

İÇİNDEKİLER

	Sayfa
SİMGE LİSTESİ	iv
KISALTMA LİSTESİ.....	vi
ŞEKİL LİSTESİ.....	vii
ENDÜSTRİYEL OTOMASYONDA BİLGİSAYARLA GÖRME	viii
ÖZET	viii
COMPUTER VISION IN INDUSTRIAL AUTOMATION	ix
ABSTRACT	ix
TEŞEKKÜR	x
1. GİRİŞ	1
1.1 İnsanda Görme	4
1.2 Tezin Amacı	4
1.3 Konu ile İlgili Araştırmalar	5
2. İNSANDA GÖRME VE MAKİNE GÖRMESİ	7
2.1 İnsanda Görme	7
2.1.1 İnsan Gözünün Yapısı.....	7
2.1.2 Nasıl Görürüz?	8
2.1.3 Gözde görüntünün oluşumu	10
2.1.4 Nasıl Mesafeyi Tahmin Edebiliyoruz	12
2.2 Makine Görmesi	13
2.2.1 Bilgisayarla görme sisteminin bileşenleri	15
2.2.1.1 Kamera	15
2.2.1.2 Lensler.....	16
2.2.1.3 Aydınlatma	16
2.2.1.4 Nesne Sensörü	18
2.2.1.5 Görüntü Yakalayıcı.....	18
2.2.1.6 Görüntü tanıma (Makine Görme) yazılımı	18
2.2.1.7 Veri İletişimi	19
2.2.1.8 Bilgisayar	20
2.2.1.9 Işık	20
2.2.2 Sayısal Görüntü	21
2.2.3 Görüntü Algılama ve Elde Etme	26
2.2.3.1 Algılayıcı dizileri (sensor arrays) kullanarak görüntü elde etme	28
3. GÖRÜNTÜ OLUŞTURMA VE İŞLEME	29
3.1 Yöntemler.....	29

3.1.1	Sayısal Görüntünün İfade Edilmesi.....	29
3.1.2	Temel İkili Düzey (Binary) Görüntü İşlemleri	32
3.1.3	Genişleme (dilation) Operatörü.....	34
3.1.4	Aşınma (erosion) Operatörü.....	34
3.1.5	Kenar Belirleme.....	35
3.1.6	Geometrik Dönüşümler.....	37
3.1.7	Renk Sistemleri	38
3.1.7.1	RGB renk sistemi.....	39
3.1.7.2	CMY ve CMYK renk modelleri.....	40
3.1.8	HSI renk sistemi	41
3.1.9	Renkleri RGB'den HSI'ya çevirme.....	42
3.1.10	Renkleri HSI'den RGB'ye çevirme.....	42
4.	STEREO GÖRME.....	44
4.1	Mesafe Tahmini.....	44
4.1.1	Karakteristik (Feature) Noktaları Bulma	46
4.1.2	Harris Operatörü.....	46
4.1.3	Harris Operatörünün Özellikleri.....	49
4.2	Karakteristik Noktaların Eşleştirilmesi.....	49
4.2.1	Eşleştirmenin Özellikleri.....	51
5.	UYGULAMA DÜZENEGİ	54
5.1	Mekanik Aksamı	54
5.2	Elektrik Aksamı.....	54
5.2.1	Doğru Akım Servo Motoru	54
5.2.2	PLC.....	55
6.	DENEYSEL SONUÇLAR.....	58
7.	SONUÇLAR VE ÖNERİLER	62
	KAYNAKLAR.....	64
	EKLER.....	65
	Ek 1 Yazılım Kaynak Kodu	66
	ÖZGEÇMİŞ.....	72

SİMGE LİSTESİ

I	Aydınlatma şiddeti
I_c	Zemin aydınlatması
(x_p, y_p)	Görüntü düzleminde bir p noktasının koordinatı
(x_p, y_p, z_p)	Nesneye ait bir P noktasının gerçek koordinatı
λ	Dalga boyu
c	Işık hızı
h	Planck sabiti
ν	Frekans
e	Elektromanyetik spektrum enerjisi
f	Odak uzaklığı
f	İki boyutlu görüntü fonksiyonu
y	Aydınlatma fonksiyonu
yan	Yansıma fonksiyonu
(x, y)	Görüntü koordinatları
l	Gri seviye değeri
L_{min}	Minimum gri seviye değeri
L_{max}	Maksimum gri seviye değeri
$(M \times N)$	Görüntü matrisinin boyutları
A	Özdeş görüntü matrisi
i	Matris satır göstergesi
j	Matris sütun göstergesi
k	Renk sayının bit olarak ifadesi
B_s	Bit olarak resim veri boyutu
N_D	Diyagonal komşuluk
N_4	4'lü komşuluk
N_8	8'lü komşuluk
q	Görüntü üzerindeki herhangi bir nokta
D	dönüşüm işlemi
g	İki boyutlu işlenmiş görüntü fonksiyonu
s	Çıkış gri seviyesi
r	Giriş gri seviyesi
c_l	Logaritmik dönüşüm sabiti
n_k	Bir gri seviyedeki piksel sayısı
h	Histogram fonksiyonu
n	Toplam piksel sayısı
p_r	Giriş görüntüsü MYF
p_s	Çıkış görüntüsü MYF
ϑ	İntegral sabiti
R	Filtreleme işlemi merkez piksel değeri
$(m \times n)$	Filtre maskesinin boyutları
z_i	Resim alt bölümü pikseli
w_i	Filtre maskesi pikseli
σ	Standart sapma
∇	Gradient operatörü
G_x	Yatay yönde gradient vektör
G_y	Dikey yönde gradient vektör
G	Gradient vektörün genliği

E_T	Gradient vektörün esik değeri
α	Gradient vektörün açısı
ρ	Polar koordinatlarda genlik
θ	Polar koordinatlarda açı
a	Doğru denklemi eğimi
b	Doğru denklemi sehimi
T	Genel esik değeri
T_1	Alt esik değeri
T_2	Üst esik değeri
C_3	Yansıma yüzeyi sabiti
η	Normalize edilmiş görüntü
G_1	Birinci grup pikseller
G_2	İkinci grup pikseller
E	Yanlış sınıflandırma olasılığı
p_1	Nesne için MYF
p_2	Geri plan için MYF
$\bar{\mu}$	Ortalama gri seviye değeri
Prop	Bölge özelliği
Im	Resim bölgesi
A_k	Piksel alanı
B	Etiketlenmiş resim
C	Yuvarlaklık
P_φ	Çevre
(X, Y)	Kütle merkezi koordinatları
x	Kalıp vektörü
d_i	Karar fonksiyonu

KISALTMA LİSTESİ

DSP	Sayısal Sinyal İşleyici (Digital Signal Processor)
USB	Evrensel Seri Veri Yolu (Universal Serial Bus)
OPC	Proses kontrolü için Obje bağlantısı (Object-Linking and Embedding for process control)
SCADA	Veri toplama ve Denetimsel Kontrol (Supervisory Control And Data Acquisition)
MYF	Muhtemel yoğunluk fonksiyonu
RGB	Kırmızı, yeşil, mavi (red-green-blue) renk modeli
HSI	Renk özü, doygunluk, ışık şiddeti renk modeli
CMY	Camgöbeği, menekşe, sarı (cyan-magenta-yellow) renk modeli
CMYK	Camgöbeği, menekşe, sarı, siyah (cyan-magenta-yellow-black) renk modeli
D.A.	Doğru Akım
PLC	Programlanabilir Lojik Kontrolör
IP	İnternet Protokolü

ŞEKİL LİSTESİ

	Sayfa
Şekil 2-1 Gözün yapısı.....	8
Şekil 2-2 Mikroskopla çekilmiş koniler.....	9
Şekil 2-3 Görüntü açısına göre koni ve çubukların sayısı grafiği	10
Şekil 2-4 Uzaktaki ve yakındaki cisimlere odaklanma.....	11
Şekil 2-5 Bir nesne görüntüsünün gözde oluşumu. (C göz merceği merkezi).....	11
Şekil 2-6 Gördüğümüz görüntüler beyinde birleştirilerek tek bir görüntü haline getirilmesi..	12
Şekil 2-8 Göz yanılması, hangi adam daha büyük?.....	13
Şekil 2-9 Görüntü sistemleri ilişkisi ve uygulama alanları [2].....	14
Şekil 2-10 Pinhole kamera projeksiyon modeli.....	22
Şekil 2-12 Tüm elektromanyetik spektrum ve görünür spektrum [3]	24
Şekil 2-13 Elektromanyetik dalga formu ve dalga boyu.....	25
Şekil 2-14 Pikel sensoru ve sensor dizilimleri.....	27
Şekil 2-15 Algılama elemanı ile sayısal resmin elde edilmesi.....	28
Şekil 3-1 Sayısal resim koordinat yapısı.....	30
Şekil 3-2 Bazı ikili düzey işlemler ve sonuçları [Gonzalez R.C. 2002]	33
Şekil 3-3 Genişleme İşlemi örneği	34
Şekil 3-4 Aşınma ve Genişleme örneği [Gonzalez R.C. 2002].....	35
Şekil 3-6 Sayısal kenarın birinci ve ikinci türevi	36
Şekil 3-7 (a) lens yüzünden bozulmaya uğramış resim, (b) düzgün resim.	38
Şekil 3-8 Renk sistemi küpü.....	39
Şekil 3-9 (a) Katot Işın Tüplü (CRT) televizyon ve bilgisayar ekranı	40
Şekil 4-1 Stereo sistemin geometrik yapısı	44
Şekil 4-2 R nin aldığı değerler göre bölgenin sınıflandırılması	48
Şekil 4-3 Karakteristik noktaları bulunmuş sol ve sağ resimler.....	49
Şekil 4-4 Korelasyon penceresi ve arama penceresi.....	50
Şekil 4-5 Eşleşen noktaların gösterimleri	51
Şekil 4-6 3D Studio Max ile hazırlanmış projenin üstten görünüşü.....	52
Şekil 4-7 Sol ve Sağ resimler	52
Şekil 5-1 3B uygulama düzeneği çizimi	54
Şekil 5-2 ILC 150 ETH.....	56
Şekil 5-3 IB IL RS 232	57
Şekil 5-4 PC Worx Merdiven Diagramı	57
Şekil 6-1 Yeşil topun görüntüsü	58
Şekil 6-2 Yeşil topun histogram görüntüsü.....	59
Şekil 6-3 Mavi topun yazılımdaki görüntüsü.....	59
Şekil 6-4 Mavi topun histogram değerleri	60
Şekil 6-5 Megenta topun görüntüsü.....	60
Şekil 6-6 Megenta topun histogramı.....	61

