ESNEK ÜRETİM SİSTEMİNDE GÖRÜNTÜ İŞLEME TEKNİĞİ İLE ROBOTİK EĞİTİM MODELİ GELİŞTİRİLMESİ

Mehmet TEKEREK

DOKTORA TEZİ
ENDÜSTRİYEL TEKNOLOJİ EĞİTİMİ

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

EKİM 2006
ANKARA

Mehmet TEKEREK tarafından hazırlanan ESNEK ÜRETİM SİSTEMİNDE GÖRÜNTÜ İŞLEME TEKNİĞİ İLE ROBOTİK EĞİTİM MODELİ GELİŞTİRİLMESİ adlı bu tezin Doktora tezi olarak uygun olduğunu onaylarım.

Prof. Dr. Mehmet EROĞLU
Tez Yöneticisi

Bu çalışma, jürimiz tarafından oy birliği / oy çokluğu ile Endüstriyel Teknoloji Eğitimi Anabilim Dalında Doktora tezi olarak kabul edilmiştir.

Başkan : Prof. Dr. H. Güçlü YAVUZCAN

Üye : Prof. Dr. Mehmet EROĞLU

Üye : Prof. Dr. Mahmut İZCİLER

Üye : Prof. Dr. Selahittin ÖĞÜLMÜŞ

Üye : Prof. Dr. Güngör BAL

Tarih : 25.10.2006

Bu tez, Gazi Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına uygundur.

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada orijinal olmayan her türlü kaynağa eksiksiz atıf yapıldığını bildiririm.

Mehmet TEKEREK

ESNEK ÜRETİM SİSTEMİNDE
GÖRÜNTÜ İŞLEME TEKNİĞİ İLE
ROBOTİK EĞİTİM MODELİ GELİŞTİRİLMESİ
(Doktora Tezi)

Mehmet TEKEREK

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
Ekim 2006

ÖZET

Modern toplum yaşamına belirgin bir şekilde giren robot sistemler ve benzeri mekatronik yapıların üretim sistemlerindeki etkin kullanımları, günlük yaşamda ki yerleri ve bu doğrultuda yeni alanların oluşumunun teknolojik açıdan ele alındığı bu çalışmada, esnek üretim sistemlerinde dijital görüntü işleme tekniklerinden renk tanıma yöntemi aracılığıyla deneysel bir eğitim robotu olan XR3 ve yan ekipmanlarının kinematik kısıtlar altında kontrolü sağlanarak bir robotik eğitim modeli geliştirilmesi amaçlanmıştır.

Renk bilgisinin kullanıldığı görüntü işleme alanında renkli nesnelerin tanınarak esnek üretim sisteminde istenilebilecek işlemlerin altı eksenli robot manipülatör aracılığıyla gerçekleştirilmesi üzerinde durulmuştur. Altı serbestlik dereceli bir deneysel robot manipülatör ve esnek üretim sistemi mantığına sahip bir konveyör bant, sistemin mekanik aksamını oluşturmaktadır. Gerçek uzayda hareketli nesnelerin yerlerinin belirlenmesi, kamera ile elde edilen görüntü üzerinde, çalışma bünyesinde geliştirilen kalibrasyon yöntemi kullanılarak sağlıklı renk bilgisinin işlenmesi sureti ile sağlanmıştır.

Çalışma kapsamında belirlenen amaçların tamamını bünyesinde barındıran ve robot sistem üzerinde eğitsel amaçlı deneylerin yapılabilmesine izin veren görsel nitelikli ve kullanıcı etkileşimli bütünleşik gaziXR3.exe olarak adlandırılan bir

yazlım da geliştirilmiş ve bu tez çalışması ile birlikte sunulmuştur. Yazılım bünyesinde robot kontrolünün sağlanmasının yanında bu yapı matematiksel olarakta eğitsel materyal olması yönüyle sunulmuştur.

Aynı zamanda her bir eksen hareketi görsel olarak tanımlanmış ve robotun hareketiyle eş zamanlı olarak simüle edilmiştir. Farklı seviyelere yönelebilmek amacıyla Türkçe robot programlama dili geliştirilerek yazılım bünyesine katılmıştır.

Bilim Kodu : **705**
Anahtar Kelimeler : **Bilgisayarlı görüntüleme, Görüntü işleme, Robotik Eğitimi, Robot Denetimi, Esnek Üretim Sistemi**
Sayfa Adedi : **165**
Tez Yöneticisi : **Prof. Dr. Mehmet EROĞLU**

DEVELOPING A ROBOTIC EDUCATION MODEL
BY USING DIJITAL IMAGE PROCESSING
ON FLEXIBLE MANUFACTURING SYSTEM
(Ph.D. Thesis)

Mehmet TEKEREK

GAZİ UNIVERSITY
INSTITUTE OF SCIENCE AND TECHNOLOGY
October 2006

ABSTRACT

The effective usage of robot systems that incoming modern society life markedly and similar mechatronic constructions in production systems, their roles within daily life and in this direction the formation of new areas taken technological point of view, in this study it is aimed to develop a robotic training model taking control of XR3 which is an experimental educational robot and its equipments under kinematics constrains by means of the colour recognition method a technique of digital image processing in flexible manufacturing systems.

It is stated that processes can be required within flexible manufacturing systems recognizing colourful objects in image process area used colour information performed by means of six axis an experimental robot manipulator and a conveyor band with logic of flexible manufacturing system constitute mechanical structure of the system.

To define moving objects in real space is provided with on image taken with camera using calibration method developed during the process of this study and processing right colour code information.

It is developed a software called gazixr3 includes all aims defined within the context of study and visual quality that allows making educational experiment

on the robot system and it is submitted with this thesis. This structure is submitted educational material as mathematical side besides providing robot's control within software. In addition to each axis transaction is defined as visual and synchronously. To reach different levels a Turkish robot programming language is developed and added the software.

Science Code : **705**
Key Words : **Computer Vision, Image Processing, Robotic Education, Robot Control, Flexible Manufacturing Systems**
Page Number : **165**
Adviser : **Prof. Dr. Mehmet EROĞLU**

TEŞEKKÜR

Çalışmalarım boyunca değerli yardım ve katkılarıyla beni yönlendiren Danışman Hocam Prof. Dr. Mehmet EROĞLU'na, laboratuar çalışmaları için kaynaklarını sunan Hocam Prof. Dr. Serpil EROL'a ve yazılım geliştirme konusunda katkılarından dolayı Makine Yüksek Mühendisi Utku AVGAN'a teşekkürü bir borç bilirim.

İÇİNDEKİLER

Sayfa

ÖZET i

ABSTRACT iii

TEŞEKKÜR v

İÇİNDEKİLER vi

ÇİZELGELERİN LİSTESİ xi

ŞEKİLLERİN LİSTESİ xii

RESİMLERİN LİSTESİ xv

SİMGELER VE KISALTMALAR xvii

1. GİRİŞ 1

2. BİLGİSAYARLI GÖRME 12

2.1. Bilgisayarlı Görme Temelleri 12

2.2. Bilgisayarlı Görme Uygulamaları 14

2.2.1. Robotlar 14

2.2.2. Görüntü sıkıştırma ve iyileştirme 15

2.2.3. Tıbbi uygulamalar 15

2.2.4. Görüntü anlama 15

2.3. Tanıma Yöntem Bilimi (Metodolojisi) 16

2.3.1. Düzenli duruma getirme 16

2.3.2. Sınıflandırma 16

2.3.3. Gruplara ayırma 17

2.3.4. Çıkarım 17

2.3.5. Eşleştirme 18

Sayfa

2.4. Görüntü Dijitalleştirme 18
2.4.1. Dijitalleştiricinin unsurları 18
2.4.2. Görüntü dijitalleştiricinin özellikleri 19
2.4.3. Görüntü dijitalleştirici tipleri 20
2.4.4. Dijitalleştirme elemanları 21
2.5. Dijital Görüntü İşleme 22
2.6. Dijital Görüntü İşlemenin Kökenleri (Kaynağı) 24
3. GÖRÜNTÜ İŞLEME 30
3.1. Renkli Görüntü İşleme 30
3.1.1. Renkli görme 31
3.1.2. Üç renk görüntüleme 32
3.1.3. Renk belirtme 33
3.1.4. Renk koordinat çevrimi 36
3.2. Renk Dengesi 40
3.2.1. Zıtlık ve iyileştirme 41
3.3. Renk Temelleri 41
3.4. Renk Modelleri 44
3.4.1. RGB renk modeli 45
3.4.2. CMY ve CMYK renk modelleri 48
3.4.3. HSI renk modeli 49
3.5. Yalancı Renk Görüntü İşleme 50
3.5.1. Yoğunluk dilimleme 50
4. ESNEK ÜRETİM SİSTEMLERİ 53

Sayfa

4.1. Esnek üretim sistemlerinin tanımı 54
4.1.1. Esnek üretim sistemlerinin genel özellikleri 55
4.2. Esnek Üretim Sistemlerinin Alt Sistemleri 55
4.2.1. Esnek üretim eleman ve teknolojileri 56
4.2.2. Bilgisayar tümleşik üretim ve esnek üretim sistemleri 56
4.2.3. Malzeme taşıma / stoklama eleman ve teknolojileri 58
4.2.4. Malzeme işleme eleman ve teknolojileri 59
4.3. Esnek Üretim Sistemi Tasarım Problemleri 60
4.3.1. Başlangıç kararları 61
4.3.2. Tamamlama kararları 61
4.4. Matris Tabanlı EÜS Tasarım Modeli 62
4.4.1. Örnek matris-tabanlı EÜS tasarım modeli 64
4.5. Esnek Üretim Sisteminde Makine Görmesi Alanları 66
5. SİSTEM ELEMANLARI VE PROGRAMLAMA ESASLARI 67
5.1. Donanım 67
5.2. Genel Arayüz Komutları ve Elektriksel Bağlantılar 72
5.2.1. Veri formatı 72
5.2.2. Veri yolu açma 73
5.2.3. Bilgisayar ile bağlantı kurma 73
5.2.4. İletişim 73
5.3. Programlama Esasları 74
5.3.1. Motorları hareket ettirme 74
5.3.2. Durum bilgisi alma 75

Sayfa

5.3.3. Durdurma ve sorgulama 80

5.3.4. Donanım başlangıç pozisyonu (Hardhome) 86

5.3.5. INPUT/ OUTPUT kullanımı 86

5.4. Kinematik İnceleme 92

5.4.1. XR3 için kinematik inceleme 92

5.4.2. Eksen ilişkilendirme 98

6. SİSTEMİN ÇALIŞMASI VE BULGULAR 101

6.1. Sistemin Tanımlanması 101

6.2. Bilgisayar Programının Çalışması 101

6.2.1. Mark3 iletişimi 102

6.2.2. Görüntü yakalama 104

6.2.3. Görüntü işleme süreci 105

6.2.4. Renk ayrımları 110

6.2.5. Renk filtreleri 125

6.2.6. Renk tanıma 126

6.2.7. Robot hareket uzayının belirlenmesi 128

6.3. Sistem İş Akışı 130

6.3.1. Kamera iletişim iş parçacığı 130

6.3.2. Görüntü yakalama ve hazırlama iş parçacığı 131

6.3.3. Robot denetleyicisi iletişimi iş parçacığı 132

6.4. Sistem İş Döngüsü 133

6.5. Türkçe Robot Programlama Dili 143

6.5.1. Komut seti 144

Sayfa

6.6. XR3 Robotun Anlık Konumunun İki Boyutlu Görüntü İle İfade Edilmesi .. 148

6.6.1. Üç boyutlu görüntünün iki boyutlu düzlemde iz düşümünün gösterilmesi 150

6.6.2. Ortak düzlemdeki eksen gruplarının ayrı ayrı gösterilmesi 150

7. TARTIŞMA VE SONUÇLAR 154

KAYNAKLAR 159

ÖZGEÇMİŞ 164

ÇİZELGELERİN LİSTESİ

Çizelge **Sayfa**

Çizelge 3.1. Her güvenli RGB elemanının geçerli değerleri 46

Çizelge 5.1. XR3 Fiziksel özellikler 67

Çizelge 5.2. Her bir eksendeki çözünürlükler 68

Çizelge 5.3. Motor dişli oranı ve durma azalması 68

Çizelge 5.4. Enkoder adımları/Eklem dereceleri 68

Çizelge 5.5. Her eksendeki maksimum hız 69

Çizelge 5.6. Mark III denetleyici switch okuma ve komut sistemi 80

Çizelge 5.7. Mark III denetleyici switch okuma ve komut sistemi 80

Çizelge 5.8. Bir switch kapalı iken diğer switchlerin durumu 81

Çizelge 5.9. Her switch için uygun tam sarı değerleri 84

Çizelge 5.10. Girişlerin bit değerleri 86

Çizelge 5.11. Beş eksenli mafsallı robot'un kinematik parametreleri 94

ŞEKİLLERİN LİSTESİ

Şekil **Sayfa**

Şekil 2.1. Renk skalası 23

Şekil 2.2. Elektromanyetik spektrum 24

Şekil 3.1. Üstten insan gözü çizgisel kesiti 32

Şekil 3.2. İnsan fotoreseptör hücrelerinin hassaslık spektrumu 33

Şekil 3.3. Dikdörtgen renk aralığı 34

Şekil 3.4. Renk çemberi 35

Şekil 3.5. Silindirik renk uzayı 36

Şekil 3.6. Renk skalası 37

Şekil 3.7. RGB küpü dönüşümü 37

Şekil 3.8. X,Y düzlemindeki renk uzayı 38

Şekil 3.9. RGB 24 bit renk küpü 39

Şekil 3.10. Renk spektrumu 41

Şekil 3.11. Elektromanyetik spektrumun görünür kısmının karşılaştırması 42

Şekil 3.12. Azalan RGB değerleriyle düzenlenmiş 216 güvenli renk 46

Şekil 3.13. Işık ve Işıklılığın birincil ve ikincil renkleri 48

Şekil 3.14. Yoğunluk dilimleme tekniğinin geometrik açıklaması 50

Şekil 3.15. (a) Gri seviye, (b) Kodlamanın sonucu 51

Şekil 4.1. Casa çemberi 57

Şekil 4.2. Deneysel EÜS 64

Şekil 5.1. XR-3 Boyutları 67

Şekil 5.2. Bilgisayar ve Mark III denetleyici arasında kullanılan veri yolu 71

Şekil 5.3. QUESTION komutu uygulama örneği 74

Şekil **Sayfa**

Şekil 5.4. 95 adım üzerinde adım sayısı uygulama tekniği 76

Şekil 5.5. Sürekli hareketi optimum zamanda gerçekleştirmeye yönelik döngü 78

Şekil 5.6. STOP ve SORGU komutları 82

Şekil 5.7. G eksen motorunu, örneğin bir taşıma bandı, INPUT #2 açılana kadar (YÜKSEK TTL düzeyi) hareket ettirir 87

Şekil 5.8. AUX veri yolu #2'nin açılıp kapanması 89

Şekil 5.9. AUX veri yolu #2'nin uzun program uygulamasında açılıp kapanması 90

Şekil 5.10. Beş eksenli mafsallı robot'un link koordinat diyagramı (XR3) 92

Şekil 5.11. Denavit- Hartenberg algoritması 93

Şekil 6.1. Sınıflandırma sistemi blok şeması 100

Şekil 6.2. Robot iletişim arayüzü 101

Şekil 6.3. Görüntü kaynağı seçimi 104

Şekil 6.4. Görüntü işleme, görüntü yakalama arayüzü 105

Şekil 6.5. Sürekli görüntü yakalamada renk filtresi uygulanması 106

Şekil 6.6. Kırmızı rengin sürekli görüntü yakalama modunda algılanması 114

Şekil 6.7. Mavi rengin sürekli görüntü yakalama modunda algılanması 115

Şekil 6.8. Sarı rengin sürekli görüntü yakalama modunda algılanması 116

Şekil 6.9. Siyah- beyaz renk dönüşümü 118

Şekil 6.10. (a) Kameradan alınan orijinal görüntü (b) Aynı görüntünün monokromatik renk filtresi uygulanmış şekli 123

Şekil 6.11. HSL aralıklarının belirlenmesi 125

Şekil 6.12. Köşe kalibrasyonu 128

Şekil 6.13. Esnek köşe kalibrasyonu oluşturmaya yönelik olarak hazırlanan kullanıcı etkileşimli arayüz 129

Şekil **Sayfa**

Şekil 6.14. Kamera iletişim ve görüntü yakalama arayüzü 130

Şekil 6.15. Sürekli hareketi optimum zamanda gerçekleştirmeye yönelik döngü ... 132

Şekil 6.16. Ters kinematik denklem çözümünün yapılması ve her bir eksen için radyan cinsinden sonuçların verilmesi 135

Şekil 6.17. Düz kinematik denklem çözümünün yapılması ve kol matrisinin oluşturulması 138

Şekil 6.18. Üzerinde örnek program yer alan Türkçe Robot Programlama arayüzü 143

Şekil 6.19. Kullanıcı etkileşimli robot simülasyonu (home pozisyon) 147

Şekil 6.20. Robot simülasyonunda artı ve eksi yönde eksen hareketlerinin gösterimi..... 148

Şekil 6.21. Eksen gruplarının ayrı ayrı olarak iki boyutlu düzlemde gösterimi 150

RESİMLERİN LİSTESİ

Resim **Sayfa**

Resim 1.1. Ebul-iz İsmail bin ar-Razzaz el-Cezeri'ye ait bir robot tasarımı 5

Resim 1.2. İnsansı Robot ASIMO'nun gelişim sürecindeki değişimi 8

Resim 2.1. İlk dijital görüntülerden biri 25

Resim 2.2. 1922 de yapılmış bir dijital resim 25

Resim 2.3. 1929 da 15 farklı gri tonun kullanıldığı dijital resim 26

Resim 2.4. Ay'ın Ranger7 tarafından 31 Haziran 1964 saat 9.09 da çekilen fotoğrafı 27

Resim 2.5. X ışını görüntüleme örnekleri. a) Göğüs X ışını. b) Atardamar anjiyosu c) Kafa tomografisi 28

Resim 5.1. XR3 Rhino robot kol 66

Resim 5.2. Rhino Mark III denetleyici 69

Resim 5.3. Rhino konveyör bant 70

Resim 5.4. Hareketli taban 70

Resim 5.5. Beş eksenli mafsallı robot (RHINO XR3) 91

Resim 6.1. XR3'ün "home pozisyonda" konumu 102

Resim 6.2. Sınıflandırma sisteminin genel görünümü (home pozisyon) 133

Resim 6.3. Kırmızı için renk tanımlamasının yapılarak konveyörün durdurulması 134

Resim 6.4. Tanıma işlemi tamamlanan nesnenin konveyör durdurulduktan sonra alınarak taşınması (a) Yakalama (b) Alma (c) Taşıma 140

Resim 6.5. Tanımlanan renkli nesnenin önceden belirlenen alana bırakılması 141

Resim 6.6. Renklerine göre ayrıştırma işlemi 142

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış bazı simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler	**Açıklama**
Dec	10'luk sayı sistem
Hex	16' lıksayı sistemi
mA	mili amper
mm	milimetre
nm	nanometre
V	Volt

Kısaltmalar	**Açıklama**
MÖ	Milattan Önce
R.U.R.	Rossumun Evrensel Robotları (Rossum's Universal Robots)
ENIAC	Elektronik Nümerik Tümleşik Bilgisayar (Electronic Numerical Integrator and Computer)
MIT	Massachusetts Teknoloji Enstitüsü (Massachusetts Institute of Technology)
NASA	AdministrationAmerikan Ulusal Hava ve Uzay İdaresi (National Aeronautics and Space)
CCD	Çift Denetimli Cihaz (Charge-Coupled Device)
RGB	Kırmızı-Yeşil-Mavi (Red-Green-Blue)

Kısaltmalar	Açıklama
CPU	Merkezi İşlem Birimi (Central Processor Unit)
CT	Bilgisayarlı Tomografi
3D	3 Boyutlu
HSI	Ton-Doygunluk-Yoğunluk (Hue-Saturation-Intensity)
CIE	Uluslar arası Renk Standardizasyonu Komitesi (Commission Internationale de l'Éclairage)
CMY	Türkuaz- Pembe Sarı (Cyan Magenta Yellow)
CASA / SME	Bilgisayar ve Otomasyon Sistemleri Derneği / Üretim Mühendisleri Birliği (Computer and Automated Systems Association of the Society of Manufacturing Engineers)
EÜS/ FMS	Esnek Üretim Sistemleri/ Flexible Manufacturing Systems
EURON	Avrupa Robot Ağı (European Robotic Network)
USB	Evrensel Veri Yolu (Universal Serial Buffer)
CNC	Bilgisayarlı Sayısal Kontrol (Computer Aided Numeric Control)
CIM	Bilgisayar Bütünleşik Üretim (Computer Integrating Manufacturing)
LCD	Dijital Ekran (Liquid Crystal Display)
ASCII	Karakter reprezentasyonu icin kullanilan karakter kod standardı (American Standard Code for Information Interchange)
I/O	Giriş/Çıkış (Input/Output)

Kısaltmalar	Açıklama
Baud	Sinyal gönderim hızı
DC	Doğru Akım (Direct Circuit)
BB	Bilgisayar Bilimleri
EM	Elektrik Mühendisliği
MM	Makine Mühendisliği
vb.	ve benzeri

1. GİRİŞ

İnsanoğlu doğumundan itibaren çevresini algılamaya başlar. Nesneler ve olaylar arasındaki benzerlikleri ve farklılıkları algılayarak sınıflandırma yapar. Bilim ve teknolojinin amacı insan hayatını kolaylaştırmak olunca algılama teknolojileri de her geçen gün biraz daha ileri aşamalara ulaşmaktadır. Robotlardan tıpkı bir insan gibi cisimlerin belirgin özelliklerinin algılanarak o özelliğe göre davranışlar sergilenmesi istenebilmektedir.

Nesne tanıma ve görüntü işleme kavramlarının, mekatronik sistemlerle birlikte işlevsel hale geldiği görülmektedir. Öyle ki, tarım, tıp, otomasyon, yeni dönem üretim sistemleri, güvenlik vb. alanlarda benzer nesne tanıma ve görüntü işleme teknikleri kullanılmaktadır. Nesne tanıma sistemi, nesneleri ebatlarına, renklerine ve diğer özelliklerine göre birbirinden ayırmak, nesnelerin tam olarak hangi koordinatlarda olduğunu belirlemek veya çalışma için en uygun alanları tespit etmek gibi birçok uygulama alanına sahiptir. Nesne tanıma sistemlerinde dijital görüntü işlemenin önemli bir yeri vardır. Dijital görüntü işleme metotlarının temeli, başlıca iki ana uygulama alanından oluşur. Görüntü bilgilerinin insanlar için yorumlanması ve özerk makine algılaması için görüntü işleme verisinin depolanması, dönüşümü ve gösterilmesidir. Nesne tanıma görevi, iki alt göreve bölünür. Bu görevler, nesnenin yörüngesinin izlenmesi ve tahmini ve arka arkaya devam eden görüntülerden nesnenin tanınmasıdır[1].

Görme duyusu, duyularımızın en üst seviyesidir. İnsanların algılamasında resmin (görüntünün) en önemli rolü oynaması şaşırtıcı değildir. Elektromanyetik spektrumun görsel bandıyla sınırlanmış görüntüleme cihazları, elektromanyetik spektrumun tümünü gama ışımasından radyo dalgalarına doğru sıralayarak kaplar. Bunlar ultrason, elektron mikroskobu ve bilgisayar tabanlı görüntülerdir. Bu da dijital görüntü işlemenin geniş ve çeşitli alanlardaki uygulamaları kapsayabilmesini sağlar.

Görüntü analiz alanı, görüntü işleme ve bilgisayarlı görme arasındadır. Görüntü işlemenin en başından, bilgisayarlı görüntülemenin sonuna kadar takip edildiğinde,

süreçler arasında kesin bir sınır yoktur. Yine de bilgisayarlaştırma işlemi düşük - orta – yüksek seviye işleme olarak kullanışlı bir değerler dizisidir. Düşük seviye işlemleri gürültüyü azaltmak için görüntü önişleme, zıtlıkları artırmak ve görüntü şekillendirme gibi basit operasyonları içerir. Orta seviye işleme görüntü üzerinde bölümlendirme (görüntüyü parçalara bölme), parçalara ayrılan bölümlerin bilgisayar işlemleri için kullanışlı hale getirilmesi ve özel objelerin tanımlanmasıdır[2]. Tanıma sürecinin sonunda, görüntü analizinde olduğu gibi, yüksek seviye işleme ile tanınan bir nesne grubunun "algılamasını" içerir. Buna göre, görüntü işleme ve görüntü analizi arasında üst üste binmiş mantıksal alanlar vardır, bu alanlar görüntü üzerinde tanımlanmış belirli bölge ve nesne alanlarıdır. Görüntü işleme olarak adlandırılan yapı, girdi ve çıktıları görüntü olan işlemleri kapsar, çevrelenen işlemeler görüntüden elde edilmiş, doğal, tanımlanmış belirli objeleri içerir.

Robotlar ortamdan aldıkları verileri, dünyası hakkında sahip olduğu bilgiyle sentezleyerek, anlamlı ve amaçlarına yönelik bir şekilde hareket edebilen ve bunu güvenli bir biçimde yapabilen makinelerdir[3]. Bu tanıma girebilmesi için öncelikle robot diye adlandırılan makinenin fiziksel bir varlığının olması gerekir. Yani sadece bir bilgisayar simülasyonu değil, gerçek bir ortamla iletişim halinde olan bir makine kastedilmektedir.

Bu nedenle bir robot tasarlamak çok boyutlu, zor bir kontrol problemidir. Robotu oluşturan dört ana sistemi birbirine uygun bir biçimde seçmek ve hepsini birlikte geliştirmek söz konusudur. Bu dört sistem şunlardır[4]:

Robotun ortam hakkında gerçek-zamanlı bilgi edinmesi için kullanılan alıcılar, robota amacına yönelik fonksiyonları gerçekleştirmesi için yerleştirilen efektörler, robotun hareket aksamı ve kontrolünü sağlayan elektronik sistemdir. Ancak bu dört unsur birleştiğinde ortaya kendi başına hareket eden, bilgi toplayan, yapacağı işin niteliklerine göre donanmış bir robot çıkabilir.

Robotik bilim ve mühendislik eğitim sürecinin vazgeçilmez bir parçasıdır. Ancak robotiği oluşturmanın pedagojisi sanat ve bilimdir. Bu güne kadar robotik eğitimi

alanında yapılanların hepsi yeni doğmuş bebek kadar kırılgan ve yetersizdir. Bugüne kadar getirilen bilgi birikimi ortak projeler ve teknoloji transferleriyle, hepsi farklı eğitim kademelerinde uygulanabilir programlar haline getirilebilir[5].

Türk Millî Eğitim Bakanlığının robotik alanında çalışmak isteyenler için bir tanımlaması mevcuttur. Buna göre bu alanda çalışmak isteyenlerin görme ve işitme duyu organlarının mesleğin gerektirdiği derecede işlevini yerine getirmesi gerekir. Matematik ve fen bilimlerine yatkın, analitik geometri ve görsel düşünme yeteneğine sahip olmaları istenir. Bunun yanında soğukkanlı, dikkatli, titiz ve ayrıntılara özen gösteren, ince ayrıntıları yakalayabilen, el becerisi gelişmiş, yeniliğe açık, araştırıcı, sorun giderici, alet ve makinelerle çalışabilen, bedenen ve ruhsal olarak sağlıklı olan, estetik görüşlü ve meslekî eğitim almış kimseler olmaları gerekir[6].

Robotik eğitimi alanında okul içi ve okul dışına yönelik projelerde amaç; eğitimcilere bütünleştirici bilim ve teknoloji ile zengin bir robotik müfredatı ve öğrenmenin standartlarıyla bağlantılı eğitim kaynakları sağlamaktır. Öğrencilerin web tabanlı veya karşılıklı etkileşimli ortaklıkları aracılığıyla hem kendi aralarında hem de öğreticilerle konu hakkındaki fikirlerin, ders programlarının ve robotik hakkında öğrenim deneyimlerinin paylaşımına zemin hazırlayacağı ve gelişmiş teknoloji uygulamalarında öğretmen ve öğrenci eğitimi aracılığıyla gelecek işgücünün gelişimine katkıda bulunacağı varsayılır[7].

Öte yandan robotik eğitimi ve otomasyon alanında yapılan çalışmalar, pahalı teknoloji transferinin yetişmiş insan gücü kısıtlarından dolayı sekteye uğramakta olduğu yönünde sonuçlar vermektedir. Öyle ki işletme sahipleri eğitim politikalarının yeniden gözden geçirilmesi ve mesleki eğitim veren kurumlardan mezun olanların niteliklerinin önceden kestirilebilir olması yönünde taleplerde bulunmaktadırlar[8].

Robotik, öğrenmesi zor bir daldır. Bu bilimde uzmanlık birkaç enstitüde toplanmış durumdadır. Geçerli bir müfredat, sağlıklı ve ucuz öğretim platformları ve derli toplu laboratuarlar ve çözümlü projeler yoktur. Robotiği iyi öğretmek MM, EM ve BB programlarının bütünleşmesine dayanmaktadır ve dengeli bir müfredat programı çıkartmak büyük bir rekabet yaratacaktır[9].

Robotlar üzerinde yapılan araştırmaların son yıllarda yoğunlaşmasının temel sebebi bu konuda üretilmiş olan bilgi birikiminin daha güvenilir tasarımlara daha çabuk ulaşılmasını sağlamasıdır. Endüstride robotlar hızlı ve görece hatasız çalışmalarıyla verimi artırırken, insanların çalışmasını kısıtlayan organik faktörlerden etkilenmemeleri onlara bazı işlerde önemli avantajlar sağlamaktadır. Örnek vermek gerekirse, elektronik devrelere pirinç tanesi büyüklüğündeki yüzlerce parçanın yerleştirilmesi insanlar tarafından yapıldığında yorucu, zahmetli, sıkıcı ve hata oranı yüksek bir iştir. Oysa bir robot kol bu işi süratli ve hatasız bir biçimde gerçekleştirebilmektedir.

Rutin işlerin yanı sıra insanların uzun süreler çalışamayacakları ortamlarda da robotlara iş düşmektedir. Deniz dibinde yüksek basınç altında, yüksek sıcaklıkta, Mars yüzeyi gibi elverişsiz ortamlarda, hatta insan vücudunun sığamayacağı genişlikteki havalandırma borularında robotlar kullanılmaktadır.

Robot biliminin günümüze kadar geçirdiği kronolojik evrelere bakıldığında insanoğlunun sürekli olarak böyle bir yapı üzerinde durduğu görülecektir. İnsana benzeyen ama insandan eksik olan varlıkların aslında çok eski bir düşüncenin ürünü olduğu ortaya çıkmaktadır. Bu düşünce, ortaya çıkışından beri insandan daha aşağı olan bu yapının insana hizmet için var olduğu varsayımıyla birlikte yürümüştür.

Eski bir Yunan mitinde tanrı Hephaestos som altından iki dişi hizmetli yaratır.

Bir diğer eski efsane de Ortaçağ Yahudilerinin Golem'idir. Golem topraktan yaratılmış ve Yahudi halkını tehlikelerden korumakla görevli bir hizmetkârdır. Ağzına (veya alnına) yerleştirilen komutlara uyar, bu komutlar yerinden çıkartıldığında durduğu hakkındadır.

Yine ortaçağ inanışlarına göre güçlü büyücüler *"Homunculus"* adı verilen ufak insancıklar yarattıkları ve bu yaratıkların sahiplerine büyük bir sadakat ile hizmet ettiklerinden söz edilmektedir.

MÖ 270: Ctesibus (Robotiğin Babası) adlı eski bir Yunan bilgini hareketli parçalardan oluşan organ ve su saatleri üretmiştir.

MÖ 100: Otomatik açılan tapınak kapılarına rastlanmaktadır.

1136–1206: Ebul-iz İsmail bin ar-Razzaz el-Cezeri' ye ait çeşitli otomatik makineler bulunmaktadır. İlk sibernetikçi olarak kabul edilen Ebul-iz İsmail bin ar-Razzaz el-Cezeri 1205–1206 yıllarında yazdığı *"Kitab-ül'-Camü Beyne'l-İlmi-i ve'l-amelen-Nafi' Fi Sınaati'l-Hiyel"* adlı kitabın içinde, 300'e yakın otomatik makine ve sistemleri ile ilgili bilgi verdikten sonra çalışma özelliklerini şemalarla göstermiştir. Sadece suyun kaldırma ve basınç gücünü kullanarak tamamen yeni bir teknik ve sistem kurmuş, çok yönlü otomatik hareketler elde edebilmiştir. Kurmuş olduğu otomatik sistemlerde ses (kuş, davul, zurna, ıslık vb.) ya da çığlık çıkması gerektiği anda bu sesleri de sağlayabilmiştir[10].



Resim 1.1 - Ebul-iz İsmail bin ar-Razzaz el-Cezeri'ye ait bir robot tasarımı

1800 yılında Jacques de Vaucanson, Pierre& Henri-Louis Jackuetdroz, Henri Maillerdet Otomatik yazı yazan ve müzik enstrümanı çalan makineler geliştirmişlerdir. 1801'de Joseph Jacquard ilk kez delikli kart kullanarak çalıştırılan otomatik dokuma makinesi geliştirmiştir. 1830'da Christoper Spencer mekanik kam denetimli otomatik bir torna tezgâhı geliştirmiştir.

1892'de Seward Babitt sıcak metal parçaları fırından almak üzere motorlu tutucuya sahip vinç tasarlamıştır.

1920–1921: Robot kelimesi ilk olarak 1920'lerin başında yazılmış bir kitapta karşımıza çıkar. Karel Capek'in R.U.R. (Rossum's Universal Robots) kitabında mekanik ve otonom, ama arzulardan yoksun yaratıklar olarak kullanılan robot, daha sonra birçok bilimkurgu romanına konu olmuştur.

1938'de De Vilbis firması için Willard Pollard ve Harold Roselund programlanabilir boya püskürtme makinesi geliştirdiler. 1940'da Grey Walter tarafından ışığa yönelen ilk gezer robotlar (machina speculatrix) üretilmiştir.

1941–1942: Isaac Asimov robot kelimesinden Robotik kelimesini türeterek ilk kez kullandı. Robotik, robot teknolojisiyle ilgili bütün alanları kapsar. Isaac Asimov ünlü robot serisiyle teknolojik açıdan tutarlı bir robot kavramı oluşturmuş ve robotların amacının insana hizmet olduğunu, bir robotun kendi amaçlarını insanların amaçlarına hiçbir zaman tercih edemeyeceğini koyduğu üç robot kuralı ile belirlemiştir[11]:

* Bir robot bir insana zarar veremez veya pasif kalmak suretiyle zarar görmesine izin veremez.

* Bir robot kendisine insanlar tarafından verilen emirlere 1. Kural ile çelişmediği sürece itaat etmek zorundadır.

* Bir robot 1. ve 2. Kurallar ile çelişmediği sürece kendi varlığını korumak zorundadır.

Bu model öylesine etkili olmuştur ki, günümüzde robot araştırmaları endüstride kullanımı hedefleyen fonksiyonel tasarımların ötesinde insansı, hayvansı ve davranış diye adlandırılabilecek özelliklere sahip robotlar üzerinde yoğunlaşmaktadır.

1946'da George Devol, genel amaçlı manyetik kayıt yapabilen ve tekrar çalıştırılabilen bir cihaz geliştirmiş ve makinelerde kullanmıştır. J. Presper Eckert ve John Mauchy, Pennsylvania Üniversitesinde ilk bilgisayar olarak bilinen ENIAC'ı geliştirmişlerdir.

1948'de MIT'ten Norbert Wiener Elektronik mekanik ve biyolojik sistemlerin denetim ve iletişimini inceleyen "Sibernetik" başlıklı kitabı yayımladı. 1951'de Raymond Goertz, ABD Atom Enerjisi Komisyonu için uzaktan işletilen bir kol tasarladı. 1954'de George Devol programlanabilir genel amaçlı robotu tasarladı ve patent başvurusu yaptı.

1961'de endüstride ilk robot (UNIMATE) General Motors'un New Jersey'deki araba fabrikasında devreye girmiştir. 60'lı yıllarda iki üniversite Stanford ve MIT robot araştırmalarında başı çekmiştir. Daha sonra birçok özel kuruluş ve üniversite robotlarla ilgili çalışmalara başlamıştır[12].

1963'de Bilgisayar denetimli, altı eklemli ilk yapay kol (Rancho Arm) geliştirildi. 1970'de Stanford Üniversitesince bir robot kol geliştirildi ve bu robot kol Stanford Kolu adıyla araştırma projelerinde bir standart olarak yerleşti.

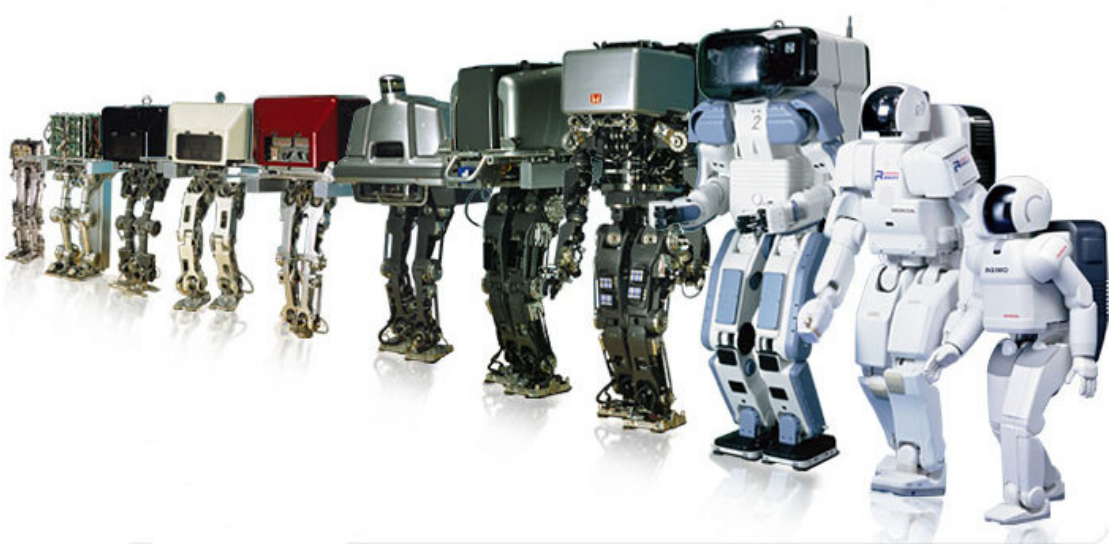
1973'de Richard Hohn, Cincinati Milaron Corporation adına ilk mini bilgisayar denetimli robotu geliştirdi. Geliştirilen robot T3 (The Tomorrow Tool) olarak adlandırıldı.

1976'da dokunma ve basınç algılayıcılarını kullanarak küçük parçaların montajını yapabilen ilk robot üretim hattında kullanılmaya başlandı. Bu dönemde vizyona giren, Star Wars sinema filmindeki C3PO ve R2D2 adlı robot animasyonlarıyla robot kelimesi geniş insan kitlelerine yayıldı.

1979'da Stanford Chart isimli gezer robot üzerine monte edilmiş bir kameradan alınan görüntüleri kullanarak engellerle dolu bir odayı baştan sona aştı.

1997'de NASA tarafından Mars araştırmalarında kullanılmak üzere Pathfinder isimli robot yapıldı. 2000 yılına gelindiğinde Honda firması tarafından üretilen insansı robot ASIMO bugünkü haline geliştirildi[13].

Günümüzde robotik adına en çarpıcı gelişme Amerikalı robotikçi ve fizikçi Dr. Mark W. Tilden tarafından NASA için interaktif çalışmalar yapabilen tamamen programlanabilir insansı Robosapien'in üretilmiş olmasıdır[14].



Resim 1.2 İnsansı robot ASIMO'nun gelişim sürecindeki değişimi

Robotik ve otomasyon, fabrikasyon ve üretim sektöründeki gelişmesinin yanında birçok insanın günlük yaşamlarının bir parçası olmaya başlıyor. Ama insanlar bunun farkında değiller; çünkü robotik genelde birçok filmde ve televizyon programında gösterilen alışılmış antropomorfik biçimlerde insanların algılarında yer alıyor. Oysa insanların hayatını kolaylaştırıcı ürünler halindedirler.

Yaygın otomasyona doğru bu yeni ve yükselen eğilim sebebiyle eğitim teşkilatlarının sadece robotik araştırmacıları ve tasarımcıları değil robotik aygıtlarının pazara dikkatli bir şekilde girmelerini sağlayacak teknik altyapı ve yetişmiş insan gücünü de hazırlamaya başlamaları gerekmektedir. Son birkaç yıldır, üniversiteler çok çeşitli robotik dersleri vermekteler ve bu derslerde hareketlilerden olduğu kadar sabit robotlardan, kinematikten ve kontrolden de bahsedilmektedir. Buna rağmen son eğitim aşamaları dışında (yüksek lisans tezi veya doktora araştırması gibi), öğrencilerin robotik aygıtlarının üretimi ve tasarımı üzerinde uygulamalı eğitim fırsatları olmamaktadır.

Mekanik olmayan alanlar için kinematik sadece matematikseldir ama öğrenciler kitap, tahta gibi iki boyutlu malzemeleri ve temel ilkeleri tanımlama ve atamalar, bağlantı döngüsü, çevrimler ve yörünge hesaplaması gibi yapıların üç boyutlu malzemeler kullanarak öğrenme zorluğunun üstesinden gelebilirler. Bununla birlikte, bu kavramlar laboratuvar deneylerinde, öğrencilerin zorluklarla karşılaşmalarını engelleyen bir robotik sistemle (tam anlamıyla düzenlenmiş bir kontrol sistemi ve programlama ortamıyla) gösterilmektedir.

Yalnızca üretimin yeterli olmadığı günümüz dünyasında; neyin, ne kadar, ne sürede, ne kalitede, hangi girdilerle üretildiğinin bilinmesi gerekmektedir. Buna bağlı olarak günümüzde üretim yapan işletmeler endüstriyel otomasyon teknolojilerine ve üretimde esnekliğe büyük önem vermektedirler. Endüstriyel otomasyon, üretim süreçlerinin yürütülmesinde insan müdahalesinin bir ölçüde veya tamamen ortadan kaldırılması ve otomatikleştirme işlemidir.

Makine görmesi birçok endüstriyel işlemde potansiyel uygulama sahasına sahip olması sebebi ile robotik alanında önemli bir algılama teknolojisidir. Makine görmesinin şu anki uygulamalarının çoğu henüz araştırma aşamalarında karşımıza çıkmaktadır ancak görüntü teknolojisinin robotik alanının geleceğinde giderek artan önemde bir rolü olacağı çok açıktır.

Bu çalışmada, Esnek üretim sisteminde görüntü işleme tekniği ile altı eksenli deneysel eğitim robotu kullanılarak bir robotik eğitim modeli geliştirilmesi amaçlanmıştır. Bu bağlamda ikinci bölümde, genel bilgisayarlı görme teknikleri üzerine literatür taraması yapılmış ve günümüzdeki yapısı üzerinde durulmuştur. Üçüncü bölümde bu tez çalışmasında uygulama tabanını teşkil eden renkli görmenin temelleri üzerinde durulmuştur. Dördüncü bölümde ise esnek üretim sistemi modelleri üzerinde durularak üretim tekniğinin ilkeleri tanımlanmaya çalışılmıştır.

Esnek Üretim Sistemleri Laboratuarında gerçekleştirilen uygulamaların çalışıldığı sistem elemanları altıncı bölümde tanıtılmış ve programlama kısıtları belirlenmiştir. Bunun için programlama dilleri arasındaki algoritmik bağlantılar kullanılmıştır. Yine bu bölümde kullanılan deneysel eğitim robotu olan XR3 robotun kinematik olarak

incelemesi yapılmış çözülen kinematik yaklaşımlar içerisinde bu tez çalışmasında kullanılan kinematik model üzerinde durulmuştur.

Çalışmanın materyal metot kısmını oluşturan sistem elemanları ve programlama esasları ile uygulama bölümlerinde temel işleyişi ifade edilen esnek üretim sisteminde görüntü işleme tekniği ile robotik eğitim modeli geliştirilmesi, bu şekilde disiplinler arası bir çalışmadır. Robotik eğitimi açısından robotların çevre elemanlarına örnek olarak esnek üretim sistemi ile bilgisayarlı görme ve görüntü işleme unsurlarının kullanılması yanında bir deneysel eğitim robotu üzerinde mekanik yapı ve işleyiş hakkında görsel ve matematiksel olarak, algılamayı kolaylaştırıcı şekilde hazırlanan bir yazılım desteği ile ortaya konması önem taşımaktadır. Böyle bir yapı hazırlanırken, laboratuvar uygulaması ortaya koymanın yanında öğretim materyali oluşturulması da amaçlanmıştır.

Bu tez çalışmasının uygulama kısmını oluşturan yedinci bölümde RHINO firması tarafından üretilmiş olan 5+1 eksenli deneysel eğitim robotu XR3 ve MARK3 denetleyici kullanılmıştır. Yine RHINO firması tarafından sunulan BASIC derleyici tabanlı RoboTalk robot programlama dili incelenmiş ve DELPHI 6 derleyici kullanılarak yeni bir görsel eksen kontrolü sağlanmış ve eğitim amaçlı bir robot kontrol programı geliştirilmiştir.

Bilgisayarlı görme unsuru olarak farklı disiplinlerden oluşan birimler bir araya getirilerek bir cismin ayırt edici özelliklerinden biri olan renk bilgisi ele alınmıştır. Deneysel özellikli bir manipülatörün kinematik özellikleri çerçevesinde fiziksel olarak ayırt edilmesi sağlanmıştır.

Geliştirilen yazılım bir eğitim materyali olarak kullanılabilecek şekilde tasarlanmıştır. Böylelikle küçük bir insan hareketi taklit edilirken arka planda nelerin gerçekleşmekte olduğu da ortaya konmaya çalışılmıştır. Yazılım dört ana modülden oluşmaktadır. Bunlar görüntü işleme modülü, robot iletişim modülü, kinematik analiz modülü ve Türkçe programlama modülüdür.

Sistem iki ana bölümde ele alınabilir: ilki algılama ve görüntü işleme, ikincisi ise manipülatörün hareketi ve kontrolüdür. Bu şekilde eğitim aracı olarak kullanılan Esnek Üretim Sistemleri Laboratuarına, görüntü işlemeye dayalı deneysel çalışmalar yapılabilecek bir yapı da kazandırılmaya çalışılmıştır.

2. BİLGİSAYARLI GÖRME

2.1. Bilgisayarlı Görme Temelleri

Bilgisayarlı görme, kavramsal ve algoritmik temelleri geliştiren bir bilim dalıdır. İstenen bilgiler; incelenmiş görüntüler, görüntü kümelerinden ya da görüntü sonuçlarından, özel amaçlı veya genel amaçlı bilgisayarların yaptığı hesaplamalar aracılığıyla çıkartılabilir ve çözümlenebilirler. Bu bilgiler; genel nesnelerin tanımı, bilinmeyen nesnelerin üç boyutlu tanımlanması, incelenmiş nesnelerin pozisyonu ve yönelimi ya da nesnenin ayrılmış iki noktası arasındaki mesafe veya dairesel kesitinin yarıçapı gibi herhangi bir uzaysal özelliğinin ölçümü ile ilişkili olabilir.

Görüntü, bir nesnenin, iki veya üç boyutlu bir manzaranın ya da başka bir görüntünün uzaysal resmidir. Optikte olduğu gibi gerçek ya da hayali olabilir. Bilgisayarlı görmede görüntü, genelde video görüntüsü, dijital görüntü ya da resim gibi kaydedilmiş görüntü anlamına gelir.

Soyut olarak görüntü bazı sınırlar içerisinde tanımlanmış, iki değişkenli, sürekli I fonksiyonu olarak ifade edilir. Uzaysal koordinatları (r,c) olarak tayin edilmiş bir görüntünün değeri, I (r,c) fonksiyonuyla gösterilir. Optik ve fotoğrafik sensörler için I (r,c) fonksiyonu, elektromanyetik spektrumdan alınmış ışık yayan güçle orantılıdır. Sıra bulucu sensörler için I (r,c) fonksiyonu, (r,c) ile üç boyutlu ortamdaki nesne arasındaki gözlem çizgisi mesafesinin fonksiyonudur ve bu görüntüye sıra görüntü denir[15]. Görüntü bir harita olduğunda, I (r,c) fonksiyonu renk, toprak çeşidi veya kaya çeşidi gibi bazı kategorilerle ilişkilendirilmiş bir simge veya bir gösterge de olur. Bu tarz görüntüler sembolik görüntü olarak adlandırılır. Kaydedilmiş görüntü, fotoğrafik, video sinyali veya dijital formatta olabilir.

Gözlem birimleri analiz birimleri olmadığından bilgisayarın tabiatında zaten var olan sorunlar ortaya çıkar. Gözlenmiş dijital görüntü birimlerine *piksel* adı verilir. Bir pikselin bir pozisyonu ve belli değer özellikleri vardır. Belirli herhangi bir pikselin pozisyonunun ve piksel değerinin bilgisi, hemen hemen hiçbir zaman nesnenin tanınması, nesnenin biçiminin tanımlanması, pozisyonu ve yönelimi, nesnenin

üzerindeki herhangi bir mesafenin ölçümü ya da nesnenin eksik veya kusurlu olup olmaması ile ilgili bir bilgi ifade etmez.

Yoğun görüntülerin piksel değeri gri seviye olarak bilinir. Her piksel değerini saklamak için 8 bitlik tamsayılar kullanıldığında, grinin bazı tonlarını gösteren her derece 0 (0 = siyah)'dan 255 (255 = beyaz)'a kadar sıralanır. Nesnenin yerini çıkarmak için, gri seviyenin uzaysal görünüşünün yönelimi veya sınıfının, hangi piksellerin nesnenin parçası olup hangilerinin olmadığını çıkartma kapasitesi olması gerekmektedir. Ardından, nesnenin parçası olan bu piksellerin arasından özel işaretler, çizgiler, eğriler ve sınırlar gibi gözlenmiş nesne özelliklerini ayırt edebilmek için de kapasite gerekmektedir. Bu özelliklerin kendileri, görüntü ve nesne üzerinde olan bir uzaysal ilişki içinde düzenlenir. Nesne şeklinin, pozisyonunun ve yöneliminin analitik çıkarımı, matematiksel çözümü içindeki seçilmiş bir sensör koruma geometrisi ve benzer nesne özellikleri ile ayırt edilmiş görüntü özelliklerine bağlıdır[15].

Bilgisayarlı görmedeki farklı birimleri özetlemek gerekirse;

* Gözlem birimi(Piksel)

* Üç boyutlu nesne özellik birimi(Nokta, doğru parçası veya yay parçası ve bükülmüş veya düzlemsel yüzey)

* İki boyutlu nesne özellik birimi(Nokta, doğru parçası veya bölge)

* Nesne özellikleri ve görüntü özellikleri arasındaki eşleştirme ve benzerlik birimi

* Nesne özellikleri arasındaki ilişki birimi

* Görüntü özellikleri arasındaki ilişki birimi

* Sensör pozisyonu ve yönelimi birimi

* Nesne pozisyonu ve yönelimi birimi

Nesnenin türü, arka planı, görüntüleme sensörünün türü ve bilgisayar görmesi probleminin kolay mı yoksa zor mu olduğunu tanımlayan sensörün görüş noktası, piksel biriminin karmaşık veya direkt olarak nesne birimine dönüşümünü mümkün kılmaktadır[15].

Bölge birimlerinin, çizgi birimlerinin ve nokta birimlerinin, bölgelerin bitişikliği veya doğru parçalarının yakınlığı, aynı doğru üzerinde olma durumu ve paralelliği gibi karşılıklı ilişkilerini tanımlamak için uzaysal ilişki birimlerine dönüştürülmeleri gerekebilir[16]. Hangi tür birim dönüştürmelerinin kullanılması gerektiği, görme işinin özel doğası, görüntünün karmaşıklığı ve mevcut bilgilerin çeşitliliğine bağlıdır.

