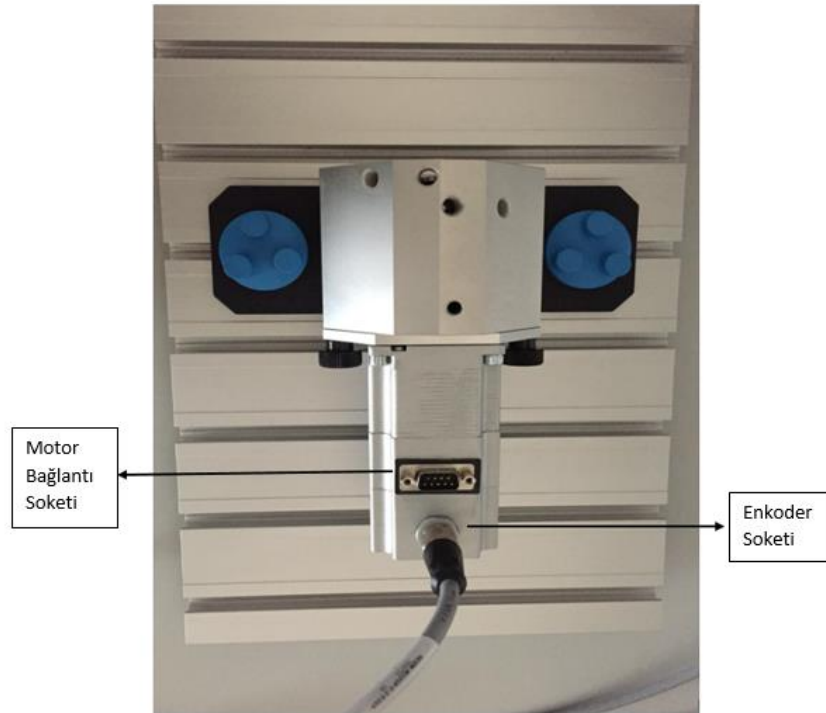
#### 3.4.1.3. Step Motor

Motor, herhangi bir enerji türünü mekanik enerjiye çeviren makinedir. Elektrik motoru ise elektrik enerjisini dönme hareketine çeviren elektro-mekanik bir makinedir. Step motor da bir elektrik motorudur. Step motorlara aynı zamanda adım

### 3. MATERYAL ve YÖNTEM

motorları da denmektedir. Adım motorları, açısal konumu adımlar halinde değiştiren, çok hassas sinyaller ile sürülebilen motorlardır. Adım motorları belirli adımlarla hareket eder. Herhangi bir uyarımda motorun rotor kısmının yapacağı hareketin ne ölçüde olacağı, motorun adım açısına bağlıdır. Adım açısı, motorun yapısına bağlı olarak 90°, 45°, 18°, 7,5° gibi değerler alabilir. Motora uygulanacak sinyallerin frekansı değiştirilerek motorun hızı kontrol edilebilir. Adım motorlarının dönüş yönü ise, uygulanan sinyallerin uygulanış sırasına bağlı olarak saat yönünde dönüş (CW) veya saat yönünün tersi yönde dönüş (ACW) şeklinde olabilir. Adım motorlarının dönüş yönü, dönüş hızı, tork değeri gibi parametreler bilgisayar veya mikroişlemciler yardımıyla kontrol edilebilir (Anonim, 2011). Bu sistemde de motorların kontrolü, PLC ve step motor sürücüsü yardımıyla yapılmaktadır.

Lineer robotik sisteminde FESTO firmasına ait bir ürün olan EMMS-ST-57-S-SE-G2 kodlu step motor kullanılacaktır. EMMS ürünün motor olduğuna dair belirlenmiş olan koddur. ST step motor olduğunu, 57 mm flanş uzunluğu, S küçük boyutlu olduğunu, SE ise Doğrudan Bağlantı ile Enkoder eklentilerinin olduğunu ve G2 ise ikinci jenerasyon ürün olduğunu göstermektedir. Motora ait görüntü şekil 3.19'da ve motora ait teknik bilgiler çizelge 3.5 ve çizelge 3.6'da verilmiştir.



Şekil 3.19. Step Motor ve bağlantı kısımları(Anonim, 2012).

Size of Motor		57-S
Nominal Voltage	[V DC]	48
Nominal Current	[Amp]	5
Max. rotational speed	[rpm]	2720
Holding torque	[Nm]	0.8
Stepper angle	[°]	1.8±5%
Winding resistance	[Ω]	0.15±10%
Winding inductance	[mH]	0.5
Output mass moment of inertia [ $kgcm^2$ ]		0.29/0.30
Radial load on shaft	[N]	52
Axis load on shaft	[N]	10
Mass moment of inertia of rotor [ $kgcm^2$ ]		0.29

Çizelge 3.5. 57 mm flanş uzunluğuna sahip S tipi step motorun teknik özellikleri

Encoder Type		Optical
Operating Voltage	[V DC]	5
Pulses/Revolution	[1/rev]	500
Zero Pulse		Yes
Line Driver		RS422 protocol

Çizelge 3.6. Motorlardaki dahili enkoderin özellikleri

#### 3.4.1.4. Step Motor Sürücüsü

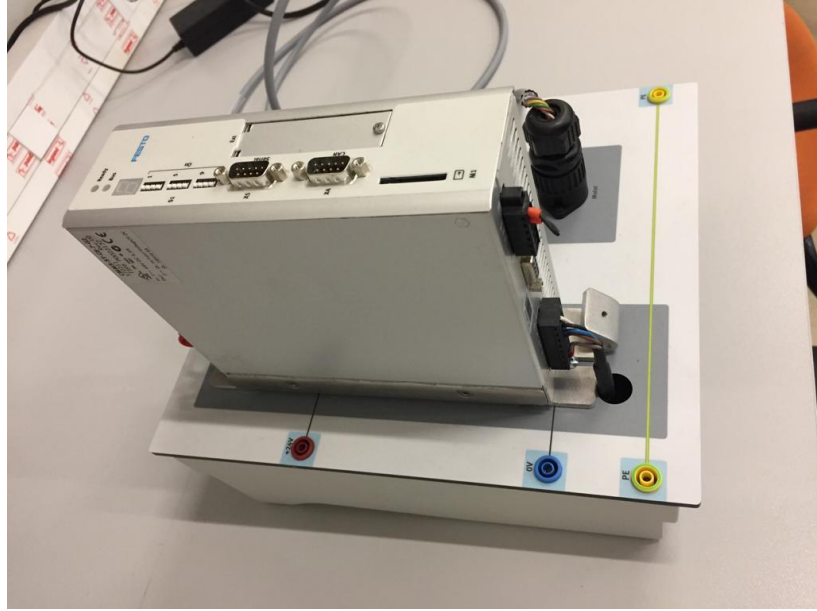
Endüstriyel uygulamalarda adım motorları oldukça kullanışlı bir cihazdır. Bu step motorların pozisyon, hız, dönüş yönü, adım sayısı gibi parametrelerin kontrolü için sürücü devresi mutlaka gereklidir. Bunun için geliştirilmiş olan step motor sürücüler, istenilen parametrelerin düzenlenmesini veya kontrolünü gerçekleştirmeye yardımcı birer elemandır. PLC cihazı ile gerekli haberleşme protokolleri sağlanmış

### 3. MATERYAL ve YÖNTEM

---

olan motor sürücüler, istenilen bir motorun sürülmesi konusunda büyük kolaylıklar sağlar.

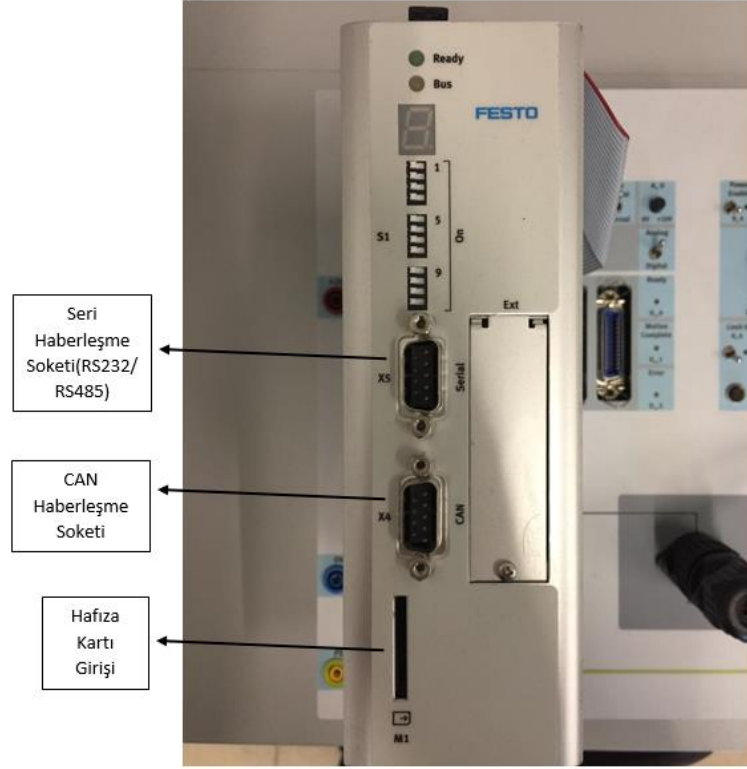
Lineer robotik platform sisteminde de kullanılan 2 adet step motor için 2 adet motor sürücü kullanılmıştır. Bu sürücüler de yine step motorların üretici firması olan FESTO tarafından üretilen CMMS-ST-C8-7-G2 kodlu step motor sürücülerdir. Ürüne ait görsel, şekil 3.20’de gösterilmiştir. CMMS kısmı standart motor kontrolcüsü olduğunu, ST kodu step motor kontrolcüsü olduğunu, C8 Nominal Akım değerinin 8 Amper olduğunu, 7 kodu giriş voltajının maksimum 48 V DC olduğunu ve G2 kısmı ise ikinci jenerasyon bir ürün olduğunu göstermektedir.