ENDÜSTRİYEL OTOMASYONDA BİLGİSAYARLA GÖRME

ÖZET

Bu tezin amacı kamera ve bilgisayar ile çeşitli görüntü işleme tekniklerini kullanıp, endüstriyel otomasyonun ihtiyacını karşılayabilecek ekonomik, modern ve gerçek-zamanlı görüntü sistemi oluşturmaktır.

Çalışmanın ilk bölümünde hazırlanan yazılım ile iki kamera kullanarak alınmış görüntülerden, nesnelerin kameraya olan uzaklığını bulmak amaçlanmıştır. Bu amaç için köşe bulma, piksel karşılaştırma ve kayma miktarı hesaplaması algoritmaları kullanılmıştır. Çalışmanın ikinci bölümünde konveyör bant üzerindeki 6 farklı renkteki topların istenen renk sırası ile dışarı atılması amaçlanmıştır. Konveyör bandı üzerinde hareket eden toplara ait görüntüler sayısal kamera kullanılarak görüntü yakalama yazılımı ile bilgisayar ortamına aktarıldı. Elde edilen görüntülere iyileştirme, filtreleme, renk tanıma algoritmaları uygulandı. Algılanan renkler seri haberleşme ile PLC'ye bildirildi. PLC'nin G/Ç bağlı pistonları yardımıyla topların dışarı atılması sağlandı. Uygulamanın üçüncü bölümünde ise maket helikopter yardımı ile havadan çekilen görüntülerde aranan rengin görüntünün hangi bölgesinde oluştuğunu bulmaktır. Bu işlem için hazır bir görüntü işleme ürünü olan CMUCam2 kamerası kullanılmıştır.

Tez kapsamında yapılan araştırma ve uygulamalarla, otomasyon sistemlerinin görüntü sistemi ile çözülebilecek ihtiyaçları göz önünde tutulmuştur.

Anahtar Kelimeler: Stereo görme, görüntü işleme, köşe bulma, piksel karşılaştırma, renk tarama.

COMPUTER VISION IN INDUSTRIAL AUTOMATION

ABSTRACT

This study focuses on the visual needs of automation processes. In application, some type of image processing algorithm use with camera and computer to make economic, modern and real-time solutions.

In the first application, analyses of images captured via stereo cameras for detect the distance of objects. For this aim, we use corner detection, pixel comparison and disparity of cross-correlation algorithm. In the second application, we achieve to detect 6 different colors of ball on conveyer bands and throw outside in required order. Image of balls that moving over conveyer bands captured via digital camera and capturing software. Images analyses in computer with improvement, filtering and color detection algorithms. Sequence of the balls are send to PLC in serial communication and PLC's digital I/Os use for active pistons. In the third application, we take images from air model helicopter. Helicopter carries a CMUCam2 image processing product. The product can detect region of searched color.

During the research and developments of the thesis, the needs of the automation systems which can be solved with vision systems are taken into consideration.

Keywords: Stereo vision, image processing, corner detection, pixel comparison, color detection.

TEŐEKKÜR

Çalıőmalarım boyunca bilgi birikimiyle beni yönlendiren ve sınırsız bir sabır gösteren danıőmanım Yrd. Doç. Dr. Őeref Naci ENGİN'e ve Yrd. Doç. Dr. Songül ALBAYRAK'a, teknik bilgisiyle bana yardımcı olan arkadaşım Enver YAĐCI'ya bu sürede iő yerindeki sorumluluklarımın aksamaması için bana yardımcı olan mesai arkadaşlarıma ve müdürlerim Mustafa TANER ve Güngör BİNGÜL'e, her zaman maddi ve manevi olarak yanımda olan ailem Cansel GÜRCAN ve Seniha GÜRCAN'a teşekkürü bir borç bilirim.

1. GİRİŞ

Günümüzde yaşanan hızlı rekabet şartları bir ürünün satışında fiyat-performans ilişkisini, üretimde ise maliyet-kalite ilişkisini sürekli gündemde tutmaktadır. Diğer bir deyişle endüstriden istenen düşük maliyetle üretime karşılık yüksek ürün kalitesidir. Modern üretim endüstrilerinde üretilen her ürün başlangıcından son kullanıcıya ulaşana kadar oldukça çeşitli süreç ve kontrollerden geçirilir. Bu yollarla yaratılan iyi bir tasarım kalitesine paralel olarak test, ölçüm ve otomasyon teknolojileri üretim süreci boyunca vazgeçilemez uygulamalardır. Bu sürecin adına genel tanımıyla Endüstriyel Otomasyon denmektedir[1]. PC tabanlı test, ölçüm ve otomasyon sistemlerinin ayrı ayrı ya da birlikte bir tümleşik sistem olarak kullanılmasıyla üretimde verimlilik ve kalite yükseltilmekte; maliyet giderleri azaltılarak rekabet gücü arttırılmaktadır.

Bugün endüstride otomasyon tekniklerinin en popülerlerinden biri de bilgisayarla görme teknolojisidir.

1980'lerin başlarında ilk görme sistemleri uygulama alanı bulmaya başladığında bu gelişme yüksek kurulum maliyetleri ve esnek olmayan tasarımlarla gölgelenmişti. Bu işlemler yalnızca görüntü işleme konusunda uzman sistem mühendisleri tarafından ve sadece birkaç kısıtlı uygulama için mümkün olmaktaydı. Karmaşık matematik ağırlıklı ve iyi bir mühendislik bilgisi gerektiren bu tür sistemlerin kurulmaları hem uzun zaman almakta hem pahalı olmakta ve sonrasında sürekli bakım gerektirmeleri birer olumsuzluk olarak nitelenmekteydi. Yeni ürün tanımlamaları veya üretim hattı değişikliklerine uyum sağlamak için yapılan düzenlemeler ise çoğu zaman uzayıp giden üretim aksaklıklarına sebep olmaktaydı. Bugün bilgisayar teknolojisinde yaşanan son gelişmeler, bütün bu olumsuzlukların değişmesine neden olmuştur. Yüksek performanslı işlemcilerin her yıl yeni gelişmeler göstermesi, PC tabanlı makine görüş sistemlerinin gereksindiği işlem hızını sağlamıştır. Bilgisayarlarda ISA veri yolundan sonra PCI veri yolunun kullanılmaya başlanmasıyla kameradan alınan görüntülerin görüntü toplama kartlarıyla hızlı bir şekilde bilgisayar ortamına alınması sağlanmıştır. Bütün bu gelişmeler sonucu sistemin kurulması esnek, hızlı, düşük maliyetli, PC tabanlı, çok fonksiyonlu (endüstriyel ve bilimsel amaçlı) makine görüş sistemlerinin gerçekleştirilmesi olanaklı hale gelmiştir.

Ancak bir makine görüş sistemi her şeye rağmen bir insan beyni kadar mükemmel esneklik ve karmaşık fonksiyon kabiliyetine sahip olamayacağı unutulmamalıdır. Böyle bir sistemin performansı ölçülürken bu önemli bir kriter olarak göz önünde bulundurulmalı. Tasarlanan en

mükemmel sistem bile yalnızca programlandığı şekilde çalışabilir. Kurulacak sistemin istenen gereksinimleri karşılmasını garanti etmesi için doğru bir algoritma kurulması kesinlikle şarttır. Sistemin neleri görüp göremediğinin önceden belirlenmesi sistemin işleyişinde birçok hatayı önleyecektir. Bir projenin yapım aşamasında;

- Parça rengindeki oynamalar
- Çevre ışıklandırması
- fon rengi
- parçanın konumu ve yönelimi
- üretim hızı
- parçanın fiziksel detayları

gibi değişkenler göz önünde bulundurulması gereken en önemli faktörlerdir. Doğru kamera ve ışık kaynağı konumlaması, parçanın düzenli bir şekilde ve aynı konumda gelmesi, çevreden gelebilecek kontrolsüz ışığın engellenmesi, üretim hızının daima sabit olması, operatörlerin eğitimi, coğrafi ve ortam koşulları, çevrede azami dikkat pek çok olası hatayı engelleyecektir. Kontrol edilmesi istenen nesnenin rengini değiştirilmemesi, kamera ve aydınlatma ünitelerinin doğru bir yere ve titreşimlerden etkilenmeyecek şekilde monte edilmesini sağlanması ve çevreden yansıyan ışığın sisteme girişini bloke edecek düzenlemeleri gerçekleştirilmesi sağlanmalıdır.