2.2. Bilgisayarlı Görme Uygulamaları

2.2.1. Robotlar

Endüstri ve savunma, otomasyonun gelişiminde iki önemli zorlayıcı fonksiyondur. Otomasyonun güvenirliliği ve etkinlik değeri her iki alanda da rekabetçi yanını geliştirmiştir. Sonuç da geliştirici geçerli standartlara yol gösterdi. Bir görüşe göre endüstriyel devrim hala bizimle; otomatik görüntülemeyi içeren ileri teknoloji içerikli uygulama geliştirmek için sensör gereksinimleri geliştiren robotik uygulamalarını dikkate almaya değerdir. Bu uygulamaların %75'inden fazlası görüntüleme gerektirmektedir[17].

Bazı uygulamalarda görüntüleme cihazı kullanılarak yapılan denetimin insan gözetimli denetimden daha uygun olduğu görülmektedir. Örneğin, baskı devre kartının görüntüsü verildiğinde bilgisayarlı görüntüleme sistemi, kartın doğru bir şekilde yapıldığını hesaplamak zorundadır.

Robot uygulamalarında manipülatör tanınması, hareket etmesi, kavranması ve görsel rehber kullanarak dağıtılmış bir çalışma alanında bir nesneyi hareket ettirmesinin istenmesi (hareket sınırlandırılmışsa) oldukça yaygındır. Bu süreç parçaya uygun bir yol ile kavranmış olabileceği için yönlendirme belirlemeyi gerektirir[18].

Sürücüleri veya pilotları tehlikeli ortamlardan kurtarma ihtiyacı, özerk araç sistemleri gereksinimleri ile daha anlaşılır olmuştur. ‘Ateş et ve unut,’ terimi sürekli alınan görüntü bilgisine göre kendi yörüngesini ayarlama yeteneğine sahip silah dağıtım sistemlerini tanımlar. Bu tarz bir uygulama, görüntü dinamiklerini modellemeyi ve zaman değişimli sensör yönleri, görüntü ve durum değişimini gerektirir[18].

2.2.2. Görüntü sıkıştırma ve iyileştirme

Bilgi taşıma, taşıma aracının bilgi taşıma kapasitesiyle ve yaklaşımıyla sınırlıdır. Özel uygulamaların gerektirdiği bant genişliğini minimize etmek için taşınan sinyalin karakteristiğinde önemli bir değişiklik yapmadan birçok strateji geliştirilmiştir. Fiber optikler gibi geniş elektriksel bant genişlikleriyle gelişen medya bu çabanın önemini azaltmıştır. Yüksek çözünürlüklü televizyon görüntülerinin, örneğin 1000 satırlı çözünürlüğün memnun edici görünüşü, izleyicilere bu görüntü kalitesini ulaştırma yeteneğine sahip bant genişliği olan sistemlere talep yaratır. Görüntü iyileştirme sonuçların öznel bir şekilde görüldüğü diğer bir alandır[19].

2.2.3. Tıbbi uygulamalar

Biyolojik mekanizmaları etkileyen enerji kaynakları görüntü benzeri şeyler oluşturur. Bu, fizikçiler tarafından tıbbi tesislerde kullanılan X ışınları, radyo frekansları, manyetizma ve ultrasonik enerjiyi içerir. Sıklıkla, görüntüleme çevresinin ideal olmayan karakteristikleri ideal olmayan görüntülere katılır. Bu görüntüleri dijital formda işleme görünüşlerini geliştirebilir. Örneğin, bir doktorun teşhisine yardım etmek için tıbbi X ışınında uç bilgisine dikkat çekmek istenebilir[19].

2.2.4. Görüntü anlama

Yüksek düzeyli biyolojik görsel sistemlerin, kavrama kabiliyetini geliştiren sistemlerin geliştirilmesi dijital görüntü işlemenin bilinen uygulama alanlarından biridir. Daha önce bahsedilen uygulamaların birçoğu doğal olarak bu yeteneği gerektirir veya bu yetenekten yararlanılır[17].

2.3. Tanıma Yöntem Bilimi (Metodolojisi)

Bilgisayarlı tanıma ve nesnenin kontrol işlemi genelde görüntüye ait verilerin tanıma bilgilerini sırayla dönüştüren çeşitli basamaklar gerektiren karmaşık bir işlemdir. Düzenli duruma getirme, sınıflandırma, gruplara ayırma, çıkarım ve eşleştirme, tanıma probleminin ayrışımını meydana getirir. Her bir basamak kendisinden bir sonraki basamak için veri hazırlar ve bu verilerin hatasız bir şekilde dönüştürülmelerini sağlar.

Bu basamaklar, dönüşüm işlemi biriminin herhangi bir aşamasında çalıştığından, bilgileri dönüşüm ünitesi için hazırlarlar ve birinci yüksek aşama birimini teşhis edip yorumlarlar. Yorum, ikinci yüksek-aşama birimini de içerir. Birçok görme algoritması bu basamakları içine alır.

2.3.1. Düzenli duruma getirme

Düzenli duruma getirme, ilgisiz dönüşümler tarafından değiştirilen, bilgi verici örnekten ibaret olan, gözlenmiş görüntü olarak ileri sürülen modele dayalıdır.

Düzenli duruma getirme, gözlenmiş görüntünün temelleri üzerinde bilgi verici örneği tahmin eder. Böylece düzenli duruma getirme basamağı, bütün ölçümleri etkileyen, rastgele kopyalanmış değişimler olarak da düşünülebilen bozucu sinyalleri önler. Düzenli duruma getirme, aynı zamanda ilgisiz sistematiği ve kopyalanmış değişimleri önleyerek zemin normalleştirmesini sağlar. CCD(Charge-Coupled Device) kameradan elde edilmiş görüntüler de uygulanmış morfolojik düzenli duruma getirme algoritmalarının sonuçlarıdır[20].

2.3.2. Sınıflandırma

Sınıflandırma, bilgi vericinin, örnek olayların uzaysal düzenlemesi gibi bir yapıya sahip olduğu öne sürülen bir modele dayalıdır. Sınıflandırma her bir pikselin hangi tür uzaysal olaylara katılacağını belirler. Örnek olarak, eğer bilgi verici örneğin ilişkili uzaysal olayları sadece yüksek değerli ve düşük değerli piksellerin olayı ise başlangıç işlemi sınıflandırma işlemi olarak da düşünülebilir. Diğer tür sınıflandırma

işlemi, farklı şekil denklemlerine katılan kenar bulma, köşe bulma ve piksellerin teşhis edilme işlemlerini içerir[21]. Sınıflandırma işlemi, en basit uzaysal olayların çeşitleriyle pikselleri sınıflandırır.

2.3.3. Gruplara ayırma

Gruplara ayırma işlemi, oluşumları bir araya toplayarak ya da aynı tür oluşumlara katılan, en fazla birleştirilmiş piksel kümelerini belirler.

Eğer sınıflandırma sembolik ise gruplara ayırma gerçekten birleştirilmiş parçaların işlemidir. Uygun veri yapısının değişimini içerir. Gözlenmiş, düzenli duruma getirilmiş ve sınıflara ayrılmış görüntülerin hepsi dijital görüntü veri yapılarıdır. Tamamlamaya bağlı olarak gruplara ayırma işlemi, ya ait olduğu uzaysal olayla birleştirilmiş, içinde verilen her bir pikselin göstergesinin olduğu görüntü veri yapısı, ya da kümelerin toplanmasıyla oluşan bir veri yapısı meydana getirir. Her bir küme, bir uzaysal olayı karşılamaktadır ve olaya katılan bir çift pozisyon (satır ve sütun) içerir. Her iki durumda da değişim uygun veri yapısında oluşmaktadır[21].

Gruplara ayırma basamağından önceki ilgilenilen yapı, piksellerdir. Gruplara ayırma basamağından sonraki ilgilenilen yapılar, piksel kümeleridir. Gruplara ayırma işlemi yeni bir varlık kümesi saptar. Ama gruplara ayırma basamağından sonra elde edilen varlıklar yalındır. Sahip oldukları tek şey, özellikleridir.

2.3.4. Çıkarım

Çıkarım işlemi her bir piksel grubu için bir özellik listesi çıkarır. Örnek olarak piksel grubunun merkez noktasını, alanını, yönelimini, uzaysal momentini, gri ton momentini, uzaysal-gri ton momentini, çevreleyen çemberini ve bunun gibi birkaç özellikleri tanımlanır. Diğer özellikler grubun bölge olarak mı yoksa yay olarak mı düşünüldüğüne bağlıdır. Eğer grup bölge olarak düşünülüyorsa boşluk sayısı faydalı bir özellik olabilir. Eğer grup yay olarak düşünülüyorsa ortalama eğrilik faydalı bir özelliktir[21].

Çıkarım işlemi aynı zamanda iki veya daha fazla grubun arasındaki topolojik ve uzaklık ilişkilerini de karşılaştırabilir. Mesela, çıkarım işlemi iki grubun bitişik olduğunu veya birbirlerine ne kadar uzak olduğunu ya da bir grubun diğerinin üstünde olduğunu belirlemeyi sağlar.

2.3.5. Eşleştirme

Çıkarım işleminin tamamlanmasından sonra, olaylar teşhis edilmiş ve karşılaştırılmış görüntüler üzerinde meydana gelebilir. Gözlenmiş uzaysal olayların önemi, algoritmayla ilgili düzenlemeler oluştuğunda ortaya çıkar. Gözlenmiş uzaysal olaylar, düzenlemede, uzaysal olaylar kümesinin, sandalye veya "A" harfi gibi daha önceden bilinen bazı görüntülenmiş örnekler teşkil etmesi gibi algılamayla ilgili düzenlemeler ortaya çıktığında oluşur[21].

İlişkilendirilmiş görüntü kümelerinin yorumunun ve bu kümeleri verilen bazı üç boyutlu nesnelerle veya iki boyutlu şekillerle birleştirme işleminin tanımlaması eşleştirme işlemi tarafından yapılır. Görüntü işlemlerinin büyük bir bölümü eşleştirme işlemidir. Klasik eşleştirme çeşidi, model eşleştirmesidir. Örneğin, boyut değişimleri çok küçük olmalıdır. Arka plan hemen hemen aynı olmalıdır. Rastgele şekil bozulmaları minimum olmalıdır.

2.4. Görüntü Dijitalleştirme

Görüntü dijitalleştirme için çok çeşitli cihaz birleşimleri kullanılmaktadır. Dijitalleştiricinin bozulma ve çarpıklığını azaltmak ya da ortadan kaldırmak, dijital görüntü oluşturmanın temel bir fonksiyonudur.

2.4.1. Dijitalleştiricinin unsurları

Bir görüntü dijitalleştirici, görüntüyü fotoğraf unsurlarına (piksellerine) ayırmalı ve her birini ayrı ayrı konumlandırabilmelidir. Yine görüntünün her pikselinde grilik düzeyini ölçebilmeli, bir tamsayı dizisi üretmek için ölçümlerini sürekli olarak dijitalleştirebilmeli ve tamsayı dizisini veri depolama birimine yazabilmelidir. Bütün

bunları gerçekleştirebilmek için bir dijitalleştiricinin beş farklı unsura sahip olması gerekir:

Birinci unsur, örnekleme diyafram aralığıdır. Bu aralık görüntü parçacıklarına ulaşılmasını sağlayan düzenektir.

İkinci unsur, görüntüyü taramak için gerekli mekanizmadır.

Üçüncü unsur, ışık algılayıcıdır.

Dördüncü unsur, algılayıcıdan çıkan verileri sayısal değerlere dönüştüren kısımdır. Bu unsur analog dijital dönüştürmeyi gerçekleştirir.

Beşinci unsur, çıktı depolama ortamıdır. Çıktı depolama, sabit hafıza, manyetik disk vb. olabilir[22].

2.4.2. Görüntü dijitalleştiricinin özellikleri

Dijitalleştiriciler fonksiyonlarını yerine getirmek için kullandıkları cihazlar açısından farklılaşmakla birlikte standart bazı özellikleri vardır.

Piksel boyutu, cihazın boyutu ve komşu pikseller arasındaki aralık iki önemli özelliğidir. Eğer dijitalleştirici optik bir sisteme monte edilmişse örnekleme boyutu ve aralık isteğe göre değişir.

Görüntü boyutu, cihazın görüntü girdisi imkânlarıdır. Bir film tarayıcı örneğinde girdi boyutu en fazla 35 mm'lik film ya da 11–14 inch x ışını olabilir. Çıktıda görüntü boyutu çizgilerin maksimum sayısı ile belirlenir ya da her satıra düşen piksel sayısı ile ölçülür.

Ölçülen alan boyutu, dijitalleştiricinin gerçekte ne kadar yer ölçtüğü ve dijitalleştirdiğidir.

Doğrusallık, dijitalleştirmenin doğrusallık derecesi de bir diğer önemli faktördür. Örneğin, cihaz ışık yoğunluğunu dijitalleştiriyorsa, grilik düzeylerinin gerçekleşme

derecesi bilinmelidir ki görüntünün gerçek parlaklığı tahmin edilebilsin. Alt süreçlerin güvenilirliği doğrusal olmayan bir dijitalleştiriciyle tehlikeye atılabilir.

Benzer şekilde cihazın görüntüyü dijitalleştirdiği grilik düzeyleri de bilinmelidir. İlk dijitalleştiriciler sadece iki grilik düzeyine sahipti; beyaz ve siyah. Günümüzde tekrenkli (monochrome) dijitalleştiricilerde 8 bit (256 renk) veri yaygındır ve ulaşılabilir aracı ile birlikte göreli olarak yüksek çözünürlüğe sahiptir.

Bozulma, eğer tek tip gri bir görüntü dijitalleştiriciye iletilirse girdi parlaklığı sabit olsa bile sistemde var olan bozulma düzeyi çıktı grilik düzeylerinde görüntü boyunca değişmelere neden olur. Dijitalleştirici tarafından tanınan bozulma, görüntü renk geçişinin bir kaynağıdır ve az da olsa görüntüde var olan zıtlığa benzer[22].

Bu özellikler bir görüntü dijitalleştiricinin özet bilgilerini içerir. Bunlar, özel bir dijitalleştirici tasarlamak ya da farklı araçları karşılaştırmak için bir temel oluştururlar. Bununla beraber birçok önemli dijital görüntü işlemede tam doğrusallık ve düşük bozulma düzeyinde geniş görüntüleri birçok grilik düzeyine bölebilen yüksek çözünürlükte görüntü dijitalleştirici gerekebilir.

2.4.3. Görüntü dijitalleştirici tipleri

Çok yönlü ve önemli bir görüntü dijitalleştirici, dijital kameradır. Kamera, bir lens sistemine sahiptir ve önündeki herhangi bir nesnenin bir bölümünü dijitalleştirebilir. Bilgisayara karşı yerleştirilen bir televizyon kamerası örnek olarak verilebilir. Böyle bir cihaz sadece fiziksel nesneleri değil, aynı zamanda fotoğrafik filmleri de dijitalleştirebilir.

Diğer bir sınırlı görüntü dijitalleştirici tipi ise film tarayıcıdır. Bu cihaz film üzerindeki fotoğrafik görüntüleri taramak amacıyla geliştirilmiştir. Tarihsel süreçte film tarayıcılar görüntü işleme konusunda öncü bir rol oynamışlardır. Fakat günümüzde, doğrudan dijitalleştirebilen kameralar üretilmektedir.

2.4.4. Dijitalleştirme elemanları

Bir görüntü dijitalleştirici, ışık kaynağına, ışık algılayıcısına ve tarama sistemine sahip olmak zorundadır. Ayrıca ışık kaynağı ya da ışık algılayıcısı, örnekleme boşluğunun önünde olmalıdır.

Işık kaynakları: Akkor lambalar, en yaygın yapay ışık kaynağıdır. Lazer, yüksek derecede odaklanmış ışınlar üretilebilmeyi sağlar. Geniş, yoğun, birbiriyle uyumlu, ışık demetini; argon, helyum, neon gibi aktif bir maddenin atomlarının ilk yükselişinde bir yüksek enerji olarak ve normal duruma eş zamanlı bir geçişi tahrik ederek açığa çıkarır. Bu geçiş, kolayca odaklanabilen ve yansıtılabilen uyumlu ışığın yüksek yoğunluklu dalgalar olarak ortaya çıkmasını sağlar. Yansıtıcılar, belirli fosforlar elektronlarla ışınlandığında, ışık yayar. Eğer bir elektron demeti fosfor kaplı bir cam yüzeydeki küçük bir bölgeye odaklanırsa, ışık bu bölgeden yansır. Ledler, yoğun ve elverişli ışık kaynağı oluştururlar. Ledler yarı iletken galyum arsenikten üretilirler. Işığı kontrollü bir yoğunlukta belirli küçük kaynaktan yayarlar[23].

Sabit kameralar: Son zamanlarda geliştirilen bu algılayıcıların üç temel biçimi; Çift Denetimli Cihaz (Charge-Coupled Device- CCD) düzeneği, Enjeksiyon Denetimli Cihaz (Charge- Injection Device- CID) ve foto diyot düzeneğidir. Bütün bu cihazlar, ilgili görüntü tarafından yayılan elektron dolgusunu okumaya yarayan elektrik devresiyle tamamlanan doğrusal ya da dikdörtgen biçiminde ışık algılayıcı bir düzeneğe sahiptir.

CCD (Çift Denetimli Cihaz) yongaları ışığa duyarlı kristal silikon yonga üzerinde üretilirler. Dikdörtgenel foto dedektör alanları (potansiyel kaynak) silikon alt katmanlarında inşa edilir. Belli bir alanda üretilen foto elektronlar en yakın potansiyel kaynakta tutulurlar ve bir dış terminale ulaşıncaya kadar şarj paketi olarak bir dizi kaynağa dönüştürülürler. Çift denetimli görüntü dijitalleştirici cihaz, dışında toplanan şarjı okumak amacıyla üç farklı yapı kullanır[20].

Tam çerçeve CCD, pozlandırmadan sonra, tam çerçeve CCD okuma süreci boyunca poz, karanlıkta tutulmalıdır. Görüntüyü algılama işi bütün satırlardaki enerji bir satıra

indirilip en altta toplanmak sureti ile en alttaki satıra kadar gerçekleştirilir. Görüntü işleme olarak adlandırılan yapı, girdi ve çıktıları görüntü olan işlemleri kapsar; çevrelenen işlemeler görüntüden elde edilmiş, doğal, tanımlanmış belirli objeleri içerir.

İç hat taşıma CCD, bu yapıda her iki kolondan biri opak bir maskla kaplanmıştır. Bu yapı sadece okuma sürecinde kullanılır. Pozlandırmadan sonra pozlanmış her alandaki enerji paketi bir sonraki masklı alana gelir. Bu taşıma işlemi bütün enerji paketleri dönüşünceye kadar çok kısa bir sürede gerçekleştirilir.

Çerçeve taşıma CCD, bu yapı ise ikili uzun bir algılayıcı düzeneğe sahiptir. Üstteki yarı görüntüyü standart bir şekilde algılar ve alttaki kısım ise - depolama düzeneği- opak mask yöntemi ile aydınlanmadan korunur. Bütünleşme periyodunun sonunda algılama düzeneğinde toplanmış olan görüntü enerjisi satır satır depolama düzeneğine aktarılır[20].

2.5. Dijital Görüntü İşleme

Dijital görüntü işleme metotlarının temeli, iki ana uygulama alanından oluşur. Görüntü bilgilerinin insanlar için yorumlanması ve özerk makine algılaması için görüntü işleme verisinin depolanması, dönüşümü ve gösterilmesidir[16].

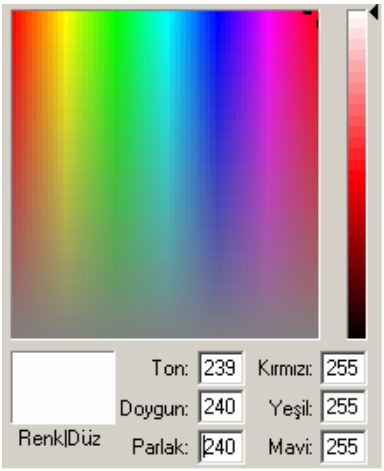
Makine görmesi, birçok endüstriyel işlemde potansiyel uygulama sahasına sahip olması sebebi ile robotik alanında önemli bir algılama teknolojisi durumundadır.

Nesne tanıma ve görüntü işleme kavramlarının mekatronik sistemlerle birlikte işlevsel hale geldiği görülmektedir. Öyle ki; tarım, tıp, otomasyon, yeni dönem üretim sistemleri, güvenlik vb. alanlarda benzer nesne tanıma ve görüntü işleme teknikleri kullanılmaktadır.

Yapılan deneysel çalışmalarda genelde karşımıza çıkan yöntem, birbirine bütünleşik belirli bölümlerden oluşmaktadır.

İşlemin ilk safhasında, kamera tarafından yakalanan görüntü RGB Skalasında (Şekil 2.1) bir frame-grabber kartı ve özel bir yazılım yardımıyla, ileriki ayrıştırma işlemleri esnasında nesnenin tanınabilmesi için dijital bir görüntü bilgisine çevrilerek bilgisayarın hafızasında depolanır.

Nesne tanıma sistemi, nesneleri ebatlarına, renklerine ve özelliklerine göre birbirinden ayırmak, nesnelerin tam olarak hangi koordinatlarda olduğunu belirlemek veya çalışma için en uygun alanları tespit etmek gibi birçok uygulama alanına sahiptir.



Şekil 2.1. Renk skalası

Alınan gerçek dijital görüntünün bu üç kanala ayrılmasıyla görüntüler üç ayrı karakteristikte değerlendirilebilir hale gelir. Burada amaç, farklı parlaklık ve ışıklandırmanın yapacağı etkileri elemek ve ayrıştırma işlemi için en uygun karakteristik eşik değerlerini elde etmektir. Ton görüntü kullanılmasıyla orijinal renk değerleri korunurken bir tür gri skala dönüşümü de elde edilmiş olur.

Bir görüntü uzayda koordinatları bulunan iki bilinmeyenli bir f(x,y) fonksiyonu ile tanımlanabilir ve f fonksiyonunun herhangi bir (x,y) koordinatındaki genliği (genişliği) bu noktadaki görüntünün gri seviyesi veya yoğunluğu olarak isimlendirilir.

Fonksiyonun herhangi bir x,y ve genlik değerleri sınırsız olduğunda bu görüntüye dijital görüntü adı verilir. Dijital görüntüyü oluşturan sonsuz sayıdaki elemanların her biri belirli bir değere ve alana sahiptir. Bunlar; görüntü elemanları, resim elemanları ve pikseller anlamına gelmektedir. Piksel, dijital görüntünün elemanlarını dile getirmek için kullanılan en yaygın terimdir[22].



Şekil 2.2. Elektromanyetik spektrum

Elektromanyetik spektrumun (Şekil 2.2) görsel bandıyla sınırlanmış görüntüleme cihazları, elektromanyetik spektrumun tümünü gama ışımasından radyo dalgalarına doğru sıralayarak kaplar. Bunlar ultrason, elektron mikroskobu ve bilgisayar tabanlı görüntülerdir. Böylece dijital görüntü işleme, geniş ve çeşitli alanlardaki uygulamaları kapsar[24].

Yazı içeren görüntü alanının elde edilmesinin işlemesi, görüntünün özel karakterlerinin bilgisayar işlemesi için tanımlanması ve bir form halinde kullanışlı hale getirilmesidir ve bu özel karakterlerin bilgisayar işlemesi için tanımlanması dijital görüntü işleme olarak adlandıran yapı ile ilgilidir. Görüntü analizi alanında ve hatta bilgisayarlı görmede sayfanın içeriğini algılama, algılama ifadesinin karmaşıklık seviyesine bağlı olarak görüntülenebilir[25].

2.6. Dijital Görüntü İşlemenin Kökenleri (Kaynağı)

Dijital görüntü uygulamalarından ilki, resimlerin ilk defa deniz altı kabloları ile Londra'dan New York'a gazete endüstrisinde gerçekleşmiştir. Bartlane kablo resim transfer sistemi, 1920'lerin başında bir resmin Atlantik'i geçmesi için gerekli 1 haftadan fazla olan zamanı, 3 saatten aza indirmiştir. Kablo transferi için kod yazma donanımı geliştirilmiştir. Resim 2.1 bu yolla transfer edilmiş ve bir harf numunesini taklit ederek bir telgraf yazıcısıyla tekrar yapılmıştır. İlk dijital resimlerin görsel kalitesinin arttırılmasındaki problemler, seçilen yazma prosedürüyle ve yoğunluk

dağılımının seviyesiyle ilişkilendirilmiştir. Resim 2.1'de kullanılan yazma metotlarından 1920'nin sonlarına doğru telgraf alma terminalinde şeritlerin delinmesiyle oluşturulan fotoğraf yapma tekniğiyle vazgeçilmiştir. Resim 2.2 bu metodun kullanıldığı bir fotoğrafı göstermektedir. Resim 2.1 üzerine yapılan ton kalitesi ve çözünürlükteki gelişmeler kendini göstermektedir[24].



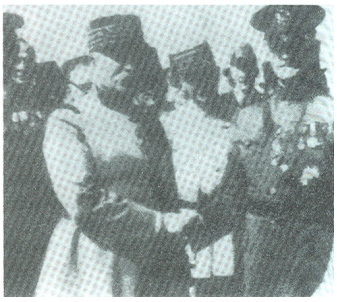
Resim 2.1. İlk dijital görüntülerden biri

İlk Bartlane sistemleri grinin 5 ayrı tonunda görüntü kodlayabiliyordu. Bu kapasite 1929'da 15 farklı gri tona yükseldi. Resim 2.3 15 farklı gri tonun kullanıldığı tipik bir görüntüdür. Bu dönemde, çoğaltma işlemiyle geliştirilen kodlanmış resim şeridiyle düzenlenmiş ışık sinyalleri aracılığıyla bir film klişesi geliştirmek için sistem gelişti.



Resim 2.2. 1922 de yapılmış bir dijital resim

Örnekler dijital görüntü örneği olsalar da, burada tanımlanan bağlamda dijital görüntü işleme sonuçları olarak sayılmamışlardır, çünkü onların bilgisayar kayıtları yoktu. Bundan dolayı dijital görüntü işleme tarihi, dijital bilgisayarların gelişmesine bağlanmıştır. Dijital fotoğrafların çok fazla disk alanı ve hesaplama gücü gerektirmesi sebebiyle dijital fotoğraf işleme alanındaki gelişmeler, dijital bilgisayarların yapılmasını ve bilgi saklama, görüntüleme ve aktarma gibi destek teknolojilerinin geliştirilmesine bağlıdır.



Resim 2.3. 1929 da 15 farklı gri tonun kullanıldığı dijital resim

Bugünkü bilgisayarın temeli son iki yüz yıllık gelişmelerle oluşturulmuştur. Ama modern dijital bilgisayarların tarihi John Van Nevmann'ın program ve bilgi saklamak için disk alanı ve şartlı dallanma gibi iki ana içeriği tanıtmasıyla, 1940'lara kadar gider. Bu iki fikir bugün bilgisayarın kalbi olan Merkezi İşlem Biriminin (CPU) temelini oluşturur. Bu gelişmeler kısaca şu şekilde özetlenebilir:

Transistörlerin 1948 yılında Bell laboratuvarında icat edilmesi, 1950 ve 1960'larda COBOL (Comman Business Oriented Languages) ve FORTRAN (Formula Translator) gibi yüksek seviyeli programlama dillerinin geliştirilmesi. 1958 de Texas instrument's'da bütünleşmiş devrelerin icadı. 1970'lerin başında Intel tarafından mikro işlemcilerin geliştirilmesi (merkezi işlem birimi, hafıza ve giriş-çıkış kontrolleri içeren tek bir yonga). IBM tarafından 1981 de kişisel bilgisayarların yapılması. Geniş ölçekli bütünleşmeyle birlikte, bileşenlerdeki minyatürleşme 1970'lerin sonlarına doğru başlamıştır[24].

İlk bilgisayarlar 1960'ların başında görüntü işleme görevi için yeterince güçlüydü. Dijital görüntü işleme olarak adlandıran şeyin doğrusu bugün bu makinelerin ve bu periyotta uzay programlarının kabiliyetiyle takip edilebilir.

Görüntünün geliştirilmesi için bilgisayar tekniklerinin kullanılması, 1964'te, Jet Progulsian Laboratuvarında, Ranger7 tarafından gönderilen fotoğrafların televizyon kamerasından kaynaklanan bozulmaların düzenlenmesiyle olmuştur. Resim 2.4 Ay'ın Ranger7 tarafından 31 Haziran 1964 saat 9:09 da çekilen fotoğrafını göstermektedir. Ranger7 ile öğrenilen görüntüleme dersleri Ay'a yapılan Surveyorlar, Mars'a yapılan Mariner uçuşları ve Ay'a yapılan insan uçuşlarının

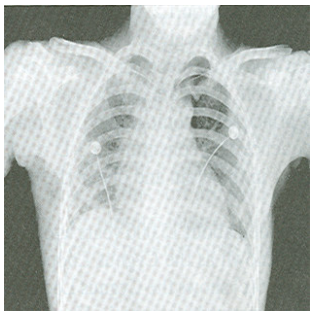
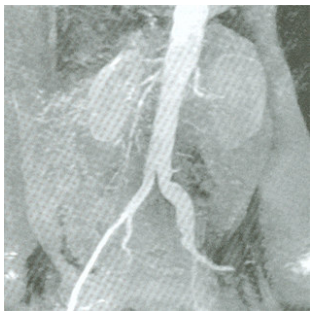
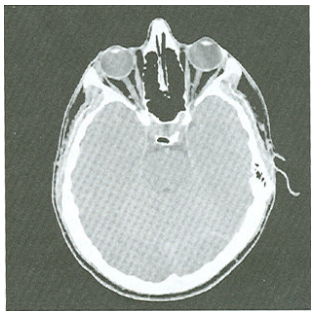
görüntülerinin işlenmesi ve saklanması metotlarının geliştirilmesinin temelini oluşturur. Dijital görüntü işleme teknikleri uzay uygulamalarının paralelinde 1960'ların sonlarında ve 1970'lerin başlarında tıbbi uygulamalarda, uydudan Dünya kaynaklarının gözlenmesi ve astronomide kullanılmaya başlanmıştır.



Resim 2.4. Ay'ın Ranger7 tarafından 31 Haziran 1964 saat 9.09 da çekilen fotoğrafı

1970'lerin başında, kısaca bilgisayarlı tomografi (CT) olarak adlandırılan ve tıp alanında görüntü işleme uygulamalarının en önemlisi, bilgisayarlı eksenli tomografinin icat edilmesidir. Bilgisayarlaştırılmış eksenli tomografi, bir dedektörün bir nesneyi (veya hastayı) sarması (kontrol etmesi) ve X ışınlarının dedektörle eş merkezli olarak bir prosedür dâhilinde nesne etrafında dönmesi sürecidir. Dedektör çalışır vaziyetteyken x ışınları nesnenin içinden geçer ve karşı tarafta toplanır. X ışın kaynağı dönerken, bu prosedür tekrar edilir. Tomografi, objenin bir kısmını algılayıp görüntü oluşturmak için algoritmalar içerir[24]. Nesnenin dedektöre dikey yöndeki hareketi objenin üç boyutlu (3D) temsilini oluşturan, bir dizi parçalar üretir.

Tomografi, 1979 Nobel Tıp Ödülü'nü bu icatlarıyla paylaşan Sir Godfrey N. Hounsfield ve Prof. Allan M. Gormoct tarafından icat edilmiştir. X ışınlarının 1901 Nobel Fizik Ödülü'nü alan Wilhelm Conrad Roentgen tarafından icat edildiğini not etmek de ilginçtir[24].

a)  b)  c) 

Resim 2.5. X ışını görüntüleme örnekleri. a) Göğüs X ışını. b) Atardamar anjiyosu c) Kafa tomografisi

Yaklaşık yüz yıl arayla gerçekleşen bu buluşlar bugün görüntü işleme alanında çok aktif uygulamaların yapılmasına olanak sağladı. Uzay ve tıp alanındaki uygulamalara ek olarak dijital görüntü işleme teknikleri, uygulamaların çoğu alanında şimdi kullanılmaktadır.

X ışınları, endüstri, tıp ve diğer biyoloji bilimlerinde görüntü yorumunu kolaylaştırmak için renk düzeylerindeki zıtlık ve kod yoğunlukları, bilgisayar prosedürleri kullanılarak geliştirildi.

Coğrafyacılar havadan ve uydudan alınan görüntülerde kirlilik numuneleri üzerinde çalışmak için aynı veya benzer teknikleri kullandılar. Görüntü geliştirme ve onarma prosedürleri kopya oluşturmak için deneysel sonuçları çok pahalı olan ve kopyalanamayan obje görüntülerinin küçültme işleminde kullanıldı. Fizikte ve bağlantılı alanlarda, görüntü alanındaki bilgisayar teknikleri yüksek enerji plazmaları ve elektron mikroskobu gibi alanlarda rutin bir şekilde gelişti. Benzer olarak görüntü işleme içeriğindeki başarılı uygulamalar astronomi, biyoloji, nükleer tıp, hukuk, savunma ve endüstriyel uygulamalarda da bulunmaktadır[24]. Bu örnekler insanların yorumu için işleme sonuçlarını açıklar.

Tarım alanında görüntü işleme teknolojileri geliştikten sonra, doğal geçiş gerçek zamanlı uygulama alanı haline gelmiştir. Gerçek zamanlı uygulamalara geçişi başarabilen birkaç makine görme sistemi vardır. Normal çevre koşulları altında, sıranın merkezini saptayabilen, gerçek zamanlı bir rehberlik sistemleri geliştirilebilmiştir. Bu rehberlik sisteminde, görme gücü, tohum kanalının yerini

saptayabilmiş ve ardından panel yana doğru hareket ettirilerek o anki durum ile istenilen durum arasında bir denge sağlanabilmiştir[26].

Dijital görüntü uygulama tekniklerinin ikinci ana uygulama alanı problemlerin çözümünün makine algılamasıyla ilgilidir. Makine algılamasında kullanılan görüntü işleme tekniklerindeki bazı problemler, otomatik karakter tanıma, ürün düzenleme ve denetimi için endüstriyel makine tasarımı, askeri görüntülemeler, parmak izi otomatik işleme, x ışınlarının ve kan örneklerinin görüntülenmesi, çevre düzenleme ve hava tahmini için anten ve uydudan alınan sinyallerin makine işlemesi gibi sınırlandırılabilir[22].

Günümüzde alternatif insan makine etkileşimi tekniklerinden biri de bilgisayarlı görme tekniklerine dayalı sistemlerdir. Örnek olarak bilgisayarlı görmeye dayalı yazı girme sistemleri hem bugünkü bilgisayarlarda kullanılabildikleri gibi hem de giyilebilir bilgisayarlar için geleneksel girdi cihazlarının yerine daha esnek ve çok yönlü alternatifler oluşturmaktadırlar. Bu tekniklerin kullanılması sayesinde klavye, fare, vb. geleneksel sistemlerden daha esnek ve daha kullanışlı sistemler geliştirmek mümkün olmuştur. Bu tür sistemler genellikle insan el ve yüz hareketlerin tanımlanması üzerine kurulmuştur[27].

Yine kara mayınlarının toplanması konusunda uluslararası boyutta çalışmalar yürütülmektedir. Bu amaca yönelik mayın tarama robotları üzerine ülkelerin devam etmekte olan görüntü işlemeye dayalı çalışmaları vardır. Mayın tarama robotunun bulunduğu ortamı algılaması ve yörünge planlamasına hedef nokta vermesi için görme gerekmektedir. İki kamera ve frame grabber kartlarıyla, üç boyutlu görüntü işleme mümkündür. Görüntü bilgisi uzaktan kumanda merkezine de gönderilerek, merkezin ortamı algılaması ve karar vermesini kolaylaştırması sağlanır[28].

Bilgisayar tarafından algılanan görüntü, görüntü işleme teknikleri ile analiz edilir. Bir sonraki bölümde dijital görüntünün işlenmesi ve depolanması üzerinde durulacaktır.

3. GÖRÜNTÜ İŞLEME

3.1. Renkli Görüntü İşleme

Renkli görüntü işleme yöntemleri dört ana kategoriye ayrılır. Birinci kategoride, her bir görüntüyü tek başına işler daha sonra tek tek işlenen bileşenlerden oluşan işlenmiş renkli görüntü oluşturulur. İkinci kategoride, doğrudan renk pikselleri ile çalışılır. Renkli görüntüler en az üç bileşene sahip olduğundan, renkli pikseller aslında vektörlerdir. Örneğin, RGB(Kırmızı-Yeşil-Mavi) sisteminde, her bir renkli nokta RGB koordinat sisteminde merkezden o noktaya uzanan bir vektör olarak yorumlanabilir.

"c" RGB renk alanında gelişigüzel bir vektör olsun:

$$c = \begin{bmatrix} c_R \\ c_G \\ c_B \end{bmatrix} = \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{3.1}$$

Denklem c'nin bileşenlerinin renkli bir görüntünün bir noktadaki RGB bileşenleri olduğunu göstermektedir. Renk bileşenlerinin *(x,y)* koordinatlarının fonksiyonu aşağıdaki gibidir:

$$c(x, y) = \begin{bmatrix} c_R(x, y) \\ c_G(x, y) \\ c_B(x, y) \end{bmatrix} = \begin{bmatrix} R(x, y) \\ G(x, y) \\ B(x, y) \end{bmatrix} \tag{3.2}$$

M x *N* boyutundaki bir görüntü için, şu *MN* vektörleri mevcuttur, c*(x, y), x* = 0, 1, 2, …, *M* – 1; *y* = 0, 1, 2, …, *N* – 1.

x ve *y* de formüle edilen görüntü işleme teknikleriyle ilgilenilmektedir. Piksellerin artık renkli pikseller olduğu gerçeği en basit formülle, standart gri ölçekli görüntü işleme yöntemleri kullanılarak, bileşen görüntülerinin her birini ayrı ayrı işleme yoluyla renkli bir görüntünün işlenmesini sağlayan bir faktör ortaya çıkarır.

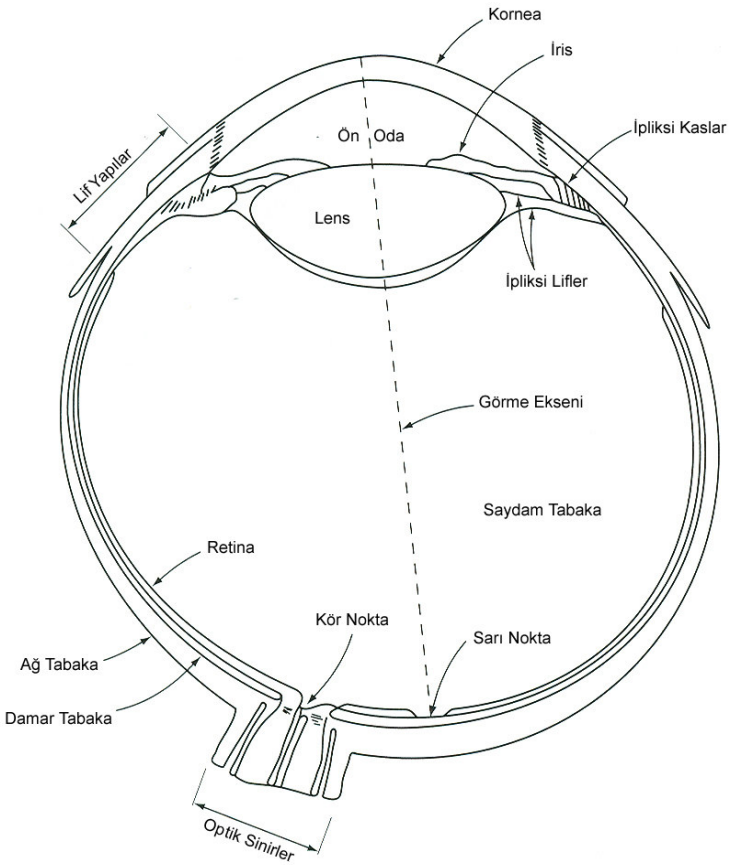
Renkli bileşen ve vektör esaslı işlemenin bir birine eşit olması için, iki şartın karşılanması gerekir: Birincisi, işlemin hem vektörlere hem de sayılara uygulanabilmesi gerekir. İkincisi, bir vektörün her bir bileşeni üzerindeki işlemin diğer bileşenlerden bağımsız olması gerekir. Ortalama, komşu piksellerin gri seviyeleri toplanarak ve komşu piksellerin toplam sayısına bölünerek elde edilir[29].

Ortalama, bütün komşu vektörler toplanıp her bir bileşen komşu vektörlerin toplam sayısına bölünerek hesaplanır. Fakat ortalama vektörün her bir bileşeni, ortalamanın tek renk bileşeni temelinde yapılarak, vektörün oluşturulması ile elde edilen sonuçla aynı olacak olan, bu bileşene karşılık gelen görüntüdeki piksellerin toplam sayısına eşittir[30].

3.1.1. Renkli görme

Çoklu spektral görüntünün en yakın formu normal renkli görmedir. İnsan gözünün retinası, fonksiyon olarak CCD yongadaki piksellere benzeyen foto reseptör hücrelerle kaplanmıştır. Foto reseptör hücreler lens ve kornea tarafından retinada odaklanmış görüntüden ışığı emer. Optik sinirler de yaklaşık bir milyon fiber yoluyla beyne giden sinir sinyali üretirler. Bu sinyallerin sıklığı ışığın parlaklığını belirtir[24].

Foto reseptör hücreler çubuk ve koni olarak iki tipin karışımıdır. Bu isimleri fiziki görüntülerinden almışlardır. Çubuk hücreler daha hassastır ve yüksek ışık hassasiyetiyle tek renkli gece görüşü sağlarlar[31]. Koniler sadece yüksek ışık seviyesinde renkli görmeyi sağlayabilirler. Koniler üç tipte görülür, farklılaşma çoğunlukla fotokimyadadır. İşleri ışığı sinir darbelerine dönüştürmektir. Koniler, kırmızı, yeşil ve mavi üç bantta elektromanyetik spektrumun görünür parçasını ayırır. Bu nedenden, bu üç renk insan görmesinin birincil renkleri olarak ifade edilir[22].

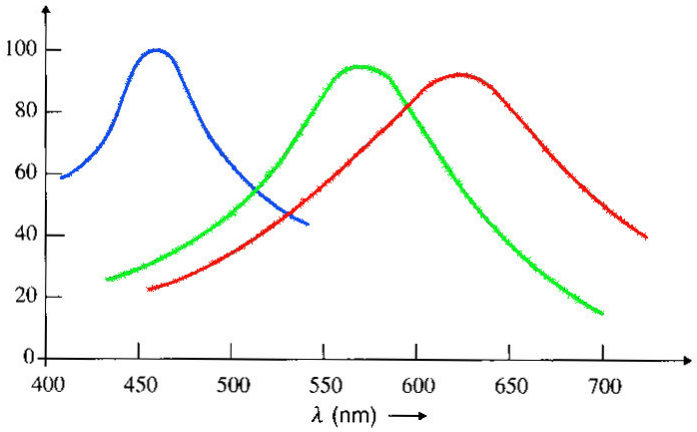


Şekil 3.1. Üstten insan gözü çizgisel kesiti

Sinir sinyalleri gangliyon hücre tabakası ve kutupsuz hücre tabakaları içinden geçen ışık karşılığında foto reseptör hücreler tarafından üretilir. Bir milyondan fazla gangliyon hücresi beyne görüntü verisi taşır[24]. İnsan fotoreseptör hücrelerinin hassaslık spektrumu Şekil 3.2 de gösterildiği gibi oluşmaktadır.

3.1.2. Üç renk görüntüleme

İnsan görme sisteminin doğası gereği, elektronik görüntülemede, ürün geliştirme emek ve harcamalarının çoğu üç renkli sistemlere adanmıştır. Bu sistemler arasında özellikle televizyon kameraları, sayısallaştırıcılar, görüntüleyiciler ve yazıcılar sayılabilir. Bu yüzden üç renk model özel bir önem arz eder. Bir üç renk uygulaması yalnızca renkli görüntü analizi yapmaz aynı zamanda üç renk donanımı üzerinde nicel renkli görüntü analizi yapar. Bu yüzden yüksek miktarları düşük maliyetlerle üretilir[31].



Şekil 3.2. İnsan fotoreseptör hücrelerinin hassaslık spektrumu

Görüntü işlemede renk kullanımı iki temel etmen tarafından yönlendirilir. Birincisi, renk genellikle nesne tanımlamayı ve bir bütün içinden çıkartmayı basitleştiren güçlü bir tanımlayıcıdır. İkincisi, insanlar sadece birkaç düzine gri tonlarına karşılık binlerce renk tonunu ve yoğunluğunu ayırt edebilirler. Bu ikinci etmen özellikle manuel (yani insanlar tarafından gerçekleştirilen) görüntü analizinde önemlidir[31].

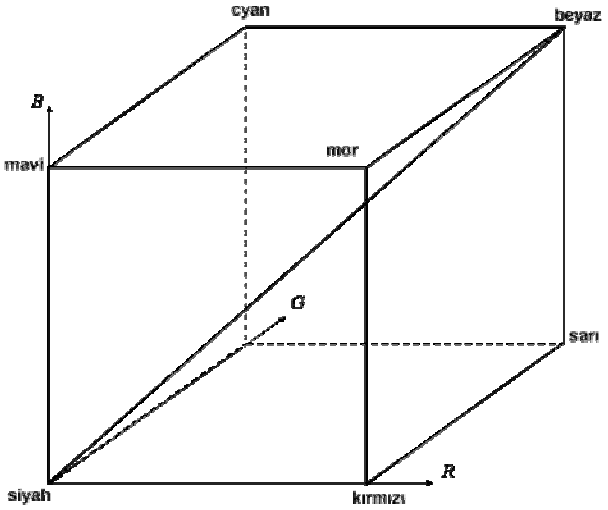
Renkli görüntü işleme iki ana bölüme ayrılır: *tam renk* ve *yarı renk* işleme. Birinci kategoride, söz konusu renkler tipik olarak renkli TV kamerası ya da renkli tarayıcı gibi bir tam renkli algılayıcı ile elde edilir. İkinci kategoride, problem belirli bir monokrom yoğunluk ya da yoğunluk aralığına bir renk atanmasıdır.

Üç renk görüntüleme sistemlerinin yaygın örnekleri içinde renkli fotoğrafçılık ve renkli televizyon yer alır. İki durumda da görülebilir spektrum üç banda –Kırmızı, Yeşil ve Mavi- bölünür. Spektrum miktarının ayarlanması insan gözü tarafından gerçekleştirilir. Tüm bu yapımlar retina üzerinde orijinal görünüm olarak aşağı yukarı aynı etkiyi bırakır. Böylece normal bakış oluşur.

3.1.3. Renk belirtme

RGB (Kırmızı-Yeşil-Mavi) format: Bir rengin nicel olarak belirtilebilmesi için renkli dijital görüntüde piksel gibi birkaç yol vardır. En açık yol, kırmızı, yeşil ve mavinin parlaklık değerlerinin kullanılması, aralarında örneğin 1 ve 0 gibi derecelendirilmeleridir. Bu uzlaşmaya RGB format denir. Her piksel gerçektir, her

rengi görüntülemek mümkündür. Şekil 3.3'teki renk küpünde gösterildiği gibi üç alanın ilk çeyreğindeki bir nokta ile gösterilebilir. RGB alanındaki noktaların dağılımı üç renkli görüntünün gri yüzey histogramıdır[22].



Şekil 3.3. Dikdörtgen renk aralığı

RGB renk aralığının kaynağı ana renklerden birinin parlaklığının sıfır olması ile gösterilir ve bu siyahtır. Üç ana rengin tam parlaklığı beyaz olarak görünür. Üç renk bileşeninin eşit miktarındaki daha az parlaklık grinin gölgesini oluşturur. Renk küpünün köşegeninin tüm bu noktalarına düşen geometrik yer gri çizgi olarak isimlendirilir. Renk küpünün üç köşesi –kırmızı, yeşil ve mavi- ana renklere uyar. Kalan üç köşe, sarı, cyan ve mor'dur[24].

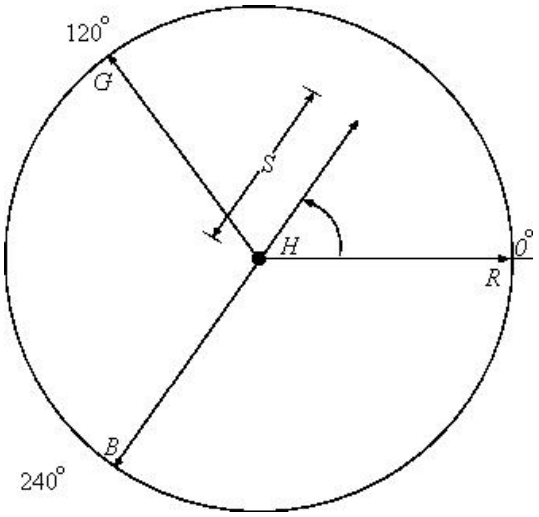
HSI (Ton-Doygunluk- Yoğunluk) format: Bir diğer kullanışlı yapı HSI format olarak adlandırılan, çoğunlukla sanatçılar tarafından kullanılan ve Munsell tarafından geliştirilen renk sisteminin formülasyonudur[24].

HSI formatın tasarımı insanın gördüğü rengi yansıtmanın yoludur ve görüntü işlemenin iyi olabilmesi için avantajlar sunar.

HSI formatta "I" yoğunluk ya da parlaklık anlamı için kullanılır. Yoğunluk değerinde rengin ne olduğuna bakılmaksızın pikselin toplam parlaklığı açıkça belirtilir. Böylece renk bilgisi atılmak suretiyle, renkli görüntüyü RGB bileşenlerinin

hepsinin ortalamasını alarak tek renge dönüştürebilir. Renk bilgisini içeren iki parametre (H-Hue) Ton ve (S- Saturation) doygunluk olmasına rağmen, eşdeğer terimler de bazen kullanılmaktadır.

Bu iki parametre Şekil 3.4'teki renk çemberinde gösterilmiştir. Ton bir açı olarak gösterilmiştir. Bir rengin tonu spektral dalga boyuna en yakın benzerliktir. (Örn: Gökkuşağının renkleri) Rastgele seçilen ton değerleri 0° kırmızı, 120° yeşil ve 240° mavidir. Ton 0° den 240° ye giden görünür spektrumun renkleri üzerinden geçer. 240° ile 360° arasına spektral olmayan (mor) renkler düşer. Bu göz algısıdır[22].



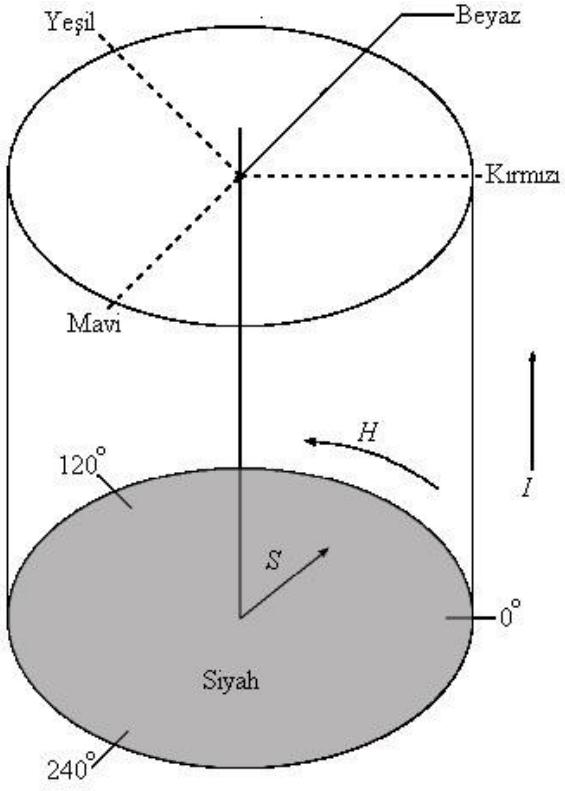
Şekil 3.4. Renk çemberi

Doygunluk parametresi renk çemberinin merkezinden noktanın yarıçapıdır. Yaklaşık dairenin çevresine düşen saf ya da doygun renkler ve onların doygunluk değerleri aynıdır. Ortada bulunan gri gölge sıfır doygunluk oranındadır[32].