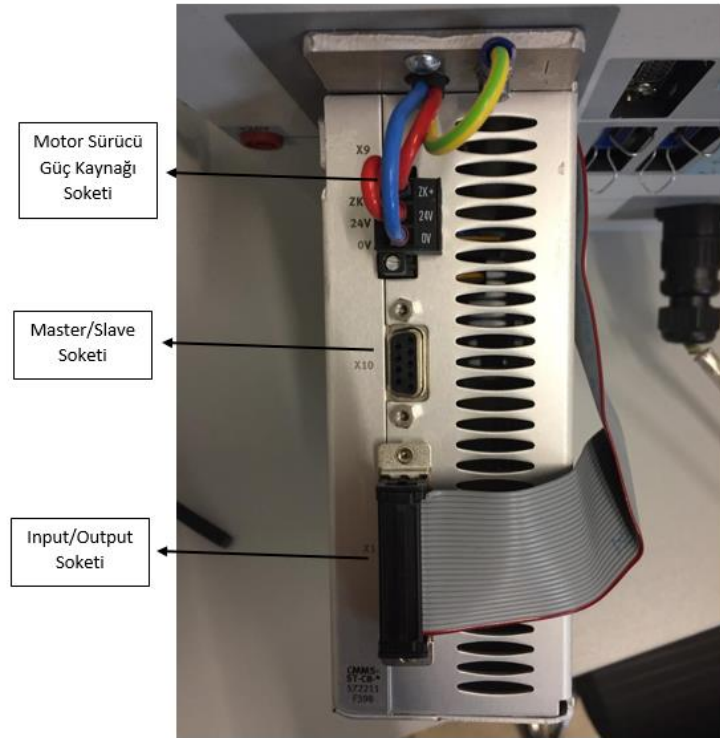
Şekil 3.20. Sistemde kullanılan motor sürücüsü

Şekil 3.21, şekil 3.22 ve şekil 3.23’de ise kullanılan sürücünün ön, üst ve alt görüşleri ile üzerinde bulunan bağlantı portlarının isimlerinin bulunduğu görsel verilmiştir.

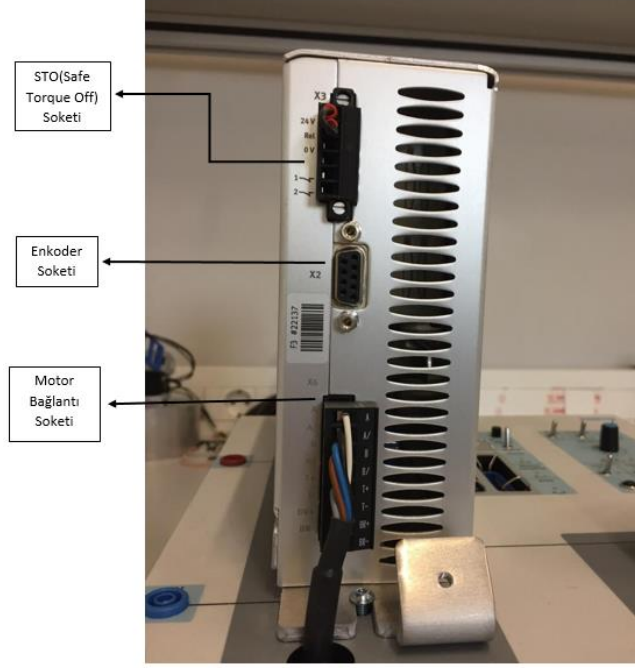




Şekil 3.21. Sürücü ön görünüşü ve bağlantı soketleri

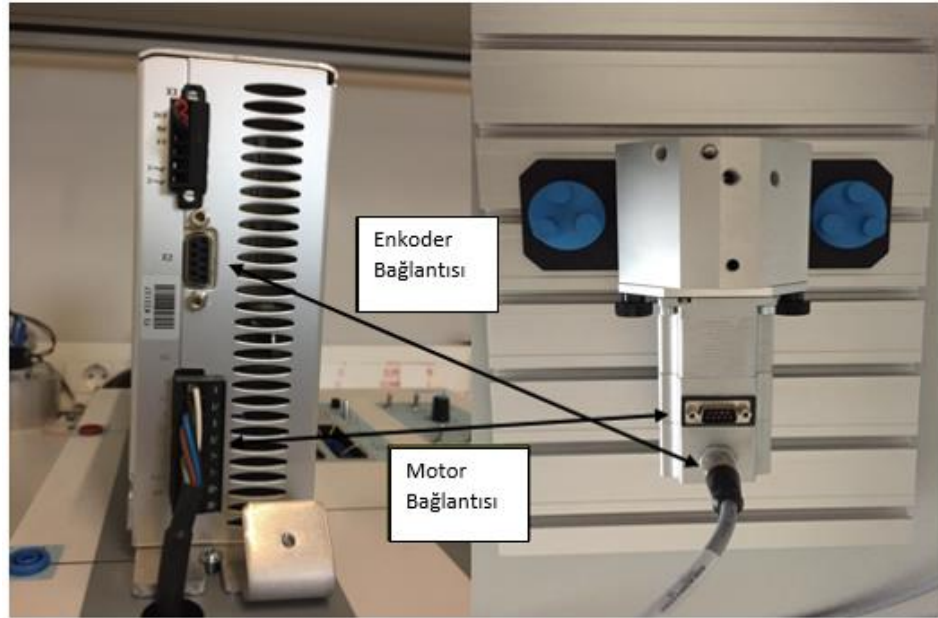


Şekil 3.22. Sürücü üst görünüşü ve bağlantı soketleri



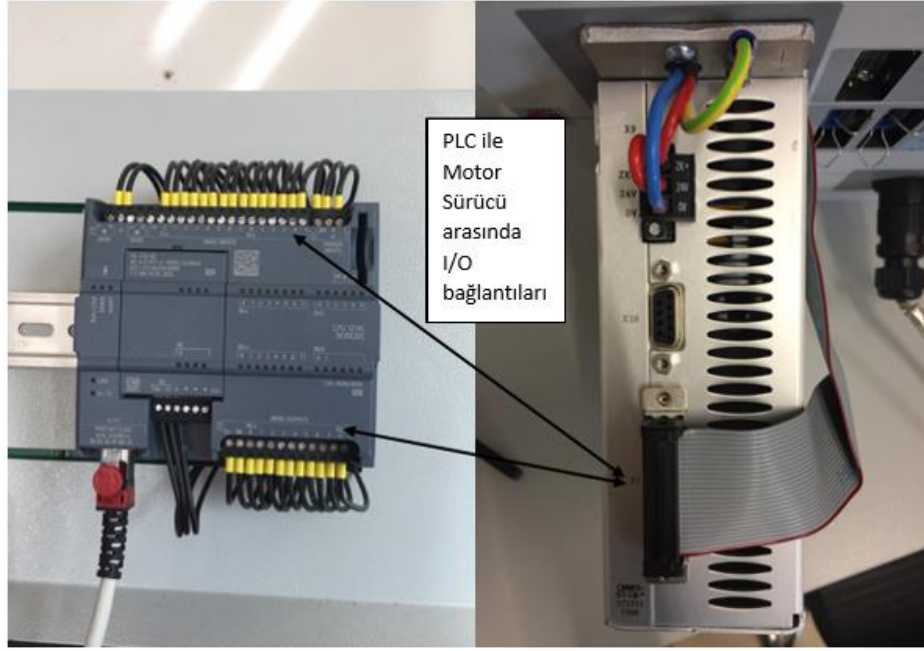
Şekil 3.23. Sürücü alt görünüşü ve bağlantı portları

Sistemde motor sürücü ve motorun bağlantısı şekil 3.24’de gösterilmiştir.



Şekil 3.24. Motor sürücü ile motor arasındaki bağlantı

PLC cihazı ile motor sürücü arasındaki bağlantılar ise şekil 3.25’de gösterilmiştir.