Bilgisayarla görme, endüstride imalat süreci içerisindeki önemi büyümektedir. Kalite kontrol amacı ile üretilen ürünün kusurlarının algılanmasında, ürünün en, boy ve yüksekliğinin ölçülmesinde, pozisyon bulmada, robot kollarının yönlendirilmesinde kullanılmaktadır.

Önemli kullanım alanlarından biri ise sağlık sektörüdür. Mikroskopla alınan görüntüler, X-ışını görüntüleri, anjiyografiler, ultrasonik fotoğraflar, tomografiler görüntü işlemede kullanılan kaynaklardır. Bu kaynaklar bir takım görüntü işleme algoritmalarından geçirilip, aranan bilgi ayıklanır ve tümör, damar sertliği gibi hastalıkları taranabilir, organların ebatları, kan akışı ölçülebilir.

Askeriyede ise düşman askeri veya aracı tanımlama, füze yön gösterimi gibi daha birçok alanda kullanımı bulunmaktadır.

Yeni bir kullanım alanı ise bağımsız çalışabilen tekerlekli robotlar, arabalar, denizaltı araçları, uçan araçlar, insansız uçan araçlardır. Bu insansızda çalışabilen araçlar bilgisayarla görme sayesinde yönünü bulur, gittiği yerin haritasını çizebilir ve engelleri algılayabilir. Bunun yanında üretim amacına göre olayları algılayabilir (ör orman yangınlarını algılama) .

Diğer uygulamaları ise:

- Sinema veya TV yayınlarında görsel efektlerin hazırlanması
- Güvenlik sistemleri (CCTV)

Bir bilgisayarla görme sistemine ait bileşenler ve özellikleri aşağıdaki gibi sıralanabilir.

- Kamera: Uygulamaya uygun seçilmiş analog, dijital ya da LineScan Kamera
- Optik: Uygulamaya göre seçilmiş mercek
- Aydınlatma: Uygulamaya göre seçilmiş mercek ve yerleştirilmiş ışık kaynakları
- Nesne Sensörü: Malzemenin kontrolünü sağlamak için seçilen uygun sensör.
- Görüntü yakalama kartı: Uygulamaya ve öngörülen kameraya göre seçilmiş görüntü yakalama kartı
- İmge toplama yazılımı: uygulamaya özel yazılım platformu
- Veri iletimi: Konveyör hattının denetlenmesi, senkronizasyonu, pnömatik, motor kontrol vb
- PC Platformu: Amaca uygun endüstriyel PC ve koruyucu kabin
- Diğer: Genel mekanik yapı, ihtiyaca özel diğer istekler, diğer çevre koşulları ve istenmeyen detayların elimine edilmesi.

1980 li yıllarda birçok görüntü işleme firması, endüstride kullanılmak üzere çeşitli ürünler üretmeye başladılar. Bunlardan en önemli bir kaçı ve ürünleri aşağıda kısaca tanıtılmıştır.

Cognex Şirketi: 1981 yılında Amerika'da kurulmuştur. İlk ürünleri DataMan adında Endüstriyel Optik Karakter Tanıma (OCR) sistemidir. Bu ürün sayıları, harfleri, sembolleri tanıyabilen bir üründü. Dijital ve analog kameralar, aydınlatma sistemleri, imge toplama kartları, bilgisayarla görme sistemleri üretimine devam ettikleri ürünlerdir. Pazarında lider firma pek çok patentin de sahibidir. En çok ilgi gören ürünü ise ilk defa 1994 yılında üretilen *Checkpoint* adındaki kullanımı kolay programlanabilir Bilgisayarla Görme sistemidir.

Keyence Şirketi: Japonya'da kurulmuştur. Bilgisayarla Görme Sistemleri üretiminin dışında çok çeşitli sensor üretimi de mevcuttur. En son üretimini başlattığı görme sistemi CV-5001 adındaki 4 adet 2MP renkli kamera bağlanabilen dakikada 20,000 kare görüntü işlemesi yapabilen bir sistemdir.

MVTec: 1996 da Amerika da kurulan firma, şu anda Bilgisayarlı Görme Sistemleri için yazılım araçları geliştirmektedir. Halcon ve ActivVisionTool adında iki ürünü bulunmaktadır. Halcon daha gelişmiş uygulamalar için uzman yazılımcılara yönelik esnek bir yazılım aracıdır. ActivVision ise daha az esnek bunun yanında kullanımı çok kolay olan bir yazılım

aracıdır. Hazır araçların içinde mesafe ölçümü, görüntü bölgesi analizi, OCR, barkod tanımlama gibi uygulamalar vardır.

Intel: CPU üretiminde lider Intel firmasının, Intel işlemciler için ürettiği ücretsiz, açık kaynak kodlu, C/C++ dilinde yazılmış yazılım aracı olan OpenCV dir. 1999 yılında başlanmıştır ve geliştirilmeye devam etmektedir. Bu tezin hazırlanmasında da pek noktasında bu aracın hazır fonksiyonlarında yararlanılmıştır.

1.1 İnsanda Görme

Görme duyumuzun dış dünyayı algılayabilmemiz için en işlevli duyu olduğu açıktır. Zihnimize çoğu bilgiyi, kişiyi eşyayı görüntüsüyle hafızamızda tutarız. Benzetme, alakalandırma, çağrışım yaptırma yeteneği en çok görsellikte vardır. Ve olguların görselliği en çok tasvir edebileceğimiz özelliktir. Cisimlerden yansıyan ışıklar gözün önünde ki saydam tabaka (kornea) ve içindeki mercek (lens) tarafından kırılarak gözün en arka tabakasında yer alan ve sinir liflerinden oluşan tabakada odaklanır ve beyne aktarılır.

1.2 Tezin Amacı

Bu tezin amacı endüstriyel uygulamalarda bilgisayar bağlantılı kameralar sayesinde ekonomik otomasyon çözümlerinin yapılabileceğini göstermektir. İlk uygulamada kullanılan konveyör bant (taşıyıcı dönen tabla) üzerine konan cisimler renklerine göre algılanmış ve bir PLC yardımıyla motor ve pistonlar kumanda edilmiştir. İkinci uygulama da ise 50cm x 50cm boyutundaki bir insansız hava aracı (kuadrokopter - dört pervaneli helikopter) ile belirli bir renk aralığını aratarak helikopterin o noktaya gitmesini sağlamaktır. İki uygulamada da işlemler gerçek zamanlı olarak yapılmıştır.

Bu çalışma kapsamında, kameralardan alınan görüntüler PC ortamına yazılım aracıyla aktarılmıştır. Daha önceden rengi belirlenen cisimlerin görüntü üzerindeki konumu bulunmuştur. Her iki resimde de köşe algılama algoritmasıyla cismin köşe noktası bulunup, iki resim için bu noktalar etrafında piksel karşılaştırması benzerlikleri taranır. Benzerlikleri onaylanan noktalar arasında iki resim için kayma miktarları hesaplanır. Kayma miktarı ise bize cisme olan uzaklık hakkında bilgi verir. İnsansız hava aracı uygulamasında ise rengi belirli bir cismin görüntü üzerindeki konumu aranır. Bu sayede hava aracının nereye gitmesi gerektiği hesaplanır.

İlk bölümde konu genel olarak tanımlanıp, konu ile ilgili çalışmaların özeti verilmiştir.

Bu tez kapsamındaki projeler bilgisayar ortamında çalıştırılmıştır. Halen yapılan çalışmalarla görüntü işleme ürünleri, bilgisayardan bağımsız hale getirilip, Sayısal Sinyal İşleyiciler (DSP) ile yapılmaya çalışılmaktadır. Bu sayede ürünlerin daha küçük boyutlu, daha ekonomik bir hal alması amaçlanmaktadır. Günümüzde üretilen Sayısal Sinyal İşleyici tabanlı ürünler yeterince esnek bir yazılıma izin vermemekte, bir bilgisayar kadar hızlı çalışmamaktadır. Bu tip zorlulardan dolayı Sayısal Sinyal İşleyici tercih edilmemiştir. Bilgisayar tabanlı bir sistem kullanarak amacımız güvenli, esnek, kullanıcı kolaylığı olan bir yazılım geliştirerek gerçek zamanlı bir uygulama örneği göstermektir.