Doygunluk kavramı şu şekilde ifade edilebilir. Birlikte alınan üç rengin koordinatları silindirik renk aralığında (Şekil 3.5) tanımlanır. Gri gölgeler tabandaki siyahtan, tepedeki beyaza eksen boyunca düşer. Tam doygun ve tam parlak renkler dairesel yüzeyin çevresine düşer[22].

Kullanılan diğer birçok renk koordinat sistemi vardır. Bunlar belki de en yaygın kullanılan renk ve ışık için uluslararası bir standart komitesi "Commission

Internationale de l'Éclairage (CIE)" tarafından kurulmuştur. Komite standardizasyon amacıyla, 1931'de üç temel renge şu özel dalga boylarını ayırmıştır: mavi = 435,8 nm, yeşil =546,1 nm ve kırmızı = 700 nm. Bunlar insan gözlemleri üzerine yapılan renk seçme deneyimlerinden elde edilen deneysel verilere dayandırılır[24].



Şekil 3.5. Silindirik renk uzayı

3.1.4. Renk koordinat çevrimi

Dijital görüntü işlemede RGB ve HSI formatları arasında geçiş yapabilmek faydalıdır. Bazı işlemler doğal olarak bir sistemde yapıldığında diğer renk sistemine göre daha iyi sonuç verirler.

RGB den HSI sistemine dönüşüm: RGB den HSI ye dönüşüm şu şekilde açıklanabilir. Gri çizgisi RGB aralığında renk küpünün köşegeni, silindirik HSI düzleminde ise dikey eksendir. RGB küpünün köşegen z ekseni ve R ekseni xz ekseni üzerinde olacak biçimde bir dönüşüm Şekil 3.6'da görülmektedir[22].

$$x = \frac{1}{\sqrt{6}}[2R - G - B] \quad y = \frac{1}{\sqrt{2}}[G - B] \quad z = \frac{1}{\sqrt{3}}[R + G + B] \tag{3.3}$$

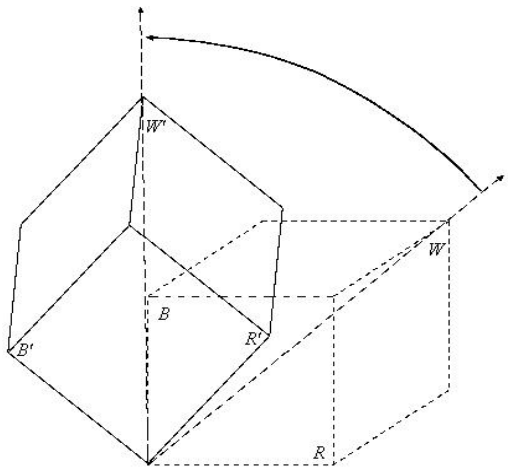


Şekil 3.6. Renk skalası

Daha sonra xy-düzlemi üzerinde polar koordinat tanımlanarak silindirik düzleme geçilir.

$$\rho = \sqrt{x^2 + y^2} \quad \phi = ang(x, y) \tag{3.4}$$

Burada (x,y) açısı merkezden (x,y) noktasına çizilen çizgiyle x ekseni arasındaki açıdır. Bu temelde tanjant yayı (Şekil 3.7) olup noktanın hangi çeyrekte olduğu önemlidir. Şimdi (Φ,ρ,z)'in (HSI) ye denk olduğu silindirik koordinatlar bellidir[24].



Şekil 3.7. RGB küpü Dönüşümü

Tam doygun renkler (iki renkten daha fazla ana renk içermeyenler) xy-düzlemi üzerinde bir daireden ziyade bir altıgen (Şekil 3.8 (a)) üzerinde yer alır. Bunun çözümü, ρ değerini üzerinde olduğu Φ açısındaki en büyük değerine bölerek normalleştirmektir. Bu doygunluk formülü;

$$S = \frac{\rho}{\rho_{\max}} = 1 - \frac{3\min(R,G,B)}{R+G+B} = 1 - \frac{\sqrt{3}}{I}\min(R,G,B) \tag{3.5}$$

şeklindedir. Tam doygun renkler artık x,y düzlemindeki birim çember üzerindedir(Şekil 3.8 (b)).

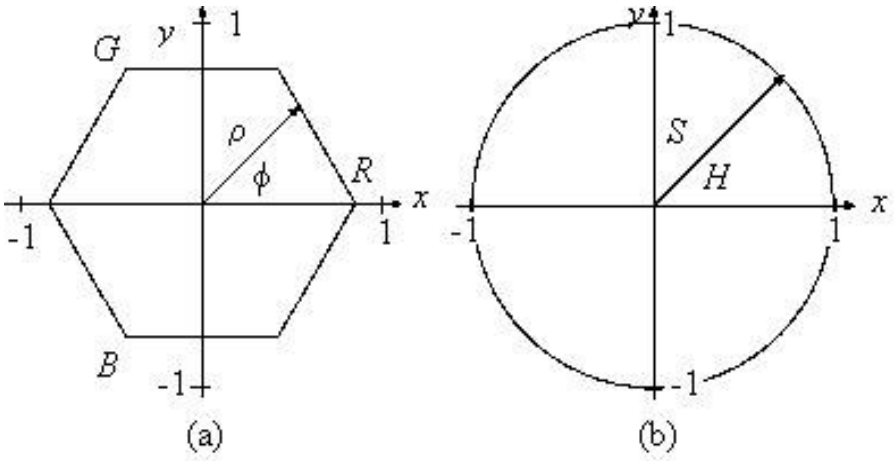
Ton denklemdeki Φ olarak alınabilse de açıyı hesaplamak için eşdeğer bir yöntemde

$$\theta = \cos^{-1}\left[\frac{\frac{1}{2}\left[(R-G)+(R-B)\right]}{\sqrt{(R-G)^2+(R-B)(G-B)}}\right] \tag{3.6}$$

ve ton

$$H = \begin{cases} \theta & G \geq B \\ 2\pi - \theta & G \leq B \end{cases} \tag{3.7}$$

Olur[24].



Şekil 3.8. x,y düzlemindeki renk uzayı

HSI den RGB sistemine dönüşüm: HSI den RGB ye dönüşümde kullanılan formüller döndürülen noktanın renk çemberinde bulunduğu bölüme göre az derecede değişir.

$0^\circ \le H < 120^\circ$ İçin,

$$R = \frac{I}{\sqrt{3}}\left[1 + \frac{S\cos(H)}{\cos(60^\circ - H)}\right] \quad B = \frac{I}{\sqrt{3}}(1-S) \quad G = \sqrt{3}I - R - B \tag{3.8}$$

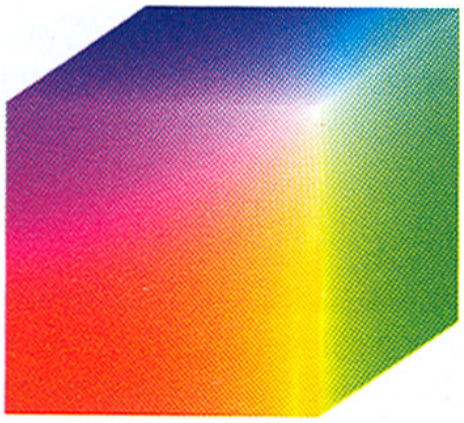
iken 120°≤H<240° için,

$$G = \frac{I}{\sqrt{3}}\left[1 + \frac{S\cos(H - 120^\circ)}{\cos(180^\circ - H)}\right] \quad R = \frac{I}{\sqrt{3}}(1-S) \quad B = \sqrt{3}I - R - G \tag{3.9}$$

ve 240°≤H<360° için,

$$B = \frac{I}{\sqrt{3}}\left[1 + \frac{S\cos(H - 240^\circ)}{\cos(300^\circ - H)}\right] \quad G = \frac{I}{\sqrt{3}}(1-S) \quad R = \sqrt{3}I - G - B \text{ dir.} \tag{3.10}$$

Uygulanabilir denklemler H değerlerine bağlıdır. Temel renklerin ayrıldığı 120^o aralıklara karşılık gelen üç ilgili sektör vardır (Şekil 3.8). H'yi 360^o ile çarparak başlanır, bu ton orijinal [0^o, 360^o] aralığına dönüştürür[24]. HSI çevriminin birçok farklı yöntemi vardır[22].



Şekil 3.9. RGB 24 bit renk küpü

Şekil 3.9 RGB değerleri için ton, doygunluk ve yoğunluk görüntülerini göstermektedir. En belirgin özelliği küpün ön (kırmızı) yüzündeki 45^{o}'lik çizgi değeri boyunca olan süreksizliktir. Küpün kırmızı köşesinden beyaz köşesine bir çizgi olduğu kabul edilirse ve bu çizginin ortasında bir nokta seçildiğinde, bu noktadan başlayarak, başlangıç noktasına geri gelinceye kadar küpün etrafında sağa doğru olduğu varsayılan diğer çizgi karşılaşılan ana renkler sarı, yeşil, cyan, mavi, magenta ve tekrar kırmızıdır. Bu yol boyunca ton değerleri 0°den 360°'ye kadar artar (yani mümkün olan en düşük ton değerinden en yüksek ton değerine kadar). Aslında, ton görüntüsü (0,1) aralığında normalleştirilmiş ve daha sonra 8 bit'e göre ölçeklendirilir: Yani, görüntü için (0, 255) aralığına dönüştürülür[24].

3.2. Renk Dengesi

Renkli bir görüntü sayısallaştırıldıktan sonra görüntülendiğinde doğru görünmeyecektir. Farklı hassasiyetler, kazanç faktörleri, ofsetler (siyah seviyesi) üç ana bileşende farklı doğrusal dönüşümlere neden olur. Sonuçta tam bileşenlerin dengesinin kaymış olduğu bir görüntü ortaya çıkar[21].

Renk dengesi kontrolünde yapılan ilk işlem tüm grilerin, gri görünüp görünmediğine bakmaktır. İkinci olarak yüksek doygunluk değerine sahip renklerin tonlarının doğru olup olmadığına bakılır. Eğer görüntü baskın oranlarda siyah veya beyaz renk içeriyorsa bu RGB bileşen histogramlarında fark edilir bir tepe oluşturur. Bu tepelerin farklı gri seviyelerinde olması renk dengesizliğini gösterir. Bu tip renk

dengesizliğinin çözümü için RGB bileşenlerinin her biri üzerinde doğrusal dönüşümler yapılır. Eğer seçilen iki bölgenin gri seviyeleri her üç renk bileşeni için aynı ise renk dengesi sağlanmış demektir[33].

3.2.1. Zıtlık ve iyileştirme

Üç renkli dijital bir görüntünün RGB bileşenleri ile çalışırken renk dengesi bozulabilir. HSI formatında bir resmin yoğunluk bileşenine uygulanırsa oldukça iyi sonuçlar verir. Birçok yönden yoğunluk bileşeni tek renk bir resim gibi kabul edilebilir. Ton ve doygunluk bileşenlerinde renk bilgisi bu durumda sorun çıkarmaz. Herhangi bir geometrik işlem RGB ve HSI formatında her üç bileşen için de aynı uygulanır[15].

Doygunluk İyileştirmesi: Bir görüntünün daha koyu olması için doygunluk bileşeni 1 den büyük bir sayı ile çarpılır. Benzer şekilde 1 den büyük bir sayı ile çarpılması da görüntünün renginin açılmasına neden olur. Doygunluk görüntüsünde 0 noktasına dönüşüm fonksiyonu 0 olan doğrusal olmayan bir dönüşümde kullanılabilir.

Ton Kaydırma: Ton bir açı olduğu için bunu değiştirmek için her piksele bir sayı eklemek mantıklı olur. Bu her objenin rengini gökkuşağı üzerinde renginin aşağı veya yukarı kayması anlamına gelir. Eklenen veya çıkartılan açı sadece birkaç derece ise bu işlem görüntüyü soğutur veya ısıtır. Daha büyük açılar şiddetli bir şekilde görüntüyü değiştirir. Ton bir açı olduğundan, ton bileşeninde yapılan işlemler, gri skalanın periyodik olduğunu dikkate almalıdır. Örneğin 8 bitlik bir görüntü için 255+1= 0 ve 0- 1= 255 tir[24].

3.3. Renk Temelleri

İnsan beyninin, renklerin algılanması ve yorumlanmasında takip ettiği işlemler henüz tam olarak anlaşılamamış fizyo-psikolojik bir fenomen olmasına rağmen, rengin fiziksel yapısı deneysel ve teorik sonuçlar tarafından desteklenen, biçimsel bir esasa bağlı olarak ifade edilebilir.



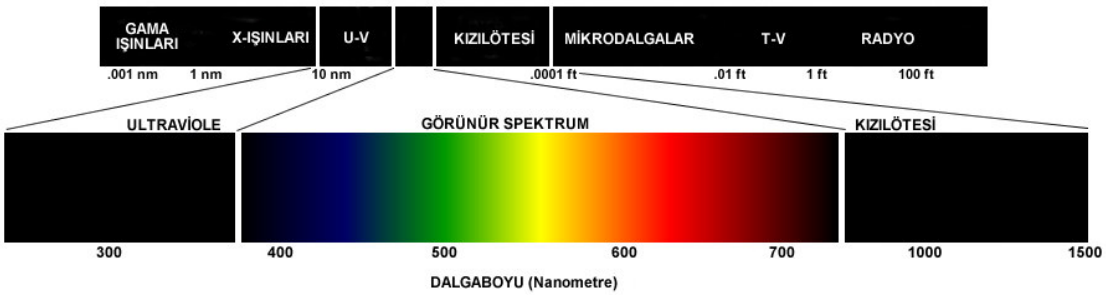
Şekil 3.10. Renk spektrumu

1666'da Newton bir güneş ışığı demeti cam bir prizma içerisinden geçtiğinde, ortaya çıkan ışık demetinin beyaz değil fakat bir uçta mor renkten diğer uçta kırmızı renge uzanan sürekli bir renk spektrumundan oluştuğunu keşfetmiştir. Şekil 3.10'da görüldüğü gibi, renk spektrumu altı genel bölgeye ayrılır: mor, mavi, yeşil, sarı, turuncu ve kırmızı. Tam renkli olarak bakıldığında spektrum içindeki renklerin hiç birisi birden bire sona ermez, fakat daha ziyade her bir renk bir sonrakine hafifçe karışır.

Temel olarak, insanların ve bazı başka hayvanların bir nesnede algıladıkları renkler, nesneden yansıtılan ışığın yapısı tarafından belirlenir. Şekil 3.10'da gösterildiği gibi, görülen ışık elektromanyetik spektrum içinde nispeten dar bir frekans kuşağından oluşmaktadır.

Bütün görünür dalga boylarının dengede olduğu ışık yansıtan bir cisim gözleyiciye beyaz görünür. Fakat sınırlı bir görünür spektrum aralığında yansıtma gösteren bir cisim bazı renk tonlarını gösterir. Örneğin, yeşil nesneler temel olarak 500 ila 570nm aralığındaki dalga boylarına sahip ışığı yansıtırken diğer dalga boylarındaki enerjinin çoğunu soğururlar[24].

Işığın karakterinin belirlenmesi renk biliminde öneme sahiptir. Şayet ışık akromatik (renksiz) ise, tek özniteliği *yoğunluğu* ya da miktarıdır. Akromatik ışık, izleyicilerin siyah beyaz bir televizyon ekranında gördükleri şeydir. *Gri ton* terimi siyahtan gri tonlara ve son olarak beyaza kadar uzanan sayısal bir yoğunluk ölçümünü ifade etmektedir[33].



Şekil 3.11. Elektromanyetik spektrumun görünür kısmının dalga boyu karşılaştırması

Kromatik ışık yaklaşık 400 ila 700 nm arasındaki elektromanyetik spektrum arasında uzanır. Kromatik ışık kaynağının niteliğini tanımlamak için üç temel nicelik kullanılmaktadır: Işınım (radiance), ışıklılık (luminance) ve parlaklık[24]. *Işınım* ışık kaynağından çıkan enerjinin toplam miktarıdır ve genellikle watt (W) cinsinden ölçülür. Lümen (lm) olarak ölçülen *Işıklılık,* bir gözlemcinin ışık kaynağından algıladığı enerji miktarının ölçümünü verir.

Örneğin, spektrumun uzak kızılötesi bölgede bulunan bir ışık kaynağından yayılan ışık büyük miktarda enerjiye (radyans) sahip olabilir, fakat bir gözlemci bunu zorlukla algılayabilir, ışıklılık neredeyse sıfır olabilir. Son olarak, *parlaklık* ölçülmesi pratik olarak imkânsız olan öznel bir açıklayıcıdır.

Temel kelimesinin kullanımı genellikle, bu üç standart temel renk çeşitli yoğunluk oranlarında karıştırıldığında *bütün* görünür renkleri üretebildikleri şeklinde yanlış yorumlanmaktadır[33].

Ton ve doygunluk bir arada türsellik (chromaticity) olarak adlandırılır ve bu nedenle, bir renk, parlaklığı ve türselliği ile karakterize edilebilir. Belirli bir rengi elde etmek için gerekli olan kırmızı, yeşil ve mavi miktarları *üç türsel (tristimilus)* değerler olarak adlandırılır ve sırasıyla *X, Y ve Z* ile gösterilir. Bu durumda bir renk;

$$x = \frac{X}{X + Y + Z} \tag{3.11}$$

$$y = \frac{Y}{X + Y + Z} \text{ ve} \tag{3.12}$$

$$z = \frac{Z}{X + Y + Z} \tag{3.13}$$

şeklinde tanımlanan üç türsel katsayı (trichromatic coefficients) belirlenir.

Yukarıdaki denklemlerden:

$x + y + z = 1$ elde edilir. (3.14)

Çeşitli spektrum renklerinin – 380nm'de mordan 780nm'de kırmızıya kadar – konumları türsellik diyagramın sınırları üzerinde gösterilir. Bunlar Şekil 3.11 spektrumunda gösterilen arı renklerdir. Sınırda olmayan fakat diyagram içinde bulunan herhangi bir nokta spektrum renklerinin bir karışımını temsil eder[15].

3.4. Renk Modelleri

Bir renk modelinin amacı genel olarak kabul edilen şekliyle bir standartta renklerin belirlenmesini kolaylaştırmaktır. Özü itibariyle bir renk modeli, bir koordinat sistemi tanımlaması ve bu sistem içinde her bir rengin bir nokta ile temsil edildiği bir alt alandır[32].

Günümüzde kullanılmakta olan çoğu renk modeli, donanım (renkli monitörler ya da yazıcılar gibi) ya da renk manipülasyonunun bir amaç olduğu uygulamalara (renkli animasyon grafiklerinin oluşturulmasında olduğu gibi) yöneliktir. Dijital görüntü işleme açısından uygulamada en yaygın kullanılan donanım yönelimli modeller, renkli monitörler ve geniş bir sınıf olan renkli video kameralardır. Bunlar için RGB (kırmızı, yeşil, mavi) modeli, renkli baskı için CMY (cyan, magenta, sarı) ve CMYK (cyan, magenta, sarı, siyah) modelleri ve insanların renkleri tanımladıkları ve yorumladıkları şekle oldukça yakın olan HSI (ton, doygunluk, yoğunluk) modelidir[22]. HSI modelini gri ölçek tekniklerinin pek çoğu için uygun hale getiren, bir görüntüdeki renk ve gri ölçek bilgisini ayırma avantajına sahip olmasıdır.

Renk bilimi pek çok uygulama alanını içine alan geniş bir alan olması nedeniyle günümüzde kullanılan çok çeşitli renk modelleri vardır.

3.4.1. RGB renk modeli

RGB modelinde, her bir renk temel kırmızı, yeşil ve mavi bileşenleri ile görünür. Bu model bir kartezyen koordinat sistemine dayanır.

RGB renk modelinde gösterilen görüntüler, her biri bir temel renk için olan üç görüntü bileşeninden oluşur. Bir RGB monitörüne gönderildiğinde, bu üç görüntü fosfor ekranda bileşik bir renkli görüntü oluşturacak şekilde birleştirilir. RGB alanı içinde her bir pikseli temsil etmek için kullanılan bitlerin sayısı *piksel derinliği (pixel depth)* olarak adlandırılır[22]. Her bir RGB renk pikselinin yani, bir değerler üçlüsü (RGB) 24-bitlik bir derinliğe sahiptir. (3 görüntü düzlemi x düzlem başına piksel sayısı). *Tam renkli* görüntü terimi genellikle bir 24-bit RGB renkli görüntüye işaret etmek için kullanılır. 24-bitlik bir RGB görüntüdeki renklerin toplam sayısı $(2^8)^3$ = 16.777.216'dır. Şekil 3.9'daki diyagrama karşılık gelen 24-bir RGB renk küpünü göstermektedir.

Şekil 3.9'da gösterilen küp, kesintisiz (solid), $(2^8)^3$ = 16.777.216 renkten oluşan bir küptür. Bu renkleri görmenin kolay bir yolu renk düzlemleri (küpün yüzeyleri ya da çapraz kesitleri) oluşturmaktır. Bu, üç renkten birisi sabitlenerek diğer ikisinin değiştirilmesi ile yapılır. Örneğin, Şekil 3.9'daki küpün merkezinden geçen ve *GB* düzlemine paralel bir çapraz kesit düzlemi G,B = 0, 1, 2, … 255 için olan düzlemdir (127, *G, B*).Burada, aralık içinde matematiksel olarak uygun olan normalleştirilmiş değerler yerine gerçek piksel değerleri kullanılmıştır. Çünkü bunlar bilgisayarda renk üretimi için kullanılan gerçek değerlerdir[32].

Yüksek-sonlu görüntü kartları ve monitörler bir 24-bit RGB görüntüsündeki renklerin makul bir tekrarını sağlar, günümüzde kullanılan pek çok sistem 256 renk ile sınırlıdır. Güncel olarak kullanımda olan sistemlerin çeşitliliğini belirttikten sonra, kullanıcının donanım kabiliyetlerinden doğru ve makul oranda bağımsız olarak üretilmesi muhtemel olan bir renk alt grubuna sahip olmak önemlidir. Bu renk alt grubu *güvenli RGB renkleri (safe RGB colors)* dizisi ya da *bütün sistemlerde güvenli renkler (all-systems-safe colors)* olarak adlandırılır[24]. İnternet uygulamalarında, *güvenli Web renkleri* ya da *güvenli tarayıcı renkleri* olarak adlandırılır.

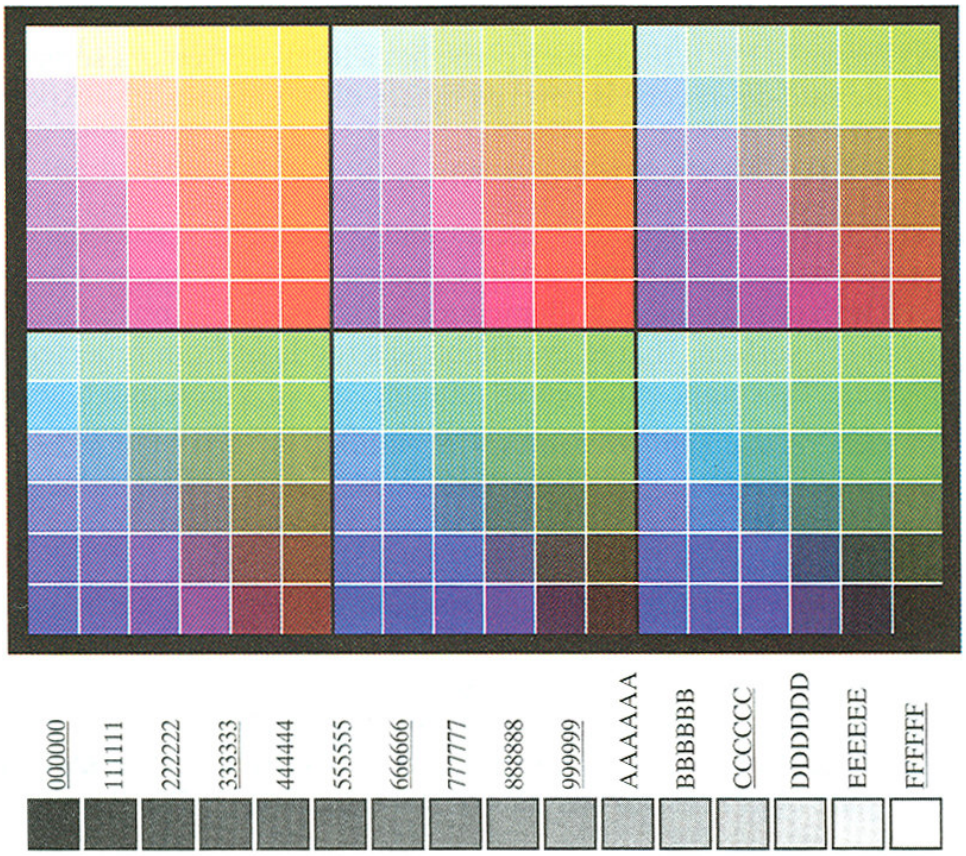
Arzu edilen sonucun görüntülenmesinin istendiği herhangi bir sistemde doğru olarak üretilecek renklerin minimum sayısının 256 olduğu varsayımına bağlı olarak, bu renklerden söz etmek için kullanılacak kabul edilmiş bir standart notasyon vardır.

216 güvenli renkten her biri öncekilerde olduğu gibi üç RGB değeri ile oluşturulur, fakat her bir değer sadece 0, 51, 102, 153, 204 ya da 255 olabilir. Böylece, bu değerlerin RGB üçlüleri $(6)^3$ =216 olası değerini verir. Bu değerlerin Çizelge 3.1'de gösterildiği gibi onaltılık sayı sistemi ile gösterilmesi gelenekselleşmiştir. 0, 1, 2,…, 9, A, B, C, D, E, F onaltılık sayıları 0, 1, 2, …, 9, 10, 11, 12, 13, 14, 15 ondalık sayılarına karşılık gelir. Ayrıca $(0)_{16} = (0000)_2$ ve $(F)_{16} = (1111)_2$ dir. Böylece, örneğin, $(FF)_{16} = (255)_{10} = (11111111)_2$ olur ve iki adet onaltılık sayı grubunun bir 8 bitlik byte oluşturur[22].

Çizelge 3.1. Her güvenli RGB elemanının geçerli değerleri

Sistem	Renk Değerliği					
Hex	00	33	66	99	CC	FF
Desimal	0	51	102	153	204	255

Bir RGB rengini oluşturmak için üç adet iki haneli onaltılık sayı gerektiğinden, her bir güvenli renk Çizelge 3.1'deki iki haneli onaltılık sayılardan üç tanesi tarafından oluşturulur. Örneğin en arı kırmızı FF0000'dır. 000000 ve FFFFFF değerleri sırasıyla siyah ve beyazı temsil etmektedir. Daha tanıdık olan ondalık sistemin kullanılmasıyla da aynı sonuçların elde edildiği unutulmamalıdır. Örneğin, en parlak kırmızı ondalık sistemde R =255 (FF) be $G = B = 0$'a sahiptir[32].



Şekil 3.12. Azalan RGB değerleriyle düzenlenmiş 216 güvenli renk

Şekil 3.12 azalan RGB değerleriyle düzenlenmiş 216 güvenli rengi göstermektedir. Sol üstteki dizinin birinci karesi FFFFFF (beyaz) değerine, sağındaki ikinci kare FFFFCC değerine, üçüncü kare FFFF99 değerine ve birinci satır üzerinde bu tür değerlere sahiptir. Aynı dizinin ikinci satırı FFCCFF, FFCCCC, FFCC99 ve benzeri değerlere sahiptir. Dizinin son karesi FF0000 (en parlak olası kırmızı) değerine sahiptir. İncelenen dizinin sağındaki CCFFFF değeri ile başlar ve aynen diğer dizilerde de olduğu gibi aynı şekilde devam eder. Sonuncu dizinin son karesi (sağ alt) 000000 (siyah) değerine sahiptir. Olası bütün 8 bit gri renk 216 güvenli renk içinde değildir[24]. Şekil 3.12 bir 216 renkli RGB sistemde mümkün olan *bütün* gri renkler için on altılık sistem kodlarını göstermektedir. Bu değerlerin bazıları güvenli renkler kümesinin dışında olmasına rağmen çoğu görüntüleme sistemi tarafından (nispi yoğunluklarına bağlı olarak) doğru gösterilirler. Güvenli renkler grubundaki griler, K = 0, 3, 6, 9, C, F için $(KKKKKK)_{16}$, Şekil 3.12'de altı çizili olarak gösterilmiştir.

Burada her bir düzlem toplam 36 renge sahiptir, böylece güvenli renk küpünün bütün yüzeyi farklı renkle kaplanmıştır.

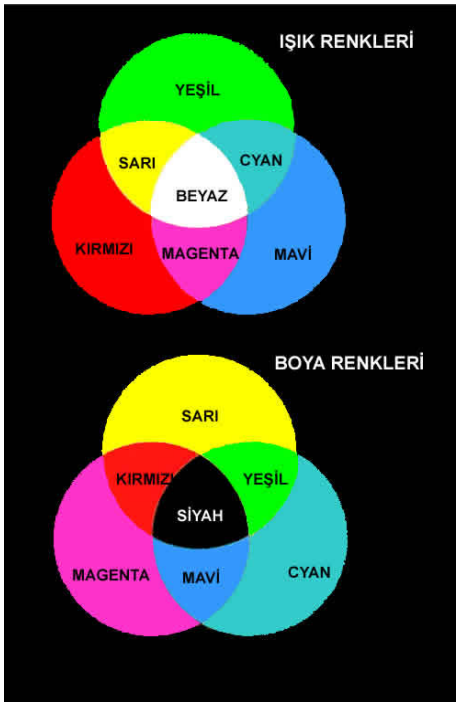
3.4.2. CMY ve CMYK renk modelleri

Cyan, macenta ve sarı ikincil ışık renkleri ya da farklı bir ifadeyle birincil pigment renkleridir. Örneğin, cyan pigmenti ile kaplı bir yüzey beyaz ışıkla aydınlatıldığında, yüzeyden kırmızı renk yansımaz. Yani, cyan, kendisi eşit miktarda kırmızı, yeşil ve mavi renklerden oluşan beyaz ışıktan yansıyan kırmızı ışığı soğurur[34].

Renkli yazıcılar ve fotokopi makineleri gibi, kâğıt üzerine renkli pigmentler püskürten çoğu cihaz, CMY veri girdisi ya da dâhili olarak RGB'yi CMY'ye dönüştürecek bir işleme gerek duyarlar. Bu dönüştürme Şekil 3.12'de verilen bütün renk değerlerinin aralık içinde normalleştirildiği varsayılan basit işlem kullanılarak yapılır.

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{3.15}$$

Saf cyan ile kaplı bir yüzeyden yansıyan ışığın kırmızı (yani, $C = 1 - R$ eşitliğindeki) içermediğini göstermektedir. Aynı şekilde, saf macenta yeşili yansıtmaz ve saf sarı maviyi yansıtmaz. Ayrıca RGB değerlerinin bir CMY kümesinden, CMY değerlerinden 1 çıkartılarak kolaylıkla elde edilebileceğini de göstermektedir[24]. Görüntü işlemede bu renk modeli basılı çıktı alma ile bağlantılı olarak kullanılmaktadır, bu nedenle CMY'den RGB'ye dönüştürme işlemi uygulamada çok az ilgi çeker.



Şekil 3.13. Işık ve Işıklılığın birincil ve ikincil renkleri

Şekil 3.13'e göre, eşit miktarda birincil pigment renkleri, cyan, macenta ve sarı siyah oluşturmalıdır. Uygulamada, bu renklerin baskı için birleştirilmesi bulanık görünümlü bir siyah oluşturur. Bu nedenle gerçek siyah (baskıda hâkim renk olan) elde etmek için dördüncü bir renk, CMYK renk modeli oluşturan *siyah* eklenir. Bu nedenle, yayıncılar "dört renkli baskıdan" bahsederken, CMY renk modelindeki üç renk artı siyahtan bahsediyorlardır.

3.4.3. HSI renk modeli

RGB ve CMY modellerinde renk üretmek ve bir modelden diğerine dönüştürmek doğrudan bir işlemdir. Bu renk sistemleri donanım uygulamalarında kullanılmaktadır. Ayrıca, RGB sistemi insan gözünün kırmızı, yeşil ve mavi temel renklerini güçlü şekilde algıladığı gerçeğine de çok uygundur. Ne yazık ki, RGB, CMY ve diğer benzer renk modelleri insanın yorumlaması anlamında renklerin tanımlanmasına çok uygun değildir. Örneğin, bir kimse bir otomobilin renginden, bu otomobilin rengini oluşturan temel renklerin her birinin oranlarını vererek bahsetmez. Dahası, renkli görüntülerin, bu görüntüleri oluşturmak üzere bir araya

gelmiş üç temel renkten oluştuğu düşünülmez. İnsanlar renkli bir nesneye baktıklarında, o nesneyi, ton, doygunluğu ve parlaklığı ile tanımlarlar.

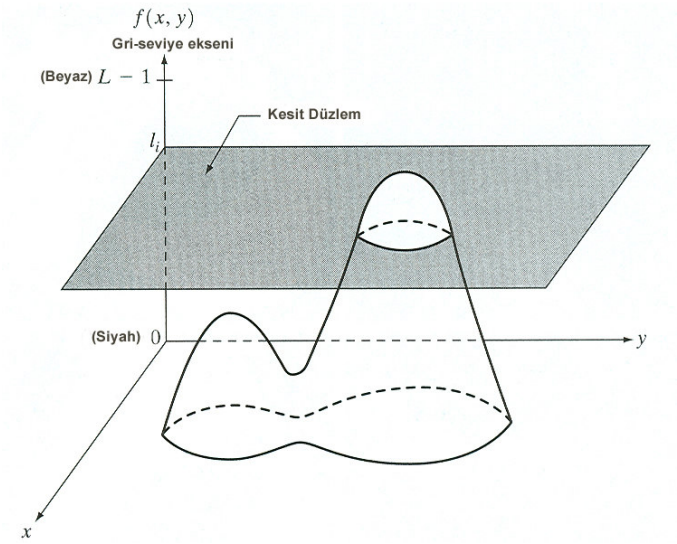
Parlaklık ölçülmesi pratik olarak imkânsız olan öznel bir tanımlayıcıdır. Akromatik yoğunluk kavramını kapsar ve renk duygusunun tanımlanmasında anahtar bir etmendir. Bu miktar tam olarak ölçülebilir ve kolaylıkla yorumlanabilir. *HSI* (ton, doygunluk, yoğunluk) *renk modeli,* bir renkli görüntüde yoğunluk bileşenini, renk taşıyan bilgiden (ton ve doygunluk) ayırır. HSI modeli, sonuçta bu algoritmaları üreten ve kullanan insanlar için doğal ve sezgisel olan renk tanımlarına dayalı görüntü işleme algoritmaları geliştirilmesi için ideal bir araçtır. Bu, RGB görüntü rengi üretimi (bir renkli kamerada görüntü yakalama ya da bir monitör ekranında görüntü göstermede olduğu gibi) için idealdir, fakat renk tanımlamasında kullanımı çok daha kısıtlıdır[22].

3.5. Yalancı Renk Görüntü İşleme

Yalancı renk (hatalı renk) görüntü işleme, belirlenen bir kıstasa göre renklere gri değerler atanmasından oluşur. Gerçek renkli görüntülerle bağlantılı işlemlerden elde edilen monokrom görüntülere renk atama işlemini ayırt etmek için kullanılmaktadır. Yalancı rengin temel kullanımı, insanların bir görüntü ya da görüntü dizisi içindeki gri seviyesindeki olayları görselleştirmesi ve yorumlamasıdır[24]. Renk kullanımının temel nedenlerinden bir tanesi insanların binlerce renk tonuna karşılık çok az gri tonunu ve yoğunluğunu ayırt edebilmesidir.

3.5.1. Yoğunluk dilimleme

Yoğunluk dilimleme ve renk kodlama tekniği yalancı renkli görüntü işlemede en basit örneklerden bir tanesidir. Eğer bir görüntü 3D fonksiyonu olarak yorumlanıyorsa (uzamsal koordinatlara karşılık yoğunluk), görüntünün koordinat düzlemine paralel düzlemler yerleştirilmesi olarak görülebilir. Bu durumda her bir düzlem kesişim alanı içindeki fonksiyonu dilimler[24]. Şekil 3.14 görüntü fonksiyonunu iki seviyeye dilimleyen bir $f(x, y)\ 0\ l_i$ düzleminin kullanım örneğini göstermektedir.



Şekil 3.14. Yoğunluk dilimleme tekniğinin geometrik açıklaması

Eğer düzlemin her iki tarafına farklı bir renk atanırsa, gri seviyesi düzlemin üstünde olan bütün pikseller bir renkle ve düzlemin altında olanlar başka bir renkle kodlanacaktır. Düzlemin kendi üstündeki seviyelere iki renkten bir tanesi rastgele atanacaktır[24]. Sonuç görüntüsü dilimleme düzleminin gri seviyesi ekseninde yukarı aşağı hareket ettirilmesi ile kontrol edilebilen iki renkli bir görüntüdür.

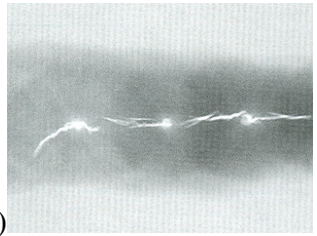
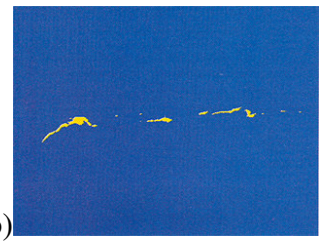
Genel olarak, bu teknik aşağıdaki şekilde özetlenebilir. [0 L - 1] gri seviyesini, l_0 siyahı [$f(x,y) = 0$] ve l_{L-1} beyazı [$f(x, y) = L - 1$] temsil etsin. Yoğunluk eksenine dik P düzlemlerinin $l_1, l_2, \ldots, l_p$ seviyelerinde tanımlandığı kabul edilir. O zaman, $0 < P < L - 1$ olduğu kabul edildiğinde, P düzlemleri gri ölçeğini $P + 1$ aralıklarında $V_1, V_2, \ldots, V_{P+1}$ olarak bölümleyecektir. Gri seviyesinden renk atamaları şu ilişkiye göre yapılır:

$$f(x, y) \in V_k \text{ ise} \qquad (f(x, y) = c_k \tag{3.16}$$

c_k $l = k - 1$ ve $l = k$ daki bölümleme düzlemleri tarafından tanımlanan k'inci yoğunluk aralığı V_k ile bağlantılı renktir[22].

Yoğunluk dilimlemesi gri ölçeğin alt bölümleri bir görüntünün fiziksel özelliklerine dayalı olduğunda çok daha fazla anlamlı ve kullanışlı bir rol üstlenir.

Şekil 3.15 (a) birkaç çatlak ve gözenek içeren (görüntünün ortasında yatay olarak uzanan parlak beyaz çizgiler) bir kaynağın (yatay koyu bölge) röntgen görüntüsünü göstermektedir. Kaynakta bir çatlak ya da gözenek olduğunda nesneden geçen X ışınlarının nesnenin diğer tarafındaki görüntü algılayıcıyı doyurduğu bilinmektedir[22]. Böylece, bu tür bir sistemden gelen 8 bitlik bir görüntüdeki 255 değerine sahip gri seviyeleri, bir problem olduğunu gösterir.

a)  (b) 

Şekil 3.15. (a) Gri seviye, (b) Kodlamanın sonucu

Analizde son kararı verecek bir insan olsaydı ve kaynakların incelenmesinde elle yürütülen işlemler kullanılsaydı, 255 seviyesine bir renk ve bütün diğer gri seviyelerine başka bir renk atayan basit bir renk kodlaması, incelemeyi yapan kişinin işini oldukça kolaylaştırırdı. Şekil 3.15 (b) bu işlemin sonucunu göstermektedir. Görüntüler Şekil 3.15 (a)'da gösterilenin yerine Şekil 3.15 (b)'deki gibi görüntülenseydi insan hatalarının daha az olacağı sonucuna varmak için daha fazla açıklama gerekmez.

Görüntü işleme, esnek üretim sistemlerinde kullanılan önemli bir temel değişkendir. Takip eden bölümde esnek üretim sistemleri irdelenmiştir.

4. ESNEK ÜRETİM SİSTEMLERİ

Teknoloji ağırlıklı sektörler bir ülkenin ekonomideki performansını belirlerler. Özellikle sanayileşmiş ülkelerin giderek artan bir önemle üzerinde durmaya başladıkları bir konu olan esnek üretim teknolojileri de, her şeyden önce bu performansın yükseltilmesiyle ilgilidir.

Bilgisayar donanım ve yazılım gücü geliştikçe esnek üretim sistemlerinde karşılaşılan problemleri çözmek için daha gelişmiş işlem ve algoritmalar kullanılabilir[35]. İşletmelerin pazardaki değişimi önceden tahmin ederek süratli bir biçimde buna uyum sağlama yetkinliğine sahip olmaları gerekliliği, günümüz teknolojisinin tetikleyicisi konumundadır. Geleceğin fabrikasının oluşturulması için, öncelikler aşağıdaki gibidir.

* Satıcı ve müşteri arasındaki bilgi akışının sağlanması için elektronik veri değişimi bağlantıları,

* Üretim sistemine yeni teknolojilerin kazandırılması,

* Bilgi transferi ve deneyimin paylaşımı,

* Yeniliğin/ yenile(n)menin geliştirilmesi,

* Ürün geliştirme sürecinin geliştirilmesi,

* Bilgisayar destekli tasarım, elektronik ürün tanımlama ve sayısal teknolojilere dayalı ürün montajı,

* Müşterinin daha fazla bilgilendirilmesi,

* Çalışma şartlarının iyileştirilmesi ve çalışma süresinin kısaltılması,

* Temiz üretim işlemleri oluşturulması,

* Üretim esnekliğinin geliştirilmesi,

* Yeni motivasyon elemanlarının tanımlanması.

Esneklik, üretim sisteminin piyasadaki değişikliklere hızlı ve etkili şekilde uyum sağlayabilmesiyle ilgili bir kavramdır. Bir üretim sisteminin esnekliği çeşitli açılardan incelenebilir. Literatürde yer alan esneklik tanımları, esneklik için bir ölçü getirmektedir ve üretim sistemleri belirli bir açıdan daha az esnek ya da daha fazla esnek olarak nitelendirilebilmektedir.

Tezgâh esnekliği, çeşitli tipte parça işlenebilmesi için tezgâhta yapılması gereken ayarların ve değişikliklerin ne kadar kolay ve çabuk yapılabildiğini gösteren bir ölçüdür. Üretim süreci esnekliği, üretim sisteminin, her biri farklı malzemeler kullanılarak ve farklı biçimlerde üretilen, çeşitli tipten bir dizi parçayı üretebilme yetisiyle ilgili bir ölçüdür. Ürün esnekliği, üretim sisteminin yeni bir ürün dizisine geçiş yapabilme yetisinin ölçüsüdür. Üretim sisteminin tasarlanıp kurulması sırasında gelecekteki ürün tasarımları belirsiz olduğundan ürün esnekliğine hangi boyutta gerek duyulacağına karar vermek oldukça zordur. İş akımı esnekliği, sistemin elemanlarından birinin, meydana gelebilecek bir arıza ya da bir bakım çalışması nedeniyle devre dışı kalması halinde, bu elemanın işlevlerinin üretimi aksatmaksızın diğer elemanlara aktarılabilmesi yetisini ifade eder. Hacim esnekliği, esnek üretim sisteminin değişik üretim hacimlerinde verimli şekilde üretim yapabilmesiyle ilgilidir. Genişleyebilme esnekliği, üretim tesisinin kolay ve modüler bir biçimde büyütülebilmesiyle (kapasitesinin artırılabilmesiyle) ilgili bir ölçüdür. İşlemsel esneklik, her bir parça tipi ile ilgili işlem sıralamasının değiştirilebilirliğinin bir göstergesidir[36].

4.1. Esnek Üretim Sistemlerinin Tanımı

Standart olarak herkesçe kabul edilebilecek bir esnek üretim sistemi tanımı yoktur. Herhangi bir esnek üretim sistemi yapılandırmasının avantajlarından yararlanabilmek için esnek üretim sistemlerinin işlemsel ve kontrol karakteristiklerinin anlaşılması ve bu karakteristiklere uygun olarak yönetilmeleri gereklidir.

Esnek üretim sistemi, işlemsel ve kontrol karakteristikleri açısından birbirinden farklı yapılardaki geniş bir üretim sistemleri yelpazesini anlatan genel bir terimdir. Bir malzeme taşıma sistemiyle birbirine bağlanmış, bilgisayar sayısal denetimli ya da sayısal denetimli tezgâhlardan ve bunların işleyişini kontrol eden bilgisayar sisteminden oluşan ve birbirinden farklı parçalar üretebilen bir üretim sistemi, olarak tanımlanabilir[36].

4.1.1. Esnek üretim sistemlerinin genel özellikleri

Esnek üretim sistemleri ürün çeşidinin fazla olduğu işletmelerde uygulanabilir. Genel amaçlı makine teçhizatı içermektedir. Mamul, yarı mamul ve hammadde otomatik bantlarla, malzeme taşıyıcılarla hareket edebilmektedir. Farklı parçaların üretilmesi makineler üzerinde gerçekleşen otomatik değişikliklerle mümkün olabilmektedir. Fabrikaya hammadde girişinden mamul çıkışına kadar kontrol, tasarım, üretim gibi tüm işlemler otomasyona dayalı olarak bilgisayarla gerçekleştirmektedir.

4.2. Esnek Üretim Sistemlerinin Alt Sistemleri

Sistem bazında ele alındığında esnek üretim sistemleri, iki alt sistemden oluşan bir üretim sistemi olarak düşünülebilir; bunlar fiziksel alt sistem ve kontrol alt sistemidir.

Fiziksel alt sistem üç ana birime ayrılabilir:

İstasyon, tezgâhlar, muayene cihazları, yıkama alanı, yükleme-boşaltma alanları.

Depolama sistemi, her istasyondaki paletler ya da iki işlem arasında parçanın geçici olarak üzerinde stoklandığı herhangi bir düzenek.

Malzeme taşıma sistemi, ışık, elektrik ya da lazer kontrollü taşıyıcılar, taşıyıcı bantlar ve diğer taşıyıcı araçlar.

Kontrol alt sistemi ise yazılım ve donanım olarak iki gruba ayrılabilir. Kontrol yazılımı, fiziksel alt sistem işletiminin yönetim mantığını oluşturan bir komut dizini ve dosyalardan oluşmaktadır. Yazılımın kullanılabilmesi için gerekli olan

bilgisayarlar, bilgi saklama sistemleri, iletişim ağları ve iletişim protokolleri de kontrol donanımı olarak adlandırılır [37].

4.2.1. Esnek üretim eleman ve teknolojileri

Bir esnek üretim sistemi, birbirine bir malzeme taşıma ağı ile bağlanmış, yarı bağımsız sayısal denetimli tezgâhlardan oluşan, simülasyon yöntemlerinden yararlanan, bilgisayar denetimli bir üretim sistemidir. Sayısal denetimli tezgâhlar parça işlemek için gerekli esnekliği sağlarken malzeme taşıma sistemleri de aynı esneklik çerçevesinde tezgâhlar arası fiziksel bağlantıyı sağlarlar. Bilgisayar sürekli olarak bu elemanları denetler. Simülasyon ise, sistemi oluşturan elemanların durumuna ilişkin olasılıkları hesaba katarak gerekli düzeltici önlemleri zamanında alabilmek için kullanılır.

Bu araçların her biri ayrı bir teknolojinin ürünüdür. Buna göre esnek üretim sisteminin temelini oluşturan teknolojileri, sayısal denetimli tezgâh teknolojisi, malzeme taşıyıcıları teknolojisi ve sistem elemanlarının birbirinden haberdar olmasını ve sistemin bir bütün olarak denetlenebilmesini sağlayan enformasyon teknolojisi olarak sınıflandırabiliriz.

Esnek üretim sistemini oluşturan fiziksel elemanlar, üretim sürecinin, uzunca bir süredir kullanılagelen, bilinen elemanlarıdır. Yeni olan, bu elemanların verimli ve esnek bir üretim sistemi oluşturacak şekilde tümleşmesinin sağlanmasıdır. 1960'ların ortalarında esnek üretim sistemi ilk şekliyle ortaya çıktığında çok kısıtlı çeşitlilikte ve birbirine çok benzeyen parçalar belirli bir düzen izleyerek işlenebiliyordu. Günümüzde, esnek üretim teknolojilerinde kaydedilen ilerlemelerin bir sonucu olarak, talep çeşitliliğine bağlı olarak birbirinden oldukça farklı karakteristikte ve çeşitlilikte üretim, gelişigüzel bir sıralamayla yapılabilmektedir.

4.2.2. Bilgisayar tümleşik üretim ve esnek üretim sistemleri

Bilgisayar tümleşik üretim sistemleri, üretimin her aşamasında bilgisayar desteği, kontrol ve tümleşik otomasyon sağlayarak, otomasyon adalarını birleştiren sistemlerdir.

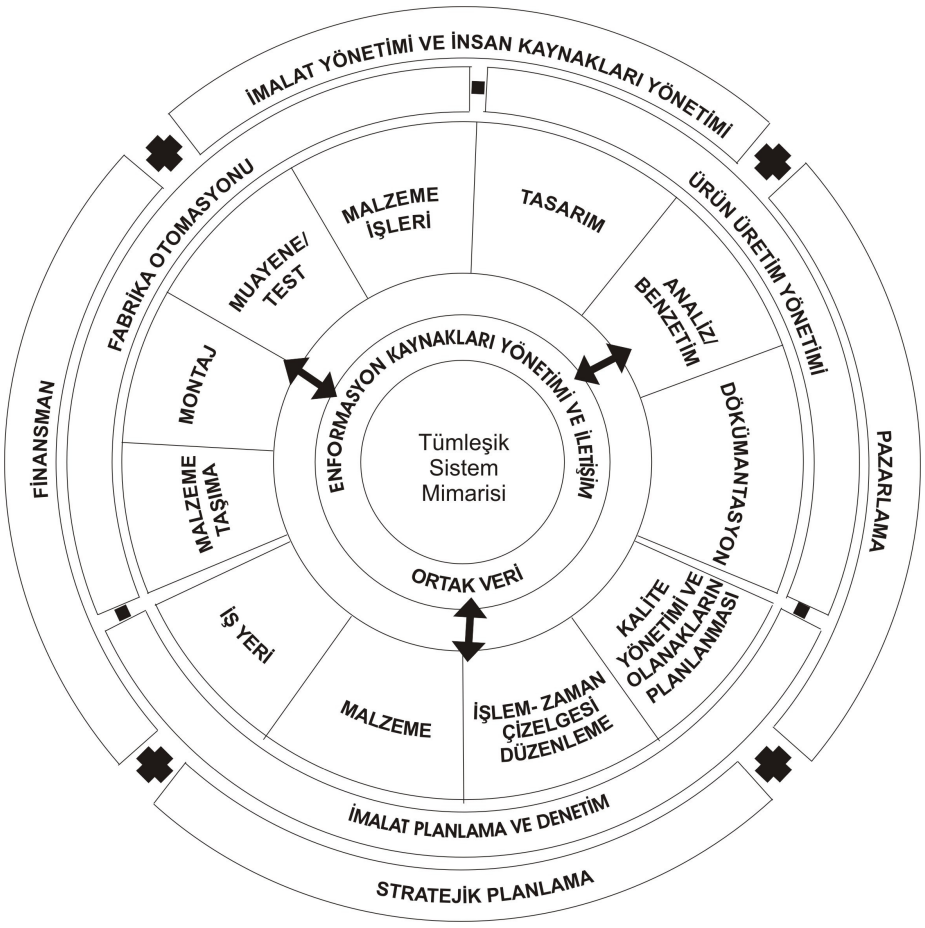
Bilgisayar tümleşik üretim sistemi üç ana alt sistemin oluşturduğu bir bütün olarak düşünülebilir:

Yönetim bilgi sistemi: yöneticilerin üretimi yönlendirmedeki karar alma işlevini destekleyen bilgisayar tabanlı bir veri iletim ve işleme sistemidir.

Bilgisayar destekli tasarım sistemi: temelde bir grafik terminalinden oluşur. Tasarlanan parçanın teknik çizimi bilgisayar ekranında görülebilir ve klavye, tablet gibi araçlarla çizim yapılabilir. Bilgisayar destekli tasarım, üretim süreciyle tasarım süreci arasında tümleşik ve sistematik otomasyonu sağlamanın önemli kilit taşlarından biridir.