Şekil 3.25. Motor sürücü ile PLC cihazı arasındaki bağlantı

Burada PLC cihazından sürücüye gönderilen çıkış sinyalleri ve sürücüden PLC cihazına gelen giriş sinyallerinin kullanıldığı soketler gösterilmektedir. Sürücü kısmında enkoderden gelen hız ve konum bilgileri sürücüye enkoder soketinden gerçekleştirilen bağlantı ile sağlanır. Enkoderden alınan bu hız ve konum bilgileri sürücüden çıkış sinyali olarak PLC cihazına giriş sinyali olarak gönderilir. PLC’de değerlendirmesi gerçekleştirilen bu bilgilerden sonra motora verilmesi gereken tahrik bilgileri, çıkış klemenslerinden sürücüye giriş olarak gönderilir ve sürücüde değerlendirildikten sonra motor soketine bağlı motora çıkış olarak gönderilir.

#### 3.4.2. PLC Cihazı İle Bilgisayarın Haberleşmesi

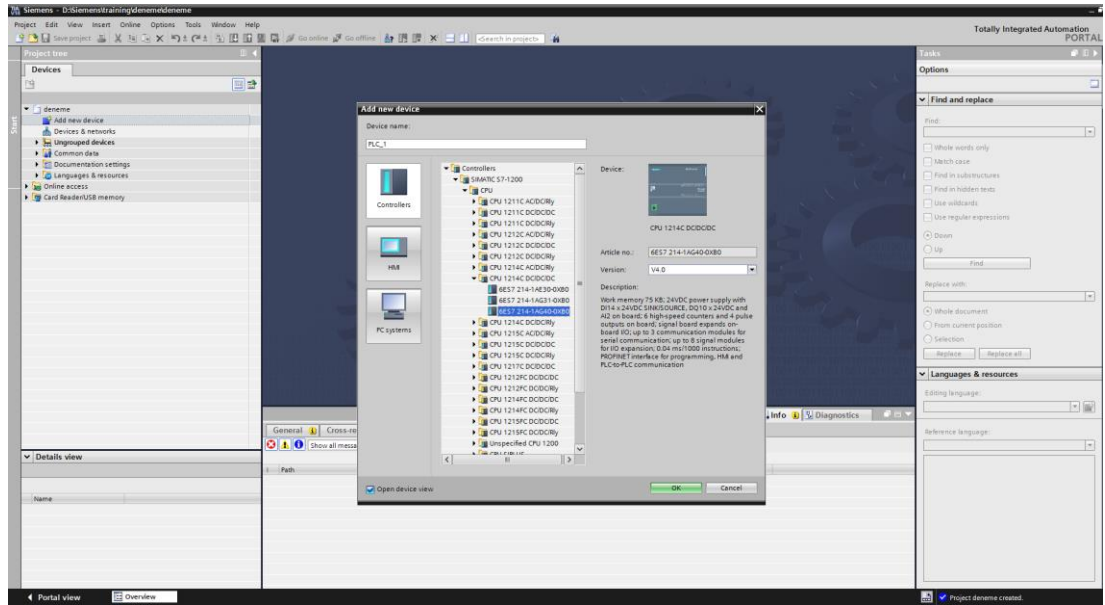
PLC cihazı ile bilgisayarın haberleşmesi, PLC üzerinde bulunan Ethernet portu sayesinde gerçekleştirilebilmektedir. Sistemin kontrol kısmına ait gerekli program kodlamaları tamamlandıktan sonra kodun derlemesi gerçekleştirilir. Derleme işleminde herhangi bir hata ile karşılaşılmaması durumunda kodların tekrardan gözden geçirilmesi gerekir. Hata olmaması veya varsa giderilmesi durumunda kod çalıştırılması aşamasına geçilir. TIA PORTAL ile yazılan program bloklarının PLC cihazı içinde bulunan hafızaya yüklenmesi gerekir. Bunun için PLC’nin bilgisayara

### 3. MATERYAL ve YÖNTEM

tanıtılması ve bağlantı yapılması gerekir. Tüm bu işlemlerin gerçekleştiği aşamalar aşağıda maddesel olarak sıralanmıştır:

- PLC cihazı ile bilgisayarın Ethernet portlarına Ethernet kablosu(CAT5) bağlanır.
- TIA ara yüzünde PLC modeli eklenir(S7-1200 CPU 1214C DC/DC/DC)
- Eklenen PLC'ye IP adres ataması yapılır. Bu yapılan atama bilgisayar ile aynı protokolde olmalıdır. (Örneğin, PLC için 192.168.0.1 iken bilgisayar için 192.168.0.2)
- TIA ara yüzünde PLC cihazına bağlanma işleminin gerçekleştirilmesi ve gerçek zamanlı olarak incelenmesi

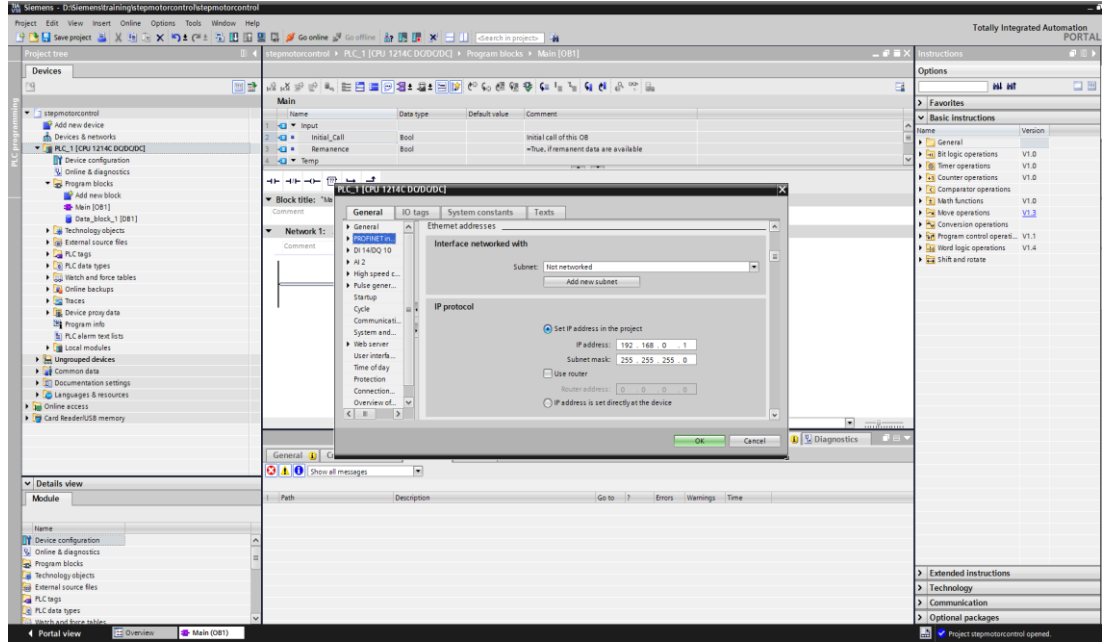
Bu bahsedilen adımlara ait görseller aşağıda sırasıyla sunulmuştur.



Şekil 3.26. PLC cihazının projeye eklenmesi

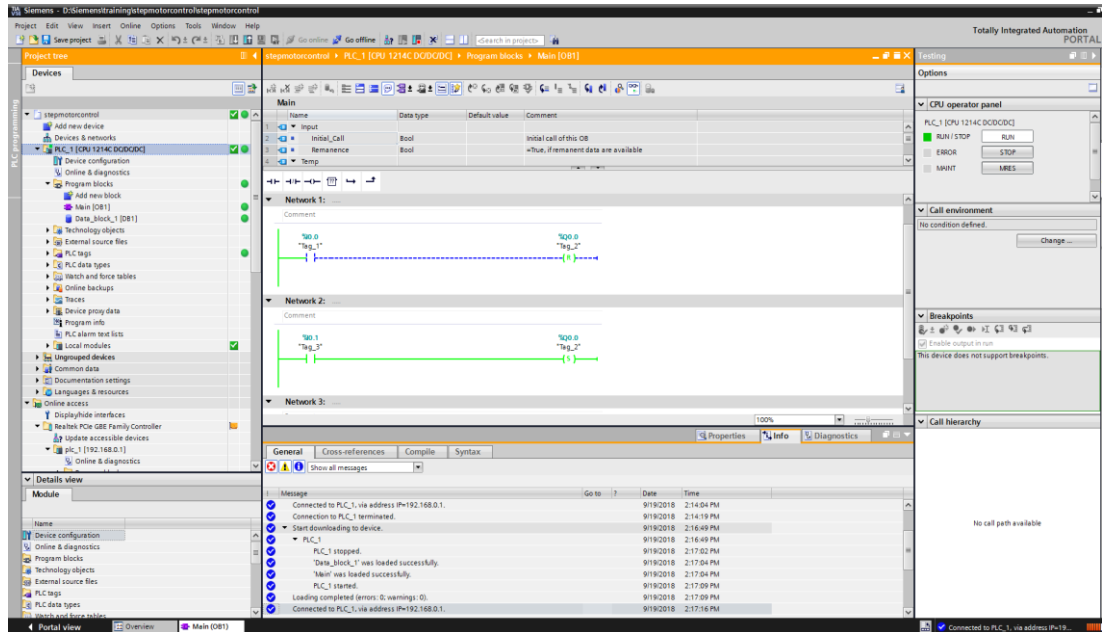
Burada sistem için kullandığımız PLC modelinin seçilerek projeye eklenmesi aşaması gerçekleştirilmektedir.