Tez kapsamında hazırlanan yazılımların çeşitli haberleşme standartlarına uygun hale getirilmesi sayesinde endüstride kullanılan diğer ağlarla haberleşmesi sağlanabilir. Örneğin *Proses kontrolü için Obje bağlantısı* (OPC) kullanılması sayesinde *Veri toplama ve Denetimsel Kontrol* (SCADA) ile haberleşmesi sağlanır. SCADA sayesinde gerekli denetimler anında masa başından uygulanabilir, ihtiyaç duyulan veriler canlı olarak izlenebilir veya bir veritabanında saklanarak yıllar öncesine bile geri dönülebilir.

1.3 Konu ile İlgili Araştırmalar

Trakya Üniversitesi'nden Filiz Soykan tarafından hazırlanan Yüksek Lisans tezinde hazırlanan bir yazılım ile temel geometrik şekiller üzerinde sınıflama yapan bir uygulama geliştirilmiştir. [Filiz Soykan, 2005]. Tezinde görüntü tanımından itibaren ele alınmış, işleme aşamaları, sınıflama yapmada ve akıllı görme sistemlerinde kullanılan nesne tanıma sistemleri incelenmiştir. Nesnelerin farklı özelliklerine tanıma yapan sistemlerden örnekler verilmiş, nesneler arası topolojik ilişkiler ve bunların mantıksal ve şematik olarak ifadeleri incelenmiştir.

Hikmet Bal, Yüksek Lisans tezinde birçok görüntü ile ilgili teorik bilgileri incelemiş. Ayrıca malzeme görüntülerinden, hazırlanan yazılım ile malzeme içerisindeki küresel grafit alan yüzdesi veya istenen fazın piksel alan yüzdesi hesaplatmıştır. Bu işlemler için görüntü iyileştirme, filtreleme, yapısal düzenleme, bölümlendirme görüntü işleme teknikleri uygulamıştır. Tezinde ele aldığı diğer bir yazılım uygulaması ile değişik doğrultularda yerleştirilmiş domino taşlarına ait görüntülerden her bir taşın değeri bilgisayar ortamında belirlenmiştir.[Hikmet Bal, 2006].

Volkan Verimin “Görüntü işleme yöntemleri ile doku sınırlarının belirlenmesi” tezinde iki boyutlu ultrason görüntülerinin analiz edilerek görüntülerdeki prostat sınırının bulunması için yeni bir yaklaşım sunmaktadır. Bu çalışmada, belirlenecek prostat sınırı için prostatlı bölgenin

prostatın dışına göre daha koyu kontrasta sahip olmasından yararlanmıştır. İki boyutlu tıbbi ultrason görüntüsündeki gürültü, Gauss alçak geçiren süzgeci ile azaltılmış, gri seviyeli görüntü eşikleme ile ikili görüntüye dönüştürülmüş, elde edilen görüntüye morfolojik kapama ve açma işlemleri uygulanmıştır. Ön işlemlerden sonra, kabaca bulunan dikey ve yatay kenarlar, uyarlanır bazı işlemlerle birleştirilerek prostat sınırı Matlab 'de hazırlanan bir yazılım sayesinde incelenmiştir [Volkan Verim, 2006].

2. İNSANDA GÖRME VE MAKİNE GÖRMESİ

2.1 İnsanda Görme

2.1.1 İnsan Gözünün Yapısı

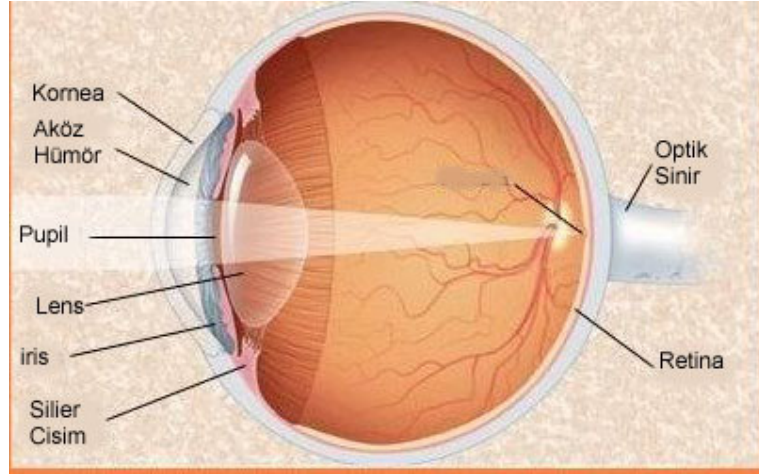
Çevremizi algılayabilmek; çevreyle ilişki kurabilmek için biz canlılara verilmiş çok önemli bir yetenektir. Gelişmiş canlılarda algı için duyu organları vardır ve hepsinin ayrı ayrı önemi bulunmaktadır. Görme duyumuzun dış dünyayı algılayabilmemiz için en işlevli duyu olduğu açıktır. Zihnimizde çoğu bilgiyi, kişiyi eşyayı görüntüsüyle hafızamızda tutarız. Benzetme, alakalandırma, çağrışım yaptırma yeteneği en çok görsellikte vardır. Ve olguların görselliği en çok tasvir edebileceğimiz özelliktir. Bu nedenle eski çağlarda insanların bilgilerini aktarabilmesi için gördüklerini mağaralara çizmişlerdir. Çünkü kişiden kişiye değişmeyen, zamandan zamana değişmeyen tek şey görselliktir.

Göz, ortalama çapı 20mm olan bir küre seklindedir ve "orbita" adı verilen göz çukurunun içinde bulunmaktadır. Ön yarısı yağ ve bağ dokusuyla çevrilidir. Ön kısmı açıkta bulunduğu için orbita kemik kenarları ve göz kapakları tarafından korunmaktadır. Ortalama ağırlığı 7,5 gr, ortalama hacmi 6,5 cc kadardır. 4 rektus göz hareketlerinden sorumludur ayrıca 2 oblik kas ile orbita tepesine tutunmuştur.

Göz küresi üç temel kattan oluşmuştur. Dışta ki katman destek katıdır. Şeffaf doku kornea, onun çevresinde yer alan opak yapı sklera, sklera üzerini ve göz kapaklarının iç yüzeyini örten zar da konjonktiva olarak adlandırılır. Orta katman damarsal tabakadır ve uvea adını alır. Koroidea, silyer cisim ve iristen ibarettir. İrisin ortasındaki açıklığa pupilla adı verilir. Son olarak iç katman sinir katmanıdır. Bu katmanda retina tabakası bulunmaktadır.

Gözün yapısı Şekil 2-1 gösterilmektedir. Ön yüzeyi saran saydam doku korneadır. Bu optik küremizin kalanını kuşatan saydam olmayan kısım ise göz akıdır. Göz akının altında koroid bulunmaktadır. Gözün beslenmesi için kan gerekmektedir. Kan damarları ağı bu zarda bulunmaktadır. Bu açıdan çok önemlidir. Önemsiz gibi gözükken yüzeysel hasar bile ciddi göz hasarlarına yol açabilir. Kan akışını engelleyen iltihaplanmalar yüzünden göz beslenmeyerek körlüğe neden olabilir. Koroidin tabakası çok pigmentlidir. Pigment; dışarıdan göze gelen ve optik kürede yayılan ışığın miktarını azaltmaya yardımcı olur. Koroid en uç noktada kirpiksi bölüm ve iris diyaframı olarak ikiye ayrılır. Göz büzülme ve genişleme ile gelen ışığın miktarını kontrol edip dengeler. İrisin merkez açıklığı (göz bebeği) yaklaşık 2 ile 8 mm arasında değişir. İrisin arka tarafı siyah pigment içermektedir. İrisin ön tarafı gözün görünen

pigmentini içererek göz rengini belirler.



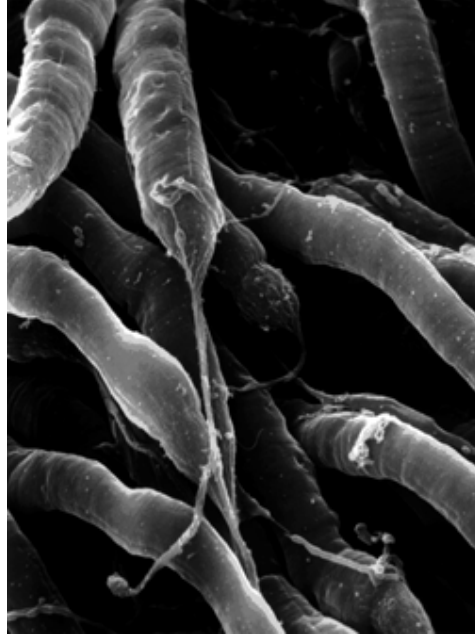
Şekil 2-1 Gözün yapısı

Eşmerkezli katmanlar halinde sıralı lifli hücreler göz merceğini oluşturmaktadır. Uç noktasında kirpiksi kısma bağlanan lifler tarafından geçici olarak kapatılır. Göz merceği yoğunluklu olarak su içermektedir (%60–70). Ve gözde ki diğer dokulardan daha fazla protein içermektedir, yaklaşık %6 oranında yağ da bulunmaktadır. Göz merceği daha kısa dalga boyunda daha yüksek emilim oranına nispeten yaklaşık olarak görünebilir ışık spektrumunun %8'ini emer. Hem kızılötesi hem de mor ötesi ışıklar mercek tarafından proteinler sayesinde emilir. Fazla miktardaki kızılötesi ve mor ötesi ışıklar göze zarar vermektedir.