Bilgisayar destekli üretim sistemi: genel olarak bir hammaddeyi satışa hazır hale gelmiş ürüne çeviren bilgisayar denetimli üretim teknikleri ve onların ön hazırlık basamaklarının tamamı olarak tanımlanabilir. Bu açıdan, esnek üretim sistemi de bilgisayar destekli üretimin içinde düşünülebilir[38].

Bir bilgisayar tümleşik üretim sisteminin işlevleri şekilde görülen casa/ sme'nin (computer and automated systems association of the society of manufacturing engineers) bilgisayar tümleşik üretim çemberi ile kolayca açıklanabilir. Bilgisayar tümleşik üretim sistemi, yürütmek zorunda olduğu işlevleri bilgisayar destekli veri akışı ile gerçekleştiren bir sistem olarak düşünülebilir. Sistem, malzeme taşıma, montaj, muayene/test ve malzeme işleme işlevlerinin otomasyonu, ürün ve üretim yöntemlerinin tasarımı, analizi, simülasyonu, dokümantasyonu, işlem-zaman çizelgesi düzenleme, kalite yönetimi ve olanakların planlanması, iş yeri ve malzeme planlama ve denetim elemanlarını içinde barındıran bir bütündür. İdeal bir bilgisayar tümleşik üretim sisteminde bu elemanların ortak bir veri tabanı aracılığıyla bağlanmış olması ve dolayısıyla sürekli olarak birbirlerinin durumundan haberdar olmaları beklenir. Bunların yanında finansman sağlama, üretim yönetimi, pazarlama ve stratejik planlama işlevleri de şirketin hedeflerine ulaşmasında önemli rol oynayan yapı taşlarıdır ve bilgisayar tümleşik üretim sisteminin içinde yer almaktadır.



Şekil 4.1. Casa çemberi[39].

Bu sistemle sipariş izleme, işlem-zaman çizelgesi düzenleme, malzeme kullanımı, fabrika denetimi, satın alma, finansman ve karar alma işlevleri birbirlerine bağlanmıştır. Ürünle ilgili tasarım, işlem planlama, üretim verileri ve finansman verileri ortak bir veri tabanında tutulmakta ve gerektiğinde tüm birimlere dağıtılabilmektedir. Parçalar bilgisayar destekli tasarım/ bilgisayar destekli üretim planlama sisteminde tasarlanmakta, üretim kaynakları planlama sisteminde hazırlanan işlem-zaman çizelgesi çerçevesinde esnek üretim sisteminde üretilmektedir.

4.2.3. Malzeme taşıma / stoklama eleman ve teknolojileri

Taşıma: modern bir malzeme taşıma sisteminin üretimde öngörülen esnekliğin sağlanabilmesi açısından sahip olması gereken temel özellik, tamamen bilgisayar

denetimli olması ve buna bağlı olarak üretim süreci içinde gerçek zamanlı tepki gösterebilmesidir.

Stoklama: stoklama sistemlerini de esnek üretim/ esnek otomasyonun gereklerine yanıt verecek bir hale getirebilmek için, otomatik malzeme taşıma sistemleriyle birlikte, otomatik stoklama ve stok yenileme sistemleri kullanılmaktadır. Otomatik stoklama ve stok yenileme sistemleri, günümüzde stok yönetim ve denetiminde ileri düzeyde otomasyonu temsil etmekte ve gerek iş gücü, gerekse yer kullanımında büyük tasarruf sağlamakta, üretim verimini artırmaktadır[40].

Kontrol: malzeme akışının gerçek zamanlı kontrolü, malzemenin izlenmesi ve veri giriş-çıkışının hızlı bir biçimde yapılabilmesiyle mümkün olabilir. Günümüzde en hızlı gelişen kontrol teknolojilerinden biri otomatik tanımlama teknolojisidir. Otomatik tanımlamayı destekleyebilecek çeşitli sensör teknolojilerinden bazıları ise barkot, optik karakter tanıma, manyetik kod okuyucular, makine görmesi, fiber optik, ses tanıma ve kimyasal sensör teknolojileridir. Ayrıca yapay zekânın malzeme taşıma, stoklama ve kontrolünde önem kazanması beklenmektedir.

4.2.4. Malzeme işleme eleman ve teknolojileri

Tezgâhlar: esnek üretim sisteminde malzeme işlemek için kullanılan bilgisayar sayısal denetimli tezgâhlar günümüzde iki ana gruba ayrılmaktadır.

İşleme istasyonları olarak adlandırılan birinci gruba giren tezgâhlarda parça sabit dururken işleme ucu döner. Karmaşık yüzeylerin işlenmesi genellikle beş ya da altı eksende hareket edebilen tezgâhların kullanılmasını gerektirmektedir. İkinci gruba giren tezgâhlarda ise parça döner ve işleme ucu sabittir. Bu tezgâhlar döner parça istasyonları olarak adlandırılırlar.

Kesici takımlar: bir taban malzemeye başka bir (ya da birden fazla) malzemeden kaplama yapılarak daha sert malzemelerin kesilebilmesine olanak veren, daha keskin ve daha dayanıklı kesici takımlar yapılabilmektedir. Bu malzemelerin ve kaplamaların birleşim olasılıkları o kadar fazladır ki birleşim seçimleri ancak bilgisayar yazılımları aracılığıyla gerçekleştirilebilmektedir.

Sabitleyiciler: sabitleyiciler, işleme ve montaj süresince parçayı belirli bir konumda tutmaya yararlar. Sabitleyicilerin tasarımında dikkat edilen en önemli nokta kurma zamanının mümkün olduğunca kısa olmasının sağlanmasıdır. Yeni sabitleme teknikleriyle bu işlem daha esnek ve programlanabilir hale gelmiştir.

Sensörler: sensörlerin üretimdeki rolü uyarlamalı kontrol sistemleri için veri toplamaktır. Robotlara yön bilgisi vermek ya da kalite güvencesi ve muayene sistemleri için ölçümler yapmak sensörlerin rolüne ilişkin örneklerdir.

Robotlar: akıllı sensörlerin en önemli kullanım alanlarından birisi robotları daha akıllı hale getirmektir. Gelişmiş bir robot sistemi akıllı sensörleri sayesinde çevresindeki değişimleri algılar ve robot yazılımı gerekli program değişikliklerini yaparak yeni şartlara uyumu sağlar.

Sistem simülasyonu ve modelleme: üretim sistemleri geliştikçe ve daha otomatize hale geldikçe sistem elemanlarının hareketlerini bilgisayar yardımıyla simüle edebilmek de önem kazanmaktadır. Üretim sistemleri tasarımında iki çeşit simülasyon kullanılır. İlki, tek bir işlemin ya da robotun grafiksel olarak simülasyonudur. Diğeri ise, bütün bir sistemin, örneğin bir esnek üretim sisteminin genellikle matematiksel olarak modellenmesidir.

Esnek üretim sistemlerinin (esnek üretim sistemi) tasarımı ve kullanımı birtakım karmaşık araştırma problemleri içerir. Örneğin, esnek üretim sistemi tasarım problemlerinin her çeşitten uygun sayıda makineyi, materyal kullanım sistemini, kapasitesini ve tamponların boyutunu saptamayı içerir

4.3. Esnek Üretim Sistemi Tasarım Problemleri

Bir esnek üretim sistemi tasarımı geliştirirken, kararların bazılarına kısmen emir verilmesi gerekir. Bazı kararlar zamanla diğerlerinin önünde olmalıdır. Bunlar başlangıç kararları ve tamamlama kararları şeklinde ayrılır.

4.3.1. Başlangıç kararları

Önce üretim gereksinimleri belirtilmelidir; üretilecek bölüm çeşitlerinin içeriklerinin sıralanmasına karar verilmesidir. Fabrikanın ürettiği tüm çeşitlerden üretilmek üzere ve/veya esnek üretim sistemine takılmak üzere bir alt grup belirleme sürecidir.

Bu, sistem üzerinde üretilmesi istenen yalnızca ilk belirlemedir. Bu zaman içinde değişecektir. Esnek üretimin yararlarından bazıları değişen piyasaların karşılanabilirliğini ve çabuk ve etkili olarak tasarımı birleştirme veya yöntem değişikliklerinin ya da yeni materyallerin kullanımını içerir.

Esnek üretim sisteminin ne üreteceği fikrinden, gerekli makine zamanı ve kesici aletler dâhilinde kapasite ve fonksiyonel gereksinimler belirlenebilir. Bilhassa; bu parçaların nasıl üretileceğine karar verilmesi. Bu düşünce gerekli makine çeşitlerini ve sayılarını neticede belirler.

Önce, her çeşit için bir yöntem planı geliştirilmelidir. Her yöntem planı sadece her bir işlemi gerçekleştirecek makine çeşidini (veya bazen çeşitlerini) tanımlamaz, ayrıca uygun kesici aletleri ve şartlarını da (yani kesiğin derinliğinin yanında her kesici aletin hızı ve ilerleme oranını da) belirler. Bunlara yöntem planındaki her işlemdeki işlem zamanını belirlemeye de gerek duyulur. Bu bilgi ile her çeşidin makine sayısı açısından gerek duyulan kapasite (hem magazin kapasitesi hem de işleyen zaman kapasitesi) belirlenebilir. Yöntem planı ayrıca işlemler arasında kısmi öncelik sıralaması da sunar; arzu edilen farklı esnekliklerin hangi türlerine gerek duyulduğunu ve her birinin miktarının belirlenmesidir[41].

4.3.2. Tamamlama kararları

Esnek üretim sisteminin düzeni saptanmalıdır; materyal kullanım sistemiyle bağlantılı düzen, makineden makineye dolaşma zamanını belirler ve toplam sistem dolaşma zamanını etkiler. Dolaşma zamanı ve mesafe bilgisi uygun çizelgeleme stratejilerini belirlemeye yardımcı olmak açısından kullanışlıdır.

Esnek üretim sistemini planlamak için stratejiler belirlenmelidir. Bütünüyle esnek üretim sisteminin fiziksel tasarımının mümkün olup olmadığını kontrol etmek açısından genel planlama ve kontrol amaçlarına gerek duyulur.

Yazılım geliştirme görevleri bir proje yönetim ayarlamasıyla belirlenmelidir ve tamamlanmalıdır. Bu görevler kontrol yapısı için yazılım yazmayı, alet yönetim envanteri kontrolünü, çizelgelemesini ve sonra her birinin tamamlanmasını içerir.

Sistem bir kere tasarlandığında, tamamlandığında ve üretime geçildiğinde yönetim endişesi sistemi en iyi nasıl yürüteceğine dönüşür[41].

Esnek üretim sistemi problemleri, esnek üretim sistemi parçaları üretmeye başlamadan önce belirlenmesi gerekenlerdir.

Esnek üretim sistemi planlama problemleri; esnek üretim sistemi planlama basamağında esnek üretim sistemi tamamlanmıştır ve üretimdedir. Problemler sırayla ve tekrar ya da birçok kez aynı anda çözülebilir. Seçilen parça çeşitlerinden birinin üretim gereksinimleri tamamlandığında bu planlama problemlerinin çözümünü yeniden tetikleyebilir.

Esnek üretim sistemi çizelgeleme problemleri; gerçek üretimin önünde olan planlama evresi boyunca kurulduğunda gerçek zaman boyunca esnek üretim sistemini çalıştırmakla ilgilidir. Sistem boyunca parçaların üretimini çizelgeleme etmek için birçok yaklaşım mümkündür. Farklı durumlara uygun çeşitli yaklaşımlar olabilir.

Esnek üretim sistemi kontrol problemleri; sistemi ürünün koruma izinin üretim gereksinimlerinin ve uygun tarihlerin çakışacağından emin bir şekilde sürekli izleriyle bağlı olarak açıklanmaktadır[41].

4.4. Matris Tabanlı EÜS Tasarım Modeli

Matris modeli bir işaretleme geçiş denklemiyle beraber bir esnek üretim sisteminin dinamiklerini tam olarak tarif eder. Bu, böyle bir sistemin çevrimiçi kontrolüne izin verir.

Bir modeli olanaklı kılmak için şu ilkeler yerine getirilmelidir. Bir kaynak, iş tamamlanana kadar o işten alınamaz. Tek bir kaynak birim zamanda yalnızca bir iş için kullanılabilir. Bir süreç zaten kendisine tahsis edilen kaynakları, bir işi gerçekleştirmek için gereken bütün kaynaklara sahip olana kadar tutar[35].

Bu ilkeler, temel olarak paylaşılan kaynaklar, montaj ilkeleri, iş yönlendirme kararları ve birden fazla parça yolu içeren parça temelinde üretim sistemi tipini tanımlamaktadır. EÜS matris tabanlı model şu biçimlere sahiptir:

Mantık matrisi durum denklemi; $\bar{x} = F_v \otimes \bar{v}_c \oplus F_r \otimes \bar{r}_c \oplus F_u \otimes \bar{u} \oplus F_D \otimes \bar{u}_D$ (4.1)

İş başlama denklemi; $v_s = S_v \otimes x$ (4.2)

Kaynak yayım denklemi; $r_s = S_r \otimes x$ (4.3)

Ürün çıktı denklemi; $y = S_y \otimes x$ (4.4)

Matris modelinde standart bir matris toplama ve çarpma "ve/veya" cebiri ile değiştirilir "veya" işleminin "ve" işleminin yerini tutar.

$F_{v,}$ $f_{r,}$ $s_{v,}$ ve s_r matrisleri aşağıdaki anlama sahiptir. F_v (i,j) = 1 eğer j işi, i işi için yakın bir öncü olarak gerekliyse, F_r (i,j) = 1, eğer j kaynağı j işi için gerekliyse, S_v (i,j) =1, eğer i işi x_j öğesinin tüm gereksinimleri karşılandığında başlatılacaksa, S_r (i,j) = 1 eğer i kaynağı mantıklı durum öğesi x_j yükseğe ayarlandığında serbest bırakılacaksa Eş. 4.1, bir kural dizisini temsil eden mantıklı bir denklem olduğundan bütün vektörler ikilidir.

Eş. 4.1'deki üst çizgi aşağıdaki gibi tanımlanan bir doğrusal olmayan işlevi (örn: mantıklı yadsıma) belirtir. Doğal bir a sayı vektörünün herhangi bir a(i) öğesi için:

$\bar{a}(i) = 0$ eğer $a(i) > 0,$

$\bar{a}(i) = 1$ eğer $a(i) \leq 0.$ (4.5)

Bir x vektörü mantıklı durum vektörünü belirtir. Bir x_i öğesi i kuralına karşılık gelir; i kuralını harekete geçirmek için gereken koşullar ("işler bitirilir" ve "kaynaklar boştur") tamamlanırsa x_i öğesi değer 1'e erişir. V_c ve v_s vektörlerindeki "1" girişleri tamamlanmış işleri ve başlayacak işleri belirtir; r_c ve r_s'deki "1" girişleri boş kaynakları ve serbest bırakılacak kaynakları belirtir; y'deki "1" ise o parçanın hariç olduğunu belirtir. U ve u_d vektörleri (çoklu parça yolu olması durumunda bu vektörler matris haline gelir) sırasıyla giren parçaları ve çatışma çözme girdisini temsil etmektedirler. F_r matrisinin aynı sütunda çoklu "1"'lere sahip olması halinde, yani paylaşılan kaynak için eşzamanlı istekler halinde, bir u_d çatışma çözme vektörü yürütülecek işi seçer. Bu, u_d'nin belirli bir anda yükseğe ayarlanan belli bir sütununun girişlerinden sadece birisine sahip olabileceği anlamına gelir. Bir f_d matrisi, kontrol işaretlerinin EÜS harekete geçirme kurallarını nasıl etkilediğini tanımlar[35].

4.4.1. Örnek matris-tabanlı EÜS tasarım modeli

Örnek olarak, aşağıdaki matrislerle tanımlanan bir sistemi kullanalım:

$$F_v = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad F_r = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

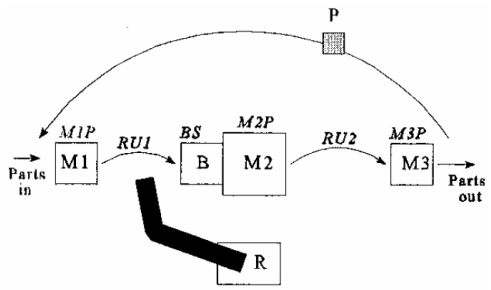
$$T_v = \begin{bmatrix} z^{-8} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & z^{-3} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & z^{-3} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & z^{-11} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & z^{-4} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & z^{-6} & 0 \end{bmatrix},$$

$$T_r = \begin{bmatrix} 0 & z^{-2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & z^{-5} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & z^{-3} \\ 0 & 0 & z^{-2} & 0 & 0 & z^{-3} & 0 \\ 0 & 0 & 0 & & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & z^{-4} \end{bmatrix}$$

Sistem matrislerinden EÜS'deki altı kaynağın bütün işlemleri gerçekleştirdiği bulunabilir ve bunlardan biri paylaşılan bir kaynaktır (f_r'deki 4üncü sütun). EÜS, Şekil 4.2'de gösterilmektedir.

Görev vektörü *v* ve kaynak vektörü *r* şu şekilde tanımlanırlar:

$$v = [M1P \quad RU1 \quad BS \quad M2P \quad RU2 \quad M3A]^T,$$
$$r = [M1 \quad M2 \quad M3 \quad R \quad B \quad P]^T$$



Şekil 4.2. Deneysel EÜS[35].

P paletinin üzerinde uzanan parça, sisteme m1 makinesi vasıtasıyla girmektedir ve m1p olarak görev yapmaktadır. Sonra bu parça (ru1 işlemi) r robotu tarafından b tamponuna taşınmaktadır ve M2 makinesinin boşalması (bs işlemi) için burada beklemektedir. M2 işleminin bitmesinden sonra robot bir parçayı alır ve bunu M3 makinesinin içine koyar. M3p görevi sona erdikten sonra parça esnek üretim sistemini terk eder ve palet serbest bırakılır.

$$m(z) = \begin{bmatrix} v(z) \\ r(z) \end{bmatrix} \tag{4.6}$$

$$m(k) = m(k-1) + W(k)^T \sigma\{F\sigma\{m(k-1)\} - (w - [1])\} \quad (4.7)$$

F_v, f_r, t_v ve t_r matrislerini ve v ve r vektörlerini Eş. 4.6'da ve daha sonra Eş. 4.7'de yerlerine koyarak zaman geçiş denklemi oluşturulabilir. Bu denklem sistem dinamiklerinin simülasyonu için uygundur[35].

4.5. Esnek Üretim Sisteminde Makine Görmesi Alanları

Esnek üretim sisteminde görüntü işleme kullanıldığı iş biriminin türüne göre şekillenmektedir. Genelde karşımıza çıkan uygulamalar sınıflandırma, kalite kontrol vb. departmanlarda yoğunlaşmaktadır.

Sınıflandırma mamul veya yarı mamul üretim bandına girerken ya da çıktıktan sonra paketleme veya tezgâh seçimi gibi işlemlerde nesnenin rengine veya diğer özelliklerine göre ayrıştırılmasıdır. Sınıflandırma sırasında ikinci ve üçüncü bölümlerde ifade edilen görüntü işleme teknikleri kullanılabilir. Kalite kontrol açısından da görüntüleme ve görüntü analizi önem taşımaktadır. Üretim bandına girecek hammadde veya yarı mamulün renk ve ebat gibi özelliklerinin kontrolünden üretim bandından çıkan yarı mamul ve mamulün benzer özelliklerinin kontrolüne kadar bir dizi süreçte görüntü işlemeye ihtiyaç duyulur.

Esnek üretimin çok küçük boyutlara ve miktarlara indiği günümüzün rekabetçi ortamında bu hızlı rekabetçi ortama uyum sağlayabilecek üretim modelleri geliştirmek ve bunları pazarlamak kadar, bu model ve ürünleri geliştirecek vasıflı iş gücünün bu alandaki eğitimi de önem taşımaktadır. Fark esneklikte farklı modellerin yaratılması geniş ufuklara açılabilme yeteneğine, teknoloji altyapısına sahip teknisyenler tarafından gerçekleştirilebilir.

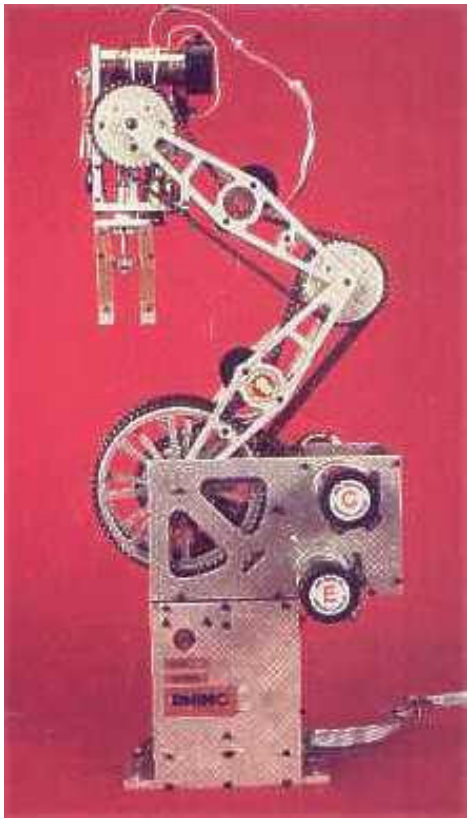
Devam eden bölümde robotik eğitimi ve bu alanda uygulamalı modeller geliştirilmesinin önemi üzerinde durulmuştur.

5. SİSTEM ELEMANLARI VE PROGRAMLAMA ESASLARI

Bu bölümde programlanan altı eksenli deneysel eğitimi robotu XR3 ve denetleyicisi Mark3'ün donanımı, elektrik bağlantıları programlama esasları ve kinematik incelemesi yapılarak sistem tanıtılacaktır.

5.1. Donanım

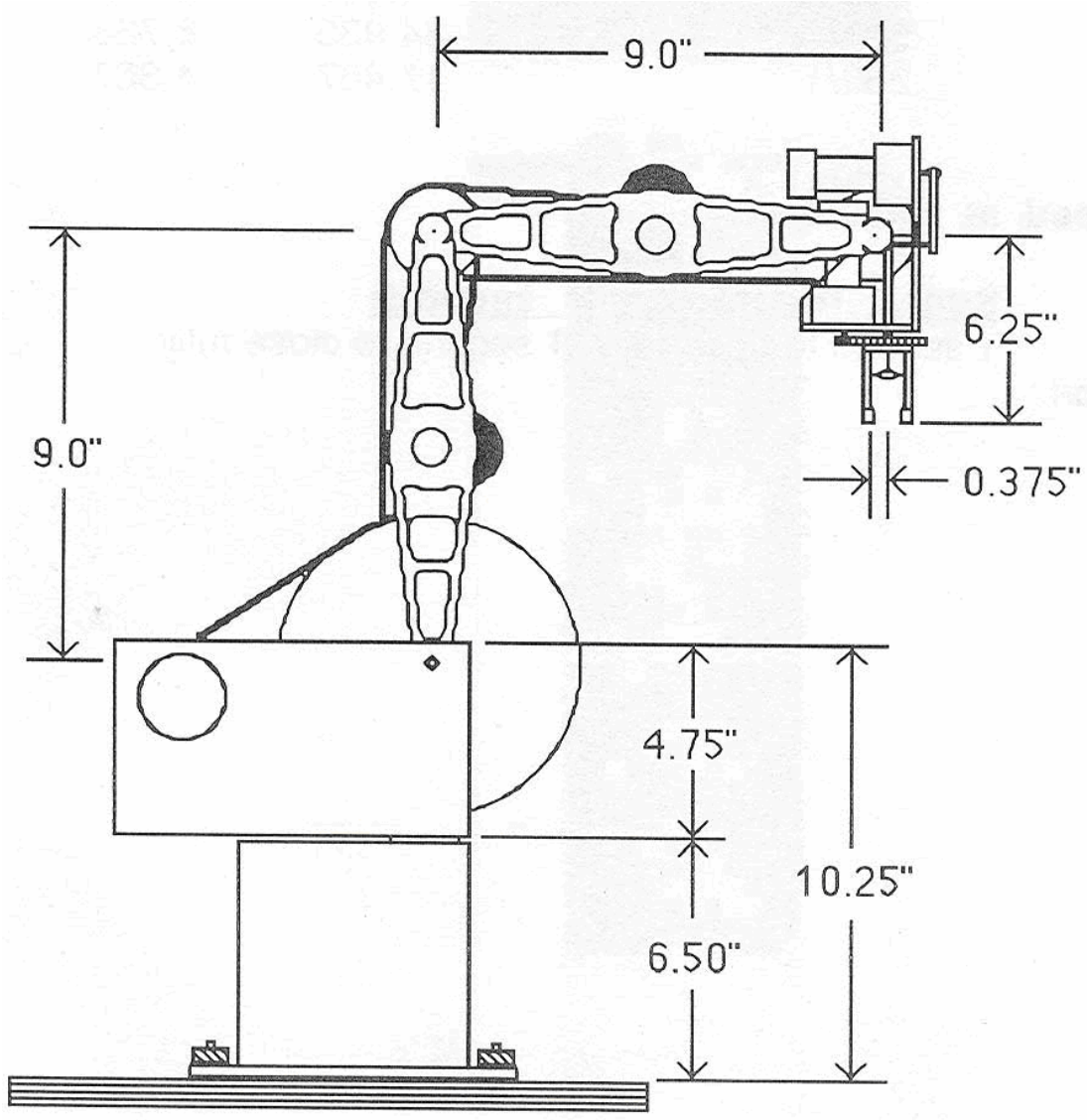
MODEL XR3: Dayanıklı, ekonomik, açık tasarım 5 eksen elektrikli tutuculu robot koldan oluşmaktadır. 5 eksenli bu robot, sorunsuz çalışabilmesi için 0,125 ve 0,250 kalınlığındaki alüminyum plakalardan inşa edilmiştir. 6 adet dişli kutulu darbe modülasyonlu ve artımlı enkoderli güç sağlanır. 0 noktası anahtarları, eksenlerde ve tutucuda bulunur[42].



Resim 5.1. XR3 Rhino robot kol

Çizelge 5.1. XR3 Fiziksel özellikler

Dikey Uzanma	87.63 cm
Dairesel Uzanma	60.96 cm
Kaldırma Kapasitesi	1.0 kg
XR–3 ün Ağırlığı	7.27 kg



Şekil 5.1. XR-3 Boyutları

Çizelge 5.2. Her bir eksendeki çözünürlükler

Eksen	Motor	MARK3	
Parmaklar	A	NIA	Ölçüme uygun değil
Bilek (dönme)	B	0.31	Teorik derece
Bilek (bükülme)	C	0.12	Teorik derece
Dirsek	D	0.12	Teorik derece
Omuz	E	0.12	Teorik derece
Gövde	F	0.23	Teorik derece

Çizelge 5.3. Motor Dişli oranı ve durma azalması

Eksen	Motor Dişli Oranı	Zincir Oranı
Parmaklar	96/1	N/A
Bilek (dönme)	96/1	4/1
Bilek (bükülme)	65.5/1	8/1
Dirsek	65.5/1	8/1
Omuz	65.5/1	8/1
Gövde	65.5/1	4/1

Çizelge 5.4. Enkoder Adımları/Eklem Dereceleri

Eksen	Toplam Azalma	MARK3
Parmaklar	N/A	N/A
Bilek (dönme)	384/1	3.200
Bilek (bükülme)	524/1	8.733
Dirsek	524/1	8.733
Omuz	524/1	8.733
Gövde	262/1	4.367

Çizelge 5.5. Her eksendeki maksimum hız

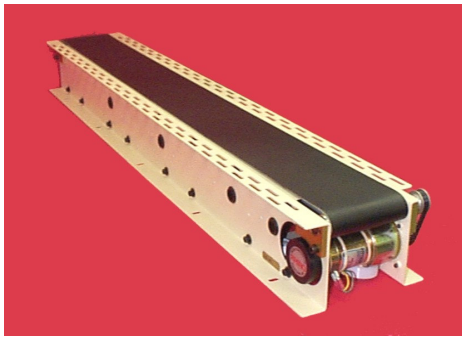
Eksen	Hız (derece/saniye)
Parmaklar	1 saniye tam açılma, 1 saniye tam kapanma
Bilek (dönme)	32
Bilek (bükülme)	45
Dirsek	30
Omuz	20
Gövde	60

Mark3 Denetleyici: 8 giriş, 8 çıkış, 2 yardımcı port 8 eksen denetleyici. El kumanda birimi destekli ve RS 232C seri bağlantılıdır. 8 TTL giriş ve 8 TTL çıkış hattına sahip olan 8 eksen kumanda ünitesidir. Aynı zamanda 20 V ve 1,5 A sağlama kapasiteli 2 yardımcı portu da kontrol eder. Bütün girişler Pull- Up ile TTL yüksek seviyede tutulur. Topraklanarak kolayca kullanılabilirler ayrıca güç gerektirmezler. Çıkış hatları TTL seviyesinde 100 mA saplar. Yardımcı portlar, motor veya röle kontrolü için kullanılabilir. MARK3 denetleyici; güç kaynakları, haberleşme, mikroişlemci, el kumanda birimi desteği, giriş-çıkış yetenekleri ve bir yazılımı içinde barındırır. Denetleyici, Rhino XR3 sistemine veya SCARA robotuna bağlanmaya hazırdır. Bütünleşik bir sistem kullanmanın faydası, kullanıcılara CIM ve FMS sistemleri kurarken kolaylık sağlamasıdır[42].



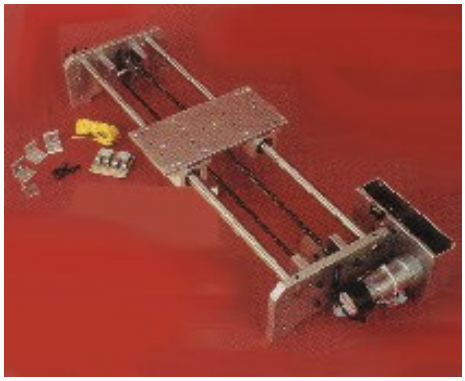
Resim 5.2. Rhino Mark3 denetleyici

Taşıyıcı Bant (Konveyör): Geniş taşıyıcılar, çalışma alanına malzeme nakli için elverişlidirler. CIM, FMS ve montaj gruplarında, 10x90cm veya 10x120cm ebatlarda konveyörler malzeme taşımada kullanılabilir. Konveyörler darbe modülasyonlu DC servo motorlar, bütünleşik dişli kutusu, artımlı enkoder içerirler[42].



Resim 5.3. Rhino konveyör bant

Hareketli Taban: Hareketli tabanlar, 5 eksenli robotlara 6. eksen olarak eklenerek çalışma alanını artırırlar. Doğrusal hareketli taban 5/8 inch'lik yataklar ve parlatılmış çubuklar üzerinde hareket eder[42].



Resim 5.4. Hareketli taban

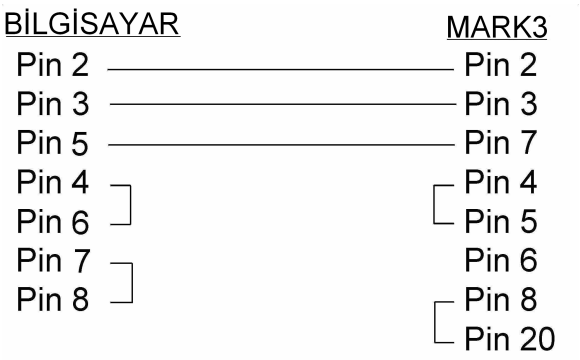
Uç elemanları: Uç elemanlar (eller) bir robotun tutma yeteneğini ve esnekliğini geliştirirler. XR3 ile birçok uç elemanı kullanılabilir. Bunlar standart uç elemanı ile yer değiştirilerek kullanılırlar. Değişik kullanımlar için değişik uç elemanları vardır. Bunlardan bazıları üçlü parmak, dar parmaklar, uzun parmaklar kürek elemanı, vakumlu parmaklar ve tarak elemanıdır.

El kumanda birimi: 7x7 LCD ekran ve 32 tuşa kadar destek verir. Kullanımı kolaydır ve Robotalk programı tarafından tamamen desteklenir. MARK3 kumanda ünitesinin standart birimidir[42].

5.2. Genel Arayüz Komutları ve Elektriksel Bağlantılar

Kullanılan bilgisayarın sunucu bilgisayar olmasına gerek yoktur. Veriyi hem gönderme hem de alma yeteneğine sahip bir RS232 seri yola sahip olmalıdır. Mark3 denetleyici, üzerindeki DB25 konnektör üzerindeki 25 iletim hattından sadece 3'ünü kullanır[43].

Sunucu bilgisayarı, Mark3 denetleyiciye bağlamak için RS232 bağlantıları aşağıdaki gibi düzenlenmelidir.



Şekil 5.2. Bilgisayar ve Mark3 denetleyici arasında kullanılan veri yolu

5.2.1. Veri formatı

Bilgisayarın RS232 veri yolu; 9600 Baud. Yedi veri biti. Eşlik biti dâhil tutulur. İki durdurma biti ile iletişim kurar.

Bilgisayara bağlı olmasına rağmen, yazılacak her bir programın başlangıcında yazılım komutlarıyla veri formatı ayarlanmalıdır[44].

5.2.2. Veri yolu açma

Komutların doğru veri yoluna yönlenmesi sağlanmalıdır. IBM uyumlu bilgisayarlarda veri ASCII I/O cihazına gönderilir gibi hazırlanmalıdır. Bu, bilgisayarın ayarlarından yapılabileceği gibi işletim sisteminin otomatik algılaması da sağlanabilir. Yazılım desteği kullanılarak da bu düzenleme gerçekleştirilebilir[45].

5.2.3. Bilgisayar ile bağlantı kurma

Bağlantı, elektrik bağlantılarının kapalı olmasına dikkat edilerek, IBM uyumlu bilgisayarın RS232 veri yoluna Dsub9 dişi konnektörün takılması ve Mark3 denetleyici üzerinde bulunan "COMPUTER" olarak gösterilen veri yoluna Dsub25 erkek konnektörün takılması şeklinde gerçekleştirilir[44].

5.2.4. İletişim

IBM uyumlu bilgisayar ile Mark3 denetleyici arasında iletişim kurmak için;

Sunucu bilgisayarın açılması,

Mark3 denetleyici "COMPUTER" konumuna getirilerek ana gücün açılması,

XR3 motorlarına güç verilmesi,

MARK3 denetleyicinin "RESET" butonuna basılması. (RESET butonu herhangi bir çarpma ve kilitlenme konumunda da kullanılabilir.)

Bilgisayar hazır olduktan sonra, BASICA, BASIC, GWBASIC, QBASIC programlarından herhangi birinin çalıştırılması. Bu işlem için DOS kullanılabilir[44].

10 OPEN "COM1:9600, E, 7, 2, CS, DS, CD" AS #1

Satırı ile programlamaya başlanır. Bu satır 1 numaralı veri yolu iletişimini 1 numaralı dosya olarak açar ve BAUD oranını ve veri formatını kurar. Aynı zamanda *zaman aşımı hatalarını* da pasif yapar. Mark3 denetleyiciye bilgisayarın hangi veri yolunun

bağlı olduğuna bağlı olarak COM2, COM3 veya COM4 portlarından biri kullanılabilir(37).

5.3. Programlama Esasları

5.3.1. Motorları hareket ettirme

Mark3 denetleyiciye gönderilebilecek dört temel komuttan biri olan START komutunu kullanarak robotu hareket ettirmek için,

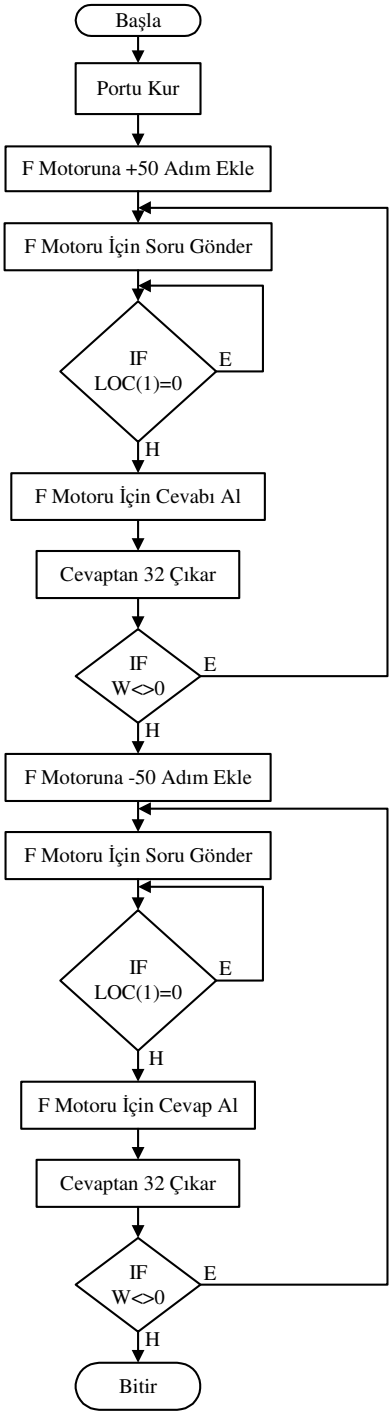
20 PRINT #1, "F+50"

30 END

Programı çalıştırarak robotun F (gövde) ekseninin pozitif yönde 50 adım ilerlediği görülebilir. Satır 20'deki işaret (+) dan (-)ye çevrildiğinde F(gövde) ekseninin az önceki hareketin tersini yaptığı gözlenecektir. Eğer herhangi bir işaret kullanılmazsa bu (+) olarak algılanacak ve o yönde hareket gerçekleşecektir. Burada "F" motoru tanımladığından diğer eksenleri hareket ettirmek için aynı döngü E, D, C, B motorları için de uygulanabilir. Bu döngüde sadece F motoru hareket etmesine rağmen, Mark3 denetleyici sekiz motorun her biri için sekiz bitlik bir hata kaydedicisi tutar. Belirli bir motor için hata kaydedicisi sıfır olduğunda denetleyici motorun gücünü keser ve motor hareketsiz kalır. Hata kaydedicisi sıfır olmadığında, denetleyici motora pozitif veya negatif gerilim uygular, polarite hata kaydedicisine göre belirlenir. Motorlar polariteye göre yönünü belirler. Herhangi bir komut öncesinde tüm motorlar için hata kaydedicisi sıfırdır[45].

START komutu, belirlenen motor için motor hareket değerini hata kaydedicisine ekler. Motor için bir değer hata kaydedicisine eklenir eklenmez, denetleyici hata kaydedicisini sıfıra götürecek yönde, motor hareketini başlatır. Motor hareket ettiği için kod çözücüler okunur ve hata kaydedicisindeki değer azaltılır. Hata kaydedicisi sıfıra ulaştığında güç kesilir. Denetleyici yazılımı sürekli olarak sıfır hata durumunu arayacak şekilde tasarlanmıştır[46].

5.3.2. Durum bilgisi alma



Şekil 5.3. QUESTION komutu uygulama örneği

Program çalıştırıldığında, F motorunun önce bir yönde ardından diğer yönde döndüğü gözlenecektir.

QUESTION komutu belirli bir motor için hata kaydedicisinin en son değerini Mark3 denetleyiciye geri döndürerek bilgi verir. Kaydedici tekrar sıfıra eşit olana dek gösterilen motorun ne kadar hareket etmesi gerektiğini sorar. Şekil 5.3'te ilk START komutu gönderildikten sonra F hata kaydedicisinin durumunu kontrol etmek için hemen bir QUESTION komutu gönderilir. QUESTION komutuna cevap (W) sıfır olana kadar döndürür. Sıfır olduğunda, hata kaydedicisinin sıfır olduğu ve bütün adımların işletildiği bilinmektedir. Sadece bundan sonra, program ters yöndeki START komutunu göndermeye geçer[43].

Sekiz bitlik bir hata kaydedicisinde saklanabilen en büyük işaretli tamsayı 127'dir (bir bit sayının işareti için ve 7 bit 127'ye kadar olan sayılar için). 127'den büyük sayılar kaydediciden taşacaktır. Bu olduğunda, kaydedici işaret bitine taşar. Bu kaydedicideki işareti değiştirir ve motor tersine döner. Diğer bir deyişle, eğer kaydedici taşarsa motor, START komutunda belirlenen yönün aksi yönde hareket etmeye başlayabilir.

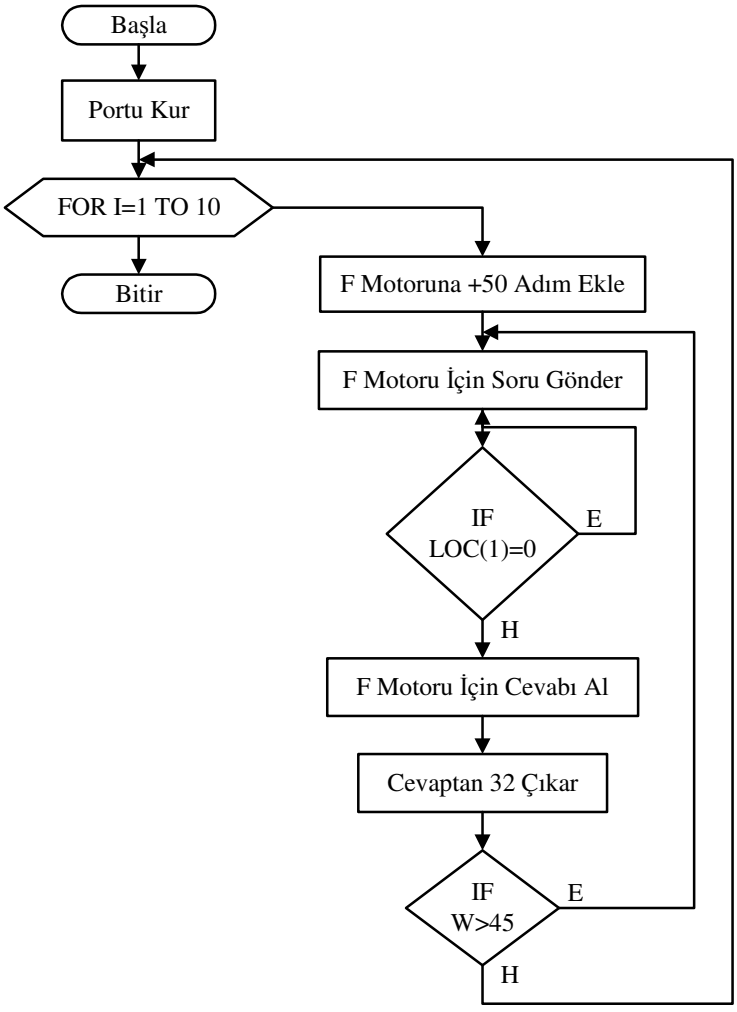
Denetleyici bir QUESTION komutu aldığında, sunucu bilgisayara cevabı göndermeden önce otomatik olarak cevaba 32 (Hex20) ekler. Çünkü bazı bilgisayarlar 32'nin altındaki değerlere veri yerine komut gibi davranır. QUESTION komutuna alınan cevaptan 32'yi çıkartır. Bu, denetleyici tarafından eklenen 32'yi karşılar.

İletişim hattı 7 bit için kurulduğundan, geri döndürebilecek en büyük değer 127'dir yani denetleyicinin kaydedicisi sıfır olabilir, fakat asla daha düşük olamaz (çünkü işaret biti motorun hareketi için kullanılmaktadır). Diğer bir deyişle bir motorun sıfır pozisyonundan ne kadar uzakta olduğu bulunabilir fakat sıfır pozisyonunun hangi yönde olduğu bulunamaz. Hata değeri her zaman pozitiftir. Denetleyici her değere 32 ekleyerek sonucu bilgisayara gönderir ve böylece bilgisayarın 32 (Hex20)'den küçük bir değer almamasını garanti eder[45].

Örneğin, 20 adımlık bir START komutu gönderildiğinde motor derhal sıfırı aramaya başlar. Motor 20 adım ilerlediğinde eğer denetleyici bir QUESTION alırsa, geriye gönderilecek cevap 100 olur (sıfıra kadar olan adım sayısı). Fakat denetleyici

sunucuya cevabı geri göndermeden önce 100'e 32 ekler. 132, 7-bit ile gösterilecek olandan fazladır ve bir hata meydana gelir.

Hata kaydedicisinin 95'i aşmasına izin verilmemelidir. Fakat 95 adımdan fazla hareketin robota nasıl yaptırılacağı sorulabilir. Cevap START komutuyla bir motorun hata kaydedicisine ilk olarak bir değer eklemektir. QUESTION komutuyla hata kaydedicisinin durumu öğrenilir. Daha sonra, kaydedici sıfıra ulaşmadan önce 95 limitini aşmayacak şekilde START komutuyla diğer kaydediciye eklenir[43].



Şekil 5.4. 95 adım üzerinde adım sayısı uygulama tekniği

Bu program 500 adımlık hareket için iki anahtar teknik kullanır.

FOR/NEXT döngüsü F motorunun hata kaydedicisine bir defada 50 adım yükleyerek 500 adımı böler ve gerçekleştirir. Bilgisayarı 10 defa döndürür.

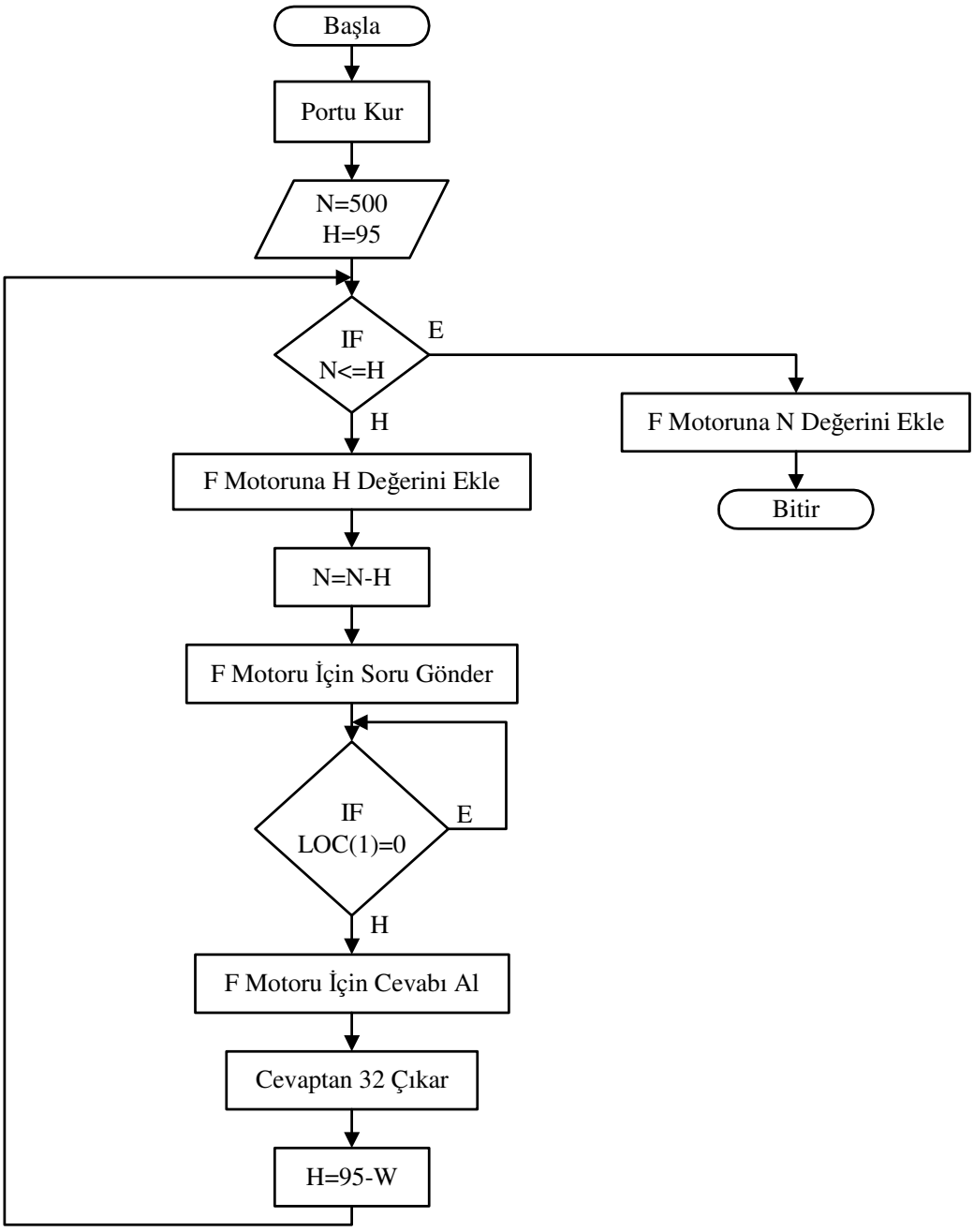
IF/THEN ifadesi, cevap 45 veya daha küçük olana kadar QUESTION komutunu döndürür. Cevap 45 veya daha küçük olduğunda, 95 limitini aşmadan sıradaki 50 adımı ekleyip FOR döngüsüne devam edilebilir.

Hata kaydedicisine ekleme yapmak için sıfıra ulaşmasını bekleme zorunluluğu yoktur. Hata kaydedicinin sıfıra ulaşması engellenmelidir. Çünkü ulaşırsa, motor duracaktır. Kaydediciye 95'i aşmadan ekleme yapılması gerekmektedir. Hata kaydedici gaz tankı olarak düşünüldüğünde, asla boşalması istenmez. Fakat taşmasına da izin verilmez. Stratejik olarak bu, gaz istasyonları arasında en uzun mesafeye gitmeyi sağlar.

Şekil 5.4 bunu yapmanın bir yolunu göstermekte. Kaydediciye 50 adım eklendi, diğer 50 için yer olur olmaz, diğer 50 eklendi. Program bir örnek olarak çalışmakta ancak sakıncaları vardır. En önemli dezavantajı daha fazla adımı eklemeden önce kaydedicinin 45'in altına düşmesini bekler. Daha karmaşık programlama yapıldığında, bilgisayar çok fazla meşgul olacak ve zamanlama daha kritik olacak. Kaydedicileri kontrol edip doldurmak için daha az zaman olacaktır.

Bunu yapmanın daha iyi bir yolu, kaydediciyi her zaman mümkün olduğunca doldurmaktır. Ne zaman ve ne kadar adım ekleneceğine önceden karar vermenin yerine, 95'i aşmadan en fazla adım için ne kadar adım yükleyebileceğine bilgisayar karar verebilir. Şekil 5.5 bu konuda fikir verici niteliğe sahiptir.

Şekil 5.5'de N, hareket ettirilmek istenen motorun toplam adım sayısını göstermektedir. Hata kaydedicisine eklenecek adım sayısına göre N değeri değiştirebilir. Hata kaydedicisine eklenebilecek adım sayısı için H değişkeni kullanıldı. H başlangıç değeri olarak hata kaydedicisine güvenli olarak eklenebilecek maksimum sayı olan 95'i atanmıştır[42].



Şekil 5.5. Sürekli hareketi optimum zamanda gerçekleştirmeye yönelik döngü

START komutunda adım sayısını belirtmek yerine H değişkeni kullanılmaktadır. F motorunun hata kaydedicisine adım sayısı H eklenir. H başlangıç değeri olarak 95 atandığı için, bilgisayar ilk olarak 95'i ekler. Aynı zamanda N'den H'yi çıkartır; hata kaydedicisine eklenen sayı kadar toplam adım sayısını azaltır.

QUESTION komutu cevabı almak için (W) altyordamını çağırır. Aynı zamanda H'ye yeni bir değer atar, 95-W. W hata kaydedicisi sıfıra ulaşmadan önce F motorunun hareket etmesi gereken adım sayısını gösterdiği için, 95-W, 95 sınırını geçmeden hata kaydedicisine eklenebilecek adım sayısıdır. Böylece, H hata kaydedicisini 95'e ulaştırmak için gereken sayıya dönüşür. Program N, H'den küçük veya H'ye eşit oluncaya kadar bu döngüyü devam ettirecektir. Böylece N, orijinal adım sayısı her defasında küçülecektir. Bu metot hata kaydedicisini mümkün olduğunca dolu tutar[43].