### 3. MATERYAL ve YÖNTEM



Şekil 3.27. PLC cihazına IP adresi atanması

Bu kısımda projeye eklenen PLC için IP adresi tanımlama işlemi gerçekleştirilmektedir.



Şekil 3.28. Çalıştırılan PLC kodunun gerçek zamanlı takibi

Bu son kısımda ise projeye eklenen PLC cihazına haberleşme yardımıyla bağlanıp, derlemesi gerçekleştirilen PLC kodunun gerçek zamanlı takibinin sağlanması işlemi gerçekleştirilmektedir.

#### 3.4.3. Visual Studio C# Yazılımı İle TIA PORTAL'ın Haberleşmesi

Visual Studio C# yazılımı, görüntü işleme aşamalarının gerçekleştirildiği yazılımdır. PLC cihazı ise kontrol kısmının gerçekleştiği donanımsal kısımdır. PLC cihazı ile haberleşme ise TIA PORTAL ile gerçekleştirilir. C# ile TIA yazılımlarının doğrudan haberleşmesi olanaksızdır. Bu problemi ortadan kaldırmak için C# diline ait birkaç kütüphane bulunmaktadır. Bu çalışmada ise S7.Net kütüphanesi kullanılmıştır ve bu kısımda bu kütüphane ile kodların derlenmesi ve TIA PORTAL ile haberleşmesi konusu ele alınacaktır.

S7.Net, sadece SIEMENS marka PLC'ler ile çalışan ve sadece Ethernet protokolü kullanılarak haberleşmesi gerçekleştirilen bir kütüphanedir. Bu kütüphane C# dili için yazılmıştır. Dolayısıyla herhangi bir ekstra dosyaya gerek kalmadan, kod kısmına sadece kütüphanenin eklenmesiyle projelerin derlenmesi gerçekleştirilebilir (Anonymous, 2016).

C# programlama dili ile yazılan;

- `Plc Plc = new Plc(CpuType.S71200, "192.168.0.1", 1, 0);`

komutuyla; PLC tanımlaması gerçekleştirilir. Burada PLC tipi, haberleşme IP adresi, Rack ve Slot parametreleri girilerek PLC tanımlanır.

- `Plc.Open();`

komutuyla; programın derlenme işlemi başladığı anda PLC'nin de haberleşmeye hazır hale gelmesi sağlanır.

- `Plc.Write("DB9.DBW0.0",axis_x);`

### 3. MATERYAL ve YÖNTEM

- Plc.Write("DB9.DBW4.0", axis\_y);

komutlarıyla; TIA ara yüzünde bilgilerin tutulması için oluşturulmuş veri bloğu olan DataBlock-9 'da, tespit edilen mantarın resimde bulunduğu konumun x eksenindeki konumu, 0.0 Input adresine Integer bir değer olarak ve y eksenindeki konumu ise 4.0 Input adresine yine Integer bir değer olarak gönderilir. Bu sistemde x ve y eksenlerinde hareket sağlayacak 2 motor bulunduğu için sadece bu eksenlere ait konum bilgileri gönderilmektedir. İhtiyaç duyulması durumunda 3. eksen veya diğer tür bilgilerin de TIA ara yüzüne gönderilmesi gerçekleştirilebilmektedir. Bu adımların gerçekleştiği C# kodu şekil 3.29'da görsel olarak verilmiştir ve EK-1 kısmına tüm kodlar eklenmiştir.

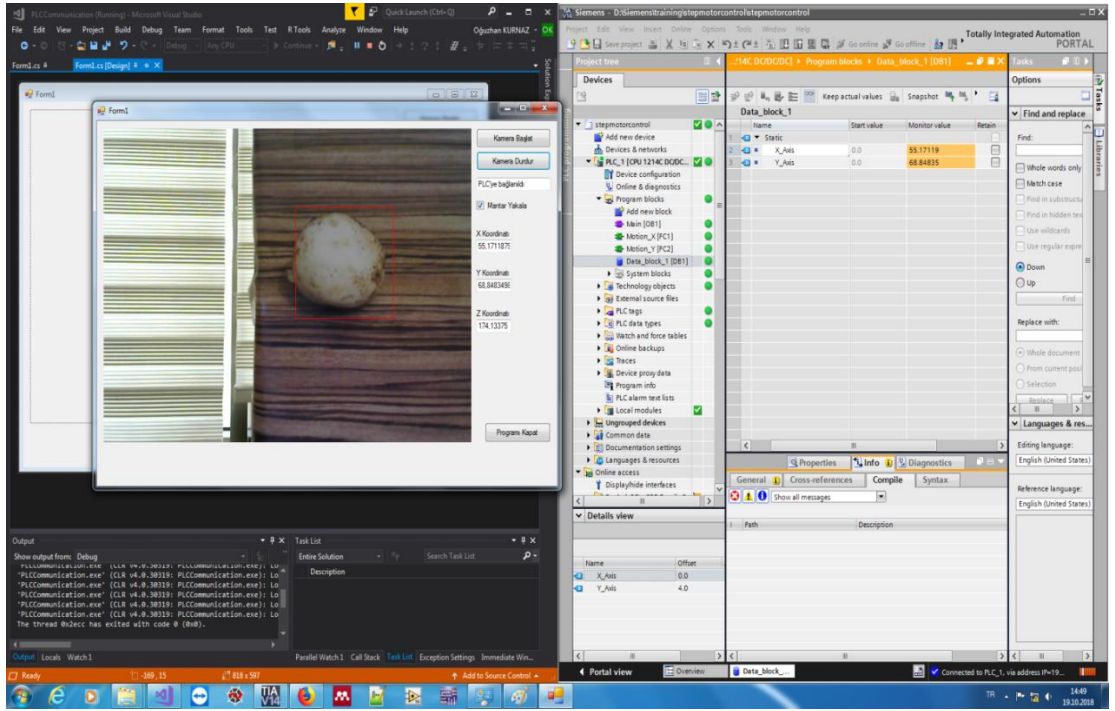
```
17
18 CascadeClassifier cascade;
19 VideoCapture capture;
20 public Mat mat = new Mat();
21 public Image<gr, byte> image;
22 double axis_x, axis_y, axis_z;
23 double focal_x=695.54, focal_y=695.54;
24 int face_length_x = 168;
25 int face_length_y = 168;
26 Plc Plc = new Plc(CpuType.S71200, "192.168.0.1", 0, 1);
27
28 public Form1()
29 {
30     InitializeComponent();
31     Control.CheckForIllegalCrossThreadCalls = false;
32 }
33
34
35 private void Form1_Load(object sender, EventArgs e)
36 {
37 }
38
39 private void FrmAcquisition(object sender, EventArgs e)
40 {
41     capture.Read(mat);
42     image = mat.ToImage<gr, byte>();
43
44     if (checkBox1.Checked)
45     {
46         var faces = cascade.DetectMultiScale(image, 1.1, 3);
47         foreach (var face in faces)
48         {
49             CvInvoke.Rectangle(image, face, new MCIScalar(0, 0, 255));
50             axis_z = (((focal_x + focal_y) / 2) * ((face_length_x + face_length_y) / 2)) / (face.Width + face.Height);
51             axis_x = (((axis_z) * ((face.Location.X * (face.Width / 2)) - (image.Width))) / focal_x);
52             axis_y = (((axis_z) * ((face.Location.Y * (face.Height / 2)) - (image.Height))) / focal_y);
53             textBox1.Text = axis_x.ToString();
54             textBox2.Text = axis_y.ToString();
55             textBox4.Text = axis_x.ToString();
56             Plc.Write("DB1.DBW0.0", Convert.ToDouble(textBox1.Text));
57             Plc.Write("DB1.DBW4.0", Convert.ToDouble(textBox2.Text));
58         }
59     }
60 }
```

Şekil 3.29. C# kodları

#### 3.4.4. PLC Cihazından Step Motor Sürücülerine Konum Bilgilerinin Gönderilmesi

Sistemin gerçek zamanlı test aşamasında örnek veri olarak kullanılan bir mantar resminde önce mantar tespit edilmiştir. Daha sonra tespit edilen mantarın konumu belirlenmiş ve Visual Studio C#.NET yazılımı ile gerçek zamanlı olarak haberleşen TIA Portal yazılımına elde edilen bu konum bilgileri anlık olarak gönderilmektedir. Test aşamasında gerçekleştirilmiş olan ürün tespiti ve konum bilgilerinin PLC'ye gönderilmesi aşaması şekil 3.30'da gösterilmiştir.