2.1.2 Nasıl Görürüz?

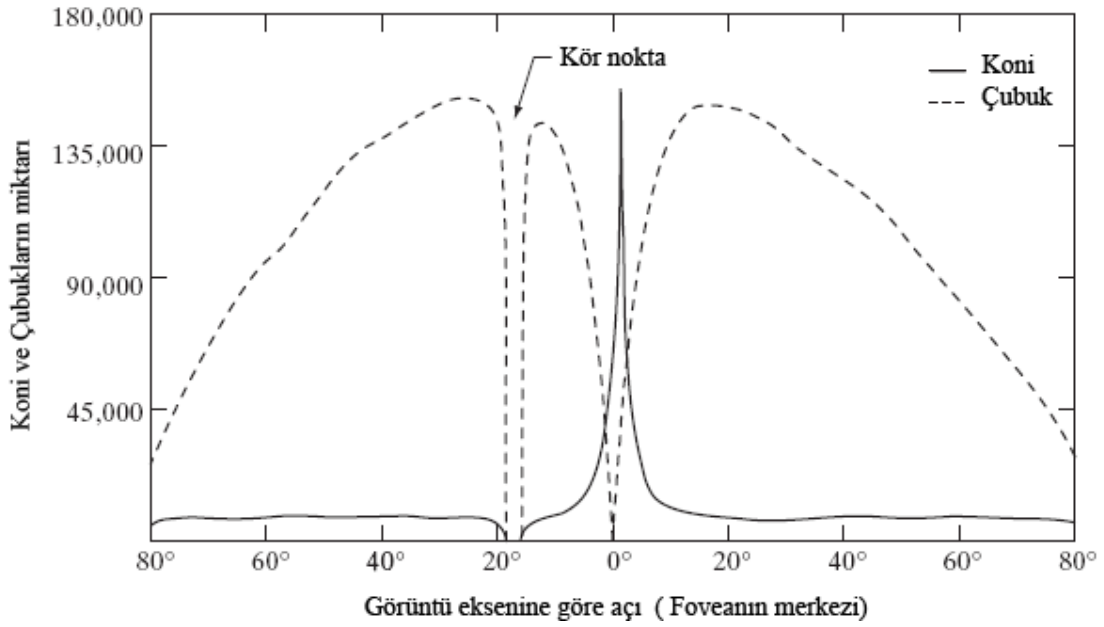
Çevremizde ki her cisim ışık yansıtır. Cisimlerden yansıyan ışıklar gözün önünde ki saydam tabaka (kornea) ve içindeki mercek (lens) tarafından kırılarak gözün en arka tabakasında yer alan ve sinir liflerinden oluşan "retina" tabakası üzerinde odaklanır. Gözün en içteki zar duvarının gerisinde ki tüm kısım retinadır. Cisimlerden yansıyan ışığı göz uygun şekilde odaklanarak toplar ve retina üzerinde görünür. Model görüşü ayrı ışık alıcılarının retina yüzeyine dağılması ile yapılır. İki tür alıcı vardır; **koni** ve **çubuk**. Her bir gözdeki konilerin sayısı 6–7 milyon arasındadır. Retinanın öncelikle fovea denen merkez kısmında yerleşmişlerdir. Renge duyarlı alıcıdır. Renk ince ayrıntıların belirlenmesine neden olduğundan bu koniler sayesinde şeklin ince ayrıntıları algılanabilir. İnce ayrıntıları her bir koninin kendi sinir ucuna bağlı olmasından dolayı algılayabiliriz. Görüntü foveaya düşene kadar göz küresini gözü kontrol eden kaslar sayesinde döndürebiliriz. Şekil 2-2 deki mikroskopla görüntüsü çekilmiş koniler alıcısı tamimiyle renk alıcısı gibi kabul

edilebileceğinden fotopik ve ya parlak-ışık görüşünü sağlar.



Şekil 2-2 Mikroskopla çekilmiş koniler

Çubukların sayısı daha fazladır: Yaklaşık olarak 70–150 milyon. Retina yüzeyi boyunca dağılmıştır. Görülebilen ayrıntı miktarı azdır çünkü bu çubuk alıcıların dağılımının geniş bölgede olması ve bazı çubukların tek sinire bağlı olması renklerinde ayırt edilememesine neden olur. Renk görüşü ile ilgili görevi yoktur bu nedenle çubuklar, görüş alanının genel, tüm bir resmini verme işini yaparlar. Fakat düşük seviyede ki aydınlanmalara karşı duyarlıdır. Örneğin; gün ışığında parlak renkli gözükken nesnelere ay ışığında renksiz gözükürler. Çünkü sadece çubuk alıcılar uyarılmıştır. Bu durum skotopik veya loş ışık görüşü olarak da bilinir.



Şekil 2-3 Görüntü açısına göre koni ve çubukların sayısı grafiği

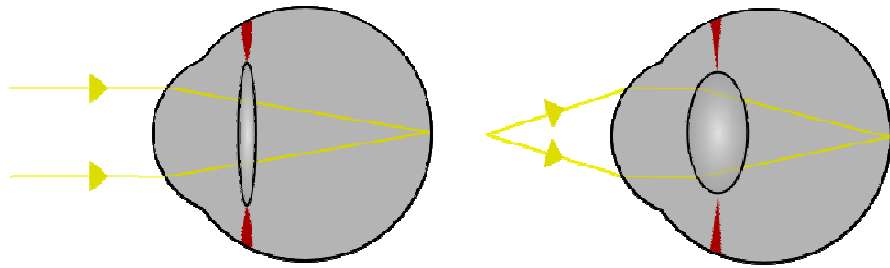
Şekil 2-3 optik sinirin çıkış alanı boyunca geçen çubukların ve konilerin yoğunluğunu göstermektedir. Bu bölgede alıcıların yokluğu kör nokta denen kısımları meydana getirir. Kör noktanın dışında, alıcıların dağılımı foveanın etrafında radyan olarak simetriktir. Alıcı yoğunluğu, foveadan derece cinsinden ölçülür (bu, eksenler arasındaki derece, göz merceğinin merkezinden geçen ve retinayı kesen bir çizgi ile görüş eksenini arasındaki açının ölçülmesidir.) Şekil 2-3'den konilerin retinanın merkezinde (foveanın merkez bölgesinde) daha yoğun oldukları görülmektedir. Ayrıca çubuk alıcıların merkezden yaklaşık 20° dışarıda yoğunluklarının arttığı ve retinanın sınır çevresinde yoğunluklarının azaldığı görülmektedir.

İnsanların, net olarak görebildiği en yakın mesafe Yakın Nokta olarak adlandırılır. Yaş ilerledikçe yakın nokta uzaklaşır. 10 yaşında 6 santimetreden net görüntü alınabilirken, 30 yaşında bu yakın nokta 15 santimetre ve 60 yaşında 80 santimetre olmaktadır.

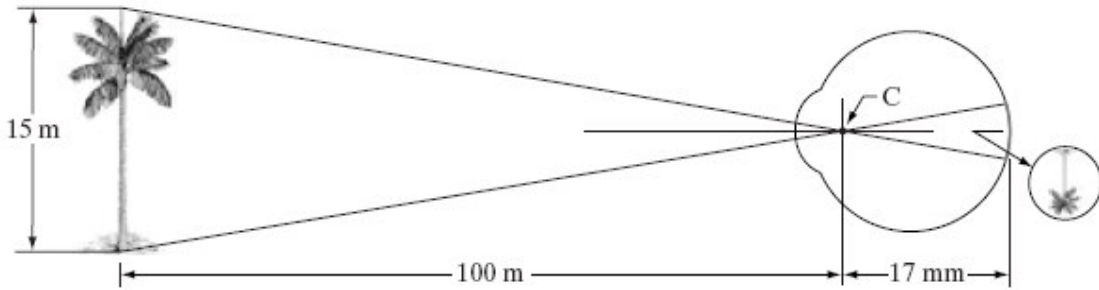
2.1.3 Gözde görüntünün oluşumu

Şekil 2-4'de göz merceğinin ön yüzünde ki kavis yarıçapının gerideki yarıçaptan daha büyük olduğu görülmektedir. Göz merceğinin yapısı, kirpiksi bölgede ki liflerin gerilimi ile kontrol edilir. Uzakta ki nesneye bakarken gözlerimizde sorun varsa eğer biraz kısma ihtiyacı duyarız gözlerimizi. Çünkü uzak nesnelere odaklanmak için, kontrol kasları göz merceğinin yassılaşmasına neden olur. Gözün yakınında ki nesnelere odaklanmak istediğimizde bu kaslar göz merceğinin kalınlaşmasına neden olur. Göz merceği ile sıradan bir optik mercek

arasındaki en temel fark göz merceğinin esnek olmasıdır. Bu esneklik kirpiksi bölgede ki lifler ve kaslarla sağlanır. Biz merceğin kalınlaşması ve yassılaşmasıyla uzak ve yakın mesafeye odaklanabilirken optik merceklerin yakın nesnelere ve uzak nesnelere için ayrılmasının nedeni de budur. Göz merceğinin merkezi ile retina arasındaki mesafe (odaksal uzaklık denir) göz merceğinin kırılma kapasitesi en küçükten en büyüğe artarken yaklaşık olarak 17mm ile 14mm arasında değişir. Göz, yaklaşık 3m'den daha uzak bir nesneye odaklanırken, göz merceği en düşük kırılma kuvvetini gösterir. Göz, yakınındaki bir nesneye odaklanırken, göz merceği en kuvvetli kırılmasını sergiler. Bu bilgi ile bir nesnenin retinal görüntüsünün büyüklüğünü hesaplayabiliriz..