5.3.3. Durdurma ve sorgulama

Durdurma: Bir servo motor, güç olduğu sürece hareketini devam ettirmeye çalışır. Hata kaydedici sıfır olmadığı sürece robotun çarpması vb. durumlarda bile motorlar çalışmalarını sürdürmek isteyeceklerdir. STOP komutu pozisyon bilgileri kaybedilmeden belirli bir hata kaydedicisini sıfıra getirmeye izin verir[45].

Mikroswitchlerle sorgulama: Robot, kol üzerindeki altı motorun her biri için bir tane olmak üzere altı mikroswitchle donatılmıştır. Her eksen belirlenen bir noktada kendi switchini kapatır. Böylece, mikroswitchin kapalı olduğu noktada, robot üzerindeki her eksen için tanımlanabilir.

Sorgu komutları (I ve J) robot kol üzerindeki herhangi bir mikroswitchin kapalı olup olmadığını saptamaya olanak sağlar. Mark3 denetleyici tüm veri yolları üzerindeki switchleri okuyabilir. Robotun bağlantısına göre denetleyici switchleri Çizelge 5.6'daki gibi okur.

Standart konfigürasyonda, denetleyici, üzerindeki A-F veri yollarına bağlı A-F motorlarına bağlıdır. I-Sorgu komutu XR–3 üzerindeki C-F motorlarını okur. A ve B veri yolu switchlerini okumak için, J-sorgu komutunu kullanılır[43].

I ve J sorgu komutları: Sorgu komutları daha çok QUESTION komutuyla çalışır. Her biri komut göndermek ve cevap almak için aynı altyordamı gerektirir. Denetleyici, ASCII komut kodlarından kaçınmak için, bilgisayara göndermeden önce cevaba 32 ekler ve alma altyordamı doğru cevabı elde etmek için 32 çıkartır.

Çizelge 5.6. Mark3 denetleyici switch okuma ve komut sistemi

Motor	Veri Yolu	Eksen	Denetleyici Switch okuması	Komut
A	A	Parmaklar	Evet	J
B	B	Bilek	Evet	J
C	C	Bilek	Evet	I
D	D	Dirsek	Evet	I
E	E	Omuz	Evet	I
F	F	Gövde	Evet	I
G	G	Ek Donatı	Evet	I
H	H	Ek Donatı	Evet	I

Fakat sorgu gönderildiğinde tek bir motoru göstermek yerine, komut gönderilir ve okumak istenen veri yoluna göre cevap yorumlanır. Denetleyici switch okuma ve komut sistemi çizelge 5.6 da verildiği gibi uygulanmaktadır.

XXX. QUESTION komutunun için kullanıldığı altyordamın numarasıdır. Altyordam bilgisayarın SORGUYA cevap almasını sağlar.

Çizelge 5.7. Mark3 Veri yolu bit atama

Veri Yolu	Bit		
C	0		
D	1		
E	2		
F	3		
G	4		
H	5		
	6	Kullanımda değil	
	7	Kullanımda değil	MSB

Cevapları yorumlama, SORGU komutunu kullanmanın anahtarı, denetleyicinin sadece 1 byte kullanarak iletişim kurduğunu bilmektir. Tek byte'lık cevap, her

mikroswitch için byte içerisinde 1 bit kullanılır. Veri yollarına atanan bitler Çizelge 5.7'de gösterildiği gibidir[44].

Bir mikroswitch açık ya da kapalı olabilir. Switch açık olduğunda, uygun gelen bit 1'dir. Bütün switchler açık olduğunda, desimal değeri 63'e karşılık gelen, altı bitin tamamı 1'dir. Cevabı alan altyordam doğru cevabı elde etmek için otomatik olarak 32'yi çıkarır). Eğer bir switch kapalıysa, uygun gelen bit 0'dır ve I- SORGU cevabı 63 eksi bitin değer karşılığıdır. Çizelge 5.8 bir switch kapalı ve diğer hepsi açık olduğunda ne olduğunu gösterir[45].

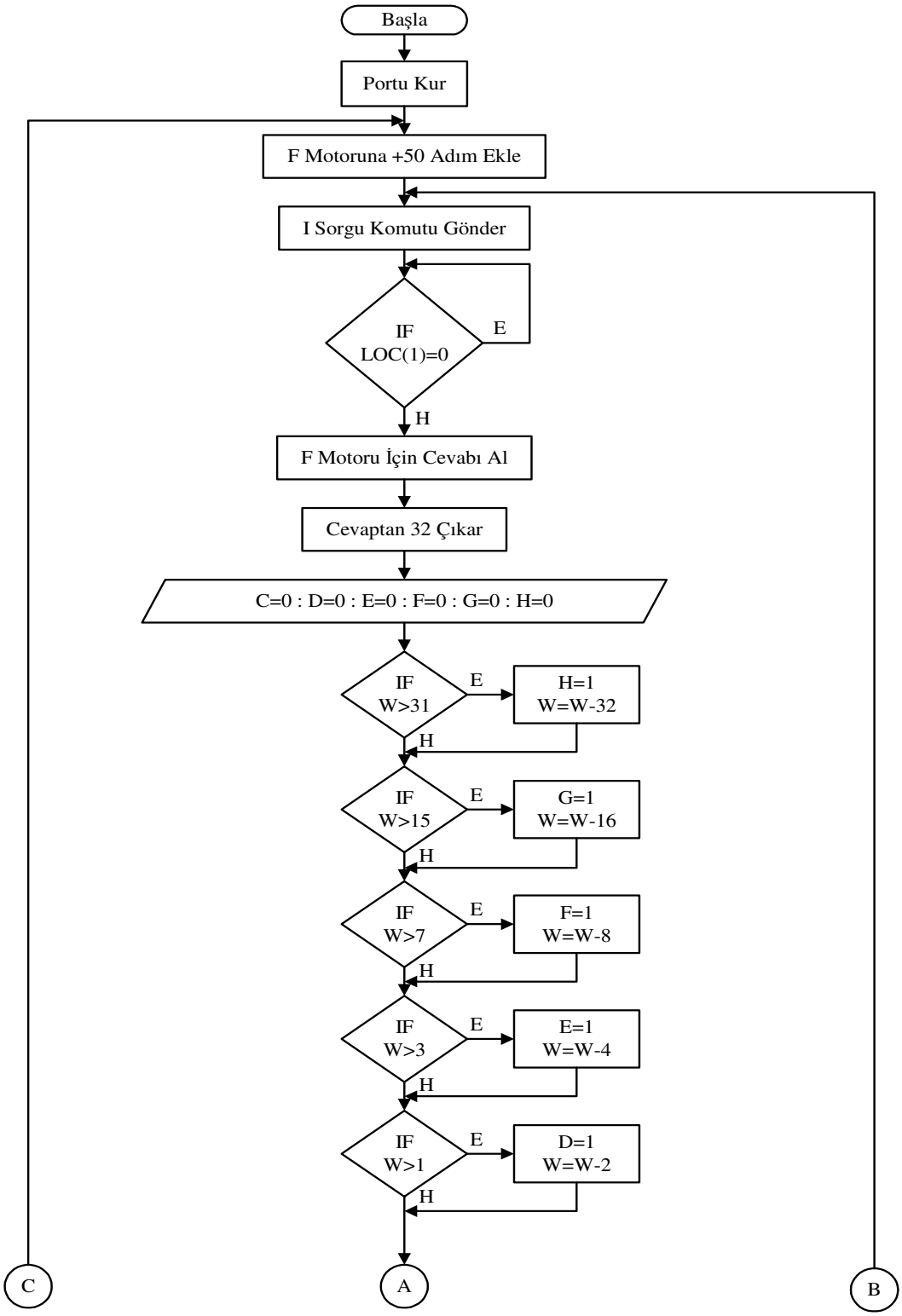
Çizelge 5.8. Bir switch kapalı iken diğer switchlerin durumu

Eğer bir switch kapalı diğerleri açıksa	SORGU komutuna dönen cevap
32 değeri göz ardı edilmekte	
IFX=C	62 (63–1)
IFX=D	61 (63–2)
IFX=E	59 (63–4)
IFX=F	55 (63–8)
IFX=G	47 (63–16)
IFX=H	31 (63–32)

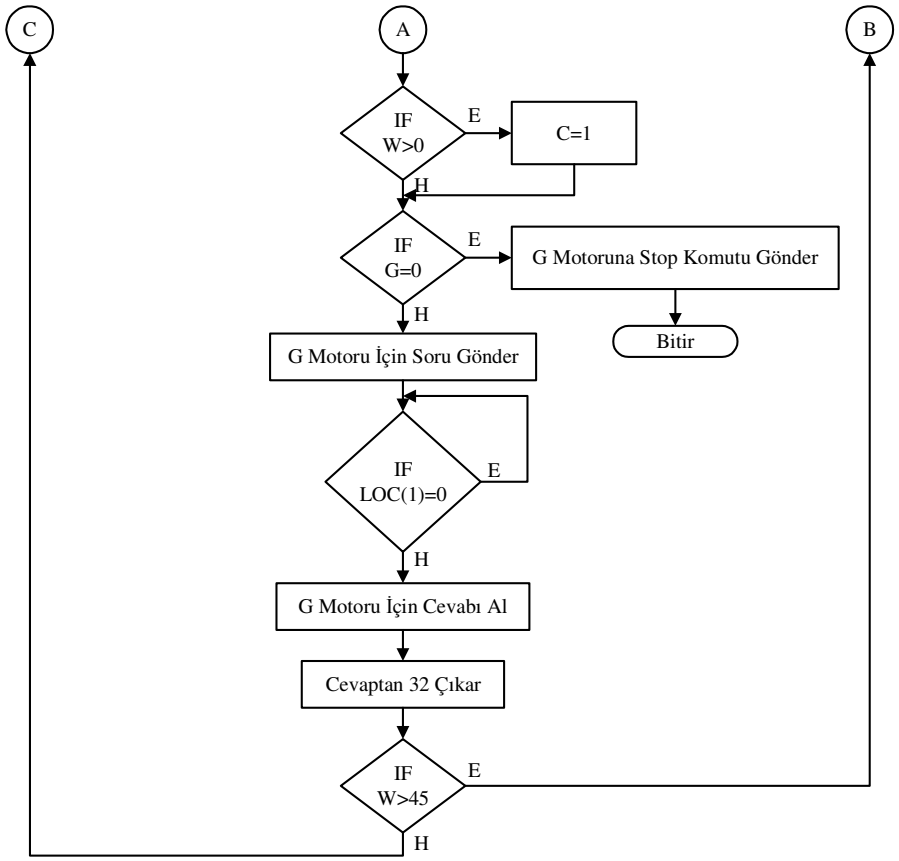
Bir switch kapalı olduğunda, o switchin değer karşılığı (yani o bitin basamak değeri) 63'den çıkartılır. Eğer herhangi bir zamanda sadece bir switchin kapalı olduğundan emin olunabilseydi her şey daha kolay olurdu. Bu nedenle I-SORGU komutunu gönderip, ilgilenilen switchin kapalı olup olmadığını belirlemek için cevabı kontrol edilir.

Herhangi bir zamanda, 1'den fazla switch kapalı olabilir. Alınan cevap 63 eksi tüm kapalı switchlerin değerleri toplamı olur. Örneğin, SORGU gönderildiğinde C, D ve E switchleri kapalıysa geri dönen cevap 56 olur. (63-(1+2+4)).

Bir switchi kontrol etmek için, kapalı olabilecek diğer switchler için cevap maskelenir. Bir sorgu gönderdikten sonra bu altyordam çağırılır ve cevabı alınır. W değişkeni cevabı gösterir. D mikroswitchi kapalı mı diye kontrol edilir[43].



Şekil 5.6. STOP ve SORGU komutları



Şekil 5.6. (Devam) STOP ve SORGU komutları

Şekil 5.6 denetleyici üzerindeki G veri yoluna bağlı ek donanıma erişimi amaçlar. Donanımı çalıştırır ve üzerindeki kameranın mikroswitchi kapattığını anlamak için SORGU komutunu kullanır. Mikroswitch kapatıldığında, G motoruna bir STOP komutu gönderilir, motoru durdurmak için G hata kaydedicisi sıfıra getirilir[44].

G motorunu başlatmak için G motorunun hata kaydedicisine 50 adım ekleyerek START komutu ve switchleri kontrol etmeye derhal başlamak için SORGU komutu gönderilir. SORGU cevabını yorumlamak için altyordam çağırılır. G switchi kapalı mı (G=0) diye kontrol edilir. Eğer kapalı ise bilgisayar STOP komutu gönderir. G motorunun hata kaydedicisinin durumu kontrol edilir ve komutu gönderip cevabı almak için altyordamı çağırır. Eğer G motorunun hata kaydedicisinde diğer 50 adım için yeteri kadar yer yok ise bilgisayar SORGU komutuna geri gönderir. Hata kaydedicisine 50 adım daha ekleyip G motorunu hareketine devam ettirmek için döngü başa döner.

G switchi bulunduğunda çalıştırılan STOP komutudur. Motoru durdurarak G motorunun hata kaydedicisini sıfıra getirir. START komutunda 50 adımlık artış kullanıldığında daha karışık programlarda 95 adımla başlamak daha iyidir ve hata kaydedicisinin her defasında mümkün olduğunca dolu tutulması gereklidir.

Bir sorgu komutuna verilen cevapları yorumlamak için daha etkili bir altyordam, BASIC'in parçası olan AND operatörünü kullanır. Kontrol edilen switche uygun bit için spesifik olarak kontrol imkanı sağlar. AND operatörünün kullanmanın ne kadar kullanışlı olduğunu göstermek için, aşağıdaki satır tüm altyordamı değiştirir,

IF W AND 16 THEN XXX ELSE XXX

AND Operatörü uzun açıklama gerektiren prensiplere dayanır AND operatörünün kullanımı için, açık olan her switchin uygun tam sayı değerleri:

Çizelge 5.9. Her switch için uygun tam sayı değerleri

Switch	Tam Sayı
C	1
D	2
E	4
F	8
G	16
H	32

2. Tek bir mikroswitchin durumunu kontrol etmek için AND operatörü her kullanıldığında, 16 yerine o switch için tam sayı değeri yerine konulur. Robot üzerinde bulunan her switch için uygun tamsayı değerleri çizelge 5.9 da gösterildiği gibidir. Örneğin, H switchi kontrol edilmek isteniyorsa satır şöyle olmalıdır.

IF W AND 32 THEN H=1 ELSE H=0

Mikroswitch kontrol edildiğinde kapalı ise, bilgisayar ELSE komutu işler (Örn. H=0) ve mikroswitch kontrol edildiğinde açıksa, bilgisayar THEN komutu icra eder (Örn. H=1).

5.3.4. Donanım başlangıç pozisyonu (Hardhome)

Sorgu komutunun mikroswitchleri okumak için nasıl çalıştığı görüldü. Birçok sebepten dolayı SORGU kullanılmak istenebilir, ama önemli neden, tekrarlanabilir başlangıç robot pozisyonunu bulmaktır.

Robotu A noktasından B noktasına hareket ettirecek bir program yazmak istenebilir. Robot her çalıştırıldığında aynı noktadan başlamadığı sürece o program bir defadan fazla çalışmaz. Böyle bir başlangıç noktası kurmanın yolu, her ekseni mikroswitchin harekete geçirdiği pozisyona gönderip orada durdurmaktır.

Bu switch tanımlı robot pozisyonuna "donanım başlangıç pozisyonu" veya kısaca HARDHOME denilir. D, E ve F motorları için mikroswitchleri bulur ve her motoru bu pozisyonda durdurur. Aslında, her mikroswitchin merkezini bulur, switchin kapandığı ve tekrar açıldığı noktalar arasındaki orta noktada durdurur. Eğer programların tam olarak tekrar etmesi isteniyorsa bu kesinlik hayatidir.

Pratikte, programlamaya başlamadan önce bir kez HARDHOME yordamı çağrılmalıdır. HARDHOME yordamı sadece üç switchi kurar. Diğer üçünü kurmak için değiştirilmek istenebilir. Bu durumda denetleyicinin A ve B veri yollarının adreslemesi. J-SORGU komutu ile yapılır[44].

5.3.5. INPUT/ OUTPUT kullanımı

Mark3 denetleyici sekiz TTL girişi, sekiz TTL çıkışı ve AUX veri yolu adında iki enkoder kullanımsız motor güç veri yolu sahiptir. Bu özellikleri tamamıyla kontrol etmek için sekiz komut mevcuttur. Bu komutların kullanımı, motor hareketlerini kontrol etmek için geliştirilen formata yakındır. Aslında, aynı altyordamları kullanırlar[43].

Mark3 denetleyicinin giriş ve çıkışları TTL uyumludur ve düşük akım aleti (5 V DC'de 15 ma'den az) gibi davranır. Bu veri yollarına harici bir voltaj uygulanmaz.

I/O ön panelinde sağlanan 5 V kaynağı kullanılır. Girişler kararlı bir mantıksal konumda beklemelerini sağlamak maksadıyla içeriden direnç üzerinden 5 V bağlanmıştır.

Girişleri kontrol etmek için, bir switch giriş veri yoluna ve toprak terminaline bağlanır. Switch kapalı olduğunda giriş KAPALI olur, switch açık olduğunda, giriş AÇIK olur.

Veri yolu açık olduğunda çıkış ALÇAK, veri yolu kapalı olduğunda çıkış YÜKSEK olur. Yani çıkışlar ALÇAK aktiftir.

J-Sorgu komutu – 1'den 4'e kadar girişleri okur.

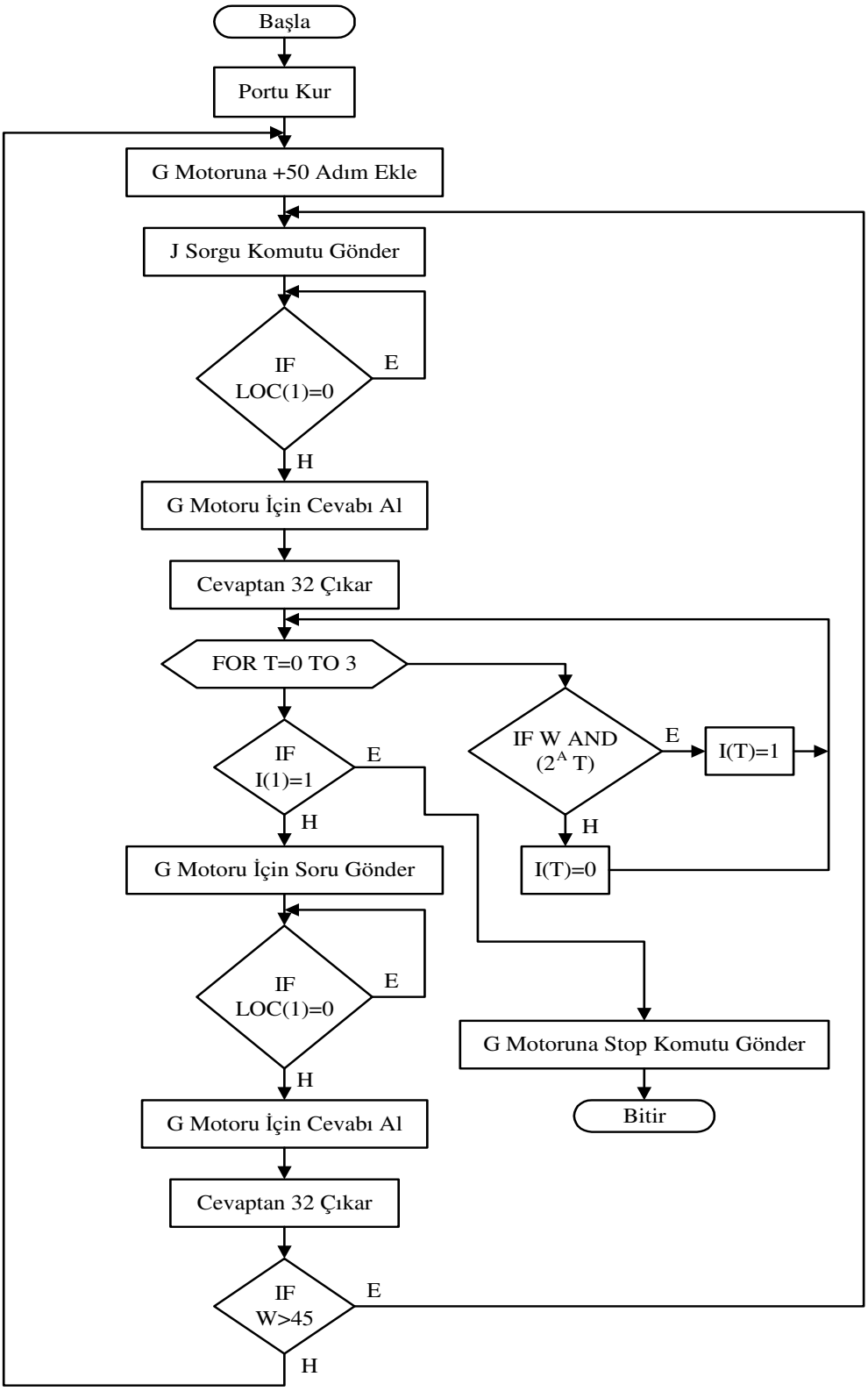
I komutu gibi J komutu da ilk dört biti 1'den 4'e kadar olan girişlerin durumunu gösteren bir byte döndürür.

I komutuyla kullanılan yordamın aynı şekli, girişin durumuna bağlı alarak bir dizi değişkeni bir'e veya sıfır'a kurmak için kullanılabilir. Burada girişleri göstermek için I değişkeni kullanılacaktır[43]. Girişlerin Bit değerleri çizelge 5.10 daki gibidir.

I(0) = Input 1, I(1) = Input 2, vb.

Çizelge 5.10. Girişlerin bit değerleri

Bit	Giriş
0	1
1	2
2	3
3	4



Şekil 5.7. G eksen motorunu, örneğin bir taşıma bandı, INPUT #2 açılana kadar (YÜKSEK TTL düzeyi) hareket ettirir.

Gerçek çalışmada, bir parça bulunduğunda YÜKSEK düzeye çıkan bir sensör giriş veri yoluna bağlı olur. Parça sensör pozisyonuna ulaştığında bant duracaktır. J komutuna denetleyici tarafından gönderilen byte ilk dört bitin her birini kontrol ederek hangi girişlerin AÇIK olduğunu belirleyen yordamı uygular. Giriş açık mı diye kontrol eder. Eğer açıksa, STOP komutu işletilir ve G veri yolu kapatılır[43].

K- Sorgu komutu – 5'ten 8'kadar girişleri okur. K komutunun uygulaması dört düşük değerlikli bitin de 5 ten 8'e kadar girişlerin durumunu içerdiği geri döndürülen byte'tan başka J komutuyla aynıdır.

Her bir girişin durumunu belirleyen yordam Şekil 5.7'deki döngü ile aynıdır. J kullanmak yerine K kullanımı ve L, M, N, O Komutları – AUX veri yollarının durumunu kurar.

Mark3 denetleyici üzerindeki iki AUX veri yoluna (+) ya da (–) 20 VDC motor gücü sağlar. AUX veri yollarının yanındaki tersleme anahtarlarının kurulmasıyla belirlenen polariteyle, dört komut (L, M, N, O) AUX veri yollarını açık ya da kapalı olarak kurar, çıkışın polaritesini kurmazlar. Bu komutlar MARK3 denetleyiciden yanıt istemeyen basit bir byte'lık komutlardır. Dört komutun tamamının uygulaması aynıdır[45]. Şekil 5.8 AUX veri yolu #2 nin açılıp kapanması hakkında fikir vermektedir.

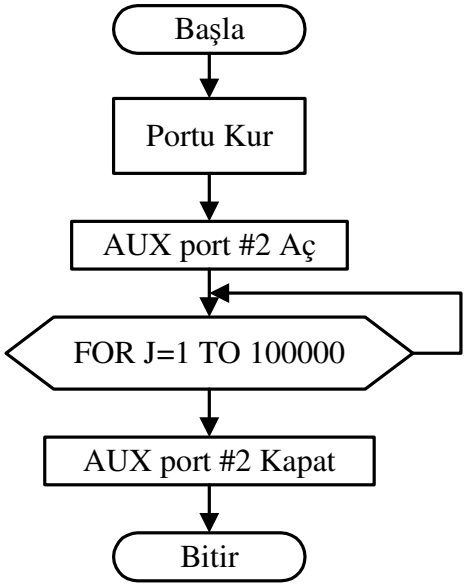
AUX veri yolu #2'yi açacak N komutunu gösterir. AUX komutlarının etkisini görmek için kullanıcıya bir gecikme zamanı sağlar. AUX veri yolu #2'yi kapatacak O komutunu gösterir.

Mark3 denetleyici, AUX veri yoluna bağlı, açık olup olmadıklarını göstermek için iki tane gösterge lambasına sahiptir.

AUX veri yollarına hiçbir şey bağlanmasa bile, bu program çalışabilir ve AUX veri yolu #2 lambasını yakıp sonra kapatır.

P ve R Komutları çıkış hatlarını yükseğe (KAPALI) ve alçağa (AÇIK) olarak kurar.

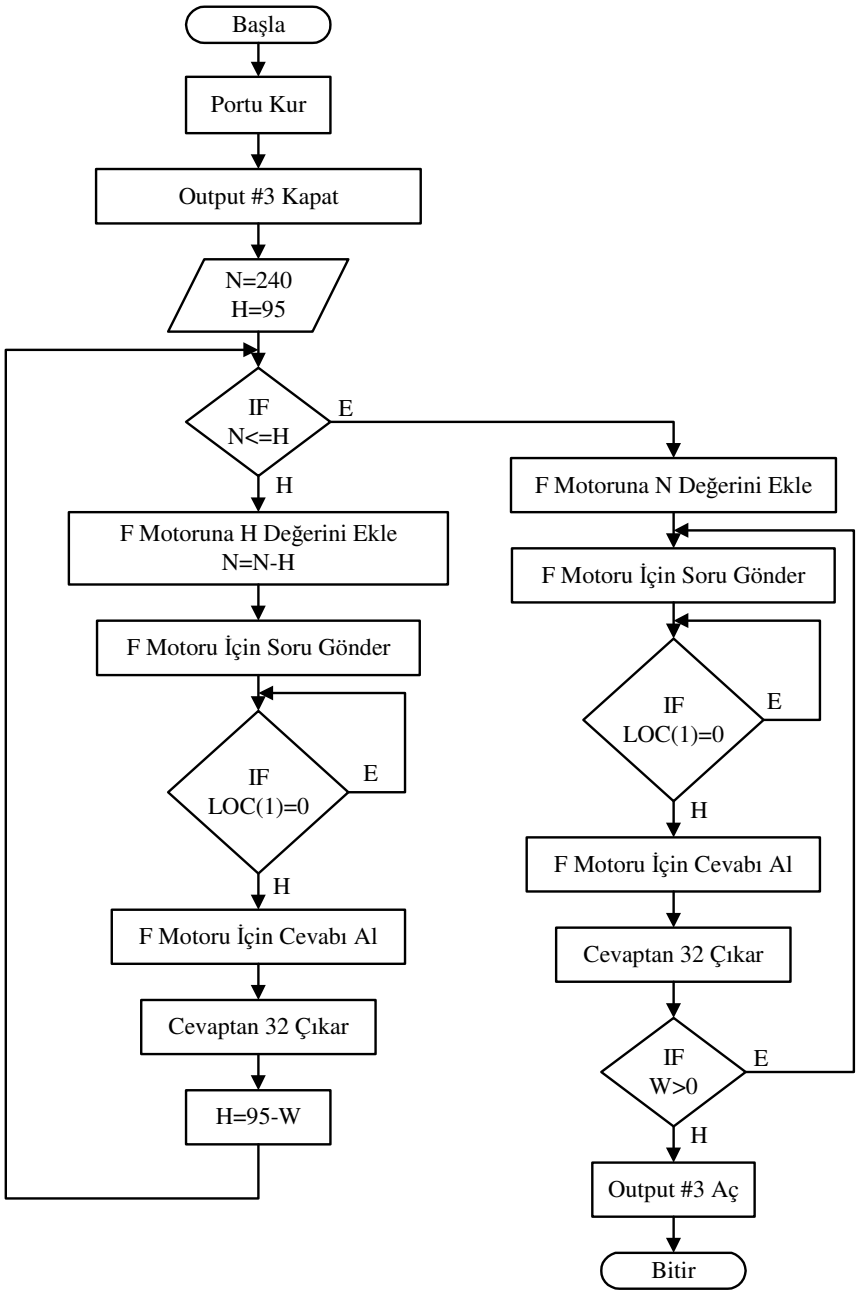
Mark3 denetleyici üzerinde her biri komutla kurulup sıfırlanabilen sekiz TTL düzeyinde çıkış kullanılabilir. P komutu denetleyiciye bir çıkışı kapatması için talimat verir (çıkışı yükseğe çeker).



Şekil 5.8. AUX veri yolu #2 nin açılıp kapanması

R komutu denetleyiciye bir çıkışı açması için talimat verir (çıkışı alçağa çeker). Her iki komutun formatı, değişebilen veri yolu numarasıyla takip edilen komut harfidir[45]. Bununla ilgili olarak oluşturulan döngü şekil 5.8 de verilmiştir.

Robotun materyal tutma işlemini tamamladığını göstermek için durduğunda, başlangıç için output #3'ü kapatılıp sonra açılmak istensin. Şekil 5.9 bunu ifade etmektedir.



Şekil 5.9. AUX veri yolu #2 nin uzun program uygulamasında açılıp kapanması

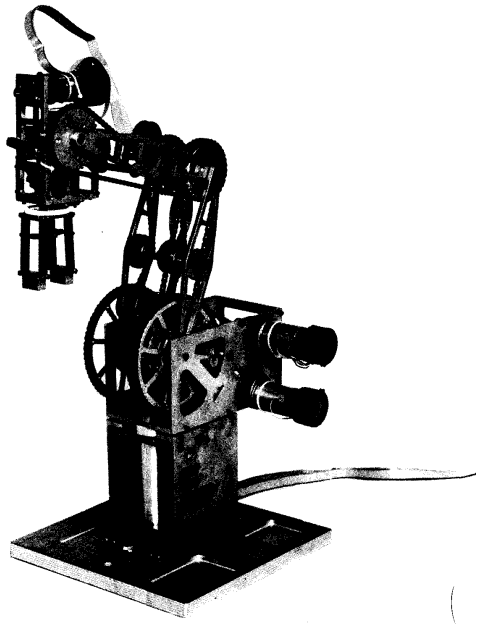
Şekil 5.9 motor hareketini başlatmadan önce output #3'ü kapatmak için denetleyici yönetilir. Uzun hareket uygulanır. Program, robot tam durmaya gelene kadar bekler. Hareketin tamamlanmasından sonra, port #3'ü açmak için komut içerir[43].

5.4. Kinematik İnceleme

Kinematik inceleme temelde iki alt problemden meydana gelir. Bunlar, düz kinematik problem ve ters kinematik problemdir. Düz kinematikte amaç açısal konum, hız ve ivme gibi kinematik parametrelerle robot elin ulaşacağı konum ve yönlenmenin veya izleyeceği yörüngenin belirlenmesidir. Ters kinematikte ise, çeşitli geometrik parametreler ve referans koordinat sistemine göre pozisyonu tanımlanan robot elin istenilen konuma ulaşabilmesi için gerekli olan eksen değişkenlerinin belirlenmesidir[47].

5.4.1. XR3 için kinematik inceleme

Bu tez çalışmasının bir parçası olan bir deneysel eğitim robotu olan RHINO XR3, Resim 5.5'de görülmektedir.

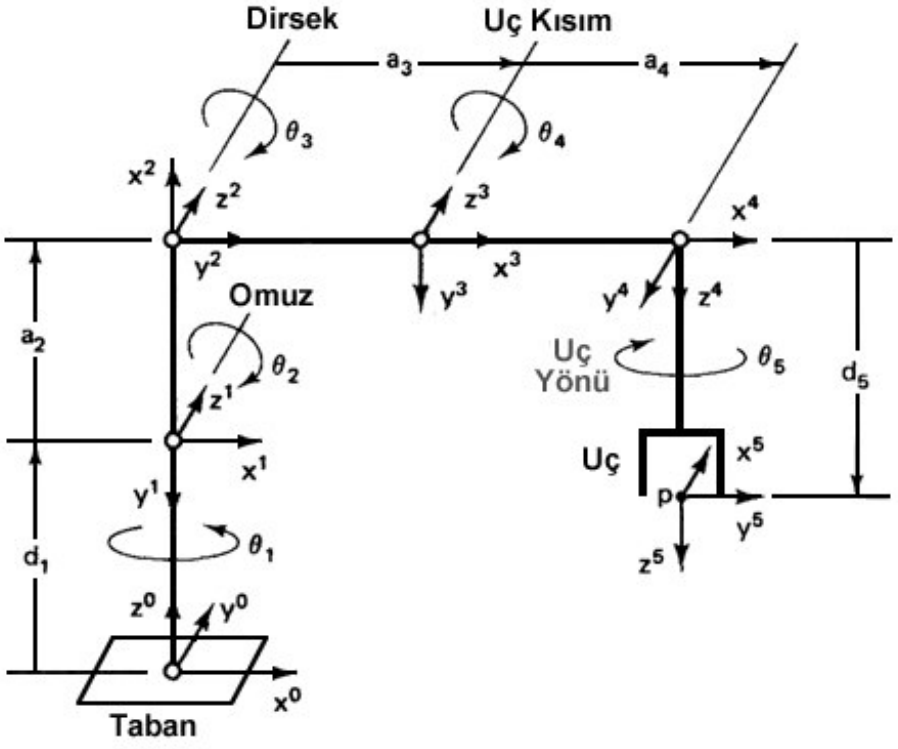


Resim 5.5 Beş eksenli mafsallı robot (RHINO XR3)

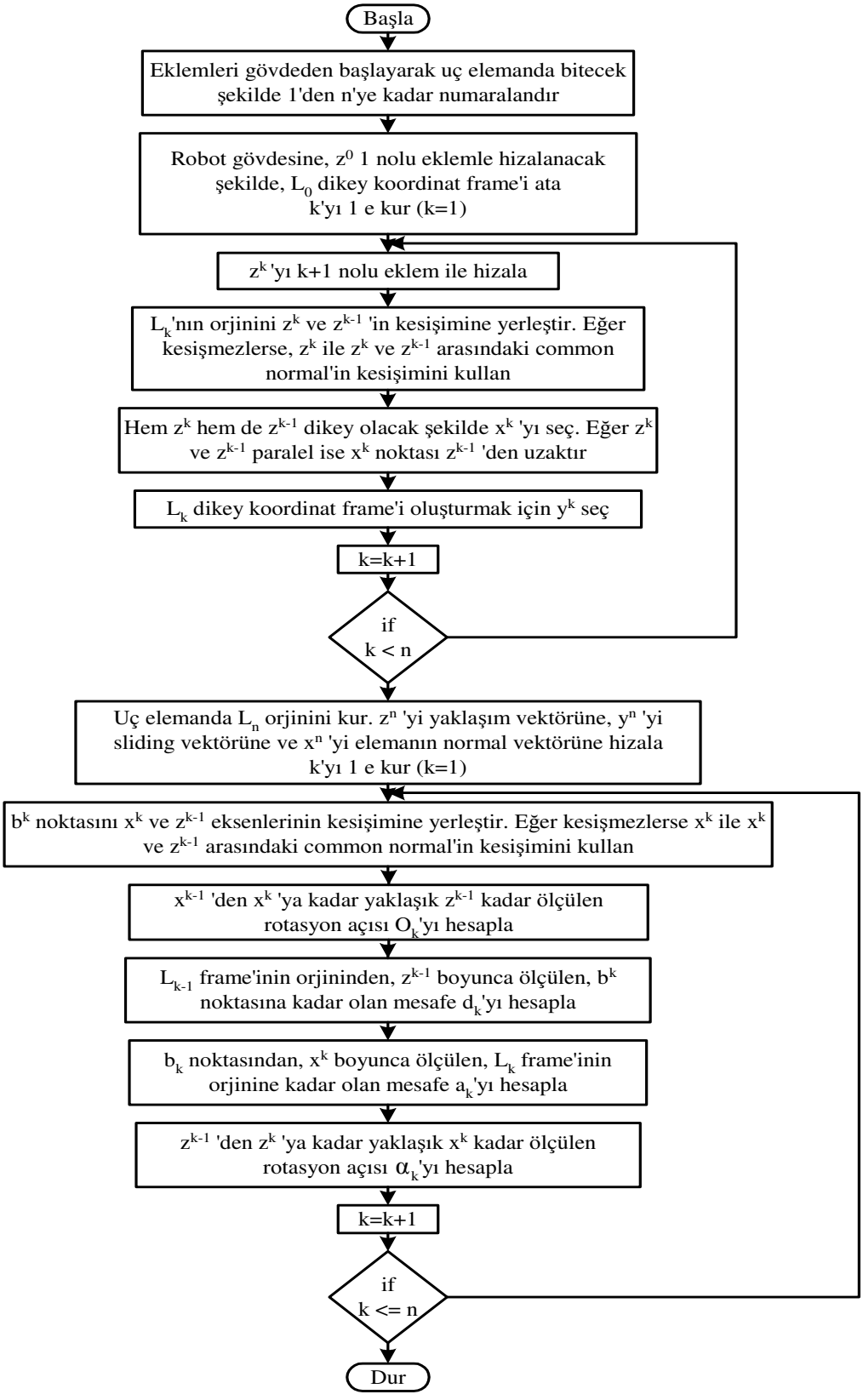
XR-3'ün fiziksel yapısında, gövde motoru düşey olarak monte edilmiştir ve sabittir. Buna karşın omuz, dirsek ve uç motorları ana eksen çevresinde dönen gövde üzerine yatay olarak monte edilmişlerdir.

Link- koordinat diyagramı: XR-3 robotun link koordinat diyagramı Şekil 5.10 da gösterilmiştir. Bu tip robotların kinematik analizleri yapılırken sıklıkla başvurulan algoritmalardan bir tanesi de Şekil 5.11'de sunulan Denavit-Hartenberg (D-H) algoritmasıdır[48]. Şekil 5.10 da verilen link koordinat diyagramı bu algoritmanın 0'dan 7'ye kadar olan adımları uygulanarak elde edilir.

Ardından, D-H algoritmasının k=1 ile başlayan 8'den 13'e kadar olan adımları uygulanır. Şekil 5.10 un kullanımı Çizelge 5.11'de gösterilen kinematik parametreler setini verir. XR–3 mafsallı robot olduğundan ortak değişkenler vektörü $q = 0$ 'dır[49]. Çizelge 5.11 in son sütununda verilmiş olan q vektörünün değeri, Şekil 5.10'daki link-koordinat diyagramında görülmekte olan soft-home pozisyonuna uygundur.



Şekil 5.10. Beş eksenli mafsallı robotun link koordinat diyagramı (RHINO XR–3).



Şekil 5.11. Denavit- Hartenberg algoritması

Çizelge 5.11. Beş eksenli mafsallı robot'un kinematik parametreleri

Açı	θ	d	a	α	Ev (Home)
1	q_1	d_1	0	$-\pi/2$	0
2	q_2	0	a_2	0	$-\pi/2$
3	q_3	0	a_3	0	$\pi/2$
4	q_4	0	a_4	$-\pi/2$	0
5	q_5	d_5	0	0	$-\pi/2$

Araç boyunu gösteren d_5, robotun ucuna hangi tür aletin yerleştirildiğine bağlı olarak robottan robota değişiklik arz edebilir. XR–3 robotunun eksen uzaklıkları d ve link uzunlukları a aşağıdaki gibidir.

$$d= [260.4,0.0,0.0,0.0,171.5]^T \text{ mm} \quad (5.1)$$

$$a= [\ 0.0,228.6,228.6,9.5,0.0]^T \text{ mm} \quad (5.2)$$

Burada 171,5 olarak alınan d_5 değeri robotun standart parmaklarına uygundur. Tercihen farklı uzunluklara sahip parmaklar da mevcuttur. Şayet alüminyum temel üzerine monte edilmesi tercih ediliyorsa o zaman $d_1 = 279,4$ mm olur.

Kol matrisi: Şekil 5.10 da robotun dayanak kolunun hesaplanması problemi bulunmaktadır. Söz konusu problem bilekten ikiye bölmek suretiyle daha basit iki alt probleme indirgenerek ele alınabilir.

Kinematik denklemlerdeki ifadelerin basitleştirilmesi maksadıyla $C_k = \cos q_k$ ve $S_k = \sin q_k$ notasyonun kullanılması uygun görülmektedir. Bu durumda $C_{kj} = \cos(q_k + q_j)$ ve $S_{kj} = \sin(q_k + q_j)$ olmaktadır[50].

T(Wrist Base) ifadesi, taban düzlemine bağlı olarak bilek düzleminin pozisyonunu ve eğimini sağlamaktadır.

$$
\begin{aligned}
T_{base}^{wrist} &= T_0^1 T_1^2 T^3 \\
&= \begin{bmatrix} C_1 & 0 & -S_1 & 0 \\ S_1 & 0 & C_1 & 0 \\ 0 & -1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C_2 & -S_2 & 0 & a_2C_2 \\ S_2 & C_2 & 0 & a_2S_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C_3 & -S_3 & 0 & a_3C_3 \\ S_3 & C_3 & 0 & a_3S_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
&= \begin{bmatrix} C_2C_2 & -C_1S_2 & -S_1 & a_2C_1C_2 \\ S_1C_2 & S_1S_2 & C_1 & a_2S_1C_2 \\ -S_2 & -C_2 & 0 & d_1 - a_2S_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C_3 & -S_3 & 0 & a_3C_3 \\ S_3 & C_3 & 0 & a_3S_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
&= \begin{bmatrix} C_1C_{23} & -C_1S_{23} & -S_1 & C_1(a_2C_2 + a_3C_{23} \\ S_1C_{23} & -S_1S_{23} & C_1 & S_1(a_2C_2 + a_3C_{23} \\ -S_{23} & -C_{23} & 0 & d_1 - a_2S_2 - a_3S_{23} \\ 0 & 0 & 0 & 1 \end{bmatrix}
\end{aligned}
\tag{5.3}
$$

T(Wrist Base) ifadesinin sağlanması, dönüşüm matrisinin soft-home pozisyonu için uygulanması ile yapılabilir. Çizelge 5.11'den, soft-home pozisyonu için $q = [0, -\pi/2, \pi/2, 0, -\pi/2]^T$ değeri T(Wrist Base) ifadesine uygulanacak olursa denklem (5.4) de verilen matris elde edilir ve link-koordinat diyagramına uyumluluk görülür[49].

$$
T_{base}^{wrist}(\text{hom}\,e) = \left[\begin{array}{ccc:c} 1 & 0 & 0 & a_3 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & d_1 + a_2 \\ \hdashline 0 & 0 & 0 & 1 \end{array}\right] \tag{5.4}
$$

Elde edilen dönüşüm matrisinin birinci sütunu, soft-home pozisyonunda, uç yelpaze hareketi çerçevesinin x^3 ekseninin taban çerçevede (1, 0, 0) koordinatlarına sahip olduğunu belirtir.

İkinci sütun R uç yelpaze hareketi çerçevesinin y^3 ekseninin taban çerçevede (0, 0,-1) koordinatlarına sahip olduğunu belirtir ve bu ikinci sütun z^0 taban eksenin karşı yönünü göstermektedir.

R'nin son sütunu uç yelpaze hareketi çerçevesinin z^3 ekseninin tabana göre (0, 1, 0) koordinatlarına sahip olduğunu belirtir.

Böylece z^3, y^0 taban ekseni gibi aynı yönü gösterir. Son olarak, vektör p pozisyonu uç yelpaze hareketi çerçevesi L_3'ün koordinatlarının merkezinin L_0 taban çerçevesine bağlı olarak $(a_3, 0, d_1 + a_2)$ olduğunu gösterir[51]. Böylece L_3 tabanın a_3 kadar önünde ve bir ve $d_1 + a_2$ kadar üzerindedir. Bu, şekil 5.10 ile uyumludur.

Bir sonraki adım bileğe bağlı aracın pozisyonunu ve oryantasyonunu belirler:

$$
\begin{aligned}
T_{wrist}^{tool} &= T_4^3 T_6^5 \\
&= \begin{bmatrix} C_4 & 0 & -S_1 & a_4C_4 \\ S_4 & 0 & C_4 & a_4S_4 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C_5 & -S_5 & 0 & 0 \\ S_5 & C_5 & 0 & 0 \\ 0 & 0 & 1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
&= \begin{bmatrix} C_4C_5 & -C_4S_5 & -S_4 & a_4C_4 - d_5S_4 \\ S_4C_5 & -S_4S_5 & C_4 & a_4S_4 - d_5C_2 \\ -S_5 & -C_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\end{aligned}
\tag{5.5}
$$

Bileğe bağlı aracın pozisyonu ve oryantasyonu için nihai ifade beklendiği gibi sadece son iki birleşik değişkene bağlıdır. Bu noktada home pozisyondaki transformasyonun bir değerlendirmesi yapılarak yeniden kısmi bir kontrol yapılabilir. Bu kontrol işlemi, L_3'e bağlı L_5 düzleminin pozisyon ve oryantasyonunu sağlamalıdır. L_5 aracın gövdeye bağlı oryantasyonunu ve pozisyonunu bulmak için olduğu gibi kalmaktadır. Bu, bölünmüş iki bilek faktörünün birbiriyle çarpılmasıyla elde edilir[52]. Aşağıda beş eksenli mafsallı robotun doğrudan kinematik probleminin çözümü olan nihai sonuç görülmektedir.

$$
\left[\begin{array}{ccc|c} C_1C_{234}C_5 + S_1S_5 & -C_1C_{234}S_5 + S_1C_5 & -C_1S_{234} & C_1(a_2C_2 + a_3C_{23} + a_4C_{234} - d_5S_{234}) \\ S_1C_{234}C_5 - C_1S_5 & -S_1C_{234}S_5 - C_1C_5 & -S_1S_{234} & S_1(a_2C_2 + a_3C_{23} + a_4C_{234} - d_5S_{234}) \\ -S_{234}C_5 & S_{234}S_5 & -C_{234} & d_1 - a_2S_2 - a_3S_{23} - a_4S_{234} - d_5C_{234}) \\ \hline 0 & 0 & 0 & 1 \end{array}\right]
\tag{5.6}
$$

Gövdeye bağlı aracın pozisyonu ve oryantasyonu için ortaya konan nihai ifadenin bütün kinematik parametrelere nasıl bağlı olduğu görülebilir. Soft-home pozisyonundaki kol matrisinin bir değerlendirmesi yapılmak suretiyle kısmi bir kontrol yapılabilir.

Çizelge 5.11 'den $q = [0, -\pi/2, \pi/2, 0, -\pi/2]^T$. Böylece:

$$T_{base}^{tool}(\text{hom}\,e) = \left[\begin{array}{ccc|c} 0 & 1 & 0 & a_3 + a_4 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & d_1 + a_2 - d_5 \\ \hline 0 & 0 & 0 & 1 \end{array}\right] \tag{5.7}$$

Şekil 5.10 incelendiğinde link- koordinat diyagramıyla uyuştuğu görülecektir. Rotasyon matrisinin birinci sütunu, soft-home pozisyonda, x^5 (normal vektör) ekseninin gövdenin y^0 ekseni gibi aynı yönü gösterdiğini işaret eder. *R*'nin ikinci sütunu y^5 araç ekseninin (hareketli vektör) gövdenin x^0 ekseni gibi aynı yönü gösterdiğini işaret eder. *R*'nin son sütunu z^5 araç ekseninin (yaklaşan vektör) gövdenin z^0 ekseninden zıt yönü gösterdiğini işaret eder. Son olarak pozisyon vektörü *p* robot soft-home pozisyondayken gövdenin önündeki $a_3 + a_4$ ve üzerindeki $d_1 + a_2 - d_5$ uzaklığına yerleştirilen uç elemanı koordinat düzleminin kaynağını işaret eder[53].

5.4.2. Eksen ilişkilendirme

Şekil 5.10'deki gibi θ_{23} açısı genel bir dirsek açısı olarak düşünülebilir. Bu dirsek açısı çalışılan yüzeye ya da x_0y_0 düzlemine bağlı olarak ölçülür. $\theta_{23} = 0$ olduğunda önkol yataydır ve doğrudan dışarıyı gösterir, oysa $\theta_{23} = \pi/2$ olduğunda önkol dikey olur ve yukarıyı gösterir. Benzer biçimde θ_{234} açısı genel bir uç elemanı eğim açısı olarak düşünülebilir. Uç elemanı eğim açısı çalışılan yüzeye veya x_0y_0 düzlemine bağlı olarak ölçülür. $\theta_{234} = 0$ olduğunda uç elemanı dikeydir ve aşağıyı gösterir, buna karşın $\theta_{234} = \pi/2$ iken yataydır ve dışarıyı gösterir[54].

XR3 gibi diğer bazı robotlardaki mekanik bağların, uç elemanı eksenlerine, dirseğe ve desteğe ait hareketlerin paralel bir kol mekanizması aracılığıyla bağlanması suretiyle tasarlandığı görülür[55].

Destek motoru $\Delta\theta_2 = B$ rotasyonu elde etmek için aktive edildiğinde, dirsek hareketleri aynı anda karşı $\Delta\theta_2 = -B$ rotasyonunda hareket eder. Böylece destek motoru aktive edildiğinde θ_3 dirsek açısı sabit kalır. Sonuç olarak gövde koordinat düzlemine bağlı önkol oryantasyonu destek motoru tarafından etkilenmez[55].

Benzer biçimde, dirsek motoru $\Delta\theta_3 = y$ rotasyonunu elde etmek için aktive edildiğinde, uç elemanı aynı anda karşı $\Delta\theta_3 = y$ rotasyonunda hareket eder. Böylece her iki durumda da yani hem destek hem de dirsek motor aktive edildiğinde θ_{234} genel uç elemanı açısı sabit kalır. Bu durum, gövde koordinat düzlemine bağlı araç oryantasyonunun destek ve dirsek motorlarınca etkilenmediği anlamına gelir[56]. Bu ilişkilendirme sonuçları eksen düzeyinde robotları kontrol ederken faydalı olabilirler. Örneğin, sıvı dolu bir kap araç tarafından taşındığında yana eğildiği için dirsek motor ya da destek motor aktive edildiğinde dökülmeyecektir.

D-H algoritması kullanan bir robot kolun kinematik analizi aralarında var olduğu varsayılan ilişkilendirmeye sahip olmayan eksenlerin bağımsız kontrolüne dayanır. Mekanik ilişkilendirme mevcut olduğunda, ilişkilendirme matrisi C ve kesin bir matris P'ye sahip eksen hareket komutu $\Delta\theta$ işlemiyle yazılım düzeyinde silinebilir. Bu işlem bir Δh komut vektörü üretir. Bu komut vektörü gövde ile başlayan ve uç elemanı yuvarlanması ile sona eren, her bir motora gönderilen sayısal kodları belirler. XR3 için, ortak alanda $\Delta\theta$ derecelerin hareketlerini üretmeyi gerektiren sayısal kodlar aşağıdaki gibi formüle edilebilirler.

$$\Delta h = \begin{bmatrix} \rho_1 & 0 & 0 & 0 & 0 \\ 0 & \rho_2 & 0 & 0 & 0 \\ 0 & 0 & \rho_3 & 0 & 0 \\ 0 & 0 & 0 & \rho_4 & 0 \\ 0 & 0 & 0 & 0 & \rho_5 \end{bmatrix}^{-1} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \Delta\theta = P^{-1} C \Delta\theta \tag{5.8}$$

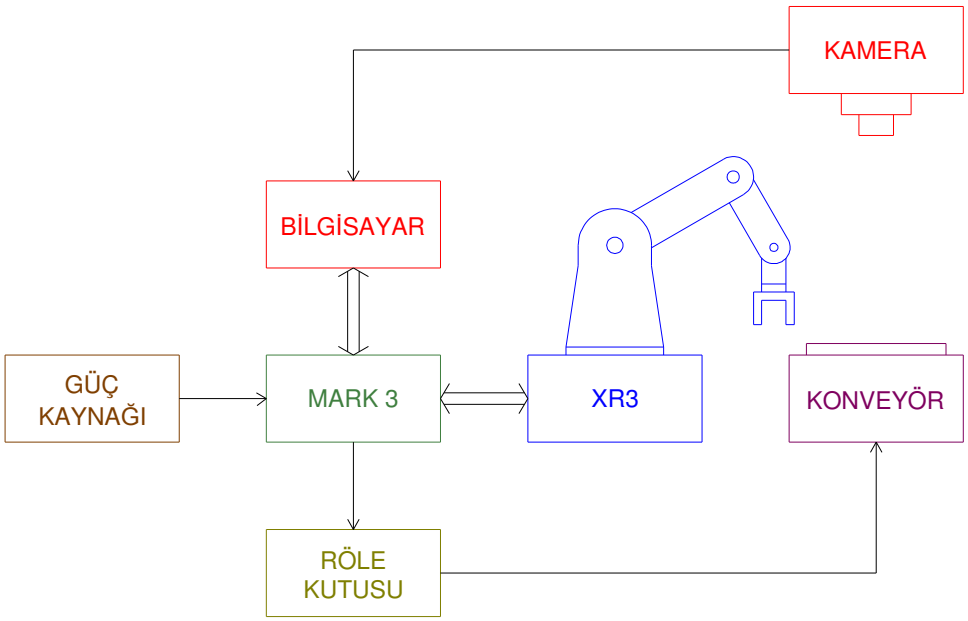
Burada $\Delta\theta$, derecelerle ifade edilen beş eksen açıdaki istenilen değişiklikleri belirleyen beş bileşene sahip bir vektördür. C matrisi katsayısı bir ilişkilendirme matrisidir.