Şekil 3.30. Mantar Tespiti ve Konum Bilgilerinin PLC'ye Gönderimi

Şekil 3.30'da gösterilen resimde sistemin çalışır haldeki durumu gösterilmiştir. Öncelikle Visual Studio platformunda C# yazılım dili ile geliştirilmiş olan form ara yüzünün derleme işlemi gerçekleştirilir. Derleme işleminde sonra kod kısmıyla alakalı herhangi bir hata ile karşılaşılması durumunda form uygulaması çalıştırılır. Form uygulaması çalıştırılınca karşımıza şekil 3.30'da sol tarafta bulunan form ara yüzü gelecektir. Form ara yüzünde bulunan 'Kamerayı Başlat' butonuna basılması durumunda kameradan görüntü alınması sağlanacak ve bu da resim ekranında gösterilecektir. Sistemde bulunan PLC ile haberleşme görevi gerçekleştirilmesi durumunda ara yüze eklenmiş olan yazı kutucuğunda 'PLC'ye bağlanıldı' yazısı görülmektedir. Bu yazı ibaresi görüldükten sonra Visual Studio ile PLC haberleşmesi gerçekleştirilmiş ve veri alışverişi için sistem uygun hale getirilmiştir. Bundan sonraki aşama ise mantarın tespit edilmesi kısmıdır. Yazı kutucuğunun altında bulunan 'Mantar Yakala' seçeneğinin işaretlenmesi durumunda sistem mantar nesnesinin tespiti için hazır hale gelmiştir. Üstteki resimde de seçenek aktif haldedir ve tespit edilen mantar nesnesi kırmızı bir dörtgen içerisine alınmıştır. Dörtgen alanın merkezi noktasının XYZ düzlemindeki konumu 'X Koordinatı', 'Y



### 3. MATERYAL ve YÖNTEM

Koordinatı’ ve ‘Z Koordinatı’ kutucuklarında gösterilmiştir. Sistemde gerçekleştirilen görev X ve Y düzleminde hareket eden motorlara konum bilgileri gönderilmesi olduğu için X ve Y koordinat bilgileri PLC’ye gönderilmektedir. Resmin sağ tarafında ise bağlanmış olan PLC cihazı görülmektedir. Sol kısımda yer alan ‘Data Block’ kısmında tanımlanan değişkenler görülmektedir. Bu tanımlanan değişkenler ise yine sol tarafta görülen ‘Motion\_X’ ve ‘Motion\_Y’ fonksiyon bloklarında kullanılmaktadır ve step motorların kontrolü gerçekleştirilmektedir. Visual Studio form ara yüzünde elde edilen ‘X Koordinatı’ değeri PLC cihazına gönderilerek ‘X\_Axis’ değişkenine atanmaktadır. Aynı şekilde ‘Y Koordinatı’ değeri de PLC cihazına gönderilerek ‘Y\_Axis’ değişkenine atanmaktadır. Bu değerler eş zamanlı olarak hem form ara yüzünde hem de PLC cihazından görülmektedir. Sağ kısımda görülen turuncu renkli kısımlar ise PLC cihazına bağlandığını ve gelen verilerin anlık olarak görüntülendiğini göstermektedir. Konum ve hız bilgilerinin bulunduğu grafik ise şekil 3.31’de gösterilmiştir.



Şekil 3.31. X ve Y eksenlerine ait konum ve hız bilgilerinin grafiksel gösterimi

Şekil 3.31’de gösterilen grafikte kırmızı renkte gösterilen kısımda, form ara yüzünden gelen konum bilgilerinden X eksenine ait olan konum verisinin zamana bağlı olarak değişimi gösterilmiştir. Mavi renkli kısımda ise Y eksenine ait olan kısımlar gösterilmiştir. Yaklaşık 70 saniyelik bir periyotta gerçekleşen lokasyonel değişiklikler grafiksel olarak gösterilmiştir. Burada PLC cihazına gönderilen konum verileri cm cinsinden hesaplanmıştır. Pembe renkli kısımda ise X eksenine ait hız

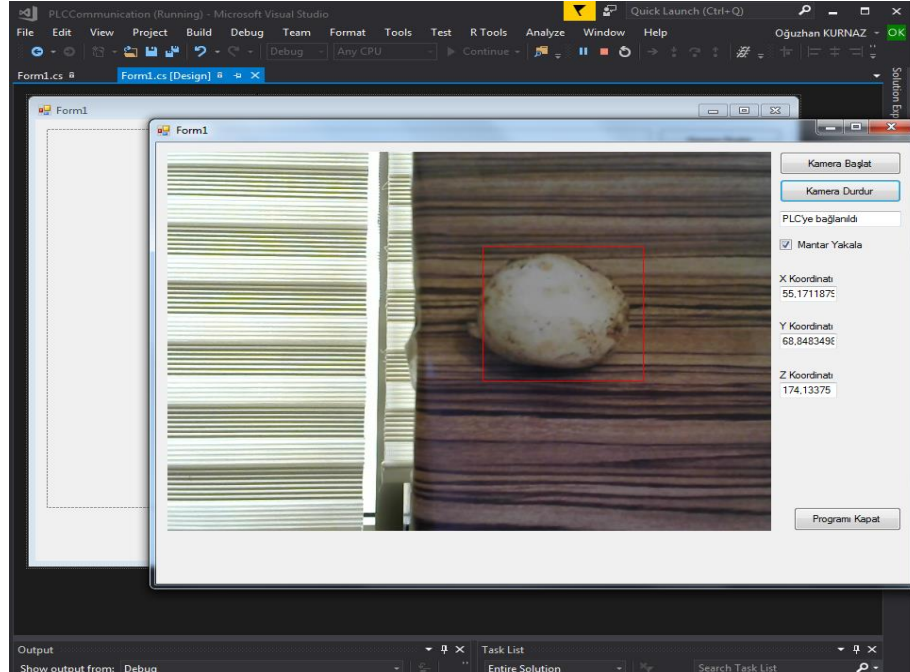
### 3. MATERYAL ve YÖNTEM

---

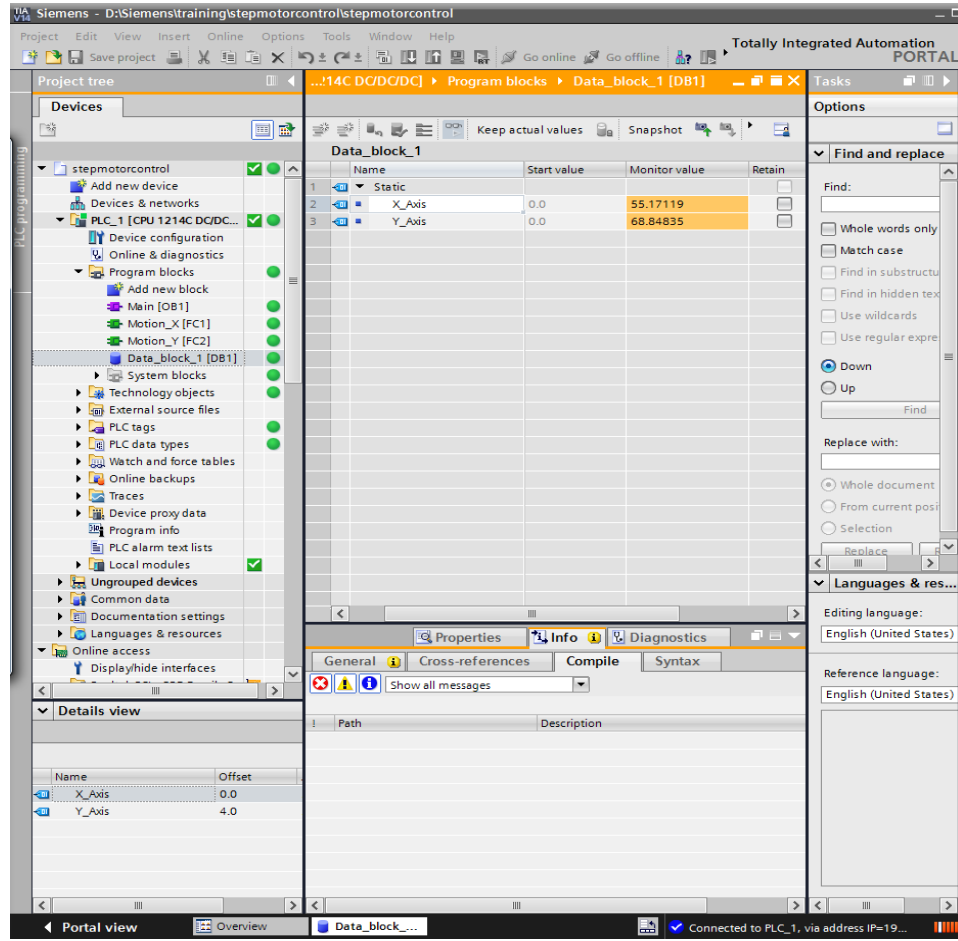
verisi gösterilmiştir. X ekseninde hareket sağlayan motor sabit hızda ve 100 mm/sn hızla hareket etmektedir ve bu motorun hızı sabit olduğu için grafikte gösterimi doğrusaldır. Aynı şekilde turkuaz renkle gösterilen kısımda ise Y eksenine ait hareketi sağlayan motorun hız verisi gösterilmiştir. Yine aynı şekilde bu ekseninde de motor 100 mm/sn hızla hareket etmektedir ve motorun hızı sabit olduğu için grafikte gösterimidoğrusaldır.