Şekil 2-4 Uzaktaki ve yakındaki cisimlere odaklanma

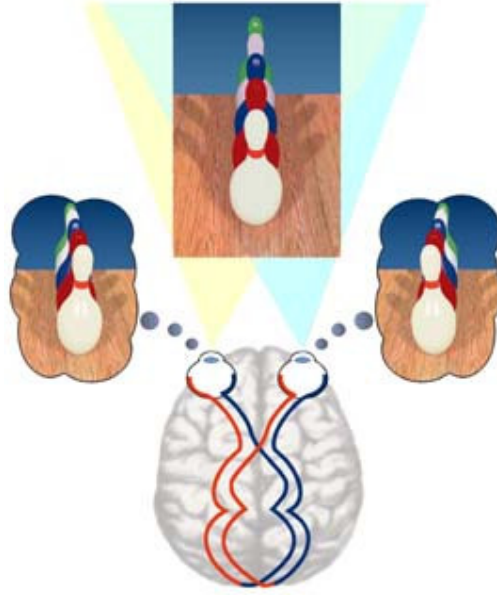


Şekil 2-5 Bir nesne görüntüsünün gözde oluşumu. (C göz merceği merkezi)

Şekil 2-5'te örneğin, gözlemci 15 m yüksekliğindeki ve 100 m uzaklıktaki bir ağaca baktığında eğer nesnenin retinal görüntüsü mm cinsinden h yüksekliği ile ifade edilirse, Şekil 2-5'ün geometrisi, $15/100=h/17$ ve $h=2,55$ mm vermektedir. Retinal görüntü öncelikle fovea alanına yansır. Daha sonra algılama, ışın enerjisini beyin tarafından çözülen elektrik darbe sinyallerini çeviren ışık algılayıcıların bağlı uyarımı ile meydana gelir.

2.1.4 Nasıl Mesafeyi Tahmin Edebiliyoruz

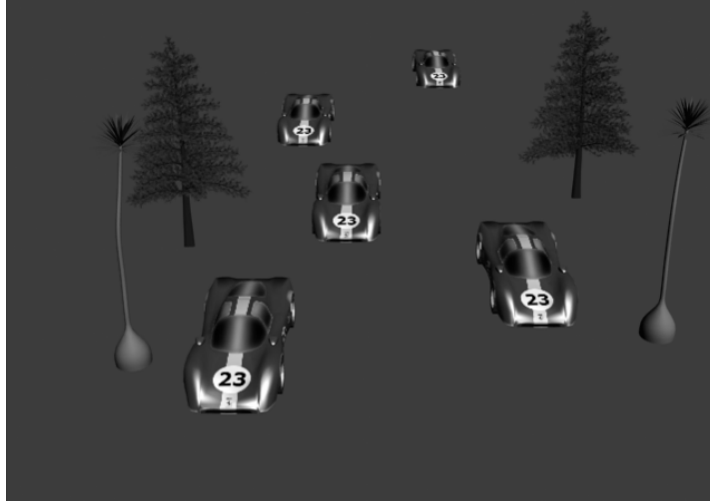
Normal bir insanın iki gözünün olmasının en önemli özelliği, cisimlerin mesafesini algılayabilmesidir. Şekil 2-6'da gösterildiği gibi iki gözümüzle gördüğümüz görüntüler beyinde birleştirilerek tek bir görüntü haline getirilir.



Şekil 2-6 Gördüğümüz görüntüler beyinde birleştirilerek tek bir görüntü haline getirilmesi

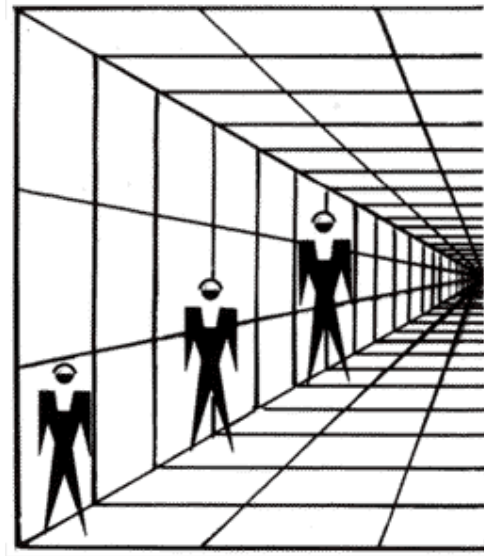
Görüntüler üzerinde ki bir nesnenin mesafesini algılamak için beyin o cismin iki görüntüde ki kayma miktarına bakar. Kayma miktarı cismin uzaklığını algılayabilmek için bir oran verir. Bu oran ile cisimler arası mesafe farkını bulabiliriz. Beyin bu hesabı bir bilgisayara göre kat kat hızlı bir şekilde yapabilmektedir.

Tek bir resimden mesafe tahmini yapmak için daha önceki deneyimlerimizi kullanarak bir tahminde bulunabiliriz. Örneğin Şekil 2-7'e bakıldığında bir ağacın arabaya göre daha büyük olacağını, arabaların boyutunun aynı olduğunu, uzakta olanın daha küçük görünmesinin nedenin en arkada olduğunu önceki deneyimlerimizden hesaplayabiliyoruz.



Şekil 2-7 Arabaların bir birine olan mesafelerini tahmin edebiliriz.

Mesafe tahmininin de, her zaman deneyimlerimiz bize yardımcı olamayabilir. Şekil 2-8 bir göz aldanması örneğidir. “Hangi adam daha yakında?” sorusunun kesin bir cevabı yoktur. Çünkü tek bir resim burada mesafeyi algılamamız için yeterli değildir. Karşılaştırdığımız nesnelerin aynı olması bu işi zorlaştırır.



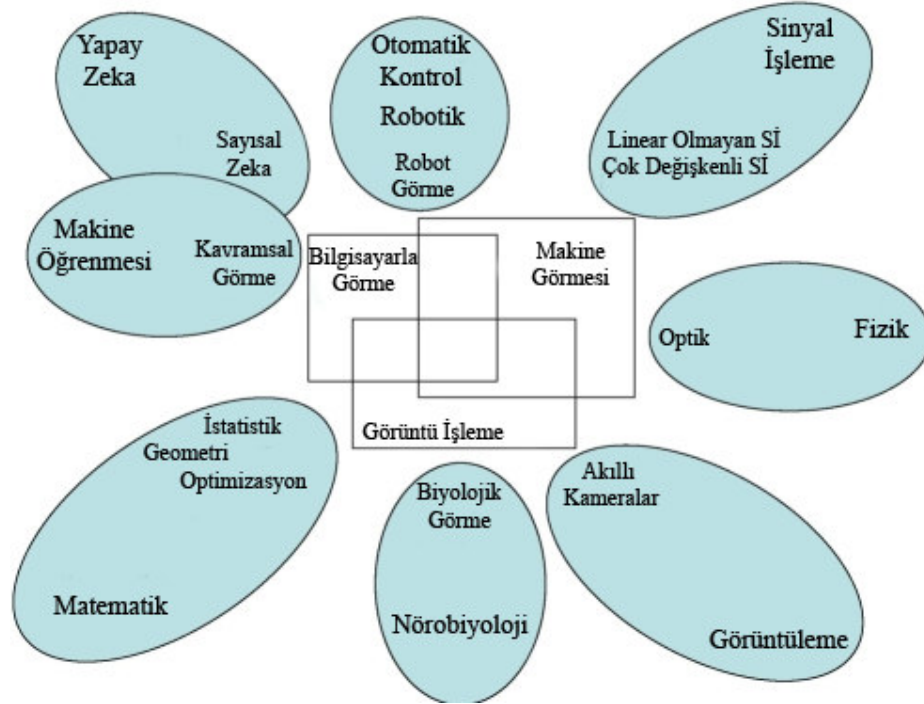
Şekil 2-8 Göz yanıltması, hangi adam daha büyük?

2.2 Makine Görmesi

Görüntü İşleme, video karesi veya bir fotoğraf imgesinin, belirli işlemlerden geçirilip, çıkışta

yine bir imge veya bir imge özellikleri kümesi elde edilmesidir. Birçok görüntü işleme fonksiyonu, sinyal işleme tekniklerinin 2 boyuta çevrilmiş halidir.