Bu kapalı diyagonal ifadeler paralel kol mekanizması tarafından getirilen ilişkilendirmeyi etkili bir biçimde yok eder. P^{-1} matrisi kesin bir diyagonal matrisin tersidir. Burada P sayısal kodlamada derecelerle ifade edilen her bir eksenin kesinliğini belirler[49].

Yukarıda özellikleri tanıtılan sistem elemanları, verili elektrik, donanım ve yazılım bağlantıları ile denklemlerde ifade edilen fiziksel hareketleri gerçekleştirir. Sonraki bölümde burada matematiksel olarak hesaplamaları ortaya konan sürecin laboratuvar uygulaması yapılarak elde edilen bulgular yorumlanacaktır.

6. SİSTEMİN ÇALIŞMASI VE BULGULAR

6.1. Sistemin Tanımlanması



Şekil 6.1. Sınıflandırma sistemi blok şeması

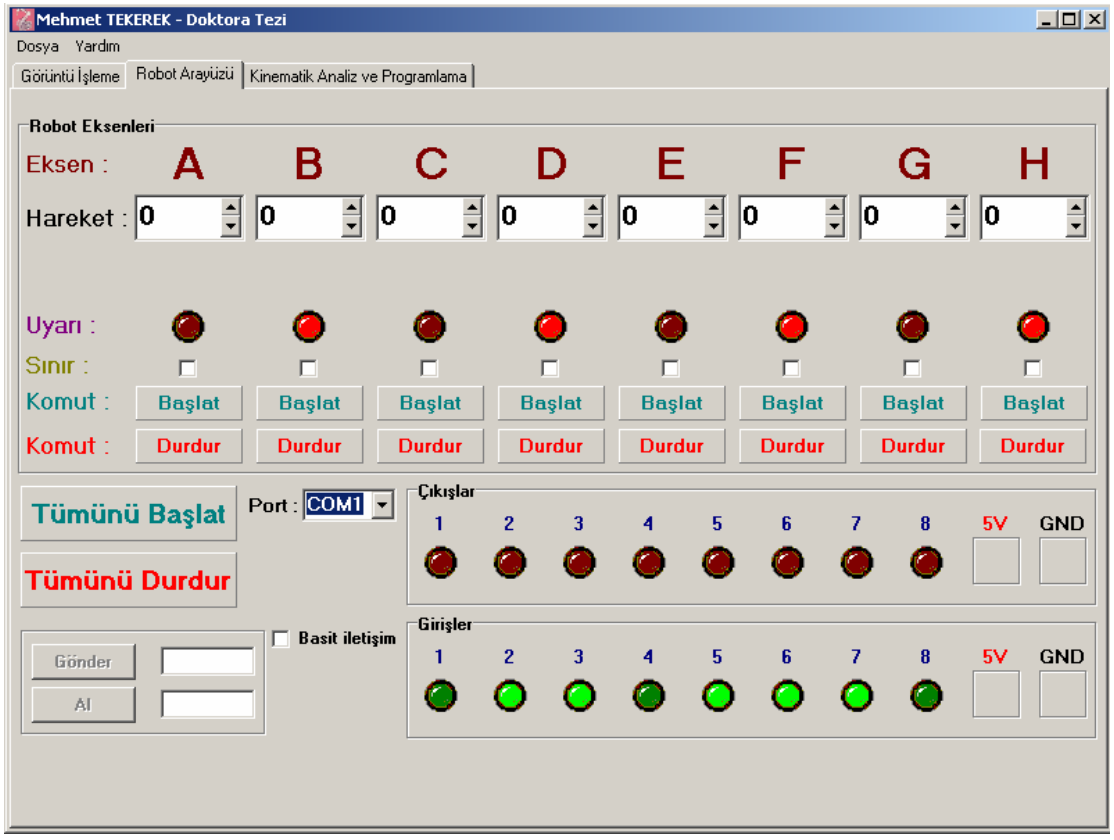
Deneysel olarak geliştirilen sınıflandırma sistemi, geliştirilen yazılımın kontrol arayüzü tarafından kontrol edilen bir konveyör bant, sürekli görüntü algılayıcı kamera, XR3 robot manipülatör ve Mark3 denetleyiciden oluşmaktadır. Sistemin işleyişi kamera tarafından görüntünün yakalanmasının ve geliştirilen yazılımla sınıflandırma işleminin tamamlanmasının ardından tanımlanan nesnenin, robot manipülatör tarafından daha önceden belirlenmiş olan alana aktarılması şeklindedir.

6.2. Bilgisayar Programının Çalışması

Görüntü tanıma ve robot kontrolü için bu tez çalışması kapsamında geliştirilmiş olan bilgisayar programı, Borland Delphi 6.0 derleyicisi kullanılarak yazılmıştır. Bilgisayar Programının yazımında, FMS laboratuarının imkânları dâhilinde bulunan 1GB Ram Belleğe ve Intel Pentium 4 işlemciye sahip bir kişisel bilgisayar kullanılmıştır. Ayrıca programda görüntü kaynağı olarak kullanılan kameradan görüntü yakalama işlemi için DirectX kitaplıklarından faydalanılmıştır.

6.2.1. Mark3 iletişimi

XR3 kontrolü üretici firma tarafından sunulan Robotalk for Windows adlı robot kontrol programının incelenmesinin ve kontrol sisteminin tanımlanmasının ardından eksenlere tam hâkimiyet sağlamak amacıyla, tez çalışmasının gereği olan kontrolün sağlanması amacıyla Basic programlama dili ortamında farklı döngüler oluşturulmuştur.

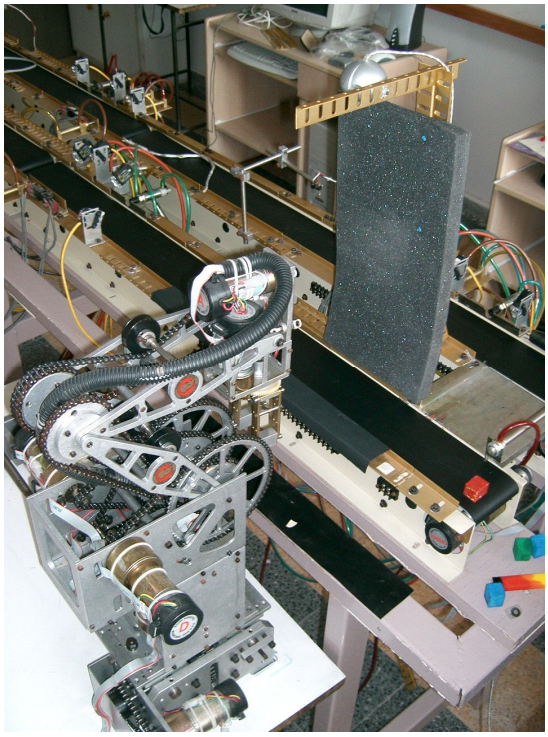


Şekil 6.2. Robot iletişim arayüzü

RoboTalk orijinal BASIC bilgisayar dilinin aynısıdır, yapısal bir dil değildir ve deneysel örnekler yapmak için kullanıldığında onu elverişsiz yapan birçok dezavantajı vardır. Birincisi Robotalk kullanıcı etkileşimli değildir, bu yüzden kullanıcı çalışan programa müdahale edemez. İkincisi, RoboTalk sadece tamsayı değişkenleri kullanabilir, gerçek değişkenleri ve trigonometrik fonksiyonları desteklemez[57]. Mark3 ile Basic derleyicisi arasındaki veri trafiğinin detaylarını belirlemek için çalışmalar yapılmış ve aynı iletişim 2 PC arasında yapılarak veri

paketlerinin ayrıntıları incelenmiştir. Bir PC'de Basic, diğerinde Delphi kullanılmış ve giden gelen veri trafiğini incelenmesi sonucunda Basic ile gönderilen veri paketlerinin sonunda "enter" karakterinin otomatik olarak konduğu, Delphi'de durumun böyle olmadığı görülmüştür. Ardından doğru çalışmayı garanti altına almak için Delphi'de geliştirilen ve Basic dilinden çeviri olan Mark3 iletişim kodlarının sonuna "enter" karakteri eklenmiştir.

Veri gönderi ve almanın sağlıklı bir şekilde sağlanmasının ardından Mark3 üzerinde denemelere başlanabilmiştir. Bunun için XR3'e ait beş eksen ve uç elemanına, yedinci eksen olarak sonradan eklenen kızak ve eklenebilecek herhangi yedinci eksen ile taşıyıcı bandın dur/başla komutlarına manuel olarak PC üzerinden erişebilmek için TTL GİRİŞ/ÇIKIŞ portlarına erişmek amacıyla klavye ve fare girişlerine izin veren bir robot iletişim arayüzü (Şekil 6.2) geliştirilmiştir.



Resim 6.1. XR3 Robotun "home pozisyon" durumu

Robot iletişim arayüzü, XR3 robotun her eksenine tek tek veri gönderebildiği gibi girilen değerlerin tamamının enkoder darbesi cinsinden aynı anda eksenel hiyerarşi doğrultusunda hareketine de izin vermektedir. Aynı zamanda limit switchler

belirlenerek robotun home pozisyona (Resim 6.1) erişmesini sağlayabilecek nitelikte tasarlanmıştır. Bahsedilen tüm işlemler hazırlanan arayüzde görsel olarak ifade edilmiş ve doğruluk kontrolleri yapılabilir durumdadır.

Robot iletişim arayüzü eğitsel açıdan eksenlerin tekil ve çoklu hareketlerinin gözlenmesi ve eksen sınırlarının etkileşimli olarak algılanması ve motorların radyan cinsinden değerlere verdikleri tepkilerin somut olarak izlenebilmesini sağlayabilmektedir. Aynı zamanda bu arayüzde grafiksel olarak XR3 simülasyonu, Robot manipülatörle eş zamanlı olarak eksen hareketini görüntülemektedir.

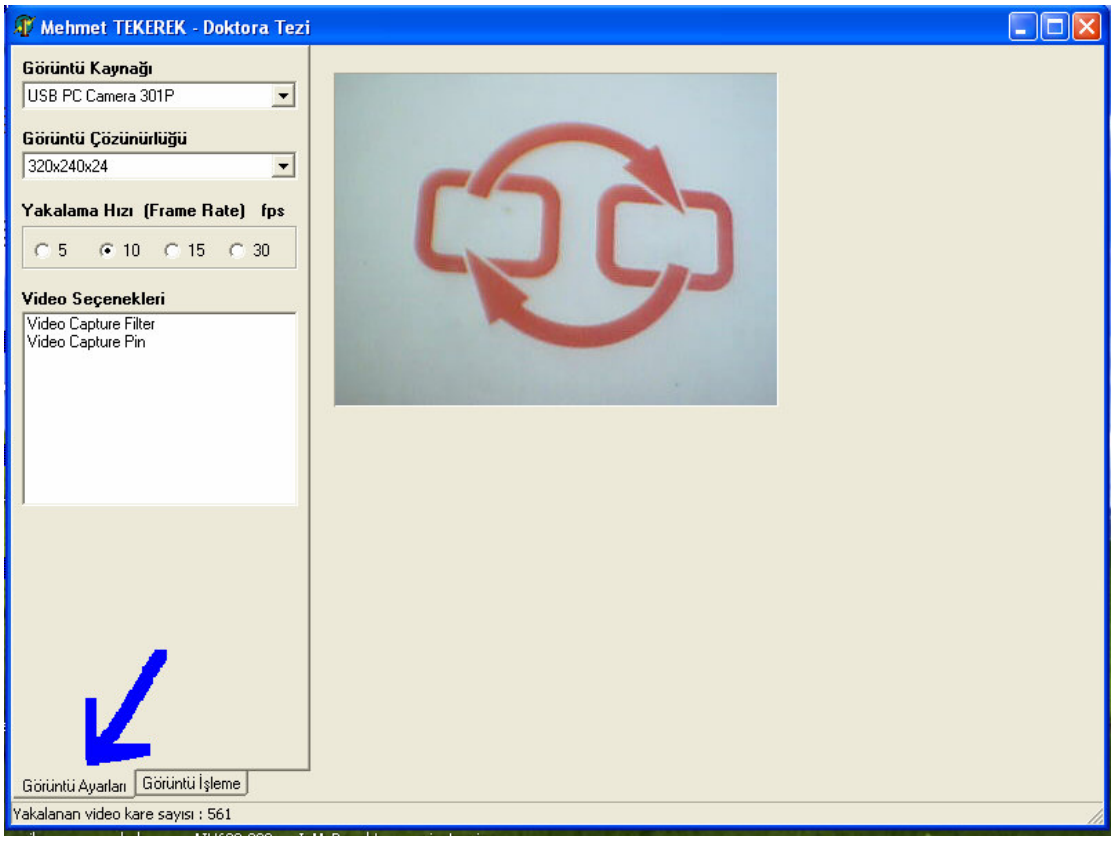
6.2.2. Görüntü yakalama

Görüntü yakalama işi için evrensel veri yolu (USB) üzerinden iletişim kurulabilen ve herhangi bir ek donanım gerektirmeden dijital görüntü sağlayabilen, halk arasında webcam olarak isimlendirilen CCD özellikli bir web kamera kullanılmıştır. CCD özelliğinden ikinci bölümde bahsedilmiştir.

Programda, görüntü yakalama algoritmalarına girdi olarak kullanılacak olan görüntünün yakalanacağı kaynağın seçimi, bir saniyede yakalanacak olan görüntü sayısının ve görüntü büyüklüğün belirlenebilmesi için bir kullanıcı ara yüzü (Şekil 6.3) yer almaktadır. Ayrıca, program açılışta sistemde kayıtlı olan tüm görüntü yakalama aygıtlarını aşağı doğru açılan bir seçim kutusu ile kullanıcının seçimine sunar ve ilk aygıtı otomatik olarak seçip görüntü yakalama işlemine başlar. İşlemin ardından belirlenen kaynaktan, programda yakalama hızı olarak belirtilen bölümden istenen tazeleme hızında görüntü yakalanması ve ekranda görüntülenmesi işlemi başlar.

Eğer listede birden fazla görüntü yakalama aygıtı varsa, kullanıcı bu kutucuk yardımıyla istediği aygıtı görüntü kaynağı olarak belirleyebilir. Bu işlemin ardından bu aygıta ait olan özel iletişim ve ayar pencereleri de kullanıcının ince ayarlar yapabilmesi için hazır hale getirilir. Elde edilen akıcı görüntü arka arkaya yakalanan karelerden oluşmaktadır. Kullanıcı yakalanan karelerin büyüklüğünü ve kare yakalama hızını belirleyebilir. Bilgisayar yardımıyla görüntü işlenmesi, bir takım

matematiksel işlemlerin hızlı bir şekilde sürekli olarak yapılmasını gerektirdiğinden herhangi bir kare üzerindeki işlemlerin, bir sonraki kare yakalanmadan tamamlanması gerekmektedir. İşlem hızı, kullanılan bilgisayarın teknik özelliklerine bağlı olduğundan başlangıçta düşük kare yakalama hızlarıyla küçük görüntü boyutlarının tercih edilmesi önerilir.



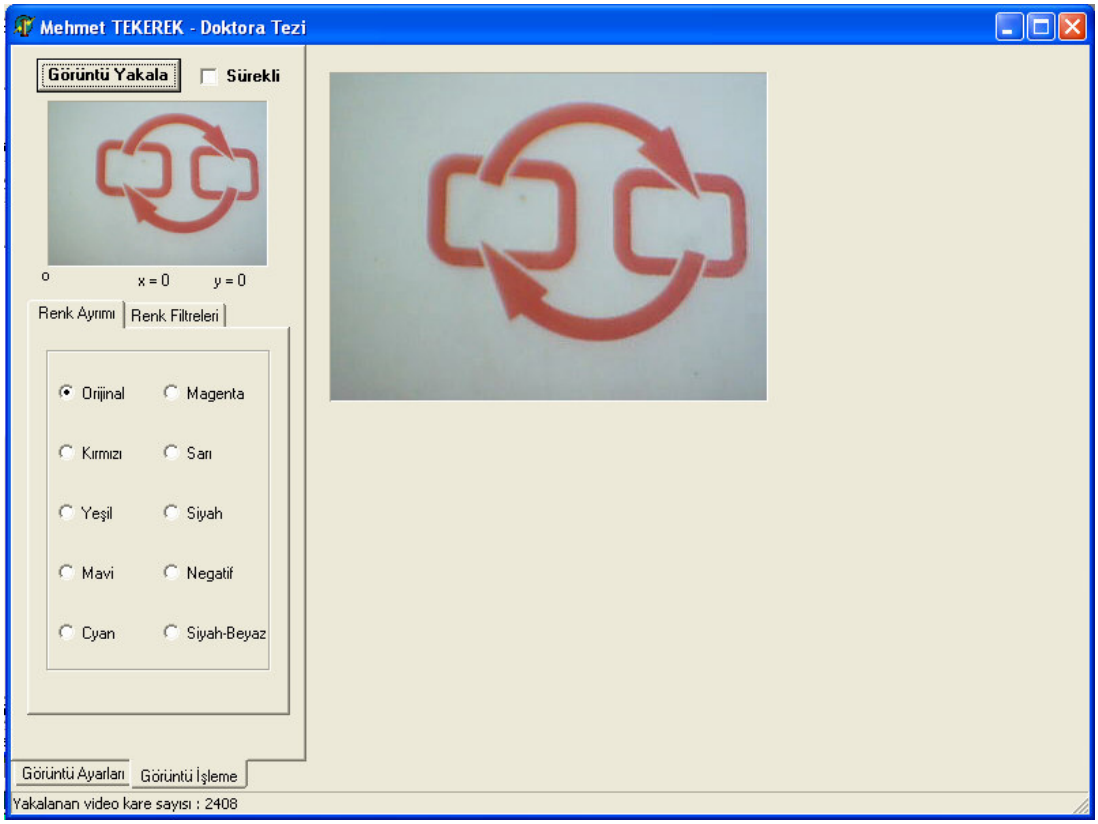
Şekil 6.3. Görüntü kaynağı seçimi

6.2.3. Görüntü işleme süreci

Programda görüntü işlemenin ilk adımı, görüntünün seçili yakalama cihazı ile yakalanmasıdır. İlk karenin yakalanmasının ardından bu kare üzerinde belirlenen işlemler uygulanır ve sistem sonraki kareyi beklemeye başlar ve işlem bu şekilde devam eder. Yakalanan karelerin veri tipi bitmap'tir.

Çok basit olarak ifade edilecek olursa, bitmap veri tipi iki boyutlu olan ve sayılardan oluşan bir matristir. Bu matrisin satır ve sütun sayıları yakalanan görüntünün piksel

cinsinden yüksekliği ve genişliğidir. Matrisin içindeki elemanların, başka bir ifadeyle piksellerin değerleri ise bilgisayar sisteminde bir rengi ifade eden sayılardır. Bilindiği üzere tüm renkler üç temel rengin (kırmızı, yeşil, mavi) birleşiminden oluşur. Bu toplamsal renk sentezi olarak da isimlendirilebilmektedir.

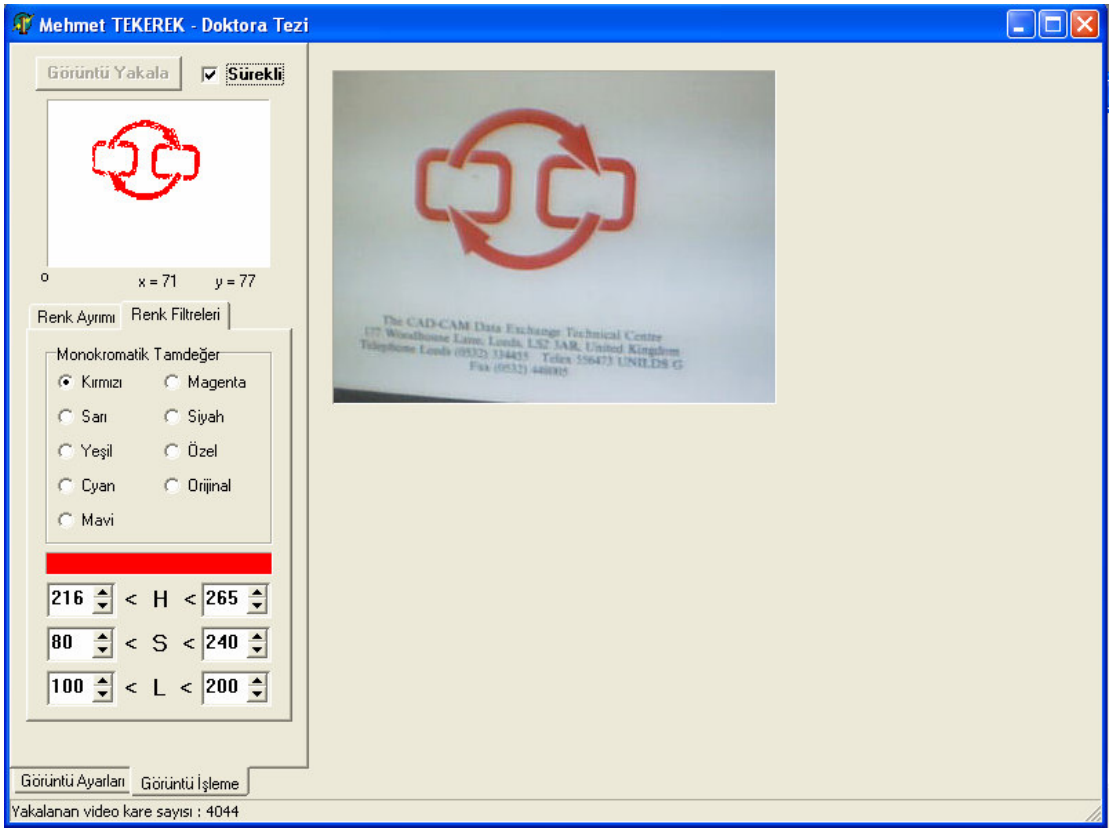


Şekil 6.4. Görüntü işleme, görüntü yakalama arayüzü

Bilgisayar sistemlerinde renklerin yoğunlukları 0 ile 255 arasında (1 byte) değerler alır. Kırmızı renk için örneklendirecek olursak; 255, tam kırmızıyken, 0 tam siyah olur ve 0 dan 255'e doğru gidilirse siyahtan başlayarak önce koyu kırmızı oradan da yavaş yavaş mutlak kırmızıya dönen renkler elde edilir; yani kırmızı yoğunluğu artar. Aynı durum diğer iki temel renk için de geçerlidir. Herhangi bir ara rengi tanımlamak için bu üç temel rengin yoğunluklarını 0'la 255 arasında değişen sayılar kullanarak vermek yeterlidir. Bir başka ifadeyle 0'la 255 arasında değişen üç tane sayı verilerek istenilen tüm renk tonları, 16777216 renk, çok yüksek bir çözünürlükle tanımlanabilir. Bu üç sayının tek bir sayı halinde paketlenmesi gereklidir. Bu işlem aşağıda verildiği şekilde yapılır.

Sayısal renk değeri= (Mavi yoğunluğu x 256 x 256) + (Yeşil yoğunluğu x 256) + (Kırmızı yoğunluğu)

Gösterimde kolaylık sağlamak maksadıyla renk yoğunluğu değerlerinin onaltılı sayı tabanındaki karşılıklarından faydalanılır. Örneğin $FA16A0 sayısı FA-mavi, 16-yeşil ve A0-kırmızı demektir. "$" sembolü sayının onaltılık sayı tabanında verildiğini ifade eder. Renkli bir görüntüyü ifade eden bir matris yukarıda verilen sayısal renk değerlerinden oluşur ve bu matrisin bir sayı olan her elemanı bir piksele karşılık gelir. Bilgisayar gerektiğinde yukarıda verilen ifade ile kodlanmış olan ana renk yoğunluklarını çözümleyip kullanır.



Şekil 6.5. Sürekli görüntü yakalamada renk filtresi uygulanması

Kullanılan bir başka renk sentezi de çıkarımsal renk sentezidir. Bu teknikte renkler, tüm renkleri eşit ve en üst yoğunlukta barındıran beyaz ışıktan ana renk bileşenlerin yoğunluklarının eksiltilmesiyle oluşturulur. Bu sistemin renk bileşenleri ara renkler diye tanımlanan turkuvaz, pembe ve sarıdır. Turkuvaz, kırmızıyı çıkartan renk olarak

ifade edilirken sarı maviyi çıkartan, pembe ise yeşili çıkartan renk olarak ifade edilir. Tam yoğunluktaki turkuvaz, pembe ve sarı koyu bir renk oluşturur. Çünkü tüm bileşenler çıkarılmıştır. Bu teknik daha çok mürekkep kullanan baskı sistemlerinde kullanıldığından koyu tonların elde edilmesi için tüm mürekkeplerin karıştırılması gerekir. Bu durum ekonomik olmadığından bu üç bileşene bir de K (siyah) bileşeni eklenir. Bu tonu koyulaştırmak içindir ve renkli mürekkep kullanımını en aza indirmek içindir.

Toplamsal renk sentezinde başlangıçta hiç renk verilmediğinden ilk renk siyahtır. Eklenen ana renk bileşenleri ile istenilen renk oluşturulur. Çıkarımsal sentezde ise başlangıçta tüm ana renk bileşenleri tam olup hiç renk çıkartılmadığından ilk renk mutlak beyazdır. Eksiltilen ana renk yoğunlukları ile ilk renk olan beyaz istenilen renge dönüştürülür. Bu iki renk sisteminin elemanları R, G, B ve C, M, Y'dir. Burada

R: Kırmızı (Red), G: Yeşil (Green), B: Mavi (Blue), C: Turkuvaz (Cyan), M: Pembe (Magenta), Y: Sarı (Yellow) olarak tanımlıdır.

Programda bu iki renk sistemi arasında çevrim yapılabilmektedir. Bir başka ifadeyle herhangi bir renk tonu, R,G,B değerleri ile tanımlanabildiği gibi C,M,Y,K değerleri ile de tanımlanabilmektedir.

Aşağıda bu renk sistemleri arasında dönüşüm için kullanılan prosedürler ve fonksiyonlar verilmiştir. Progam Delphi ile yazıldığından Pascal esaslı bir kodlama kullanmaktadır. Pascal'ı diğer programlama dillerinden ayıran en temel özelliği, yazılan kodların İngilizce'ye çok yakın olması sebebiyle rahat okunabilmesidir.

Birer byte'lık R,G,B değerlerini girdi olarak alıp, karşılığı olan birer byte'lık C,M,Y,K değerlerini hesaplamak için,

```
procedure RGBdenCMYKya(R,G,B:byte; var C,M,Y,K:byte);

begin

  C:=255-R;
```

```
 M:=255-G;

 Y:=255-B;

 if C<M then K:=C else K:=M;

 if Y<K then K:=Y;

 if K>0 then

  begin

   C:=C-K;

   M:=M-K;

   Y:=Y-K;

  end;

end;
```

prosedürü oluşturulmuştur.

Birer byte'lık C,M,Y,K değerlerini girdi olarak alıp, karşılığı olan birer byte'lık R,G,B değerlerini hesaplamak için,

```
procedure CMYKdanRGBye(C,M,Y,K:integer; var R,G,B:byte);

begin

  if C+K<255 then R:=255-(C+K) else R:=0;

  if M+K<255 then G:=255-(M+K) else G:=0;

  if Y+K<255 then B:=255-(Y+K) else B:=0;
```

```
end;
```

prosedürü oluşturulmuştur.

6.2.4. Renk ayrımları

Renk ayrımı işlemi hem RGB hem de CMYK olarak iki renk sistemi içinde ayrı ayrı yapılır. Bunlar için oluşturulan prosedürler aşağıdaki gibidir.

Renk ayrımı için bir bmp verisini girdi olarak alıp, bundan ayrı ayrı 3 bmp verisi (R- rbmp, G- gbmp, B- bbmp) oluşturan kod,

```
procedure RGBRenkAyrimi(var bmp,rbmp,gbmp,bbmp:Tbitmap);

var i,j:integer; P1,PR,PG,PB:PByteArray; R,G,B:byte;

begin

 with rbmp do

  begin

   PixelFormat:=pf24bit;

   Width:=bmp.Width;

   Height:=bmp.Height;

  end;

 with gbmp do

  begin

   PixelFormat:=pf24bit;

   Width:=bmp.Width;
```

```
 Height:=bmp.Height;
end;
with bbmp do
 begin
  PixelFormat:=pf24bit;
  Width:=bmp.Width;
  Height:=bmp.Height;
 end;
for i:=0 to bmp.Height-1 do
 begin
   P1:=bmp.ScanLine[i];
   PR:=rbmp.ScanLine[i];
   PG:=gbmp.ScanLine[i];
   PB:=bbmp.ScanLine[i];
   for j:=0 to bmp.Width-1 do
     begin
      PB[j*3]:=P1[j*3];
      PB[j*3+1]:=0;
      PB[j*3+2]:=0;
```

```
      PG[j*3]:=0;
      PG[j*3+1]:=P1[j*3+1];
      PG[j*3+2]:=0;
      PR[j*3]:=0;
      PR[j*3+1]:=0;
      PR[j*3+2]:=P1[j*3+2];
     end;
  end;
end;
```

şeklinde oluşturulmuş ve kullanılmıştır.

Renk ayrımı için bir bmp verisini girdi olarak alıp, bundan ayrı ayrı 4 bmp verisi (C–cbmp ,M- mbmp,Y- ybmp, K- kbmp) oluşturan kod,

```
procedure CMYKRenkAyrimi(var bmp,cbmp,mbmp,ybmp,kbmp:Tbitmap);
var i,j:integer; P1,PC,PM,PY,PK:PByteArray; C,M,Y,K:byte;
begin
 with cbmp do
  begin
   PixelFormat:=pf24bit;
   Width:=bmp.Width;
```

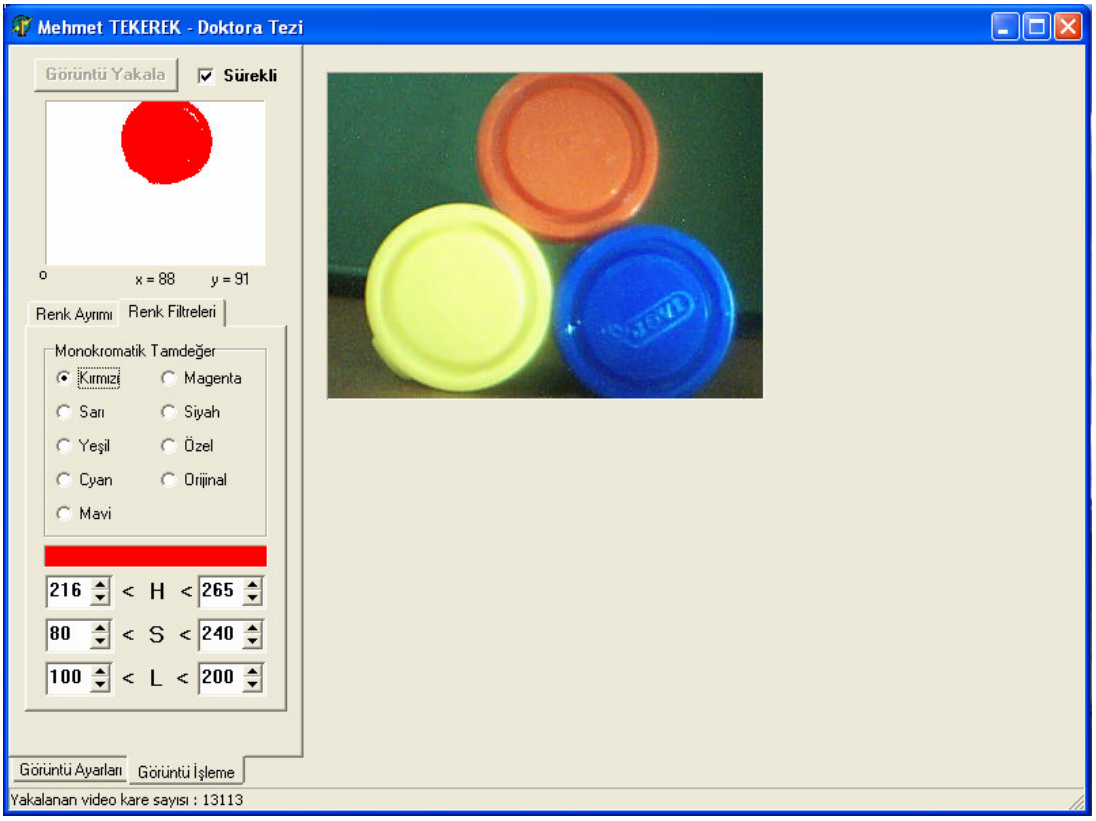
```
 Height:=bmp.Height;
 end;
with mbmp do
 begin
 PixelFormat:=pf24bit;
 Width:=bmp.Width;
 Height:=bmp.Height;
 end;
with ybmp do
 begin
 PixelFormat:=pf24bit;
 Width:=bmp.Width;
 Height:=bmp.Height;
 end;
with kbmp do
 begin
 PixelFormat:=pf24bit;
 Width:=bmp.Width;
 Height:=bmp.Height;
```

```
  end;


 for i:=0 to bmp.Height-1 do
  begin
    P1:=bmp.ScanLine[i];
    PC:=cbmp.ScanLine[i];
    PM:=mbmp.ScanLine[i];
    PY:=ybmp.ScanLine[i];
    PK:=kbmp.ScanLine[i];
   for j:=0 to bmp.Width-1 do
     begin
      RGBdenCMYKya(P1[j*3+2],P1[j*3+1],P1[j*3],C,M,Y,K);
      CMYKdanRGBye(C,0,0,0,PC[j*3+2],PC[j*3+1],PC[j*3]);
      CMYKdanRGBye(0,M,0,0,PM[j*3+2],PM[j*3+1],PM[j*3]);
      CMYKdanRGBye(0,0,Y,0,PY[j*3+2],PY[j*3+1],PY[j*3]);
      CMYKdanRGBye(0,0,0,K,PK[j*3+2],PK[j*3+1],PK[j*3]);
     end;
  end;
 end;
```

şeklinde oluşturulmuş ve kullanılmıştır.

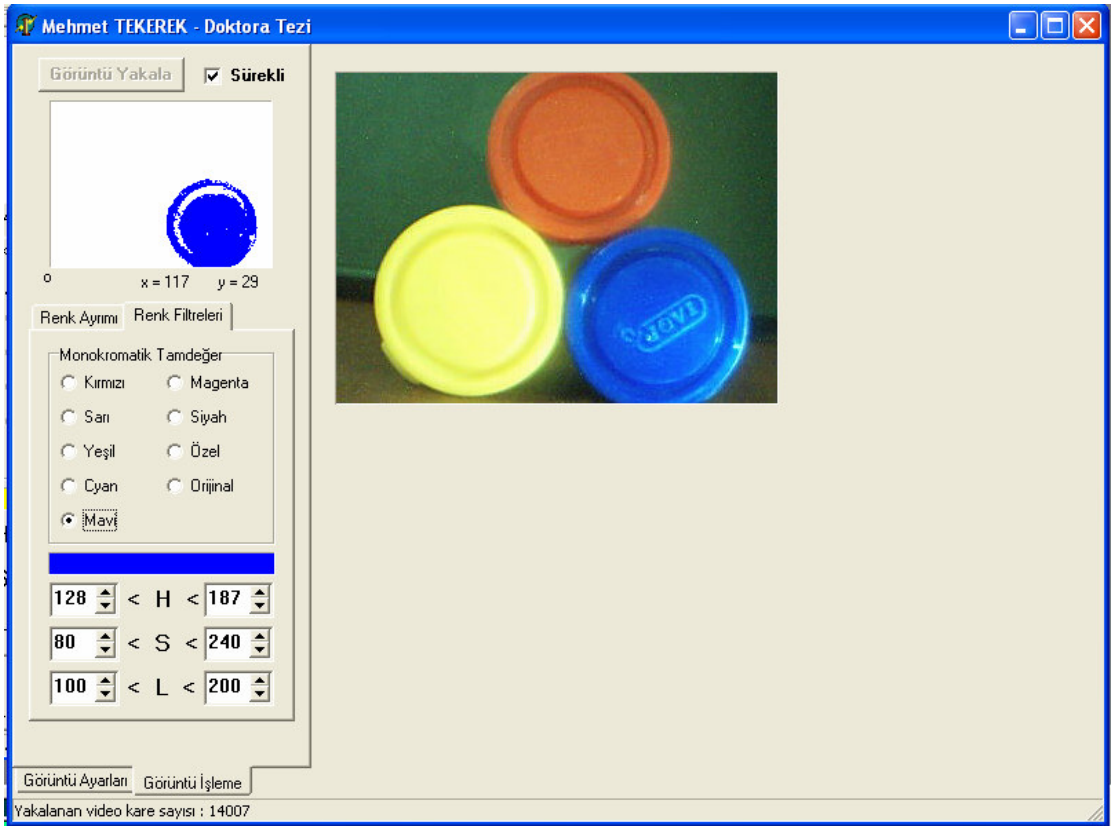
Kırmızı: Matristeki tüm sayısal renk değerleri teker teker 255 ya da $0000FF sayısı ile "ve" işlemine tabi tutulur. Bu durumda piksellerin sayısal renk değeri ifadesindeki kırmızı bileşeni sahip olduğu değeri korurken, yeşil ve mavi bileşenleri 0 olur. Geriye sadece piksellerin kırmızı bileşenleri kalır.



Şekil 6.6. Kırmızı rengin sürekli görüntü yakalama modunda algılanması

Yeşil: Matristeki tüm sayısal renk değerleri teker teker 65280 ya da $00FF00 sayısı ile "ve" işlemine tabi tutulur. Bu durumda piksellerin sayısal renk değeri ifadesindeki yeşil bileşeni sahip olduğu değeri korurken, kırmızı ve mavi bileşenleri 0 olur. Geriye sadece piksellerin yeşil bileşenleri kalır.

Mavi: Matristeki tüm sayısal renk değerleri teker teker 16711680 ya da $FF0000 sayısı ile "ve" işlemine tabi tutulur. Bu durumda piksellerin sayısal renk değeri ifadesindeki mavi bileşeni sahip olduğu değeri korurken, kırmızı ve yeşil bileşenleri 0 olur. Geriye sadece piksellerin mavi bileşenleri kalır.

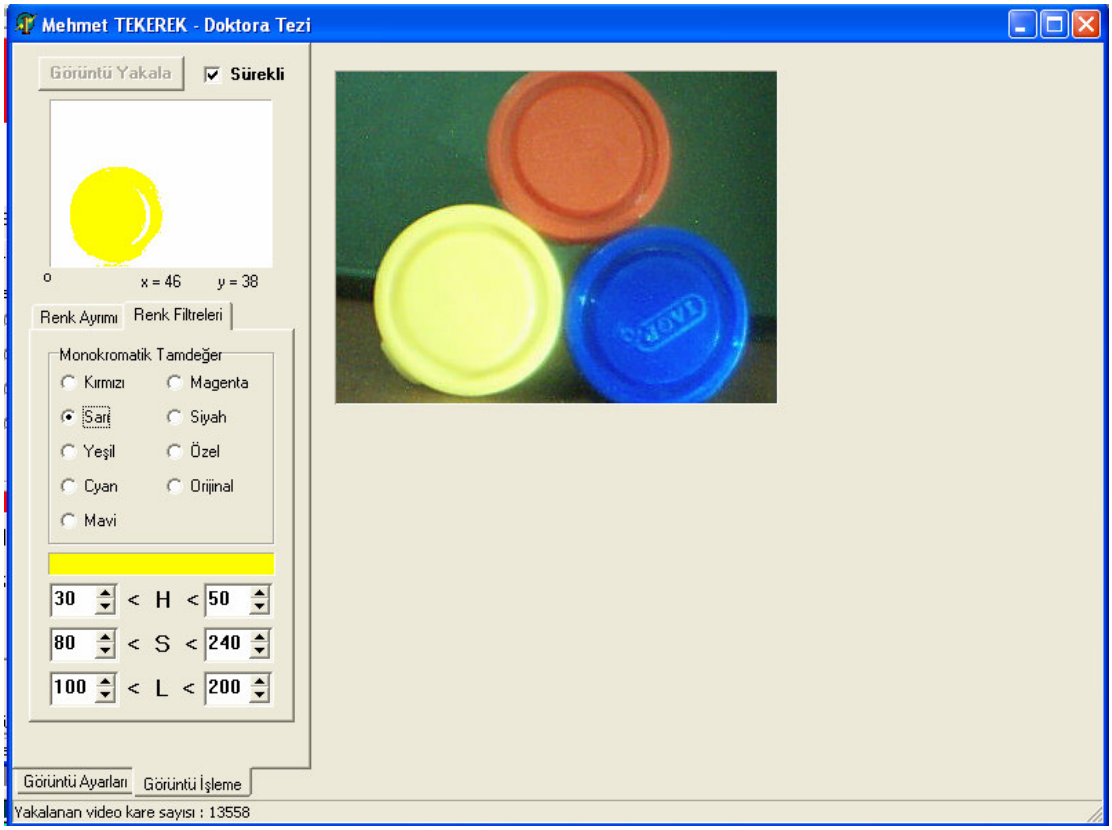


Şekil 6.7. Mavi rengin sürekli görüntü yakalama modunda algılanması

Cyan: Matristeki tüm sayısal renk değerleri teker teker "RGBdenCMYKya" işlemine tabi tutulup, Cyan haricindeki tüm bileşenler 0 yapılır ve ardından "CMYKdanRGBye işlemi" yapılır.

Magenta: Matristeki tüm sayısal renk değerleri teker teker "RGBdenCMYKya" işlemine tabi tutulup, Magenta haricindeki tüm bileşenler 0 yapılır ve ardından "CMYKdanRGBye" işlemi yapılır.

Sarı: Matristeki tüm sayısal renk değerleri teker teker "RGBdenCMYKya" işlemine tabi tutulup, Sarı haricindeki tüm bileşenler 0 yapılır ve ardından "CMYKdanRGBye" işlemi yapılır.



Şekil 6.8. Sarı rengin sürekli görüntü yakalama modunda algılanması

Siyah: Matristeki tüm sayısal renk değerleri teker teker "RGBdenCMYKya" işlemine tabi tutulup, Siyah haricindeki tüm bileşenler 0 yapılır ve ardından "CMYKdanRGBye" işlemi yapılır.

Negatif: Matristeki tüm sayısal renk değerlerinin kırmızı, yeşil ve mavi bileşenleri teker teker üst sınır değer olan 255'ten çıkarılır. Bir bmp verisini girdi olarak alıp, aynı bmp verisinin negatifini çıktı olarak almak için,

```
procedure Negatif(var bmp:Tbitmap);

var x,y:integer; P1:PByteArray;

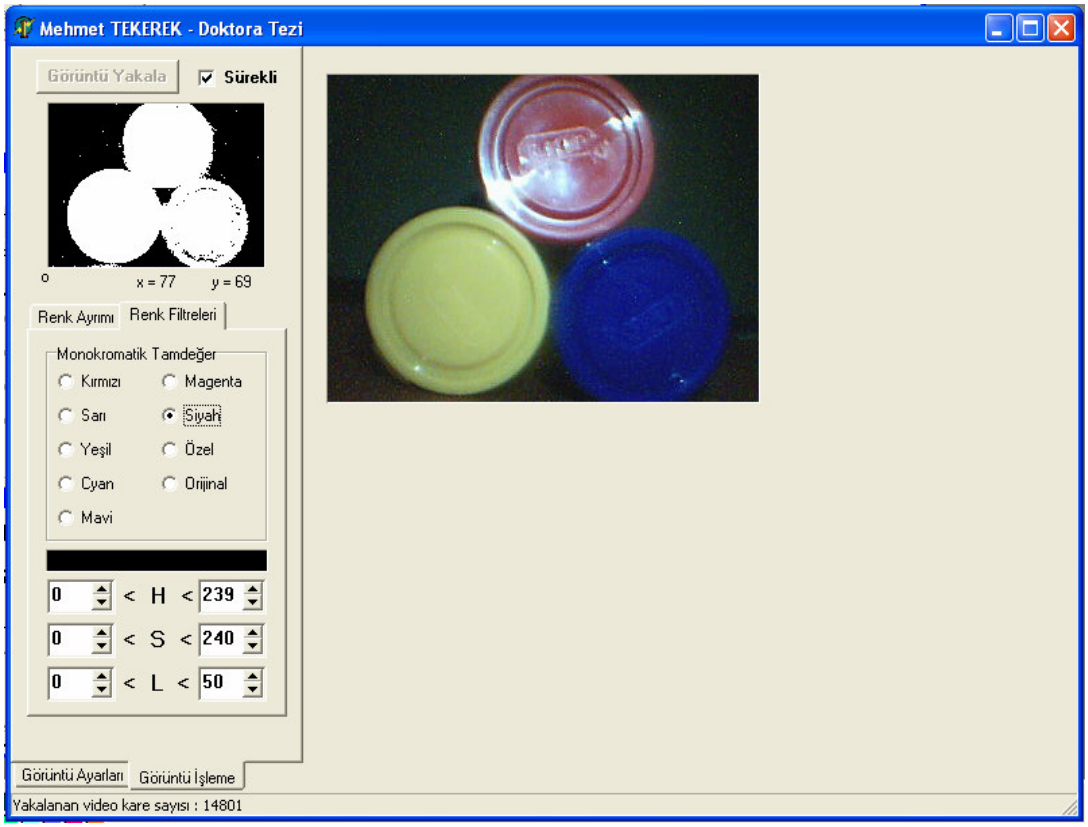
begin

 for x:=0 to bmp.Height-1 do
```

```
 begin
  P1:=bmp.ScanLine[x];
  for y:=0 to bmp.Width-1 do
    begin
     P1[y*3]:=255-P1[y*3];
     P1[y*3+1]:=255-P1[y*3+1];
     P1[y*3+2]:=255-P1[y*3+2];
    end;
 end;
end;
```

prosedürü oluşturulmuştur.

Siyah-Beyaz: Matristeki tüm sayısal renk değerlerinin ana renk bileşenleri için Mavi*0.11+Yeşil*0.59+Kırmızı*0.30 ifadesi hesaplanıp, her üç ana renk bileşeni de elde edilen bu değere eşitlenir.



Şekil 6.9. Siyah- beyaz renk dönüşümü

Bir bmp verisini girdi olarak alıp, aynı bmp verisinin siyah beyaz halini çıktı almak için,

```
procedure SiyahBeyaz(var bmp:Tbitmap);

var x,y:integer; P1:PByteArray; tmp:byte;

begin

 for x:=0 to bmp.Height-1 do

  begin

    P1:=bmp.ScanLine[x];

   for y:=0 to bmp.Width-1 do
```

```
    begin
     tmp:=round(P1[y*3]*0.11+P1[y*3+1]*0.59+P1[y*3+2]*0.30);
     P1[y*3]:=tmp;
     P1[y*3+1]:=tmp;
     P1[y*3+2]:=tmp;
    end;
  end;
end;
```

prosedürü oluşturulmuştur.

Bir diğer renk tanımlama sistemi de H, S, L sistemidir. Bu sistemde "Hue" renk tonunu, "Saturation" doygunluğu, "Luminance" ise parlaklığı ifade eder. Bu sistem, renkleri tonlarına göre sınıflandırmak veya algılamak için uygundur. R, G, B sistemi ile H, S, L sistemi arasındaki dönüşümler aşağıdaki kodlar kullanılarak yapılır.

Birer byte'lık R,G,B değerlerini girdi olarak alıp, karşılığı olan H,S,L değerlerini hesaplamak için,

```
procedure RGBdenHSLye(R,G,B:byte; var H,S,L:double);
  function Max (a,b:double):double;
  begin
    if a>b then Result:=a else Result:=b;
  end;
```

```
 function Min (a,b:double):double;
 begin
   if a<b then Result:=a else Result:=b;
 end;


var R1,G1,B1,D,Cmax,Cmin: double;
begin
 R1:=R/255;
 G1:=G/255;
 B1:=B/255;
 Cmax:=Max(R1,Max(G1,B1));
 Cmin:=Min(R1,Min(G1,B1));
 L:=(Cmax+Cmin)/2;


 if Cmax=Cmin then
 begin
   H:=0;
   S:=0;
 end else begin
```

```
  D:=Cmax-Cmin;
  if L<0.5 then S:=D/(Cmax+Cmin) else S:=D/(2-Cmax-Cmin);
      if R1=Cmax then H:=(G1-B1)/D
  else if G1=Cmax then H:=2+(B1-R1)/D
  else H:=4+(R1-G1)/D;
  H:=H/6;
  if H<0 then H:=H+1;
 end;
end;
```

prosedürü oluşturulmuştur.

H,S,L değerlerini girdi olarak alıp, karşılığı olan birer byte'lık R,G,B değerlerini hesaplamak için,

```
procedure HSLdenRGBye(H,S,L:double; var R,G,B:byte);
var M1,M2: double;
 function HuedanRenkDegerine(Hue:double):byte;
 var V:double;
 begin
      if Hue<0 then Hue:=Hue+1
  else if Hue>1 then Hue:=Hue-1;
```

```
        if 6*Hue<1 then V:=M1+(M2-M1)*Hue*6
    else if 2*Hue<1 then V:=M2
    else if 3*Hue<2 then V:=M1+(M2-M1)*(2/3-Hue)*6
    else V:=M1;
    Result:=round(255*V);
  end;
begin
  if S=0 then
  begin
    R:=round(255*L);
    G:=R;
    B:=R
  end else begin
    if L<=0.5 then M2:=L*(1+S) else M2:=L+S-L*S;
    M1:=2*L-M2;
    R:=HuedanRenkDegerine(H+1/3);
    G:=HuedanRenkDegerine(H);
    B:=HuedanRenkDegerine(H-1/3)
  end;
```

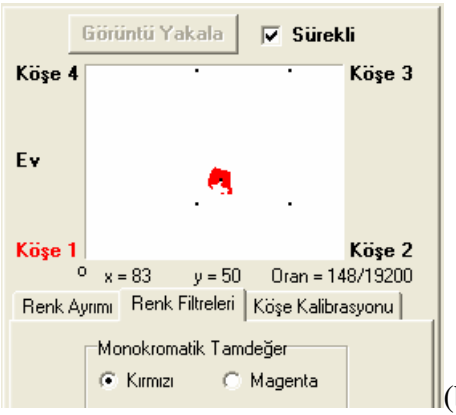
end;

prosedürü oluşturulmuştur.

H, S, L değerleri matematiksel olarak 0 ile 1 arasında elde edilmesine rağmen 0 ile 240 arasındaki gösterim yaygındır. Değerleri bu aralığa getirmek için kullanılan prosedürler de yukarıda verilmiştir.

Programda renk tanımlama işlemi için belirli bir rengin ait olabileceği H aralığı, S aralığı ve L aralığı, biri alt sınır diğeri ise üst sınır olmak üzere ikişer sayı kullanılarak tanımlanmıştır.