#### 4. ARAŞTIRMA BULGULARI ve TARTIŞMA

Robotik Mantar Toplama sisteminin yazılım kısmında öncelikle ürün tespiti gerçekleştirilir. Tespit edilen ürünün takibi sağlanır ve takip edilen ürünün konum bilgileri eş zamanlı olarak PLC cihazına gönderilir. Gönderilen bu konum bilgilerine göre motorların hareketi sağlanarak ürünün toplanma aşaması öncesi takip görevinin kolaylaştırılması sağlanmıştır. Şekil 4.1’de ürünün Visual Studio C#.NET platformu üzerinde hazırlanmış Form uygulamasında tespit edilme işlemi ve sonucu gösterilmiştir. Yine bu Form uygulamasında Visual Studio yazılımı ile TIA Portal yazılımının haberleştirildiğine dair bilgi verilen kısım bir yazı olarak görülmektedir. Bağlantı sağlanması durumunda “PLC’ye bağlandı” yazısı ekrandan görülmektedir. Son kısımda ise TIA Portal yazılımına haberleşme vasıtasıyla gönderilen bu konum bilgileri PLC cihazına anlık olarak bağlanmış durumdayken aktarılmaktadır. Bu konum bilgilerinin görüldüğü aşama Şekil 4.2’de gösterilmiştir. Gelen bu konum verileri, TIA Portal yazılımında X ve Y eksenleri için belirlenmiş olan fonksiyon bloklarına gönderilerek konum verilerinin step motor sürücülerine gönderilmesi sağlanır. Tüm bu işlemler sonucunda mantar tespiti ve mantarın konum bilgileri elde edilerek takibi işlemi gerçekleştirilir.



Şekil 4.1. Visual Studio C#.NET platformundaki Form uygulaması ve Mantar Tespiti



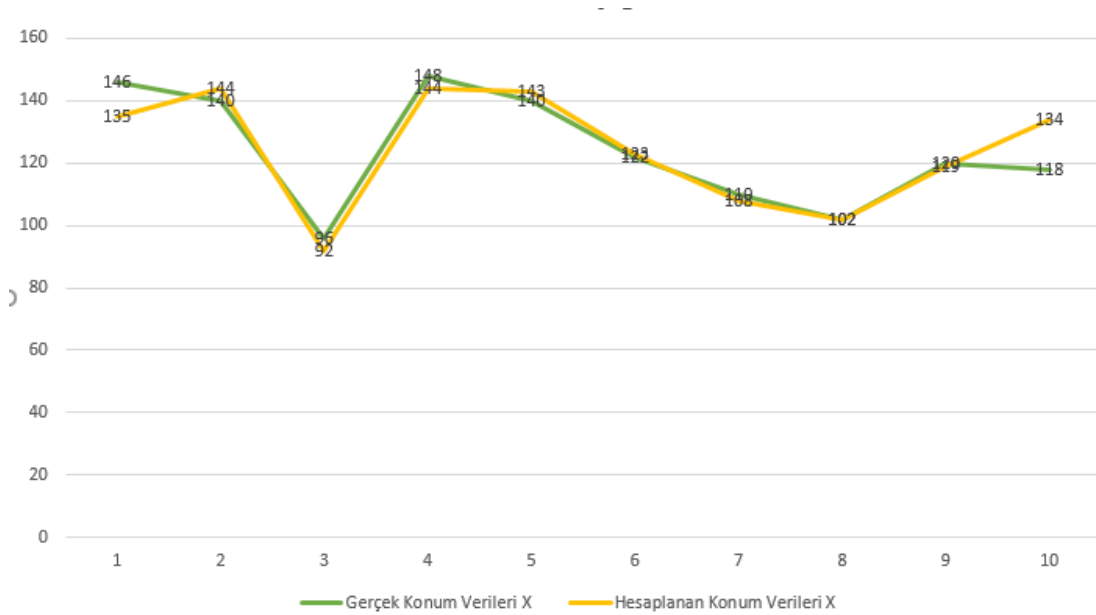
Şekil 4.2. TIA Portal yazılımında hazırlanmış olan Program Blokları ile tespit edilen mantarın konum bilgilerinin alınması

Benzer şekilde çekilmiş 10 farklı resimde, 10 farklı konumda tespit edilen mantarın tespit edildiği konum bilgileri ve global koordinat sisteminde bulunduğu noktalara ilişkin detaylar çizelge 4.1’de verilmiştir. Verilen konum bilgileri mm cinsinden hesaplanmıştır. X eksenindeki ve Y eksenindeki değişimler ise şekil 4.3 ve şekil 4.4’de gösterilmiştir.

#### 4. ARAŞTIRMA BULGULARI ve TARTIŞMA

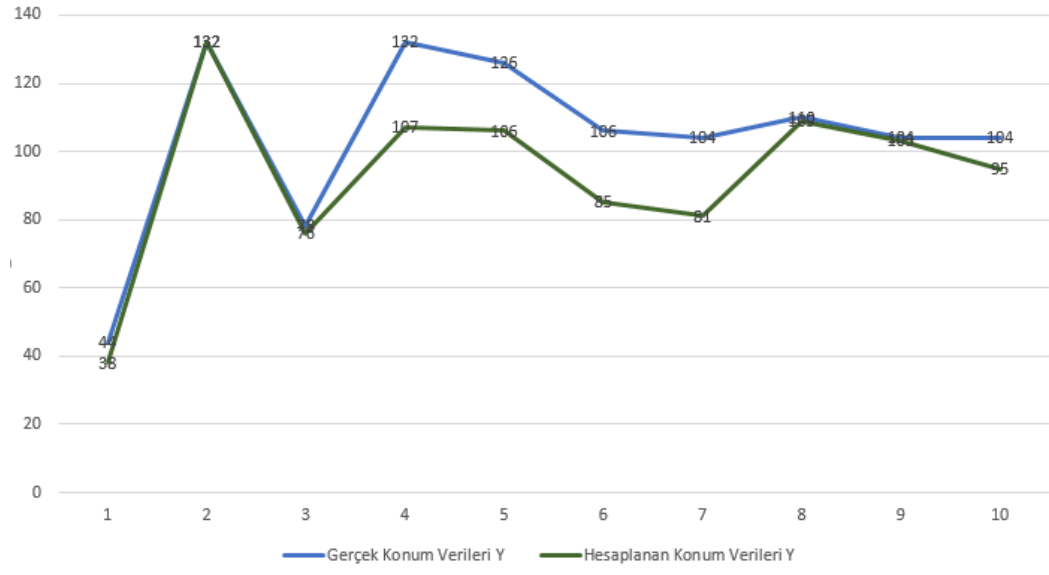
Ölçüm No	Gerçek Konum Verileri		Hesaplanan Konum Verileri		X Eksenindeki Hata Oranı	Y Eksenindeki Hata Oranı
	X	Y	X	Y		
1	146	44	135	38	7,53%	13,64%
2	140	132	144	132	2,86%	0,00%
3	96	78	92	76	4,17%	2,56%
4	148	132	144	107	2,70%	18,94%
5	140	126	143	106	2,14%	15,87%
6	122	106	123	85	0,82%	19,81%
7	110	104	108	81	1,82%	22,12%
8	102	110	102	109	0,00%	0,91%
9	120	104	119	103	0,83%	0,96%
10	118	104	134	95	13,56%	8,65%

Çizelge 4.1. X ve Y eksenlerindeki gerçek ve hesaplanan konum bilgileri ve hata oranları



Şekil 4.3. X eksenindeki gerçek konum ve hesaplanan konumların grafiksel gösterimi

#### 4. ARAŞTIRMA BULGULARI ve TARTIŞMA



Şekil 4.4. Y eksenindeki gerçek konum ve hesaplanan konumların grafiksel gösterimi

### 5. SONUÇ ve ÖNERİLER

Bu tez çalışmasında, Robotik Mantar Toplama adı verilen mantar toplama otomasyon sistemine dair ürün tespiti ve robotun ürün konumuna gönderilmesi konuları ele alınmış ve çözümler getirilmiştir. Robotik Mantar Toplama sisteminde ilk aşama olarak görüntü işleme teknikleri kullanılarak ürün tespiti yapılmıştır. İlk olarak mantar tespiti için Haar-Cascade sınıflandırıcısı kullanılmıştır. Yüksek doğruluk oranında tespit, hızlı çalışma ve hedeflenen herhangi bir nesneye eğitme yöntemiyle uyarlanabilir olması sebebiyle Haar-Cascade sınıflandırıcısı, birçok nesne tespit çalışmalarında kullanılan ve başarılı sonuçlar elde edilen bir sınıflandırıcı olmuştur. Bu çalışmada da tespiti istenen nesne mantar nesnesidir ve mantar nesnesinin eğitimi gerçekleştirilerek sınıflandırıcıya tanıtılması sağlanır. Daha sonra sınıflandırıcının kullanıldığı yazılım algoritmasının bu nesneyi tespit etmesi sağlanmıştır.