Makine Görüntü Sistemleri, Bilgisayarla Görme sistemlerinin endüstri ve imalat alanlarındaki uygulamasıdır. Bilgisayar görüşü, yapay zekâ bir sistemin davranışlarını kontrol etmek üzere resim bilgisinin metin temelli girdilere alternatif olarak bir sisteme beslendiği bir alan olan yapay zekânın bir alt alanıdır. Bilgisayar görüşünde kullanılan bazı öğrenme yöntemleri yapay zekâ alanında geliştirilen öğrenme tekniklerine dayanır.



Şekil 2-9 Görüntü sistemleri ilişkisi ve uygulama alanları [2]

Bilgisayarla görme, teknolojik ve bilimsel olarak makinelerin görebilmesidir. Kullanılan imgeler, video parçası, birden çok kameradan alınmış görüntü, ya da 3B lu tıbbi tarayıcılardan alınmış görüntüler olabilir.

Bilgisayarla görme sistemleri, bilgisayarla görme teorileri ve modellemeleri kullanılarak yapılandırılır. Bilgisayarla görme sistemleri şu sistemleri içermektedir

- 1) Kontrol işlemleri (ör. Endüstriyel robot veya şoförsüz araçlar)
- 2) Algılama olayları (ör. Görsel gözetim)
- 3) Bilgilerin düzenlenmesi (ör. İmge veritabanlarının sıralanması)
- 4) Çevre ve Obje Modellemesi (ör. Endüstriyel denetim, tıbbi imge analizi)

Bilgisayarla görme uygulamaları olarak sahne tekrar yapılandırılması (scene reconstruction), olay algılama, izleme, cisim tanıma, öğrenme, sıralama ve imge onarımı örnek olarak verilebilir.

Dijital görüntü işleme 1960'larda Jet Propulsion Laboratory, MIT, Bell Labs, University of Maryland ve diğer birkaç merkezde, uydu fotoğrafları, tıbbi fotoğraflar, görüntülü haberleşme, karakter tanımlama ve resim düzeltmesi amacı ile geliştirilmeler yapılmaya başlandı. Fakat o zamanın donanım ve işlemcileri ile işlem yapmak çok büyük maliyetler gerektiriyordu. 1970'lerde ise, daha ucuz bilgisayar ve bu işe adanmış donanımlar ile görüntü işleme hızla artmaya başladı. Artık görüntü işleme gerçek zamanlı yapılmaya başlanmıştı ve TV sektöründe kullanılıyordu.

2000 li yıllara gelindiğinde ise hızlı bilgisayarların gelişimi ile dijital görüntü işleme, görüntü işlemenin en çok kullanılan ve en az maliyetli yöntemi haline geldi.

2.2.1 Bilgisayarla görme sisteminin bileşenleri

Bilgisayarla görme sistemi birçok parçanın birleşmesinden oluşmaktadır.

2.2.1.1 Kamera

Bu kategori genel olarak incelenen parçanın resmini alan kamera ve lenslerden oluşur. Kamera ve lens seçimi uygulama gereksinimleriyle sıkı bir şekilde ilişkilidir.

Uygulamaya bağlı olarak kullanılan kameralar standart tekrenkli (monochrome) (RS-170), birleşik (composite) renkli (Y/C), RGB veya çizgi tarama (LineScan) olabilirler. Örneğin bir monokrom kamera genel uygulamalar için kullanılırken renkli kontrol işlemlerinde zorunlu olarak bir renkli kamera gerekecektir. Bunun yanında gerekli bilgiyi alabilmek için çözünürlük yeterince yüksek olmalıdır.

Tetikleme ve entegrasyon kontrol özellikleri de gerekli olabilir. Son olarak, kameraların kir, toz, ısı gibi olumsuz ortam şartlarına dayanabilmeleri için yeteri kadar kaliteli ve dayanıklı olmaları ya da en azından bu konuda özel üretilmiş IP standartlarına sahip kamera kılıfları kullanılmalıdır. Bu noktada endüstri standardı kameralar seçilmesi en uygun yoldur.

Kontrol sistemlerinde tipik olarak çizgi tarama (LineScan) ve dizi (array) olarak iki çeşit kamera kullanılır. Konvansiyonel bir CCD dizi (array) kamera dikdörtgen veya kare bir resmi tek defada alır. Çizgi tarama (LineScan) kamera ise doğrusal dizi halinde tasarlanmış algılayıcılara sahiptir ve imgeyi bir satır olarak tarar.

2.2.1.2 Lensler

Yazılım ve donanım özellikleriyle veya sadece kameraların özellikleriyle görüntü kalitesi ancak bir yere kadar iyileştirilebilir. Fakat bu noktada uygun lens seçilmesi büyük önem taşır

Günümüzde kullanılan çoğu lensler güvenlik amaçlıdır, Bunlar ancak endüstrinin kısıtlı uygulamalarına cevap verebilirler. Standard lenslerde boşlukların hesabı yapıp sabitlenerek odaklamaları yapılır. Kısa mesafede odaklanabildikleri zaman üretim hattı boyunca doğru ve esnek çalışmayı sağlarlar. Bazı firmalar sanayi uygulamalarında kullanılmak üzere özel tasarlanılan Çift Gauss Makro görüntüleme lensleri üretmektedirler. Ayrıca uygulamanın şekline göre kalitenin korunmasına yönelik özel filtre mercekler kullanılması gerekebilir.

2.2.1.3 Aydınlatma

Kontrol amaçlı bir görüntü sistemi tasarlariken genellikle kameranın özellikleri üzerinde yoğunlaşılır. Ancak incelenmek istenen nesne yeterli ve dikkatli şekilde aydınlatılamazsa alınan görüntü ile amaçlanan doğru sonuca ulaşılamaz.

Bu hayati gereksinimlere genellikle gereken özen çoğu kez gösterilmez. Işık her yerde ışık demek değildir. Kötü ışıklandırma ya da yanlış optik elemanlar kullanıldığında en iyi şekilde tasarlanmış bir görüntü sistemi dahi kapasitesinin çok altında bir performans gösterecektir.

Nesnenin gerektiği şekilde aydınlatılması mümkün olan en iyi görüntünün yakalanabilmesi için ön koşuldur. İncelenmek istenen nesnenin geometrisi aydınlatma tercihlerinin yönlendirilmesindeki en önemli faktördür. Görüntü işleme için kullanılması tasarlanan kamera tipi ışık kaynağının da nasıl düzenleneceğini belirler.

Bir aydınlatma sistemi kameranın görüş alanındaki birimlere mümkün olduğunca düzenli ışık vermelidir. Aynı zamanda aydınlatma sırasında gölgelerin ve yansımaların kontrol edilebilmesi gereklidir. Değişik geometrili alanları için farklı yoğunlukla aydınlatma yapabilen birçok özel tasarım ışık sistemi bulunmaktadır. Genelde en çok kullanılan ışık tipleri titreşmesiz flüoresan, LED ve yüksek yoğunluklu ışıktır. Aydınlatma ihtiyacı uygulamadan uygulamaya, basit aydınlatma, çevresel aydınlatma, yüksek hızlı hatlar için anlık donma etkisi veren aydınlatma gibi farklılıklar gösterir.

Aydınlatma sisteminin tasarlanmasında bazı noktalara dikkat edilmesi performans açısından önemlidir.

Örneğin nesne üzerindeki Parlak Yansımalar CCD (Charge Couple Device) kamerada doyuma ve körleşmeye (blooming) sebep olur.

Nokta kaynak gibi bazı ışıklandırma sistemleri ise nesne üzerinde gölgeler oluşturarak bir kameranın detayları görmesini zorlaştırabilirler. Bu nedenle özel uygulamalar hariç nokta kaynak kullanılmaz.

Dağınık ışık kaynakları aydınlatmayı geniş bir alana yayarak ve cisimi pek çok açıdan aydınlatarak istenmeyen yansıma ve gölgeleri önlerler. LED grupları bu iş için idealdirler. Bir başka tip uygulama için çizgi tarama kamera seçilmiş olabilir. LineScan kamera bilgiyi satır satır okur ve bu satırları PC'ye gönderir. Çizgi ışık kaynağı çizgi tarama kameranın taradığı satırı parlak, ince bir ışıkla aydınlatır.

Bir ışık kaynağının pozisyonlanması yanında, kaynağın cinsi de çok önemlidir. En çok kullanılan ışık kaynakları;

- Lazer diyotlar (Light, Amplification, Stimulated and Emitted Radiation = Uyarılmış Radyasyon Yayılımıyla Güçlendirilmiş Işık) yüksek yoğunluklu küçük ışık kaynaklarıdır ve özel bazı uygulamalarda dizi halinde kullanılabilirler
- LED'ler (Light Emitting Diodes) küçük katı-hal ışık kaynaklarıdır. Oldukça düşük maliyetli olan bu elektronik elemanlarla diziler oluşturulabilir. Geniş bant spektrumlu çeşitli LED'ler bulmak mümkündür.
- Quartz-halojen ampuller düşük maliyetle sürekli beyaz ve yoğun ışık üretirler. Fakat aynı zamanda çok fazla ısı da üretirler. İstenmeyen bu ısıyı uzak tutmak gereklidir. Quartz-halojen ampullerin 50 Hz şebeke frekansından etkilenmemesi için DC güç kaynağı kullanılmalıdır. Quartz-halojen ampullerin ömrü 200 saat ile 500 saat civarındadır.
- Strobe lambalar kısa süreli (yaklaşık 10 s), parlak ışık üreten ışık kaynaklarıdır. Lazer ve LED'lerden farklı olarak sürekli olarak geniş bant spektrumuna sahip ışık verirler. Oldukça pahalı olan bu ark-boşalım lambaları yüksek gerilim darbeleri bir güç kaynağına ihtiyaç duyarlar. Ampullerin ömürleri de (saniyede 60 flaşla) yaklaşık 500 saatle sınırlıdır.