(a)



(b)

Şekil 6.10. (a) Kameradan alınan orijinal görüntü (b) Aynı görüntünün monokromatik renk filtresi uygulanmış şekli

Sistem görüş alanındaki nesneyi tespit eder ve uygun renk eşik değerinin seçilmesi temeline dayanan özel algoritması sayesinde arka planda ayırır. Bu uygulamada üç ana renk özelliği olan doygunluk, ton ve ışık yoğunluğu (HSL) özellikleri kullanılmaktadır.

Alınan gerçek dijital görüntünün bu üç kanala ayrılmasıyla görüntüler üç ayrı karakteristikte değerlendirilebilir hale gelir. Burada amaç farklı parlama ve ışıklandırmanın yapacağı etkileri elemek ve ayrıştırma işlemi için en uygun karakteristik eşik değerlerini elde etmektir. Doygun görüntü kullanılmasıyla orijinal renk değerleri korunurken bir tür gri skala dönüşümü de elde edilmiş olur.

Bir görüntü uzayda koordinatları bulunan iki bilinmeyenli bir f(x,y) fonksiyonu ile tanımlanabilir ve f fonksiyonunun herhangi bir (x,y) koordinatındaki genliği (genişliği) bu noktadaki görüntünün gri seviyesi veya yoğunluğu olarak isimlendirilir. Fonksiyonun herhangi bir x,y ve genlik değeri sınırsız olduğunda görüntüye dijital görüntü adı verilir.

Dijital görüntüyü oluşturan sonsuz sayıdaki elemanların her biri belirli bir değere ve alana sahiptir. Bunlar, görüntü elemanları, resim elemanları ve pikseller anlamına gelmektedir. Piksel, dijital görüntünün elemanlarını dile getirmek için kullanılan en yaygın terimdir.

Görüntü işleme algoritmalarının sürekli olarak çalışması için “sürekli” seçeneği tercih edilir. Bu seçim arka planda bir zamanlayıcının çalışmasına neden olur. Bu zamanlayıcı belirlenmiş bir kodun saniyede 10 defa olacak şekilde çalışmasını sağlar.

Bu kod sayesinde ekranda görüntülenmek üzere belirlenmiş olan görüntü, görüntü işleme algoritmalarında kullanılmak üzere bir *bitmap* değişkenine kopyalanır. Alınan görüntü üzerinde bir takım görüntü işleme algoritmalarının çalıştırılmasının ardından geçen 1/10 saniyenin ardından zamanlayıcı devreye girer ve bu *bitmap* değişkeninin içeriği yakalanmış daha güncel bir görüntünün kopyalanmasıyla güncellenir. Görüntü işleme süreci, kesintiye uğratılana kadar bu şekilde devam eder.

6.2.5. Renk filtreleri

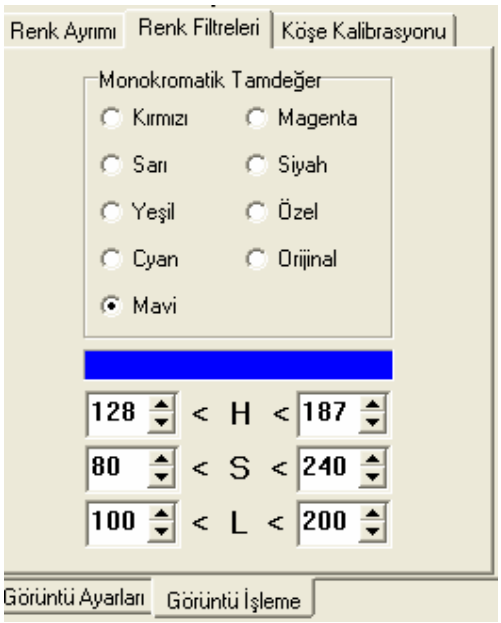
Matristeki tüm sayısal renk değerleri teker teker "RGB den HSL aralığına" işlemine tabi tutulur ve hangi renk filtresine göre ayrım yapılacaksa elde edilen H, S, L değerlerinin o renk için belirlenmiş olan sınır değerleri arasında olup olmadığı kontrol edilir. Sonuç olumluysa matrisin bu elemanı olan piksel değeri renk filtresinin tanımlı rengine dönüştürülür. Sonuç olumsuzsa bu piksel beyaz ($FFFFFF) yapılır. Bu işlem program bünyesinde monokromatik filtre olarak adlandırılmaktadır. Bu ismin kullanılmasındaki neden elde edilen bitmap görüntünün beyaz haricinde sadece filtre rengini içermesidir.

6.2.6. Renk tanıma

Bitmap değişkenleri matris yapıda dizilmiş olan piksellerden oluşur ve her bir piksel için üç temel renk bileşeni (R,G,B) için 8'er bitlik sayılardan oluşan 3 sayı tutar. Bir başka ifadeyle her pikselde 0'la 255 arası değişen 3 tane sayı vardır ve bu sayılar o piksel için kırmızı, yeşil ve mavi yoğunluklarını ifade ederler. Görüntü işleme hızını belirleyen zamanlayıcının faaliyete geçmesiyle beraber yakalanan ve *bitmap* değişkenliğine aktarılan görüntünün her bir pikseli ayrı ayrı incelenir. Öncelikle tüm piksellere RGB'den HSL'ye dönüşüm kodları uygulanır.

$$P_{ij}[R,G,B] \rightarrow P_{ij}[H,S,L]$$

Burada amaç pikselin renginin belirlenmesi ve ayrıştırılmasıdır.



Şekil 6.11. HSL aralıklarının belirlenmesi

Sistemde, Şekil 6.11 de gösterildiği gibi, tonunu kullanıcının belirleyebildiği renk seçimleri vardır. Kullanıcı bu ara yüzü kullanarak belirli bir H (Ton), belirli bir S (Doygunluk) ve belirli bir L (Parlaklık) aralığı tanımlayabilir. Eğer RGB'den HSL'ye dönüştürülmüş olan herhangi bir piksel, daha önceden belirlenmiş olan H,S,L aralığına giriyorsa, ya da bir başka ifadeyle tanımlanmış olan herhangi bir

renge çok yakınsa, o pikselin değeri, tanımlanmış olan rengin değerine dönüştürülür. Eğer bu şart sağlanmıyorsa, yani pikselin R,G,B değerlerinden elde edilmiş olan H,S,L değerleri, tanımlanmış olan aralıkların içine girmiyorsa o piksel beyaza dönüştürülür. Bu işlem, *bitmap* değişkeni içindeki tüm pikseller için gerçekleştirilir.

Eğer R_1 atanmış renklerden birisiyse, bu ifade aşağıdaki gibi özetlenebilir:

$$\left[R_1[H]_{alt} \le P_{ij}[H] \le R_1[H]_{üst}\right]$$

$$\left[R_1[S]_{alt} \le P_{ij}[S] \le R_1[S]_{üst}\right]$$

$$\left[R_1[L]_{alt} \le P_{ij}[L] \le R_1[L]_{üst}\right]$$

ifadesi doğru ise,

$$P_{ij}[H] = [R_1[H]_{alt} + R_1[H]_{üst}]/2$$

$$P_{ij}[S] = [R_1[S]_{alt} + R_1[S]_{üst}]/2$$

$$P_{ij}[L] = [R_1[L]_{alt} + R_1[L]_{üst}]/2$$

$P_{ij}[H,S,L] \to P_{ij}[R,G,B]$ doğru değilse beyaz renge dönüşecek şekilde,

$P_{ij}[R] = 255$ $\quad$ $P_{ij}[G] = 255$ $\quad$ $P_{ij}[B] = 255$ olarak belirlenir.

Herhangi bir pikselin tanımlanan aralıklar içine girmesi halinde gerçekleştirilen tek işlem renk dönüşümü değildir. Söz konusu pikselin bulunduğu koordinatların x ve y değerleri de toplam x ve toplam y değerlerinin saklandığı değişkenlere eklenir.

Eğer P_{ij} aranılan sınırlar içindeyse, P_n tanıma uyan toplam piksel sayısı olmak üzere:

$cgX = cgX + i$,

$cgY = cgY + j$,

$P_n = P_n + 1$

işlemleri tüm i, j değerleri için gerçekleştirilir. Son olarak,

$cgX = cgX / P_n$

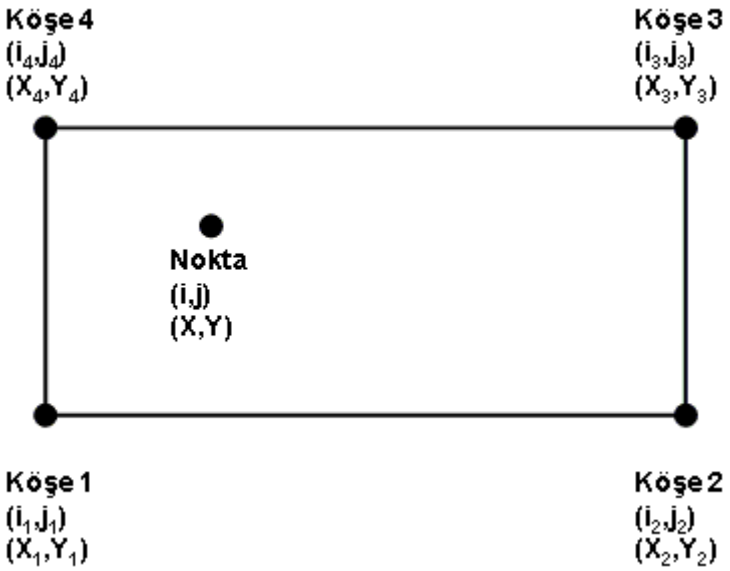
$cgY = cgY / P_n$

işlemi ile tanıma uyan piksellerin ağırlık merkezi bulunur.

6.2.7. Robot hareket uzayının belirlenmesi

Robot hareket uzayının belirlenmesinde farklı kalibrasyon yöntemleri mevcuttur. Tek kamera kullanımı söz konusu olduğunda düzlem çerçeveli özellikle üç boyutlu ölçümler için ayna yansımasından faydalanılmaktadır. Bu da stereo kameranın avantajlarını sağlamaktadır[58].

Bu çalışmada robotun hareket uzayıyla kameranın görüntü uzayı arasındaki ilişkinin sağlanması için köşe kalibrasyonu işlemine gereksinim duyulmuştur. Bu işlemde kamera tarafından gönderilen görüntü üzerinde dikdörtgensel bir bölgenin köşe noktalarının piksel koordinatları, robotun hareket uzayındaki koordinatları ile eşleştirilir. Bu işlemin ardından piksel koordinatları belirlenen herhangi bir noktanın robotun hareket uzayındaki koordinatlarının bulunması için, kalibrasyon için kullanılan 4 köşe koordinatından yararlanılarak interpolasyon yapılır.



Şekil 6.12. Köşe kalibrasyonu

Şekil 6.12'de verilen *i* ve *j* değerleri piksel koordinatları, *X* ve *Y* değerleri ise bu değerlere karşılık gelen robotun hareket uzayındaki koordinatlardır. Buna göre,

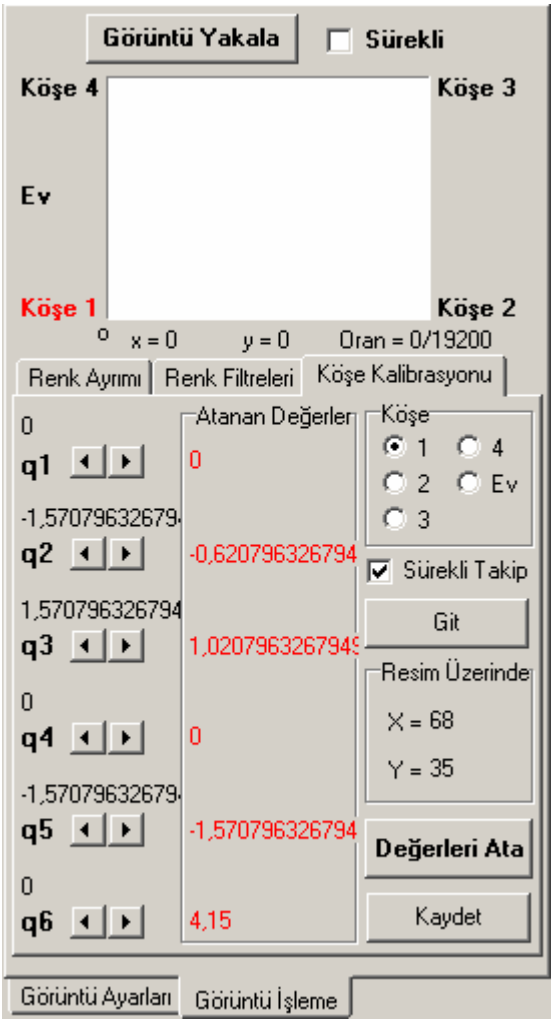
$$X = (i - i_1)/(i_2 - i_1).(X_2 - X_1) + X_1$$

$$Y = (j - j_1)/(j_2 - j_1).(Y_2 - Y_1) + Y_1$$

olarak tanımlanabilir.

Kamera ile piksel koordinatları belirlenen noktanın robotun hareket uzayındaki koordinatlarının yukarıda verilen formülle interpolasyon metodu kullanılarak hesaplanmasının ardından bulunan koordinatlar robota hedef olarak gösterilir ve robotun o noktaya gitmesi sağlanır.

Köşe kalibrasyonu kamera uzayının sınırları içerisinde simetrik dört ayrı noktanın koordinat sistemi içerisinde belirlenmesi ve bunun dışında robotun home pozisyonun belirlenmesi şeklinde gerçekleşir. Çalışma ortamının farklılaşması durumunda bu kullanıcı tarafından eldeki ortama uyum sağlayacak esneklikte tasarlanmıştır.



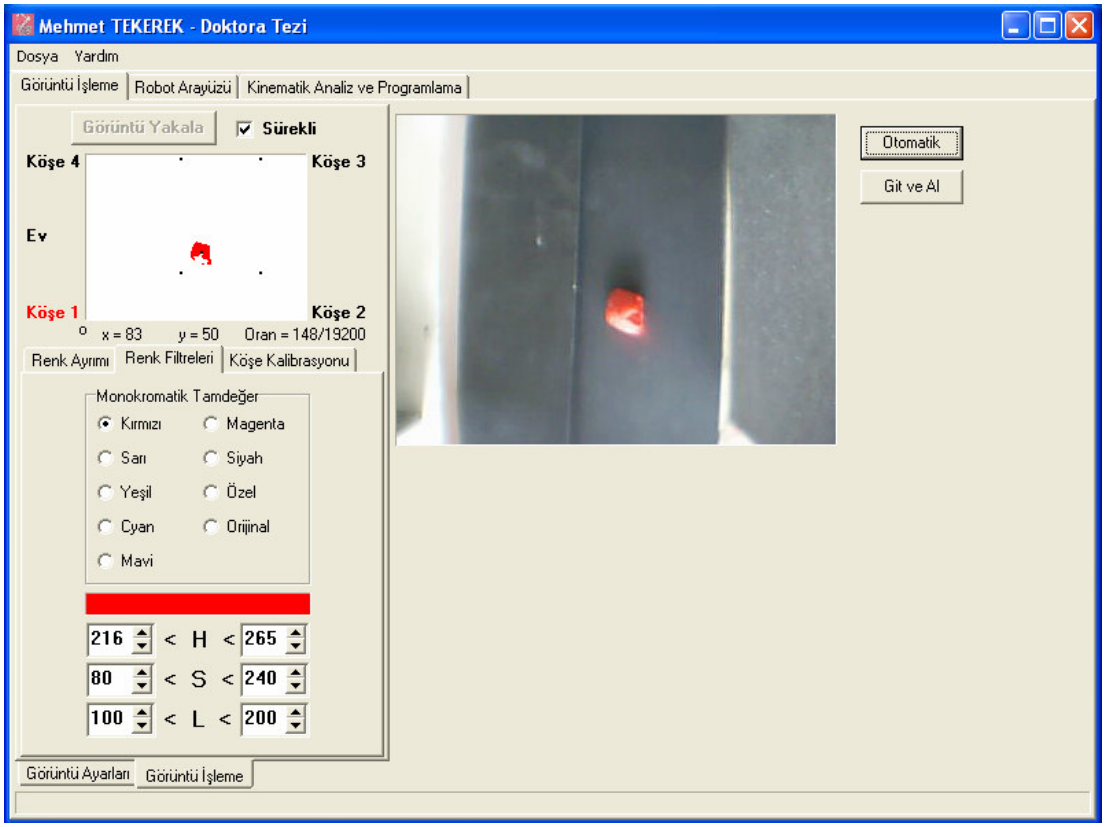
Şekil 6.13. Esnek köşe kalibrasyonu oluşturmaya yönelik olarak hazırlanan kullanıcı etkileşimli arayüz

Kalibrasyon işleminde robotun bütün eksenlerinin koordinat sistemi dâhilinde hazırlanmış olan köşe noktalarında alacağı değerin home pozisyona göre radyan cinsinden ifadesi kullanılmaktadır.

6.3. Sistem İş Akışı

6.3.1. Kamera iletişim iş parçacığı

Sistem aynı anda çalışan paralel iş parçacıkları (thread) üzerine kuruludur. Bunlardan ilki, görüntü yakalama iş parçacığıdır. Bu iş parçacığı Microsoft DirectX kitaplıkları kullanılarak ana program formu üzerine basılan sürekli akan görüntüden, işlenmek üzere görüntü alınmasını sağlar.



Şekil 6.14. Kamera iletişim ve görüntü yakalama arayüzü

DirectX kitaplıkları kullanılarak ana program formu üzerine sürekli akan görüntüyü basan program bloğu, görüntü yakalama kaynağı olarak donanım üzerinde bu amaç için kullanılabilecek olan tüm görüntü yakalama kaynaklarından herhangi biri için seçim yapılabilmesine olanak tanır ve o cihaza ait olan Windows sürücüsünün sunduğu tüm prosedürlere erişim sağlar. Bu iş parçacığı sürekli olarak arka planda çalışır.

6.3.2. Görüntü yakalama ve hazırlama iş parçacığı

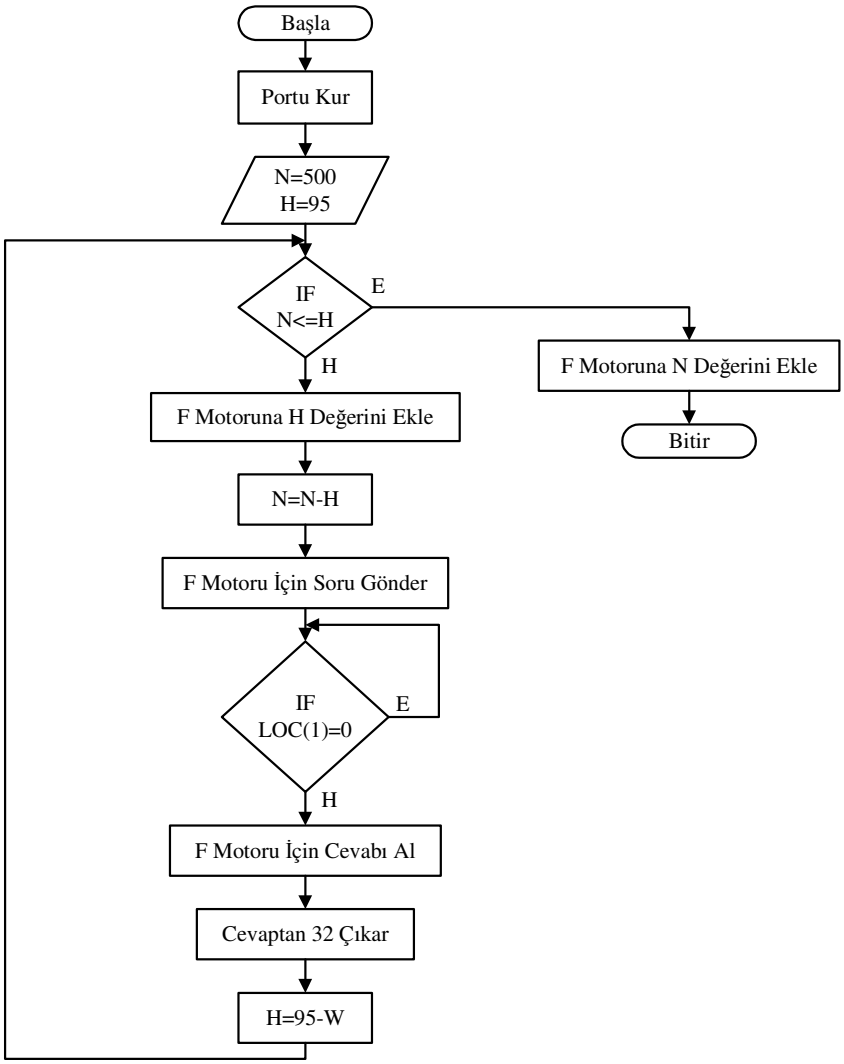
Sistemdeki bir diğer iş parçacığı ise belirli zaman aralıkları ile çalışan ve ilk iş parçacığı olan görüntü yakalama iş parçacığının sunduğu yakalanmış olan görüntüyü alarak renk ayrım ve renk teklileştirme (monochrome) işlemini yazılımla tanınması istenen tüm renkler için ayrı ayrı yapar ve geçici hafızaya kaydeder. Bu iki iş parçacığının ortak çalışması sonucu tüm sistemin çalışması için gerekli

algoritmaların ihtiyaç duyacağı yarı işlenmiş görüntüler geçici hafıza üzerinde her an kullanılmak üzere yer alır ve belirli aralıklarla tazelenir.

6.3.3. Robot denetleyicisi iletişimi iş parçacığı

Üçüncü iş parçacığı Mark3 robot denetleyicisi iletişim kod bloğudur. Daha önceki bölümlerde anlatıldığı gibi iletişim bilgisayar ve Mark3 robot denetleyici arasında çift yönlü mesaj trafiği esasına dayanmaktadır. RS232 seri haberleşme protokolünü kullanan bu mesaj trafiği sonlu bir hıza sahiptir. Bir başka ifade ile yapılan işlemler neticesinde ortaya çıkan sonuçlar doğrultusunda robot denetleyicisine aktarılmak istenen mesajlar, birden bire gönderilemezler ve bir kuyrukta sıraya alınırlar. Bu mesajlar robot hareket mesajları olabildiği gibi, denetleyici üzerindeki giriş ve çıkışlara müdahale eden mesajlar da olabilir.

Mark3 robot denetleyicisi *iletişim kod bloğu* adlı üçüncü iş parçacığı mesaj kuyruğunda robot denetleyicisine gönderilmek üzere bekleyen mesaj olup olmadığını sürekli kontrol eder. Programın herhangi bir anında otomatik çalışan algoritmalar sonucu ya da robot-insan arayüzü üzerinden kullanıcı tarafından oluşturulan mesajlar, mesaj kuyruğuna kaydedilir ve arka planda sürekli olarak çalışmakta olan üçüncü iş parçacığı tarafından işlenir ve kuyruktan çıkarılır. Robot iletişimi arayüzü Resim 6.1 de gösterilmektedir.

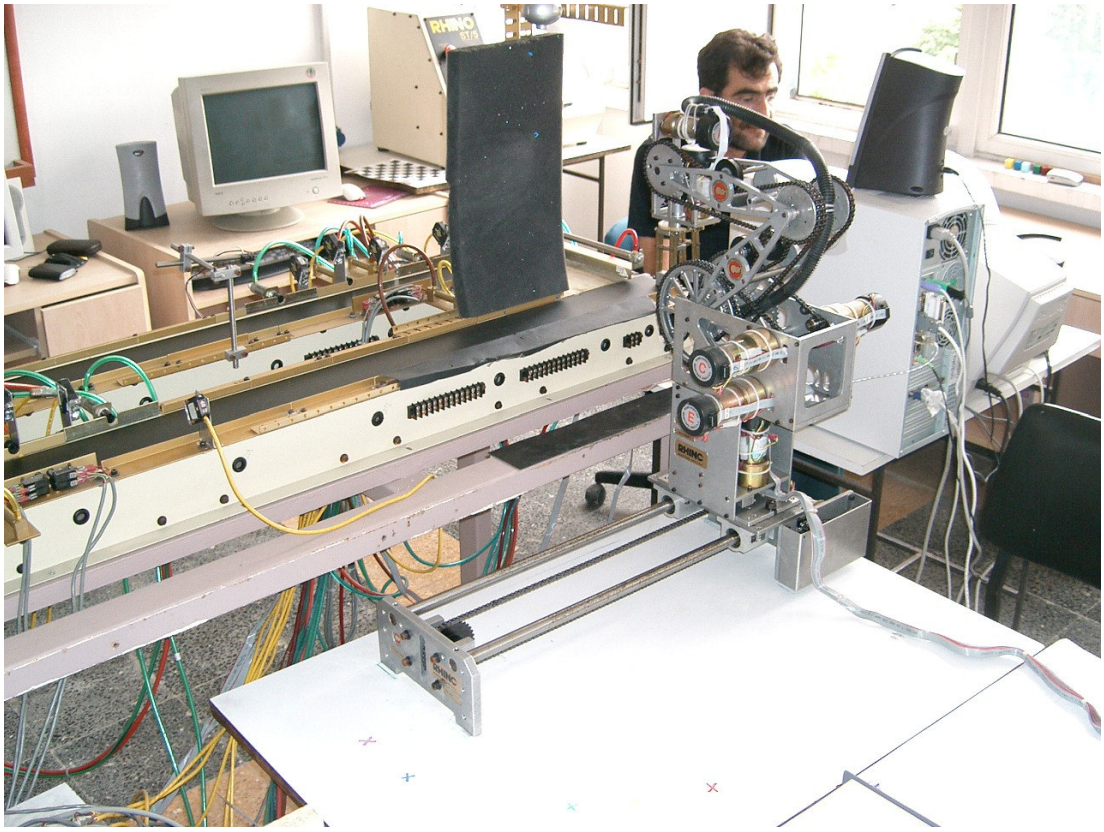


Şekil 6.15. Sürekli hareketi optimum zamanda gerçekleştirmeye yönelik döngü.

Aynı anda çalışan bu iş parçacıklarının detaylı çalışma sistemleri Şekil 6.15. de ifade edilmiştir.

6.4. Sistem İş Döngüsü

Bütün bu iş parçacıklarının genel tanımını verdikten sonra bir iş döngüsü incelenecek olursa: ilk önce konveyörün çalıştırılması için gerekli olan mesaj, ana kod bloğu tarafından mesaj kuyruğuna yazılır.



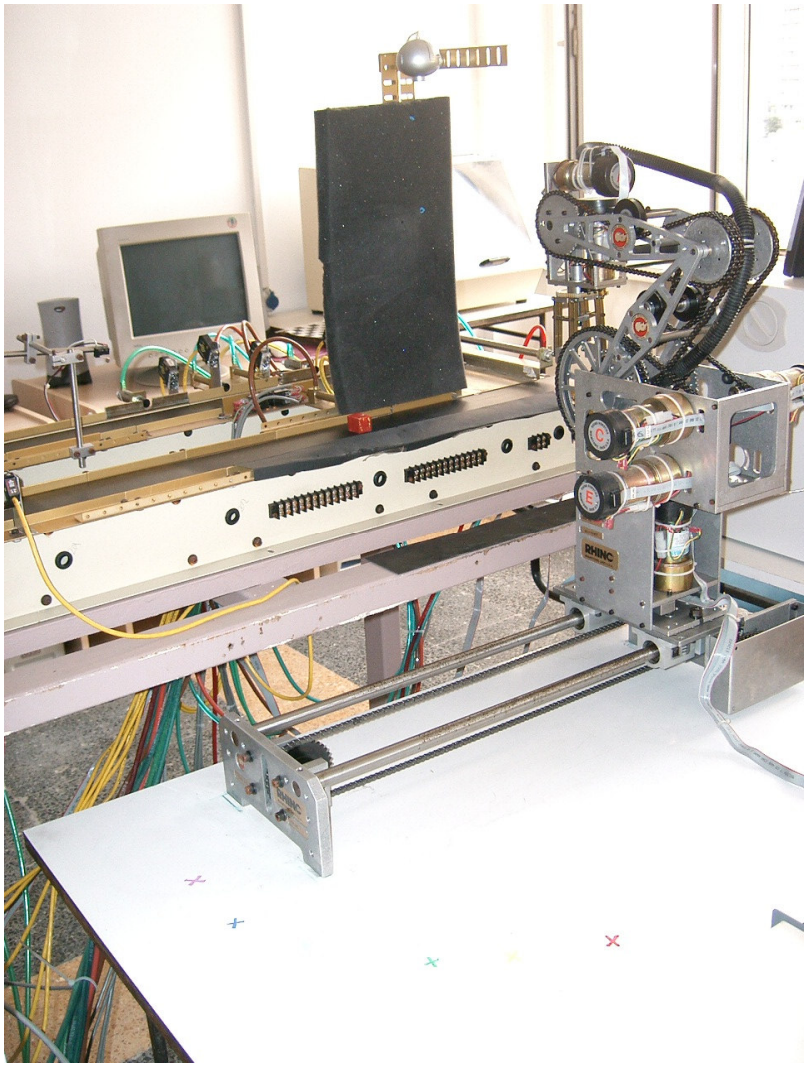
Resim 6.2. Sınıflandırma sisteminin genel görünümü (home pozisyon)

Arka planda sürekli çalışmakta olan robot denetleyicisi iletişimi iş parçacığı durumu fark eder ve gerekeni yapmak için harekete geçer. Aynı anda ana kod bloğu, görüntü yakalama ve hazırlama iş parçacığının geçici hafızada sürekli olarak güncel tuttuğu hazırlanmış görüntüleri renk sırasına göre teker teker ele alarak herhangi bir renk yoğunluğunun oluşup oluşmadığını denetleyen bir döngü içerisine girer.

Bu esnada konveyör hareket etmekte ve kamera iletişim iş parçacığının sürekli olarak sunduğu ve görüntü yakalama ve hazırlama iş parçacığının sürekli olarak güncellediği görüntü yukarıda belirtildiği şekilde renk sırası ile döngü içerisinde kontrol edilmektedir.

Ana kod bloğu sistemin tanıdığı renklerden herhangi birini kontrol ederken bir yoğunluğa rastlayacak olursa döngüden çıkar ve konveyörün durdurulması için gerekli olan mesajı kuyruğa yazar. Durumu fark eden robot denetleyicisi iletişimi iş

parçacığı ilgili mesajı Mark3 robot denetleyicisine göndererek konveyörün durdurulmasını (Resim 6.3) sağlar.

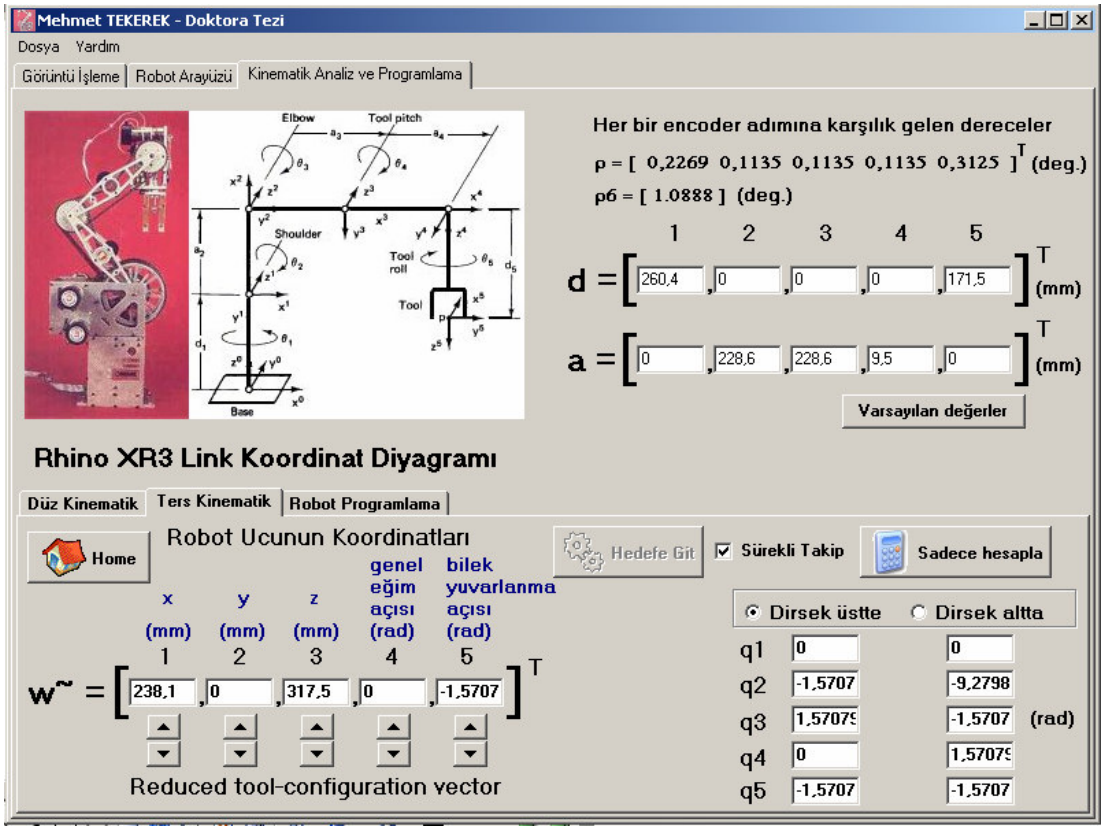


Resim 6.3. Kırmızı için renk tanımlamasının yapılarak konveyörün durdurulması

Döngüden çıkan ana kod bloğu görüntü yakalama ve hazırlama iş parçacığından durmuş olan konveyörün görüntüsünün güncel halini alır. Döngüden çıktığı andaki renge ait olan görüntü üzerinde daha önce bahsedilen görüntü işleme kodlarını çalıştırır. Ardından robotun gitmesi gereken noktanın piksel koordinatları tespit edilir. Bu işlem önceki bölümlerde ifade edildiği gibi piksel koordinatlarıyla robot koordinatları arasındaki dönüşümün köşe kalibrasyonu yardımıyla interpolasyon yapılması ile gerçekleştirilir.

Robotun ulaşması gereken noktanın koordinatlarını, robot koordinatları cinsinden hesaplayan ana kod bloğu, robota gönderilecek olan eklem açılarını hesaplamak için daha önceden hazırlanmış olan kinematik denklem çözüm algoritmalarına başvurur.

Ana kod bloğunun bu amaç için kullanabileceği iki kinematik denklem metodu mevcuttur. Bunlardan birincisi ters kinematik denklem çözümü, diğeri ise düz kinematik denklem çözümüdür. Yapılan deneyler göstermiştir ki bu tez çalışmasında kullanılan ve robot hareketlerinin özgürce yapılamadığı sınırlı çalışma alanında XR3 robotu için hazırlanan ters kinematik denklem çözümleri bir takım tekilliklere (singularity) yol açabilmektedir.



Şekil 6.16. Ters kinematik denklem çözümünün yapılması ve her bir eksen için radyan cinsinden sonuçların verilmesi

Öyle ki ters kinematik denklem çözümü sonucunda teoride elde edilen bazı açı değerleri pratikte elde edilememekte ya da elde edilen bir takım açı değerleri robota uygulandığında bazen uç noktadaki açısal yanaşma mümkün olamamaktadır.

Yukarıda bahsedilen olumsuzlukların temel nedeni, XR3 robotun eksen sayısının az olması ve çalışma alanına dik yanaşma istenildiği için sabitlenen son eksenin açısal konumundan dolayı ters kinematik denklem çözümüne gelen sınırlamadır. Tüm bu olumsuzluklara rağmen, her durumda çözüm üretebilmek ve çözümün robotun tüm işlemlerinde gösterimi maksadı ile ters kinematik denklem çözümü sadece matematiksel olarak çözülmüş ve programlanmıştır.

Tanımlı w,a matrisi elemanları ile q matrisini oluşturan ters kinematik prosedürü aşağıdaki gibi gerçekleştirilmiştir.

```
function TersKinematik2:boolean;

var q234,b1,b2:real;

begin

try

 q_ElbowUp[1]:=arctan2(w[2],w[1]);

 q_ElbowDown[1]:=q_ElbowUp[1];

 SinCos(q_ElbowUp[1],S1,C1);

 q234:=arctan2(-C1*w[4]-S1*w[5],-w[6]);

 SinCos(q234,S234,C234);

 b1:=C1*w[1]+S1*w[2]-a[4]*C234+d[5]*S234;

 b2:=d[1]-a[4]*S234-d[5]*C234-w[3];

 q_ElbowUp[3]:=arccos( (sqr(b1)+sqr(b2)-sqr(a[2])-sqr(a[3])) / (2*a[2]*a[3]) );

 q_ElbowDown[3]:=-q_ElbowUp[3];
```

```
SinCos(q_ElbowUp[3],S3,C3);

q_ElbowUp[2]:=arctan2( (a[2]+a[3]*C3)*b2 - a[3]*S3*b1 , (a[2]+a[3]*C3)*b1
+ a[3]*S3*b2 );

SinCos(q_ElbowDown[3],S3,C3);

q_ElbowDown[2]:=arctan2( (a[2]+a[3]*C3)*b2 - a[3]*S3*b1 ,
(a[2]+a[3]*C3)*b1 + a[3]*S3*b2 );

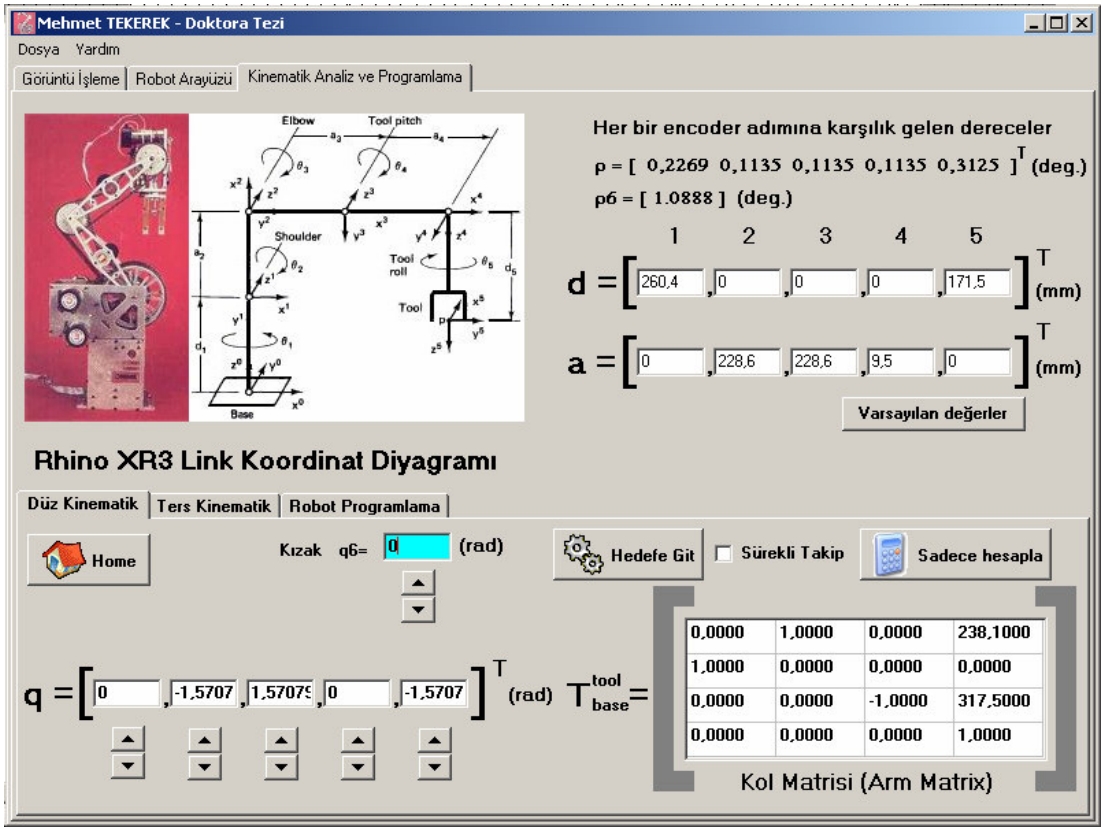
q_ElbowUp[4]:=q234-q_ElbowUp[2]-q_ElbowUp[3];

q_ElbowDown[4]:=q234-q_ElbowDown[2]-q_ElbowDown[3];

q_ElbowUp[5]:=Pi*Ln(sqrt(sqr(w[4])+sqr(w[5])+sqr(w[6])));
```

Bunun yanında problemin düz kinematikle çözülmesine karar verilmiş ve aşağıda verilen yapı oluşturularak robot kontrolü sağlanmıştır.

Daha önce piksel koordinatlarıyla robot koordinatları arasıdaki dönüşümü yapmak üzere ortaya konan köşe koordinatları kavramı burada da kullanılmakta ve köşe koordinatlarını saklamak için kullanılan veri bloğunun içine yazılmış olan eksen açıları kullanılarak ulaşılması istenilen konum için gerekli olan eksen açıları interpolasyon yapılarak hesaplanmaktadır.



Şekil 6.17. Düz kinematik denklem çözümünün yapılması ve kol matrisinin oluşturulması

Tanımlı q matrisi elemanları ile T matrisini oluşturan düz kinematik prosedürü şu şekildedir.

procedure DuzKinematik;

begin

SinCos(q[1],S1,C1);

SinCos(q[2],S2,C2);

SinCos(q[5],S5,C5);

SinCos(q[2]+q[3],S23,C23);

SinCos(q[2]+q[3]+q[4],S234,C234);

```
T_Tool_Base[1,1]:=C1*C234*C5 + S1*S5;

T_Tool_Base[2,1]:=S1*C234*C5 - C1*S5;

T_Tool_Base[3,1]:=-S234*C5;

T_Tool_Base[4,1]:=0;

T_Tool_Base[1,2]:=-C1*C234*S5 + S1*C5;

T_Tool_Base[2,2]:=-S1*C234*S5 - C1*C5;

T_Tool_Base[3,2]:=S234*S5;

T_Tool_Base[4,2]:=0;

T_Tool_Base[1,3]:=-C1*S234;

T_Tool_Base[2,3]:=-S1*S234;

T_Tool_Base[3,3]:=-C234;

T_Tool_Base[4,3]:=0;

T_Tool_Base[1,4]:= C1 * ( a[2]*C2 + a[3]*C23 + a[4]*C234 - d[5]*S234);

T_Tool_Base[2,4]:= S1 * ( a[2]*C2 + a[3]*C23 + a[4]*C234 - d[5]*S234);

T_Tool_Base[3,4]:=d[1] - a[2]*S2 - a[3]*S23 - a[4]*S234 - d[5] * C234;

T_Tool_Base[4,4]:=1;

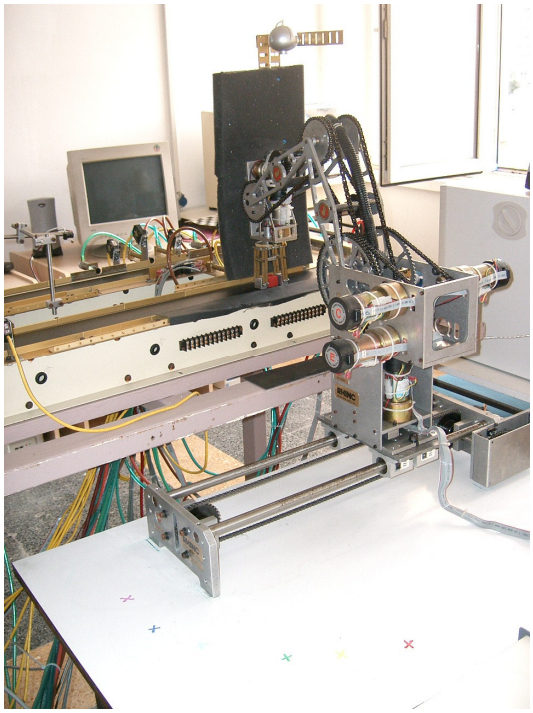
end;
```

Elde edilen bu açısal değerler son olarak karşılıkları olan ve Mark3 denetleyicisinin algıladığı enkoder darbeleri cinsinden olan karşılıklarına dönüştürülür. Ana kod bloğu, enkoder darbeleri cinsinden olan eksenin açısal konumlarını mesaj kuyruğuna

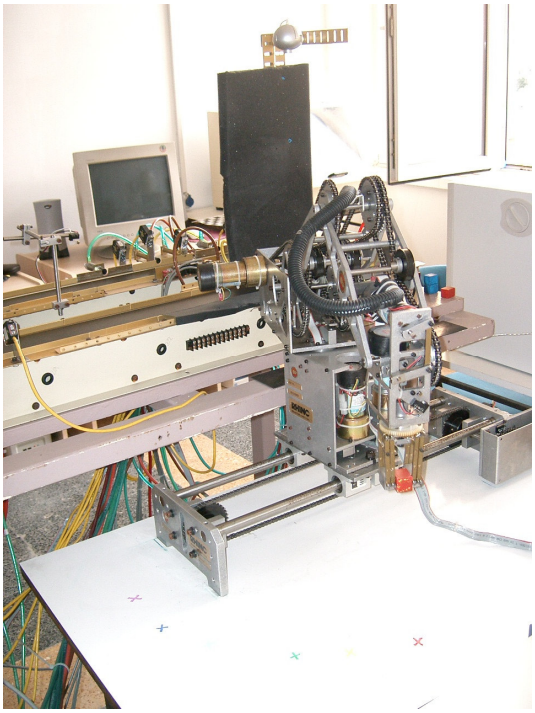
yazar ve durumu fark eden robot *denetleyicisi iletişimi iş parçacığı* devreye girerek PC ile Mark3 robot denetleyici arasındaki gerekli mesaj trafiğini başlatır.



(a)



(b)



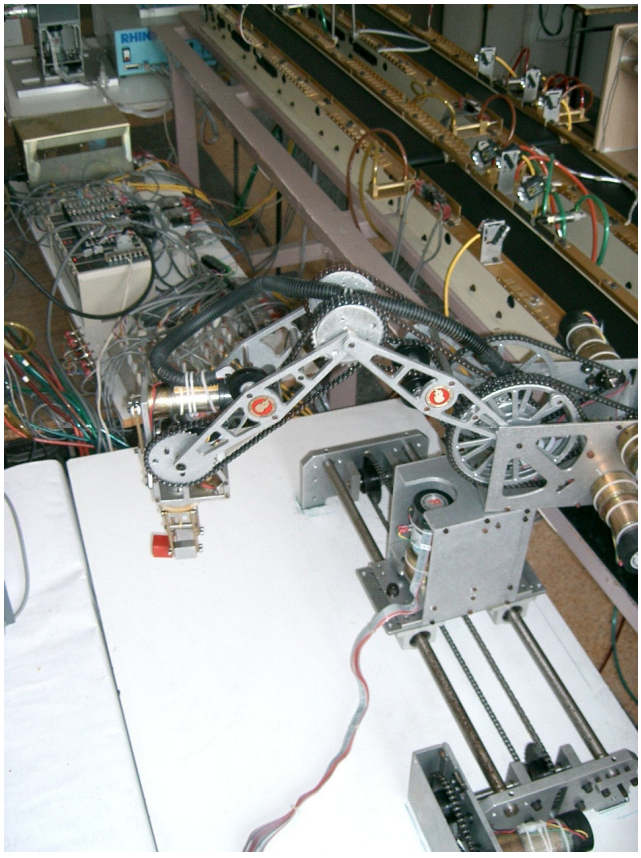
(c)

Resim 6.4. Tanıma işlemi tamamlanan nesnenin konveyör durdurulduktan sonra alınarak taşınması (a) Yakalama (b) Alma (c) Taşıma

Mark3 robot denetleyicisine bir seferde gönderilebilecek enkoder darbeleri cinsinden olan açısal değerler en yüksek 95 değerini alabilir. Çoğu zaman bu değerden çok daha yüksek olan açısal konum talepleri söz konusu olduğundan açısal değerlerin parçalı paketler halinde gönderilmesi için gerekli olan rutin işlemler de *robot denetleyicisi iletişimi iş parçacığı* tarafından ana kod bloğuna ek yük getirmeden

yapılır. 95 değeri Mark3 denetleyicinin fiziksel olarak sınırlandırdığı bir değerdir ve problemin optimizasyonu için kullanılan algoritma Şekil 6.15'te gösterilmektedir.

Hedef noktaya ulaştırılan robotun, gönderilen tutma komutuyla nesneyi yakalaması sağlanır. Yakalanan nesne çarpmaların önüne geçmek için önce bir miktar yukarıya kaldırılır, ardından da daha önceden belirlenmiş noktaya götürülür. Bu işlem yukarıda anlatılan ve robotun home pozisyondan belirlenen ve konveyör üzerinde olan ilk hedef noktaya götürülmesi gibidir. Ancak burada robota hedef nokta olarak görüntü işleme kodlarının sonucunda elde edilen nokta değil, nesneyi bırakması istenilen alanın koordinatları hedef olarak gösterilir.

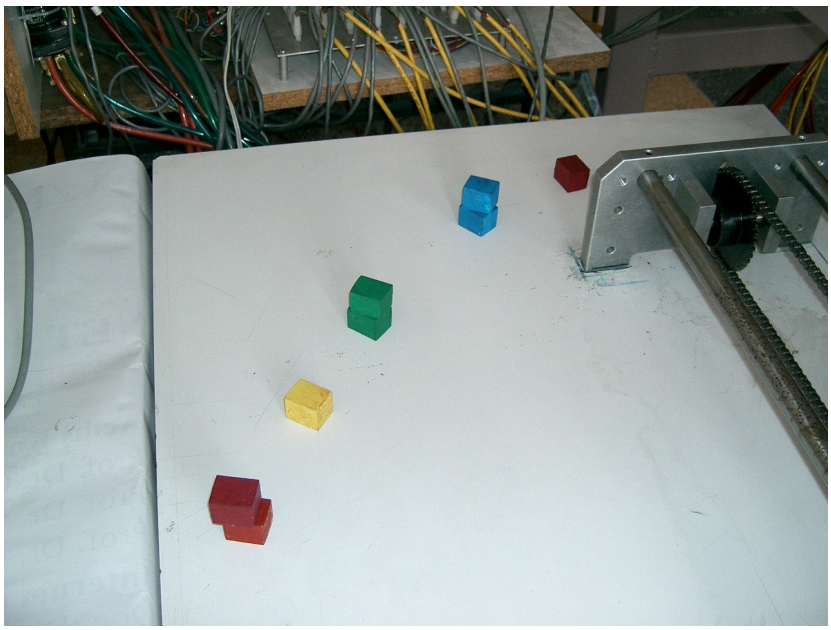


Resim 6.5. Tanımlanan renkli nesnenin önceden belirlenen alana bırakılması

Bu sistemin nesneleri renklerine ayrıştırması istenmektedir. Farklı renkli nesnelerin bırakılacakları alanlar arasında sabit bir konum farkı tanımlanmıştır. Rengi belirlenen cismin renginin renk dizisindeki sırası ile bu fark uzunluk çarpılarak cismin bırakılacağı noktanın ilk rengin bırakıldığı noktadan farkı hesaplanır ve hedef nokta

tayin edilmiş olur. Bu yapı robotun dinamikleri göz önüne alınarak oluşturulmuştur. Hedef noktaya yukarıda prensipleri detaylı bir şekilde anlatıldığı gibi ulaşan robota tutmakta olduğu cismi bırakması için gerekli olan komut gönderilir.

Bu işlemin ardından döngüyü tamamlamak maksadıyla robota son olarak home pozisyon hedef olarak gösterilir. Robotun home pozisyona ulaşmasıyla yeni bir döngünün başlatılması mümkün olabilir.