Bu sınıflandırıcının kullanılması için öncelikle sınıflandırıcı eğitiminin yapılma aşaması gerçekleştirilmiştir. Pozitif ve negatif resimlerin yazılıma tanıtılması vasıtasıyla sınıflandırıcı eğitimi gerçekleştirilir. Burada pozitif resimler bulunması istenen mantar nesnesine ait resimlerdir. Negatif resimler ise mantar nesnesinin bulunmadığı resimlerdir. Pozitif resimlerde mantarın olduğu kısımlar sisteme tanıtılarak eğitim gerçekleştirilmiş olur ve yazılım için gerekli olan .xml uzantılı dosya elde edilmiştir. Yazılıma tanıtılan bu dosya sayesinde sistemin çalışır durumunda mantar nesnesinin tespit etmesi sağlanmıştır.

Görüntü işleme teknikleri ile tespit edilen mantar nesnesinin görüntü ekranında bulunduğu konum kamera kalibrasyon yöntemi sayesinde tespit edilmiştir. Kalibrasyon için 10x7 boyutunda satranç tahtası resmi kullanılmıştır. Bu resmin üzerindeki köşe noktalarının tespiti sayesinde kameranın resmi aldığı an ve konumdaki kamera parametreleri elde edilmiştir. Bu parametreler sayesinde 2 boyutlu konumu bilinen bir noktanın 3 boyutlu uzaydaki konum bilgileri elde edilmiştir. Burada da öncelikle tespit edilen mantarın merkez noktasının piksel bazlı konumu belirlenmiştir. Kalibrasyon işleminden elde edilen kamera parametreleri de

kullanılarak 3 boyutlu global koordinat sistemindeki konumu tespit edilmiştir. Bu elde edilen konum, PLC'ye bilgi olarak gönderilmiştir.

Görüntü işleme teknikleri ve kamera kullanılarak gerçekleştirilen mantar tespiti probleminin çözümünden sonra PLC'ye gelen konum bilgileri PLC'de değerlendirme aşamalarından geçmiştir. Gerekli optimizasyonlar yapıldıktan sonra bu konumlar step motor sürücülerine gönderilerek motorların bu noktalara gönderilmesi sağlanmıştır. Sürücülere gelen bu konum bilgileri sürücüler tarafından gerekli dönüşümler yapılarak motorların bu konuma gönderilmesi sağlanmıştır.

Toplamda gerçekleştirilen 10 ölçümde elde edilen sonuçlara bakılacak olursa özellikle Y eksenindeki hassasiyetin daha da artırılması gerektiği sonucu çıkarılabilir. Sisteme yapılacak olan iyileştirmeler ile X ve Y eksenindeki hata oranlarının minimum seviyeye indirilmesi gerçekleştirilebilir.

Bu tez çalışmasında gerçekleştirilen en büyük kazanımlardan birisi, sıradan bir USB veya dahili kamera ile, görüntüdeki herhangi bir nesneye ait piksel bazlı konumdan, global koordinat sistemindeki gerçek konumuna ulaşılabilecek bir yöntem olan kalibrasyon tekniğinin kullanılmış olmasıdır. Microsoft Kinect, stereo kamera sistemi vb. sistemlerle elde edilmesi nispeten daha kolay olan bu konum bilgileri, kalibrasyon işlemi sayesinde standart bir kamera ile de gerçekleştirilmiştir.

Bu tez çalışmasının çıktısından elde edilen tecrübelerle, yapılacak olan başka bir projede bu mantar ürününün hasadının gerçekleştirilebilmesi için ürünün yetiştirme durumunun değerlendirilmesi yapılmalıdır. Buradan elde edilen sonuca göre ürün hasadının gerçekleştirilmesi veya ürünün yetiştirmemesi durumunda ekil halde kalmasının devamına ilişkin kamera tarafından kararlar alınacaktır. Birçok yöntemi bulunan sınıflandırma algoritmaları ile önce mantara dair özneliklerin çıkarılabilir. Akabinde bu çıkarılan öznelikler kullanılarak mantar ürününün yetişip yetişmediği değerlendirilmesi, gerçek zamanlı görüntü işleme yöntemleri ile tespit edilebilir ve ürün hasadı buna göre gerçekleştirilecektir.



## KAYNAKLAR

- An, A., 2008. Classification Methods., *Encyclopedia of Data Warehousing and Mining*, 196–201. doi:10.4018/978-1-59140-557-3.ch028
- Anonim., 2011. Step Motor Nedir. <http://www.elektrikrehberiniz.com/elektrik-motorlari/step-motor-nedir-543/>. Accessed 19 September 2018
- Anonim., 2012. *Stepper Motors EMMS-ST-57-S-SE-G2 DataSheet*.
- Anonim., 2013. Sınıflandırma. <http://bilgisayarkavramlari.sadievrenseker.com/2013/03/31/siniflandirma-classification/>. Accessed 11 September 2018
- Anonim., 2017a. Mantar yetiştiriciliği, 2–5. <https://istanbul.tarim.gov.tr/Belgeler/KutuMenu/Brosurler/Sebzecilik/mantar.pdf>
- Anonim., 2017b. Mantar Üretiminde Dikkat Edilmesi Gereken Hususlar. <https://www.kosgebhibe.net/2017/03/02/kultur-mantar-yetistirciligi-nasil-yapilir/>
- Anonymous., 2016. S7.Net documentation Create a PLC instance , connect and disconnect. <https://github.com/S7NetPlus/s7netplus/blob/master/Documentation/Documentation.pdf>
- Babenko, B., Ming-Hsuan Yang, & Belongie, S., 2011. Robust Object Tracking with Online Multiple Instance Learning., *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8), 1619–1632. doi:10.1109/TPAMI.2010.226
- Cyganek, B., & Gruszczynski, S., 2013. Eye recognition in near-infrared images for driver's drowsiness monitoring., *IEEE Intelligent Vehicles Symposium, Proceedings, (Iv)*, 397–402. doi:10.1109/IVS.2013.6629501
- Diaz, J. A. I., Ligeralde, M. I., Jose, J. A. C., & Bandala, A. A., 2017. Rust detection using image processing via Matlab., *TENCON 2017 - 2017 IEEE Region 10 Conference*, 1327–1331. doi:10.1109/TENCON.2017.8228063
- Eminoğlu, Y., 2015. *S7-1200 ile PLC Programlama* (3.Baskı.).
- Enzweiler, M., & Gavrilu, D. M., 2009. Monocular pedestrian detection: Survey and experiments., *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(12), 2179–2195. doi:10.1109/TPAMI.2008.260
- Feng, Q., Wang, X., Wang, G., & Li, Z., 2015. Design and test of tomatoes harvesting robot., *2015 IEEE International Conference on Information and Automation, ICIA 2015 - In conjunction with 2015 IEEE International Conference on Automation and Logistics*, (August), 949–952. doi:10.1109/ICInfA.2015.7279423
- Gonzalez, R., & Woods, R., 2002. *Digital image processing*, Prentice Hall. doi:10.1016/0734-189X(90)90171-Q
- Hiromoto, M., Sugano, H., & Miyamoto, R., 2009. Partially parallel architecture for AdaBoost-based detection with Haar-like features., *IEEE Transactions on Circuits and Systems for Video Technology*, 19(1), 41–52. doi:10.1109/TCSVT.2008.2009253
- Issac, A., Dutta, M. K., Sarkar, B., & Burget, R., 2016. An efficient image processing based method for gills segmentation from a digital fish image., *3rd International Conference on Signal Processing and Integrated Networks, SPIN 2016*, 645–649. doi:10.1109/SPIN.2016.7566776
- Jun, W., Zhou, Z., & Xiaodong, D. U., 2012. Design and Co-simulation for Tomato Harvesting Robots., *Proceedings of the 31st Chinese Control Conference*, 5105–5108.
- Kalal, Z., Mikolajczyk, K., & Matas, J., 2011. Tracking-Learning-Detection., *IEEE transactions on pattern analysis and machine intelligence*, 34(7), 1409–1422. doi:10.1109/TPAMI.2011.239
- Kanayama, A., & Tabusa, T., 2010. Auto-capturing system for facial images from video by Simulink model., *ISPACS 2010 - 2010 International Symposium on*