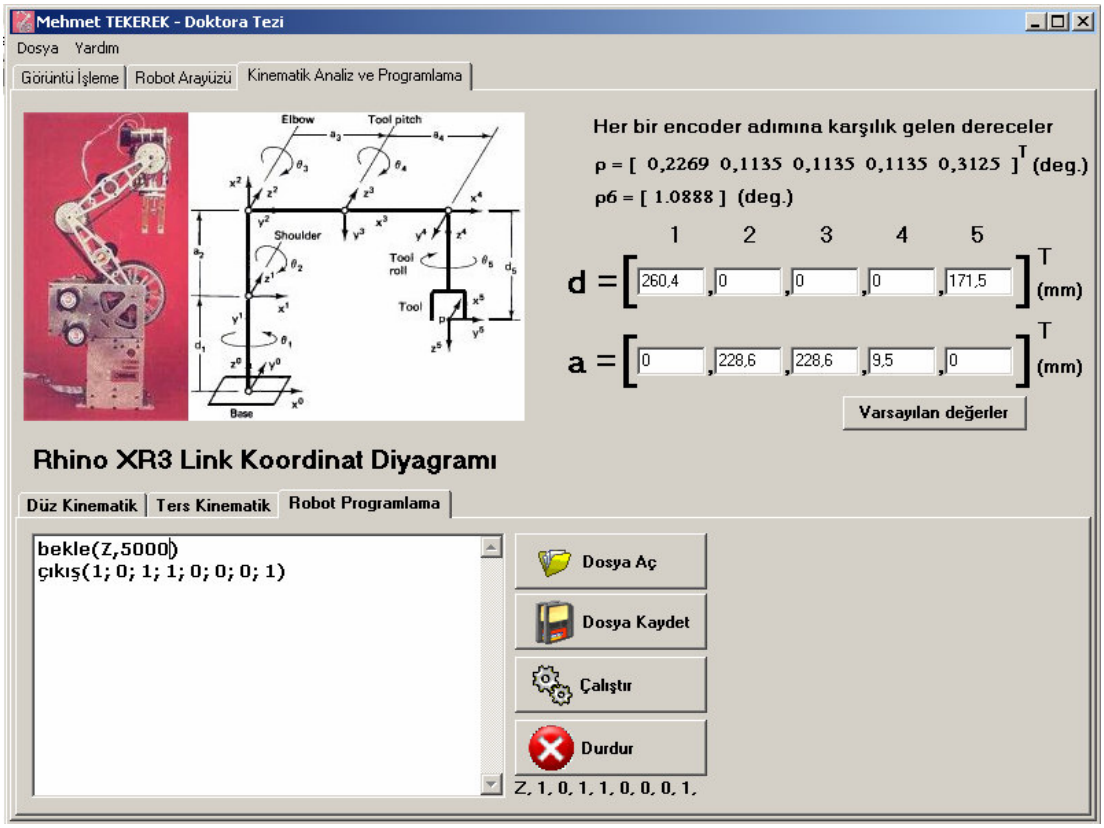
Resim 6.6. Renklerine göre ayrıştırma işlemi

6.5. Türkçe Robot Programlama Dili

Eğitim amaçlı olarak da kullanılması istenen bu robot sistemine istenilen manevraları basit bir şekilde yaptırabilmek için bir insan-makine arayüzü (man-machine interface) arayışı içerisine girilmiş ve en uygun yöntemin Türkçe komutlardan oluşan bir programlama dilinin geliştirilmesi olduğu kanısına varılmıştır.

Bu maksatla bir komut seti hazırlanmış ve sistemi eğitim amaçlı kullanacak olan öğrencilerin gerçekleştirmek isteyecekleri manevraları robot sistemine basit bir şekilde ifade edebilmeleri sağlanmıştır(Şekil 6.18).

Program, yapılmış bir döngüyü kaydetmeye ve daha önceden kaydedilmiş herhangi bir döngüyü çağırarak çalıştırmaya izin vermekte ve hata denetimi yapabilmektedir.



Şekil 6.18. Üzerinde örnek program yer alan Türkçe Robot Programlama arayüzü

6.5.1. Komut seti

Bu bölümde, tanımlanan Türkçe komutlar teker teker incelenecek ve hangi maksatla kullanılacakları açıklanacaktır. Geliştirilen komutların Türkçe olması, eğitim amacıyla bu sistem üzerinde çalışma yapacak öğrencilere kolaylık sağlaması açısından önemlidir. Konu hakkında yapılan literatür taraması çalışmalarından da görüleceği üzere XR3 robot sistemleri için uygun bir Türkçe Robot Programlama dili üzerinde yapılmış olan herhangi bir çalışma yoktur. Bu nedenle bu çalışma özgün bir çalışma olup, bir ilk niteliği taşımaktadır.

Ancak Türkçe olarak tanımlanan komutların içerebileceği Türkçe karakterlerin, bu karakterleri içermeyen klavye sistemlerinde kullanılmasının yaratabileceği zorluğu

ortadan kaldırmak ve yeni geliştirilen dilin dil uyumluluğu sorunu nedeniyle gelişiminde yaşanabilecek aksaklıkları gidermek maksadıyla komutlar çoklu olarak tanımlanmıştır. Bir başka ifadeyle, bir komut içerisinde Türkçe bir karakter bulunması halinde latin alfabesi içerisinde o karaktere en yakın olan karakterin kullanılması durumunda da sistem hata vermemekte ve komutu bu şekliyle de algılayabilmektedir.

Ayrıca bu yeni dil, komutlarda büyük küçük harf ayrımı yapmamaktadır. Türkçe robot programlama dili geliştirilmesi çalışması kapsamında sistemin kullanıcının yapabileceği dizin hatalarını da bulup göstermesi amaçlanmıştır. Bunu sağlamak için, geliştirilen yazılıma bilgisayar programı derleyicilerinde olduğu gibi hatalı yeri bulup işaretleyen bir kod yerleştirilmiştir. Bu kod, Türkçe robot programlama dili ile yazılan herhangi bir kod bloğunu, çalıştırmadan önce dizin hatalarına karşı denetlemekte ve eğer hata yoksa kodun çalışmasına izin vermektedir.

Türkçe robot programlama dilinin şu anda tanımlı olan 5 farklı komutu bulunmakta olup bu komutlardan bazıları farklı girilen parametrelere göre farklı şekillerde de çalışabilmektedir. Tanımlı olan kod sayısının ileride yapılabilecek olan çalışmalarla artırılabilecek olması dolayısıyla sistem geliştirilmeye açıktır.

Yeni geliştirilen dilde henüz herhangi bir koşulun sağlanması halinde dallanma yapabilme yetisi yoktur. Bir başka ifadeyle programlama dillerinde “IF-THEN” şeklinde bilinen programlama yapısı bu sistem için henüz geliştirilmemiştir. Aynı durum, “GOTO” şeklindeki koşulsuz dallanmalar için de geçerlidir.

Türkçe robot programlama dilinin şu anda tanımlı olan 5 farklı komutu aşağıda listelenmiştir.

- git
- bekle
- tut
- bırak
- çıkış

"Git" komutu: Bu komutla robot üzerinde bulunan ve beşi dairesel, biri de doğrusal olan toplam altı eksene istenilen açısal hareketler radyan cinsinden gönderilebilir. Komut, başlangıçta "B" ya da "M" olabilen bir parametre almakta ve bu şekilde istenilen hareketin bağıl mı (B), yoksa mutlak mı (M) olacağına karar vermektedir.

İstenilen hareket bağılsa(B), bu komutla gönderilen sayısal değerler negatif ya da pozitif oluş durumlarına göre fark olarak eksenlere yansıtılmakta ve eksenlerde belirtilen değerler kadar adımsal bir sapma sağlanmaktadır. İstenilen hareket mutlaksa(M), bu komutla gönderilen sayısal değerler doğrudan eksenlere hedef olarak gösterilmekte ve robotun eksenlerinin bu pozisyonlara gelmesi sağlanmaktadır. Komut dizilimi için iki örnek aşağıda sunulmuştur.

git(B; 0; 0; 0.2; 0; 0; -2)

git(M; 1.56; 0.3; 0.5; 2; 3; 4.1)

Bu komutlardan üstteki çalıştırıldığında sistem, istenilen hareketin bağıl hareket olduğunu anlamakta ve robot eksenlerinden 3. eksenin açısal konumunu 0.2 radyan artırmakta, son eksen olan kızağın motorunun açısal konumunu da 2 radyan azaltmaktadır. Alttaki komutta ise sistem, istenilen hareketin mutlak hareket olduğunu anlamakta ve robot eksenlerine sırasıyla, 1.56, 0.3, 0.5, 2, 3 ve 4.1 değerlerini hedef olarak vermektedir.

"Bekle" komutu: Bu komutla robotun bekleme yapması sağlanmaktadır. Komut, başlangıçta "hareketi bekle- H" ya da "zamanı bekle- Z" olabilen bir parametre almakta ve bu şekilde istenilen beklemenin, sürmekte olan hareketin tamamlanmasına kadar mı H, yoksa belirtilen bir sürede mi Z yapılacağına karar vermektedir.

İstenilen bekleme, sürmekte olan hareketin tamamlanmasına kadarsa, robot sonraki komutlara geçmeyip yapmakta olduğu hareketin bitmesini bekleyecektir. İstenilen bekleme, süre tabanlı ise, robot, Z parametresinin ardından girilen sayısal değerin milisaniye cinsinden değeri kadar bir süre boyunca sonraki komutları çalıştırmayıp bekleyecektir. Komut dizilimi için iki örnek aşağıda sunulmuştur.

bekle(H)

bekle(Z,1800)

Bu komutlardan üstteki çalıştırıldığında sistem, istenilen beklemenin sürmekte olan hareketin tamamlanmasına kadar olduğunu anlamakta ve sonraki komutları işletmemektir. Eğer bu durum zaten sağlanmışsa, bir başka ifadeyle robot zaten o esnada hareket etmiyorsa ve boştaysa bu kodun herhangi bir etkinliği olmayacaktır ve doğrudan sıradaki diğer komuta geçiş yapılacaktır. İkinci satırdaki komutta ise sistem, istenilen bekleme süresinin zaman esaslı olduğunu anlamakta ve 1800 milisaniyelik beklemenin ardından sıradaki diğer kodlara geçiş yapmaktadır.

"tut" komutu: Bu komutla robotun uç elemanının kapanıp, nesnenin tutulması sağlanmaktadır. Komut dizilim yapısı aşağıdaki gibidir:

tut

Her ne kadar bu komutun işlevi *git* komutunun içine eklenecek eksen değerini ifade edebilecek bir sayı ile gerçekleştirilebilecek olsa da kullanıcıların program akışını algılamasını kolaylaştırmak için ayrı tutulması uygun görülmüştür.

"bırak" komutu: Bu komutla robotun ucundaki elin açılıp, nesnelerin bırakılması sağlanmaktadır. Komut dizilim yapısı aşağıdaki gibidir:

bırak

Bu komutun da işlevi *git* komutunun içine eklenecek eksen değerini ifade edebilecek bir sayı ile ya da *tut* komutu ile birleştirerek gerçekleştirilebilecek olsa da kullanıcıların program akışını algılamasını kolaylaştırmak için ayrı tutulması uygun görülmüştür.

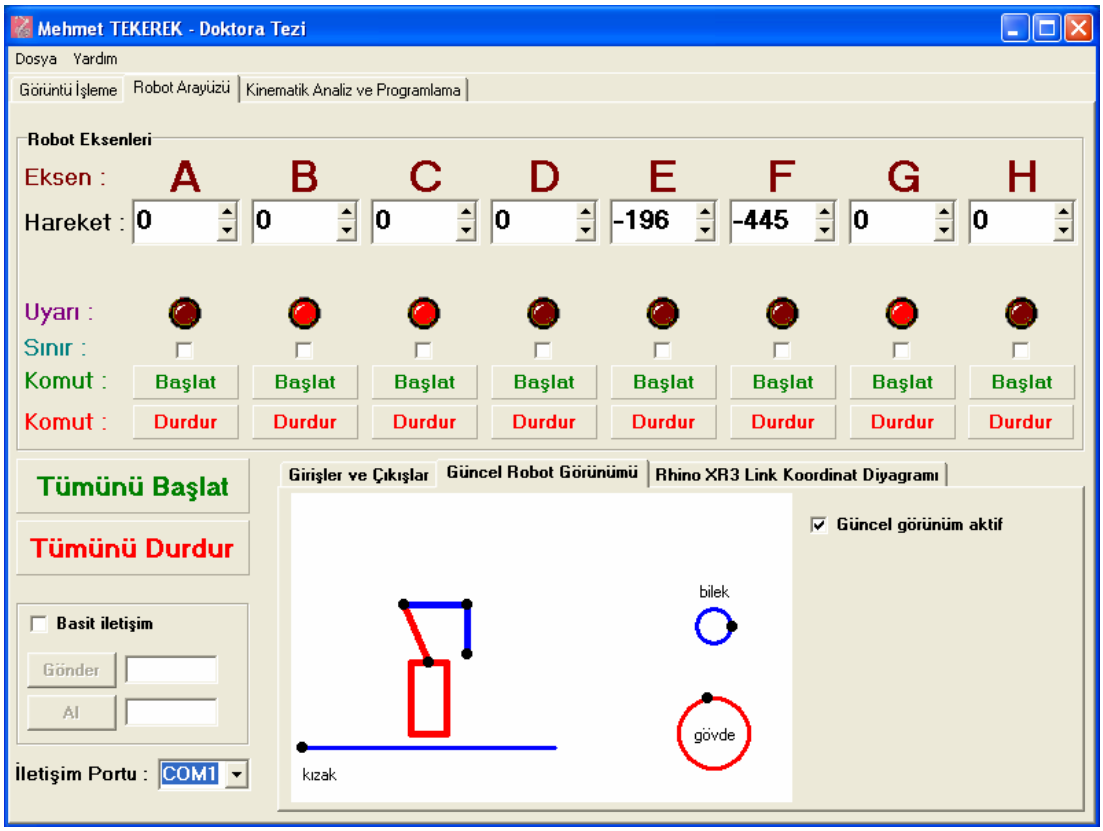
"çıkış" komutu: Bu komutla Mark3 denetleyici üzerinde bulunan 8 adet dijital çıkışın istenilen değeri alması sağlanabilmektedir. Komut dizilim yapısı aşağıda sunulmuştur:

çıkış(1; 0; 1; 1; 0; 0; 0; 1)

Komut diziliminde görünen 1'ler çıkışları açarken, 0'lar ise kapamaktadır. Yukarıda verilen örnek kod çalıştırıldığında, 1 görünen 1., 3., 4. ve 8. çıkışlar açılırken, 0 görünen 2., 5., 6. ve 7. çıkışlar kapatılacaktır.

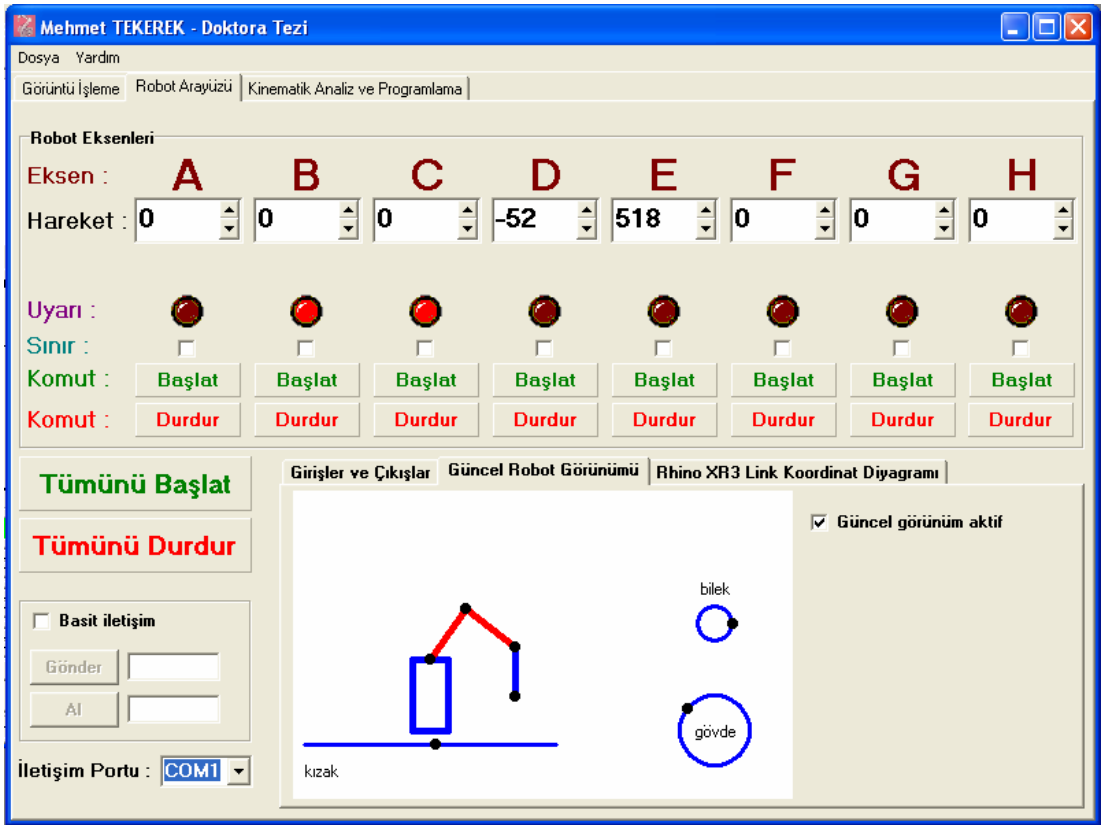
6.6. XR3 Robotun Anlık Konumunun İki Boyutlu Görüntü İle İfade Edilmesi

Bu tez çalışma kapsamında geliştirilen unsurlardan bir tanesi de Rhino XR3 robot temelli oluşturulan ve görsel takip sağlanması amaçlanan simülasyondur. Robot simülasyonu tez kapsamında geliştirilen yazılıma hem hareketlerinin gözlemlenmesi için robot kontrol arayüzüne hem de kinematik hesaplamalar sonucunda elde edilen matematiksel değerler sonucunda robotun hareket uzayının gözlenmesini sağlamak amacıyla kinematik analiz ve programlama arayüzüne yerleştirilmiştir.



Şekil 6.19. Kullanıcı etkileşimli robot simülasyonu (home pozisyon)

Bunun yanında robot hareket uzayının ve eksenlerin artı ve eksi dönüş yönlerinin belirtilerek görülebilmesine yönelik olarak, fare etkileşimli bir yapı geliştirilmiştir(Şekil 6.19). Uygulamada XR3'ün hareketini kopyalayan yapıda enkoder değerleri matematiksel olarak hesaplanarak eksen açılarına dönüştürülerek gerçekleştirilen simülasyonda hareket istenen eksen, fare ile seçilerek hareket yönü artı veya eksi olarak belirlendikten sonra robot hareketi başlatılabilmektedir(Şekil 6.20). Burada ikinci boyutta yer alan kızak ekseni ve üçüncü boyutu oluşturan gövde ve bilek eksenler ana gövdenin dışında gösterilmiştir.



Şekil 6.20. Robot simülasyonunda artı ve eksi yönde eksen hareketlerinin gösterimi

Hareket etmesi istenen eksen hareketi sırasında ve tamamlandıktan sonra mavi renkten kırmızıya dönüştürülerek görsel olarak fark edilmesi sağlanmıştır. Hareket yönü de (+) ve (-) işaretleri ile eksen adına ait kutucukta klavye ve fare etkileşimli olarak verilmiştir.

XR3 robotun hareketlerini iki boyutlu bir görüntü ile ifade etmenin iki yolu vardır. Bunlardan birincisi üç boyutlu görüntünün iki boyutlu düzlemde iz düşümünün gösterilmesi, diğeri ise ortak düzlemdeki eksen gruplarının ayrı ayrı gösterilmesidir.

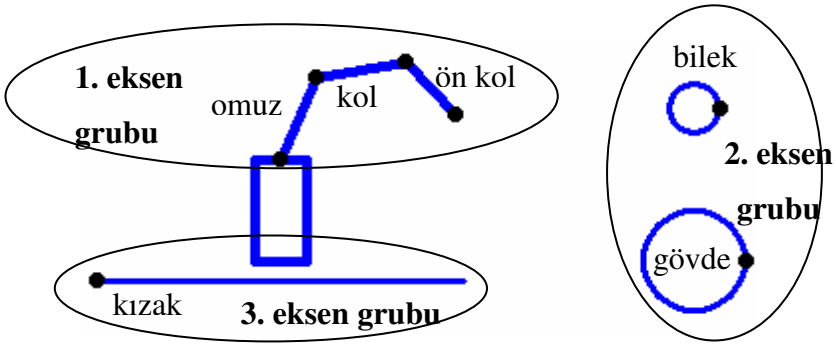
6.6.1. Üç boyutlu görüntünün iki boyutlu düzlemde iz düşümünün gösterilmesi

Bu yöntemde amaç robotun eksenlerinin anlık açısal konumlarını kullanarak eksenlerin üç boyutlu uzaydaki konumlarının iki boyutlu düzlemde iz düşümünü hesaplamak ve bu görüntüyü oluşturmaktır. Bu çalışmanın kapsamı dâhilinde olmayan detaylı gölgelendirme ve grafik kütüphanelerinin kullanılması gibi tekniklerin kullanılmaması halinde bu metodun kullanımıyla oluşacak olan perspektif görüntü nedeniyle iki boyutlu basit gösterimde görsel ifadenin zayıf olacağı ve bunun da robotun konumunun programı eğitim amacıyla kullanacak olan kişilerce algılanmasını zayıflatacağı düşünülmektedir. Bu nedenle bu çalışmada üç boyutun görüntünün iki boyutlu düzlemde iz düşümünün gösterilmesi metodunun kullanımından vazgeçilmiştir.

6.6.2. Ortak düzlemdeki eksen gruplarının ayrı ayrı gösterilmesi

Bu yöntemde ortak düzlemde yer alan robot eksenleri bir araya getirilir ve oluşan her grup ayrı ayrı gösterilir. XR3 robotunun link koordinat diyagramı incelenecek olursa robotun omuz, kol ve ön kol eksenlerinin açısal hareketlerini aynı düzlemde gerçekleştirdikleri görülür. Bu açısal düzlemde hareket eden başka bir eksen daha olmadığına göre bu üç eksen bir eksen grubu oluşturabilir. Bir diğer eksen grubu ise gövde ve bilek eksenlerinin oluşturduğu eksen grubudur. Bu iki eksenin de hareketi aynı düzlem üzerindedir. Üzerine robotun gövdesinin monte edildiği kızak da, içinde tek eksenin yer aldığı bir diğer eksen grubu olarak kabul edilebilir. Özetlenecek olursa XR3 robotunun hareketleri üç ayrı düzlemde üç ayrı eksen grubu tarafından oluşturulmaktadır. Bu üç eksen grubunun iki boyutlu düzlemde ayrı ayrı ifade gösterildiği bir görüntü, programı eğitim amacıyla kullanacak olan kişiler açısından robotun konumunun algılanmasını kolaylaştırmaktadır. Şekil 6.21'de eksen gruplarının ayrı ayrı olarak iki boyutlu düzlemde gösterimine ait görüntü verilmiştir. Bu gösterimde siyah olarak ifade edilen noktalar eklemlerin anlık konumlarını

göstermektedir. Mavi olarak ifade edilen eksenlerin hareket halindeyken kırmızı olarak gösterilmesiyle ifadenin görselliğin arttırılabileceği düşünülmektedir.



Şekil 6.21. Eksen gruplarının ayrı ayrı olarak iki boyutlu düzlemde gösterimi

XR3 robotunun eksen ölçülerini ifade eden *d* ve *a* matrisleri incelenecek olursa birinci eksen grubunu oluşturan omuz, kol ve ön kol eksenlerinin uzunluklarının a_2, a_3 ve (a_4+d_5) şeklinde ifade edildikleri görülür. Bu ifadelerden a_2, omuz eksenin robot gövdesine bağlandığı eklemden kol ekseninin omuz eksenine bağlandığı ekleme kadar olan mesafeyi; a_3, kol ekseninin omuz eksenin bağlandığı eklemden ön kol ekseninin kol eksenine bağlandığı ekleme kadar olan mesafeyi ve (a_4+d_5) ise ön kol ekseninin kol eksenine bağlandığı eklemden bilek ekseninin uç kısmına kadar olan mesafeyi ifade etmektedir. Birinci eksen grubunu oluşturan bu eksenlerin robot gövdesine bağlandıkları ilk ekleminin konumu sabit tutarak ikinci eklemin konumu, a_2 uzunluğu ve omuz ekseninin açısal konumu kullanılarak aşağıdaki trigonometrik işlemle hesaplanabilir.

$$E2x = a_2.\cos\theta_2 + E1x$$

$$E2y = a_2.\sin\theta_2 + E1y$$

Üçüncü eklemin konumu hesaplanırken kendisinden bir önce konumu hesaplanan ikinci eklem sabit varsayılıp, üçüncü eklemin konumu a_3 uzunluğu ve kol ekseninin açısal konumu kullanılarak aşağıdaki trigonometrik işlemle bulunur ve bulunan

değere ikinci eklemin daha önce hesaplanan konumunun eklenmesiyle ilk başta yapılan sabit eklem varsayımı ortadan kaldırılır.

$$E3x = a_3.\cos\theta_3 + E2x$$

$$E3y = a_3.\sin\theta_3 + E2y$$

Bu işlemin aşağıda verilen ifade ile son eklem olan dördüncü ekleme kadar sürdürülmesiyle birinci eksen grubunda yer alan tüm eklem noktalarının koordinatları bulunmuş olur.

$$E4x = (a_4 + d_5).\cos\theta_4 + E2x$$

$$E4y = (a_4 + d_5).\sin\theta_4 + E2y$$

Koordinatları cm cinsinden bulunan tüm bu noktalar, bitmap görüntü üzerine aktarılırken istenen bir ölçeklendirme katsayısı ile çarpılarak değerler ölçülendirilir ve piksel olarak kabul edilerek değerlendirilir. Bunda amaç görüntünün istenen büyüklükte olmasını sağlamaktır. Bu katsayının artması büyük bir görüntü oluşmasını sağlarken, azalması ise daha küçük bir görüntü oluşturulmasını sağlar.

İkinci eksen grubunu oluşturan gövde ve bilek eksenleri birbirinden bağımsız şekilde döndükleri için bu eksenler için kinematik bir hesaplama yapmaya gerek yoktur. Eksenlerin açısal konumlarının sadece siyah bir nokta ile görüntülenmesi yeterli olmaktadır. Bunun için sadece varsayılan daire yarıçapı kullanılarak siyah noktanın anlık konumu hesaplanır.

$$g_x = r_g.\cos\theta_1$$

$$g_y = r_g.\sin\theta_1$$

$$b_x = r_b.\cos\theta_5$$

$$b_y = r_b.\sin\theta_5$$

Bu ifadelerde yer alan r_g ve r_b ifadeleri birinci eksen grubundaki a ve d ifadelerinden farklı olarak XR3 robotun ölçüsel değerlerinden bağımsız olup, iki boyutlu gösterimde gövde ve bilek eksenlerinin gösterimi için kullanılan dairelerin çaplarını belirlemek için kullanılır. Hesaplama sonucu elde edilen değerler görüntü üzerinde gövde ve bilek konumunu sembolize eden siyah noktaların x ve y koordinatlarıdır.

Benzer bir durum üçüncü eksen grubunun tek elemanı olan kızak ekseni için de geçerlidir. Bu eksenin hareketinin tamamı 3050 enkoder darbesiyle gerçekleşmektedir. Eksen motorunun bir turu ise 165 enkoder darbesine karşılık gelmektedir. Bu eksenin hareketinin gösterimi için görüntü üzerinde ölçeklendirme katsayısı ile 3050'nin çarpımı sonucunda piksel cinsinden uzunluğu elde edilen bir çizgi oluşturulup bu çizgi üzerinde yer alacak siyah noktanın konumu da eksen motorunun tur sayısıyla orantılı olarak görüntülenir.

Robot üzerinde yer alan tüm eklemlerin konumlarının hesaplanmasının ardından hesaplanan değerlerin bitmap görüntü üzerinde gösterilmesi aşaması başlar. Bitmap değişkenlerinde *x* ekseni standart koordinat ekseni ile aynı yönde yer alırken *y* ekseni standart koordinat ekseninin tersi yönündedir. Bir başka ifadeyle y değeri sıfır olan bir nokta üstte yer alırken, *y* değeri en yüksek olan nokta altta yer alır. En yüksek *y* değeri ise bitmap görüntünün yüksekliğinden bir eksiktir. Bu nedenle bitmap değişkeni üzerine aktarılacak olan hesaplanmış koordinatlarda *x* değeri aynı bırakılırken tüm *y* değerleri bitmap görüntünün yüksekliğinden çıkarılır. XR3 robotun anlık konumunun iki boyutlu görüntü ile ifade edilmesi işlemi oluşturulan görüntünün bilgisayar ekranında görüntülenmesi ile son bulur.

7. TARTIŞMA VE SONUÇLAR

Bu çalışmada esnek üretim sisteminde görüntü işleme tekniği ile bir robotik eğitim modeli geliştirilmesi amaçlanmıştır. Bu amaçla çalışma, teorik ve uygulama kısımları olmak üzere iki farklı ana kısımdan oluşmaktadır. Teorik bölümde bilgisayarlı görme, görüntü işleme ve bu teknolojinin esnek üretim sistemlerinde kullanılabilirliği tartışılmıştır. Böyle bir sistem geliştirmenin robotik eğitimi açısından önemine de değinilmiştir.

Uygulama kısmında ise, esnek üretim sisteminde önemli bir konu olan "sınıflandırma" konusu, robotik eğitiminin ilkeleri ışığında renk ayrıştırma temelinde işlenmiştir. Bu amaçla geliştirilen robotik eğitim modelinin unsurları, MARK3 denetleyici kontrolü, robot kontrol, görüntü işleme ve Türkçe programlama dili ve XR3 robotun anlık konumunun iki boyutlu görüntü ile ifade edilmesidir.

Esnek üretim sisteminde görüntü işleme tekniği ile robotik eğitim modeli geliştirilmesi, sistem elemanlarının uyumlu hale getirilmesi ile başlatılmıştır. Bu nedenle öncelikle eldeki kaynaklar, çalışma ortamı oluşacak şekilde yeniden düzenlenmiştir. Bu düzenleme sırasında, esnek üretim sistemleri laboratuarında var olan PLC kontrol sistemi kullanılmak yerine uygulamanın bir parçası olan konveyör banda MARK3 denetleyicinin çıkışına yerleştirilen röle devresi ile kumanda edilmiştir. Böylece, sistem, ek denetleyiciye ihtiyaç duymadan kendi başına çalışır duruma gelmiştir ki röle kumandası bizim uygulamamamız için yeterli olmuştur.

Çalışma ortamının oluşturulması sürecinde görüntü işleme işi, ortamın aydınlatılmasıyla doğrudan ilişkilidir. Oluşturulan sistemin benzer amaçlar için daha sonradan kullanılması durumunda özel ayarlamalar yapılamayacağı düşüncesiyle herhangi bir özel aydınlatma tekniği kullanılmamıştır. Bunun yerine, uygulama kısmında bahsi geçen kalibre edilmiş görüntü işleme alanı, sabit zemin rengi oluşacak şekilde düzenlenmiş ve asıl hedef olan farklı renkli nesnelerin tanınması için uygun hale getirilmiştir. Sonuçta, gün ışığının veya sabit ortam aydınlatmasının etkileri doğrultusunda renk tanıma işlemi gerçekleştirilmiştir. Bu ışık etkileri, bazen görüntü işleme alanında gürültüye neden olmaktadır. Bu sorunun aşılması için

geliştirilen yazılıma kullanıcı etkileşimli özel ayarların düzenlenebileceği bir yapı eklenmiştir.

MARK3 denetleyicinin, sahip olduğu donanım ve çalışma prensibi incelendiğinde XR3'ün sahip olduğu tüm motorlara tek tek veya hiyerarşik sıra içerisinde veri gönderebilen bir yapıya sahip olduğu, bunun yanında giriş/çıkış TTL çıkışlara sahip olduğu görülmektedir. MARK3 denetleyici kontrolü için bir yazılım aracı geliştirilmiş ve MARK3'ün tüm yeteneklerine kumanda edilebilecek yapıya ulaşılmıştır. Öyle ki; programlama mantığının temelinde yer alan ve Mark3 tarafından motorlara gönderilerek işlenebilecek en yüksek değer olan 95 değeri için düzenleme, bu arayüzde de desteklenmektedir.

Robot kontrol programı tasarlanırken, sistemin üreticisi olan RHINO firması tarafından sunulmuş olan BASIC derleyici tabanlı RoboTalk yazılımı için yapılan incelemede de eğitsel ve programlama tekniği açısından bazı kısıt denebilecek unsurların olduğu görülmüştür. Bu kısıtlar şöyle sıralanabilir: Öncelikle, RoboTalk kullanıcı etkileşimli bir yazılım olmadığı için çalışan programa müdahale edilememektedir. Yazılımın bir diğer dezavantajı sadece tam sayı değişkenleri kullanabildiği için reel değişkenleri ve trigonometrik fonksiyonları desteklemiyor olmasıdır. Uygulamada geliştirilen yeni yazılımda, Mark3 denetleyicinin bufferında kalan veri sorgulanabilmiş ve taşmaların önüne geçilerek kesiklikler önlenebilmiştir. Geliştirilen yazılımda tam sayı değişkenleri dışında trigonometrik ve reel değişkenler girilebilmektedir.

Uygulama bölümünde bahsedildiği üzere, renk sınıflandırması robot manipülatör tarafından yapılırken ters kinematik ve düz kinematik denklemleri çözülmüş ancak, ters kinematik denklemlerinin uygulamasında karşılaşılan bazı tekillik (singularity) problemlerinden dolayı, işlem düz kinematikle gerçekleştirilmiştir.

Oluşturulan düzenekte taşıma malzemesi olarak farklı yoğunluktaki renk tonlarını içeren renk küpleri kullanılmıştır. Taşınan malzemenin bırakılacağı yer olarak her renk için robot uzayı içerisinde farklı bir alan belirlenmiştir. Bu da ayrıştırma işleminin gerçekleştirilmesini sağlamaktadır.

Geliştirilen bu modelde, hedeflenen yapı oluşturulmuştur. Esnek üretim sistemi için teknolojik düzenlemenin yapılabilmesi amacıyla alan uzmanlarıyla ortak çalışmalar söz konusu olduğunda alana yönelik daha verimli deneysel uygulamalar gerçekleştirilebilir.

Robotik eğitimi açısından çalışmanın önemi, geliştirilen yazılımı, oluşturulacak herhangi bir eğitim ortamında eğitim materyali olarak kullanabilmeyi amaçlamasıdır. Bu amaç doğrultusunda geliştirilen yazılımla öğrencilerin beklentilerinin, robot sistemlere ait mekanik yapının işleyişini görsel ve matematiksel olarak karşılanması hedeflenmektedir. Öyle ki ileri robotik eğitimine geçmeyi düşünen veya sınırlı düzeyde alan bilgisi olan lisans öğrencilerinin alan çalışmalarına rehberlik edecek sadelik ve anlaşılırlık ön planda tutulmuştur.

Çalışma aynı zamanda esnek üretim sistemi eğitim çalışmaları açısından robot programlamanın değil, tezgâh planlamanın öne çıkacağı bir yazılım modeli olarak sunulmuştur.

Robot programlama eğitimi bakımından yazılıma eklenmiş olan ve yapılan ön araştırma sonucunda daha önce yapılmış bir örneğine rastlanılmayan Türkçe robot programlama dili geliştirilmiştir. Burada temel bilgisayar programlama yetilerine henüz sahip olmayan kişilerin bu dili kullanarak XR3 robotun temel hareketlerini programlayabilmeleri amaçlanmıştır. Böyle bir yapının oluşturulmasında cesaretlendiren unsur, önceki bölümlerde bahsi geçen ilköğretim ve lise çağına yönelik yeni nesil LEGO kitleri programlama teknikleridir. Bu tekniklerde eğitsel açıdan savunulan yapı, kullanılan programlama terimlerinin İngilizceye en yakın terimler dizini kullanılmasıdır.

Yapılan araştırmalar göstermiştir ki, ticari kaygılarla üretilerek piyasaya sunulan ve dünyanın birçok yerinde olduğu gibi ülkemizde de uygulamalarına rastlanan LEGO kitlerinin, özellikle ilköğretim teknoloji eğitimi ve ortaöğretim fen ve matematik alanlarında kullanımı yaygınlaşmaktadır. Robotik eğitiminde en çok şikâyet edilen konulardan bir tanesi olarak karşımıza çıkan yüksek maliyetler ve materyal sıkıntısının, bu uygulamayla aşılmaya çalışıldığı görülmektedir. Ancak, teknolojik

tabanı tam olarak oturtulmadan öğrencilerin deneme yanılma mantığı ile uygulamalar yapmasının güvenilirliği henüz tam kesinleşmiş değildir. Bu yapıyla yetişecek olan ve sloganlarda "geleceğin dehaları" şeklinde özendirilen çocuk ve gençlerin gelecekteki düzeylerinin ne olacağını şimdiden kestirmek zordur. Kaldı ki sözü edilen bu yapı incelendiğinde şimdilik özellikle ülkemiz açısından, robotik eğitiminin maliyetlerini düşürmek gibi bir özelliği görülememektedir.

Robotik eğitimine yönelik ulusal ve uluslararası alanda ülkelerin üniversiteler ve kurumsal yapılar boyutunda robotiği genel bilimler arasında bir yere oturtmak, paylaşımlar yapabilmek, eğitim ortamlarını tip projeler oluşturarak verimli ve standardize etmek için çabaları söz konusudur. Ülkemizde de bu tür faaliyetlere rastlanmakla birlikte istatistikler göz önüne alındığında bu açıdan yetersiz kaldığı izlenimi uyanmaktadır.

Bu bulgular ve tartışmalar ışığında uygulamaya yönelik özel amaçlı ve robotik eğitimine yönelik genel amaçlı öneriler sunulabilir. Uygulamaya dönük olarak bilgisayarlı görme sistemlerine dayalı esnek üretim sistemleri için aydınlatma problemleri, amaca özel proje geliştirilerek ortadan kaldırılabilir.

Bu çalışmada tek bilgisayar, tek denetleyici ve tek robot kullanılmıştır. İleriki çalışmalarda var olan laboratuvar imkânları ışığında, kullanılacak robot sayısı artırılarak deneysel çalışmalar geliştirilebilir. Yine laboratuvar imkânları dâhilinde bulunan CNC torna ve freze de sisteme adapte edilerek esnek üretim sistemini genişleten yapılar oluşturulabilir. Ayrıca benzeri çalışmalar uzaktan erişim modelleriyle desteklenebilir.

Geliştirilen Türkçe programlama dilinde uygulama kısmında bahsedildiği üzere, temel programlama döngülerine ve görsel desteğe yer verilmemiştir. Bu alanda bir çalışma yapılacak olursa bu ve benzeri eğitim amaçlı programlama dilleri geliştirilebilir.

Genellikle üniversitelerde öğrenci veya araştırma toplulukları bazında eğitim ve araştırma faaliyetlerini yürüten robot topluluklarının araştırma merkezleri haline

dönüşebilmeleri ve daha büyük çaplı ulusal ve uluslararası projelere imza atabilmeleri için eğitim programı, yeterli danışmanlık hizmeti ve materyal açısından desteklenmelidirler.

AB bünyesinde oluşturulan EURON (Avrupa Birliği Robot Ağı) çerçevesinde gerçekleştirilen eğitim faaliyetleri ve proje fonları, Altıncı Çerçeve Programında oldukça geniş yer almaktadır. Bu yapıdan ülkemizin şu ana kadar aldığı pay, son derece yetersizdir. Bu alana yönelik üniversitelerin ve ilgili kuruluşların kaynak ve eğitim materyali açısından daha fazla eğilmelerinde fayda vardır.

Teknoloji eğitiminin yapısı ele alınacak olursa robotik eğitiminin, disiplinler arası yapısıyla bu eğitim sistemiyle birebir örtüştüğü görülecektir. Teknoloji eğitimi, ilköğretimden yükseköğrenime robotik eğitimine yönelik hem proje bazında hem de müfredat geliştirilmesi açısından lokomotif unsur olarak yerini alabilir. Bu bağlamda üniversitemizin Endüstriyel Teknoloji Eğitimi Bölümü merkezli araştırma ve geliştirme projeleri ve müfredat çalışmaları yapılabilir.

Robotik eğitimi konusunda daha ileri düzeyde daha geniş kapsamlı ve geniş katılımlı politika geliştirilmesine de ihtiyaç olduğu açıktır. Sonuç olarak, robotik eğitimi konusunda politika yapıcıların da ilgili kurum, kuruluş ve uzmanlarla eşgüdüm içinde bu alana yönelik politikalar üretmeleri gerekmektedir.

KAYNAKLAR

1. He, D., Hujic, D., Mills J.K., Benhabib, B., "Moving object recognition using premarking and active vision" ***International Conference on Robotics and Automation***, USA, 7 (1996).

2. Gümüşer, M., "Dijital Görüntü İşleme", Yüksek Lisans Tezi, ***Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü,*** İstanbul, 1-2 (2001).

3. Ayberk, A., "Vision and Tracking Control of a Robot Manipulator", Yüksek Lisans Tezi, ***Dokuz Eylül Üniversitesi Fen Bilimleri Enstitüsü***, İzmir, 5-8 (2001).

4. Yenitepe, R., "Mekatronik Eğitiminde MPS Üniteleri", ***the Turkish Online Journal of Educational Technology,*** 3: 18 (2004).

5. Matari´c M., J., "Robotics Education for All Ages", ***AAAI Spring Symposium on Accessible, Hands-on AI and Robotics Education***, Palo Alto, CA, 1 (2004).

6. Milli Eğitim Bakanlığı, "Otomasyon Alanı modülü", ***MEGEP, Öğrenme Faaliyeti 37***, Ankara, 234 (2005).

7. Wood, S., "Robotics In The Classroom. A teaching tool for K- 12 Educators"**,** ***"Symposium of Growing up with Science and Technology in the 21st Century"*****,** Virginia, 2 (2003).

8. Mendi, F., "Sanayi robotlarının mekanik üretimindeki rolü, performansları ve Türk sanayiinin robotlar yönünden bugünkü durumu", Doktora Tezi, ***Gazi Üniversitesi Fen Bilimleri Enstitüsü***, Ankara, 74-75, 80 (1992).

9. Rus D., "Teaching Robotics Everywhere", Massachusetts Institute of Technology, ***IEEE Robotics & Automation Magazine***, 15: 94 (2006).

10. Erden, A., "Robotik I" ***TUBİTAK Bilim Teknik Dergisi Yeni Ufuklara Eki***, Ankara, 8 (2003).

11. Eroğlu, M., "A Methodology for the Path Planning of Robot Manipulators Using AI Techniques", Degree of Doctor of Philosopy, ***University of Leeds,*** UK, 2 (1992).

12. Yıldırım, S., "Dinamik Sistemlerin Modellenmesi", ***Otomasyon Dergisi***, 135 (2003).

13. Sakagami, Y., Watanabe, R., Aoyama, C., Matsunaga, S., Hikagi, N., Fujimura K., "The Intelligent ASIMO: System overview and integration", ***IEEE/RSJ International Coneference on Intelligent Robots and Systems EPFL***, Switzerland, 1 (2002).

14. WowWee Ltd., “User's Manual of Robosaphien 8081”, WowWee Ltd., China,1-2 (2004)

15. Haralick, R. M., Shapiro, L. G., “Computer and Robot Vision v. I” ***Addison-Wesley Publishing Company, Inc***, USA, 1-5, 200-205 (1992).

16. Simon, J. C., “From Pixels to Features”, ***Elsevier Science Publishers B.V.,*** Netherlands, 1-5 (1989).

17. Schalkoff, R. J., "Digital Image Processing and Computer Vision", ***John Wiley & Sons, Inc***, Singapore, 4-6 (1989).

18. Graves, C., "Machine vision reaches top gear", ***IEE Review***, 44: 265 (1998).

19. Kuo-Yang Tu, "Design and implementation of a cheap middle size soccer robot with wide vision scope for RoboCup", ***Mechatronics, ICM '05, IEEE International Conference***, 131 (2005).

20. Chan, W. H., “CCD Camera Design and ASIC Applications”, ***ASIC Conference and Exhibit, Eighth Annual IEEE International***, 403 (1995).

21. Haralick, R. M., Shapiro, L. G., “Computer and Robot Vision v. II” ***Addison-Wesley Publishing Company, Inc***, USA, 26-29, 578-580, 597- 602 (1993).

22. Castleman, K. R., “Digital Image Processing”, ***Prentice- Hall, Inc.***, USA, 27, 35 (1996).

23. Pratt, W. K., “Digital Image Processing”, ***Second Edition, John Wiley & Sons, Inc***, Canada, 13, 21, 24-25 (1991).

24. Gonzales, R. C., Woods, R. E., “Digital Image Processing” ***Second Edition, International Edition, Prentice Hall***, USA, 3-7 (2002).

25. Erdem, İ. A., Erdem, M. E., Atalay, V., Çetin, A. E., “Bilgisayarlı Görmeye Dayalı Klavye”, ***Sinyal İşleme ve Uygulamaları Kurultayı***, Pamukkale, 1 (2002).

26. Lee, W. S., Slaughter, D. C. Giles, D. K., “Robotic Weed Control System for Tomatoes” ***Precision Agriculture, Kluwer Academic Publishers***, Netherland, 3, (1999).

27. Erdem, İ. A., Erdem, M. E., Atalay, V., Çetin, A. E., “Bilgisayarlı Görmeye Dayalı Fare”, ***Sinyal İşleme ve Uygulamaları Kurultayı***, Pamukkale, 2, (2002).

28. Ertuğrul, M., Kaynak, O., Akın, H. L, "Mayın Tarama Robotu," ***Savunma Sanayindeki Son Gelişmeler Sempozyumu***, 2 (1997).

29. Özcanlı, Ö. C., Şahin, P. D., Vural, F. T. Y., "Açıklamalı görüntü veritabanları kullanarak nesne tanıma ve erişimi", ***Bilgisayar Mühendisliği Bölümü Orta Doğu Teknik Üniversitesi***, Ankara

30. Uyanık, B., Özdemir, E., "Görüntü işlemeye dayalı robot kol hareket optimizasyonu***", IJCI Proceedings of International Conference on Signal Processing***, Vol.1, 16 (2003).

31. Kulkarni, A. D., "Computer Vision and Fuzzy-Neural Systems", ***Prentice Hall PTR***, New Jersey, 28-30, 48-54 (2001).

32. Forsyth, D. A., Ponce, J., "Computer Vision a Modern Approach", ***Prentice Hall***, New Jersey, 102-104, 110-113, 119-122 (2003).

33. Morris, T., "Computer vision and image processing", ***Palgrave Macmillan***, New York, 43-48 (2004).

34. Mettaa, G., Gasteratos, A., Sandini, G., "Learning to track colored objects with log-polar vision", ***Mechatronics,*** 14 (2004).

35. Bogdan, S., Lewis, F. L., Gürel, A., Kovacic, Z., "Timed Matrix-based Model of Flexible Manufacturing Systems", ***ISIE-99***, Bled, Slovenia, 4- 6 (1999).

36. Falkner, C. H., "Flexibility in Manufacturing Plants", ***Second ORSA/TIMS Conference on Flexible Manufacturing Systems***, Ann Arbor, MI, 13 (1986).

37. Maleki, R. A., "Flexible Manufacturing Systems" The Technology and Management, ***Prentice-Hall, Inc***., 122 (1991).

38. Nahmias, S., "Production and Operations Analysis", ***McGraw-Hill, Fourth Edition,*** Boston, 36-39 (2001).

39. Shunk, D. L., "Computer-Integrated Manufacturing", (in) Manufacturing High Technology Handbook, ***Ed. Tijunelis, D., K. E. McKee, Marcel Dekker, Inc***., New York, 42- 44 (1987).

40. Kochan, A. and Cowan, D., "Implementing CIM", ***IFS (Publications) Ltd***, Springer-Verlag, Berlin, 21- 24 (1986).

41. Stecke, K.E. "Design, Planning, Scheduling, And Control Problems Of Flexıble Manufacturing Systems", ***Graduate School of Business Administration, The University of Michigan, Ann Arbor***, Michigan, 5- 8 (1986).

42. Rhino Robotics, "Accessories for Robots and Work Cells Owners Manuel"***, FG2952***, USA, 1201- 1206 (2000).

43. Rhino Robotics, "Mark3 8 Axis Controller Owners Manuel"***, FG3771***, USA, A1-A12, D1-D3 (2000).

44. Rhino Robotics, "RoboTalk for Windows User's Manuel for Mark3 Controller"***, FG3760,*** USA, 12–14, 115, 609– 610, A31-A37 (2000).

45. Rhino Robotics, "XR-3, XR-4 and SCARA Owners Manuel"***, FG3056***, USA, 0201-0204 (2000).

46. Kovačić, Z., Ročak, N. S., Punčec, M., "Osnove Robotike", ***Upute za laboratorijske vježbe***, Zagreb, 5 (2004).

47. Sarıkaya, H., "İki serbestlik Dereceli Manipülatörlerin Ortak Manipülasyonu, Doktora Tezi, ***Erciyes Üniversitesi Fen Bilimleri Enstitüsü***, İzmir, 14 (1995).

48. Wang, F., Gao, Y., "Advenced Studies of Flexible Robotic Manipulators Modelling, Design, Contrrol and Applications Series in İntelligent Control and İntelligent Automation" ***World Scientific Publishing Co. Pte. Ltd.,*** Singapore, 4: 54 (2003).

49. Schilling, R. j., "Fundamentals of Robotics: Analysis and Control", ***Prentice Hall,*** New Jersey, USA, 3, 54, 56, 62-65, 67 (1990).

50. White, R.B., Read, R.K., Koch, M.W., Schilling, R.J., "A graphics simulator for a robotic arm Education", ***IEEE Transactions***, 32: 417, 418 (1989).

51. Zhuang, H., Roth, Z.S., Hamano, F., "A complete and parametrically continuous kinematic model for robot manipulators", ***Robotics and Automation, IEEE Transactions***, 8: 453 (1992).

52. Pradeep, A.K., Yoder, P.J., Mukundan, R., Schilling, R.J., "Crippled motion in robots", ***Aerospace and Electronic Systems IEEE Transactions***, 24: 2 (1988).

53. Negrean, I., Vuscan, I., Forgo, Z., "Inverse modelling of the kinematical errors of industrial robots", ***Intelligent Engineering Systems, IEEE International Conference***, 135 (1997).

54. Imam, I. N., Meredith, D.C., "Task level programming and graphical simulation of the Rhino XR3 robot", ***Energy and Information Technologies in the Southeast***, IEEE, 123 (1989).

55. Predko, M., "Robotic Experiments for the Evil Genius", ***Mc Graw Hill***, USA, 27, 33 (2003).

56. Kurfess, T. R., "Robotics and Automation Handbook", ***CRC Press***, USA, 9, 12 (2005).

57. Fouad, T. M., "Extending the Utility of the Rhino Educational Robot", ***IEEE Transactıons on Educatıon,*** Vol. 40, 184- 185 (1997).

58. Gluckman J., Nayar, S. K., "Planar Catadioptric Stereo: Geometry and Calibration", ***Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference***, 1: 23 (1999).

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, Adı	: TEKEREK, Mehmet
Uyruğu	: Türkiye Cumhuriyeti
Doğum Tarihi ve Yeri	: 10.10.1976 - Kahramanmaraş
Medeni hâli	: Bekâr
Telefon	: (90505) 383 79 88
E-posta	: mtekerek@yahoo.com

Eğitim

Derece	**Eğitim Birimi**	**Mezuniyet Tarihi**
Yüksek lisans	Gazi Üniversitesi	
	Endüstriyel Teknoloji Eğit.	2000
Lisans	Gazi Üniversitesi	
	Endüstriyel Teknoloji Eğit.	1996
Lise	K. Maraş Sütçü İmam Lisesi	1992

İş Deneyimi

Yıl	**Yer**	**Görev**
1996 –	T.C. Millî Eğitim Bakanlığı	Teknoloji ve Tasarım Öğretmeni

Yabancı Dil

İngilizce (ileri), Almanca (Temel)

Yayınlar

1. Tekerek, M., Gençay, S.**,** "Ergin İzcilikte Proje Uygulaması ve Yönetimi", Gazi Üniversitesi, I. Gençlik ve Rekreasyon Sempozyumu, Ankara- 2004

2. Gençay, S., Gençay, Ö. A., Marangoz, İ., Tekerek, M., "Üniversite Öğrencilerinin Rekreatif Aktivite Programlarına Katılımını Etkileyen Faktörler", Gazi Üniversitesi, I. Gençlik ve Rekreasyon Sempozyumu, Ankara- 2004

3. Tekerek, M., "Proje Tabanlı Mesleki Yönlendirme Teknoloji Eğitimi Modeli", T.C. MEB Orta Öğretimde Yeniden Yapılanma Sempozyumu, Ankara- 2005

Hobiler

Oryantiring, İzcilik, Klasik Türk Müziği