- Intelligent Signal Processing and Communication Systems, Proceedings, (ISPACS)*, 6–9. doi:10.1109/ISPACS.2010.5704635
- Kiang, M. Y., 2003. A comparative assessment of classification methods., *Decision Support Systems*, 35(4), 441–454. doi:10.1016/S0167-9236(02)00110-0
- Leaman, J., La, H. M., & Nguyen, L., 2017. Development of a smart wheelchair for people with disabilities., *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, 279–284. doi:10.1109/MFI.2016.7849501
- Li, B., Wang, M., & Li, L., 2010. A real-time pineapple matching system based on speeded-up robust features., *Proceedings - 2010 International Conference on Computational Intelligence and Security, CIS 2010*, 243–247. doi:10.1109/CIS.2010.59
- Mahmud, M. S. A., Abidin, M. S. Z., & Mohamed, Z., 2015. Development of an autonomous crop inspection mobile robot system., *2015 IEEE Student Conference on Research and Development, SCORed 2015*, 105–110. doi:10.1109/SCORed.2015.7449304
- Manolescu, D.-A., 1998. Feature Extraction, 1–5. <http://www.ncbi.nlm.nih.gov/pubmed/21689860>
- Masoudian, A., 2013. COMPUTER VISION ALGORITHMS FOR AN AUTOMATED HARVESTER (Thesis format: Monograph), (December).
- Mehta, S. S., & Burks, T. F., 2014. Vision-based control of robotic manipulator for citrus harvesting., *Computers and Electronics in Agriculture*, 102, 146–158. doi:10.1016/j.compag.2014.01.003
- Mehta, S. S., & Burks, T. F., 2016. Adaptive Visual Servo Control of Robotic Harvesting Systems., *IFAC-PapersOnLine*, 49(16), 287–292. doi:10.1016/j.ifacol.2016.10.053
- Olsson, C., 2013. Lecture 1 : The Pinhole Camera Model, 3(2), 2–5.
- Peilin, L., & Hong, L., 2017. An Automatic Sorting System for Sorting Metal Cylindrical Workpiece Based on Machine Vision and PLC Technology.
- Sari, Y. A., & Adinugroho, S., 2017. Tomato ripeness clustering using 6-means algorithm based on v-channel otsu segmentation., *5th International Symposium on Computational and Business Intelligence, ISCBI 2017*, 32–36. doi:10.1109/ISCBI.2017.8053539
- Satari, B. S., Rahman, N. A. A., & Abidin, Z. M. Z., 2015. Face recognition for security efficiency in managing and monitoring visitors of an organization., *Proceedings - 2014 International Symposium on Biometrics and Security Technologies, ISBAST 2014*, 95–101. doi:10.1109/ISBAST.2014.7013101
- Shaikh, A., Khaladkar, G., Jage, R., & Taili, T. P. J., 2013. Robotic Arm Movements Wirelessly Synchronized with Human Arm Movements Using Real Time Image Processing., *2013 Texas Instruments India Educators' Conference*, 277–284. doi:10.1109/TIIEC.2013.56
- Shao, L., Mu, Y., Liu, J., Dong, G., Liu, H., & Guo, P., 2014. The trunk of the image recognition based on BP neural network., *2014 IEEE International Conference on Mechatronics and Automation*, 53(9), 1800–1805. doi:10.1109/ICMA.2014.6885974
- SOLAK, S., & ALTINIŞIK, U., 2018. Görüntü İşleme Teknikleri ve Kümeleme Yöntemleri Kullanılarak Fındık Meyvesinin Tespit ve Sınıflandırılması., *SAÜ Fen Bilimleri Enstitüsü Dergisi*, 22(1), 1–1. doi:10.16984/aufenbilder.303850
- Soundariya, R. S., & Renuga, R., 2018. Eye movement based emotion recognition using electrooculography., *2017 Innovations in Power and Advanced Computing Technologies, i-PACT 2017*, 2017–Janua, 1–5. doi:10.1109/IPACT.2017.8245212
- Sun, T., Tang, S., Wang, J., & Zhang, W., 2013. A robust lane detection method for autonomous car-like robot., *Proceedings of the 2013 International Conference on Intelligent Control and Information Processing, ICICIP 2013*, 373–378. doi:10.1109/ICICIP.2013.6568100
- Tzokev, A., & Topalova, I., 2010. Image and data pre-processing model for real-time communication between dedicated PC and PLC neural network application in

- marble production., *Proceedings of the Mediterranean Electrotechnical Conference - MELECON*, 41–46. doi:10.1109/MELCON.2010.5476341
- Utaminigrum, F., Kurniawan, T. A., Fauzi, M. A., Maulana, R., Syauqy, D., Wihandika, R. C., et al., 2017. A laser-vision based obstacle detection and distance estimation for smart wheelchair navigation., *2016 IEEE International Conference on Signal and Image Processing, ICSIP 2016*, 123–127. doi:10.1109/SIPROCESS.2016.7888236
- Viola, P., & Jones, M., 2001. Rapid object detection using a boosted cascade of simple features., *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, 1*, I-511-I-518. doi:10.1109/CVPR.2001.990517
- Von Söhsten, D., & Murilo, S., 2014. Multiple face recognition in real-time using cloud computing, Emgu CV and Windows Azure., *International Conference on Intelligent Systems Design and Applications, ISDA*, 137–140. doi:10.1109/ISDA.2013.6920723
- Wang, Y., 2011. Digital image processing methods based on Visual C#.NET., *2011 2nd International Conference on Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC)*, 1073–1076. doi:10.1109/AIMSEC.2011.6010731
- Wang, Y. M., Li, Y., & Zheng, J. B., 2010. A camera calibration technique based on OpenCV., *3rd International Conference on Information Sciences and Interaction Sciences (ICIS 2010), Chengdu, China*, 403–406. doi:10.1109/ICICIS.2010.5534797
- Wood, E., & Bulling, A., 2014. Eytat: Model-based gaze estimation on unmodified tablet computers., *Proceedings of the Symposium on Eye Tracking ...*, 3–6. doi:10.1145/2578153.2578185
- Wren, C. R., Azarbajani, A., Darrell, T., & Pentland, A. P., 1997. Pfunder: real-time tracking of the human body., *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7), 780–785. doi:10.1109/34.598236
- Yadav, D. K., Sharma, L., & Bharti, S. K., 2014. Moving Object Detection in Real-Time Visual Surveillance using Background Subtraction Technique, 3(1), 79–84.
- Yakar, M., 2009. SAYISAL GÖRÜNTÜ ANALİZ İŞLEMİNDE KAMERA KALİBRASYON DETERMINATION OF CAMERA CALIBRATION PARAMETERS AT, 13–15.
- Zhao, Y., Gong, L., Huang, Y., & Liu, C., 2016. A review of key techniques of vision-based control for harvesting robot., *Computers and Electronics in Agriculture*, 127, 311–323. doi:10.1016/j.compag.2016.06.022

## EKLER

### EK-1

```
using System;
using System.IO;
using System.Threading;
using System.Windows.Forms;
using Emgu.CV;
using Emgu.CV.Structure;
using S7.Net;

namespace PLCCommunication
{
    public partial class Form1 : Form
    {
        CascadeClassifier cascade;
        VideoCapture capture;
        public Mat mat = new Mat();
        public Image<Bgr, byte> image;
        double axis_x, axis_y, axis_z;
        double foclen_x=697.53, foclen_y=695.54;
        int face_length_x = 60;
        int face_length_y = 60;
        Plc Plc = new Plc(CpuType.S71200, "192.168.0.1", 0, 1);

        public Form1()
        {
            InitializeComponent();
            Control.CheckForIllegalCrossThreadCalls = false;
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            capture = new VideoCapture(0);
            capture.ImageGrabbed += FrameAcquisition;
            cascade = new
CascadeClassifier("C:/Users/oguzhan.kurnaz/source/repos/PLCCommunication/PLCCo
mmunication/data/mantaryeni1.xml");

            Plc.Open();
            if (Plc.IsConnected)
            {
                textBox3.Text = "PLC'ye bağlanıldı";
            }
        }

        private void FrameAcquisition(object sender, EventArgs e)
        {
            capture.Read(mat);
            image = mat.ToImage<Bgr, byte>();

            if (checkBox1.Checked)
            {
                var faces = cascade.DetectMultiScale(image, 1.1, 3);
            }
        }
    }
}
```

```

        foreach (var face in faces)
        {
            CvInvoke.Rectangle(image, face, new MCvScalar(0, 0, 255));
            axis_z = (((foclen_x + foclen_y) / 2) * ((face_length_x +
face_length_y) / 2)) / (face.Width + face.Height)/2;
            axis_x = ((-axis_z) * ((face.Location.X+(face.Width / 2))
- (image.Width))) / foclen_x;
            axis_y = -(((axis_z) * ((face.Location.Y+(face.Height /
2)) - (image.Height))) / foclen_y);
            textBox1.Text = axis_x.ToString();
            textBox2.Text = axis_y.ToString();
            textBox4.Text = axis_z.ToString();
            Plc.Write("DB1.DBW0.0", Convert.ToDouble(textBox1.Text));
            Plc.Write("DB1.DBW4.0", Convert.ToDouble(textBox2.Text));

        }
    }
    pictureBox1.Image = image.ToBitmap();
}

private void button1_Click(object sender, EventArgs e)
{
    capture.Start();
}

private void button3_Click(object sender, EventArgs e)
{
    this.Close();
}

private void button2_Click(object sender, EventArgs e)
{
    capture.Stop();
}

}
}

```

## ÖZGEÇMİŞ

### **Kişisel Bilgiler**

Adı-Soyadı : Oğuzhan KURNAZ  
Uyruğu : T.C.  
Doğum Tarihi ve Yeri : 22/05/1993-ERZURUM  
Medeni Hali : Bekar  
Telefon : +90 536 391 70 26  
e-mail : oguzhan.kurnaz@btu.edu.tr

### **Eğitim**

<b>Derece</b>	<b>Üniversite</b>	<b>Mezuniyet Yılı</b>
Yüksek Lisans	Erzurum Teknik Üniversitesi	2019
Lisans	Erciyes Üniversitesi	2016
Lise	Mecidiye Anadolu Lisesi	2011

### **Çalıştığı Kurum/Kurumlar ve Yıl**

Bursa Teknik Üniversitesi Araştırma Görevlisi Haziran 2018-..