Değişik birkaç ışık kaynağından ve düzenlemesinden bir tanesini seçmek nispeten karmaşık bir iştir ve bir miktar hesaplama gerektirir. Konusunda deneyimli bir mühendis bütün bu alternatifleri göz önüne alarak projesini tasarlamalıdır. Ayrıca kesin uygulamaya geçmeden önce tasarlanan sistemi sınırlayabilecek olan insan ve çevre faktörlerinin de mutlaka dikkate alınması zorunludur.

2.2.1.4 Nesne Sensörü

Genel olarak bir kızıl ötesi dalga alıcı vericisinden oluşan bu algılayıcı cihazlar, görüntüsü yakalanacak parçanın kamera merceğinin tam önüne geldiğini algılayarak bilgisayara bir tetikleme sinyali gönderir. Böylece aydınlatma sistemi ve görüntü yakalayıcı uyarılmış (Trigging) olur.

Bu sayede bilgisayarın gereksiz zamanlar için işlem yapmasının önüne geçilerek sistemin performansı artırılır. Bu algılayıcılar uygulamanın şartlarına göre çeşitli yapılarda olabilir. Mekanik anahtar, Lazer beam, Ultrasonic, Kapasitif, yaklaşım gibi farklı yapılardaki algılayıcılar uygulamalarda en çok kullanılan algılama araçlarıdır.

2.2.1.5 Görüntü Yakalayıcı

Bir makine görme işleminde tasarımın kalbi görüntü toplama kartı sayılmaktadır. Görüntü yakalayıcı (frame grabber) olarak adlandırılan bu elektronik devre (PCB) analog ya da dijital bir kameradan gelen görüntüyü alarak bilgisayar kullanımı için mümkün olan en az kayıp ve bozulmayla sayısal veriye çevirir ve işlenecek hale getirir. Görüntü yakalayıcı kartlar genellikle PC içerisine yerleştirilirler ve çeşitli kamera tiplerini ve veri yolu platformlarını desteklemek için farklı yapılandırmalara sahip olabilirler PCI veri yolu en çok kullanılandır. Bunun yanında IEEE1394 ya da Firewire gibi yeni nesil veri yolu tipleri de kullanılmaktadır.

Görüntü yakalayıcı kartının amaca uygun seçilememiş olması durumunda kameradan alınan görüntü işaretlerine gürültü eklenir. Bu yanlışlık görüntü işlenmesi sırasında daha da büyüyerek kontrol işleminin istenen doğrulukta gerçekleşmemesine sebep olur. Kameranın görme alanı ve hassasiyeti, imgenin hızı ve geometrisi bir kartın seçiminde belirleyici rol oynar.

2.2.1.6 Görüntü tanıma (Makine Görme) yazılımı

Tasarlanan sistemde işin en zor kısmı bu bölümdür. Diğer bir deyimle uygulamanın kalbi bu kısımdır. Bu konuda bir standart da yoktur. Her uygulama en iyisidir diye düşünülür. Doğal olarak her uygulamanın özellikleri birbirinden çok farklı olabilir. İstekler, ilaveler ve programı yapan kişiler uygulamanın belirleyicisidir.

Tasarlanan yazılım istenen programı uygular, gelen veriyi işleyerek kararları verir. Bu tür yazılım paketleri bir zamanlayıcıya (timer) sahiptir. Bu işlemin her aşamasının izlenebilmesini sağlar. Bu veriler kullanılarak parçanın hareketine göre ve istenen zamanlama kriterlerine göre programlama yapılır. PC-tabanlı analog kameralı sistemlerde hız genel olarak

saniyede 25 parçaya kadar çıkabilir. Ancak dijital kamera uygulamalarında hız 400 Kare civarına kadar çıkmaktadır. (hatta çarpma testi gibi çok özel uygulamalarda bu hız 100 000 kare/sn 'e kadar çıkmaktadır.) Değişik uygulamalar için özel paket yazılımlar bulunmaktadır. Ancak bunlar son derece pahalıdır ve maliyeti artırır.

Yazılımlardan beklenen bazı temel özellikler:

- Köşe bulma, ölçekleme, hizalama fonksiyonları
- Uzunluk, çap, açı ve kritik boyutları ölçebilme yeteneği
- Gri tonda örnek eşleştirme, şekil eşleştirme, fonksiyonları
- ROI (region of interest) , Eşik bulma ve kenar bulma fonksiyonu
- Alan, çevre ve yer hesaplamalarında tanecik analizi
- RGB, HSL ve HSV imgelerin renk işleme
- Hizalama, ölçüm ve denetleme uygulamaları için arama ve şekil eşleştirme
- Dosyadan, analog ve dijital kameralardan imge toplama
- İmgelerin veri tabanı analizi için 'batch processing'
- OCR Yazılım fonksiyonu ile en sık kullanılan fontları tanıma, karakterleri çözme

2.2.1.7 Veri İletişimi

Makine görüş sistemleri genellikle endüstride üretim hattıyla senkronize çalışması gerekmektedir. Bu senkronizasyon genellikle seri veya paralel haberleşme veya sayısal G/Ç ile otomasyon sistemini tetikleyerek doğru eylemi gerçekleştirmesi sağlanır.

Seri haberleşmede bitler teker teker sırasıyla, paralel haberleşmede ise toplu bir şekilde gönderilir. Donanımsal maliyetleri ve pratikteki zorlukları paralel haberleşmeyi tercih edilmemesine neden olmuştur. Endüstride kullanılan bazı seri haberleşme mimarileri :

- RS-232
- RS-423
- RS-485
- Ethernet
- Fiber Kanal
- Evrensel Seri Veri Yolu (USB)

Bunlar haricinde otomasyon ürünleri üreten firmalarında kendine ait seri haberleşme mimarileri vardır. Bunlara Siemens in ürettiği ProfiBus veya ProfiNet örnek verilebilir.

Haberleşme sırasında gönderilen bitlerin daha hızlı, güvenli, ve doğru adrese gitmesi için seri

haberleşme protokolleri yapılmıştır. Endüstride kullanılan bazı protokoller:

- Modbus
- Internet protokolü (TCP/IP)
- ProfiBus
- ProfiNet
- HART Protokolü
- CANOpen
- DeviceNET
- EtherCAT

Örnekleri verilebilir. Bu örneklerden bazıları çeşitli otomasyon ürünleri üreten firmalarında üretilmiştir.

Veri iletişimi sayesinde üretim hattının rejekt ünitesini kumanda eden RTU ya da PLC birimleri ile PC arasında bağlantısı sağlanır. Bu bağlantının bilgisayar ortamında izlenmesi ve denetlenmesini sağlayan *Veri toplama ve Denetimsel Kontrol* (SCADA) yazılımları vardır. Yazılımı *Proses Kontrolü için Obje Bağlantısı* (OPC) uyumlu hale getirilirse SCADA ekranında yazılımın oluşturduğu giriş çıkışlar gözlemlenebilir. Buna ek olarak sistem bir ağa bağlanarak örneğin istatistikî süreç kontrol uygulaması ya da stok kontrolü gibi çeşitli amaçlarla başka birimlere de bilgi iletmesi sağlanabilir.

2.2.1.8 Bilgisayar

Bir makine görüş sisteminin en önemli üyesi seçilecek bilgisayardır. Sistemin her bir parçayı incelemek için ne kadar zamana ihtiyaç duyacağı uygulamanın en temel noktasıdır. Bu hem CPU seçimini hem de üretim hattının hızını belirleyecektir. Böyle bir uygulama için bilgisayar genellikle hızlı bir Pentium işlemcili ve yeterli PCI yuvası olmak zorundadır. Hız önemlidir çünkü her bir görüntüyü işlemek için gereken zaman işlemci hızlandıkça azalır; bu da hattın daha hızlı ilerlemesi ve daha çok noktanın daha kısa sürede kontrolü demektir. Böyle bir sistem bir üretim hattına kurulduğunda genel olarak ortamdaki toz, nem, sıcaklık titreşim ve elektro manyetik girişimlere karşı dayanıklı olan IP standartlarına sahip endüstriyel PC'ler kullanılır. Bu tip bir PC'nin seçilmemiş olduğu endüstriyel ortamlarda sistem uzun süreli ve verimli çalışmaz.

2.2.1.9 Işık

Işığa tepki vermek tüm canlılarda vardır. Dalga boyu genellikle angström olarak ölçülür ve bir milimetrenin milyonda biridir. Bizlerin görebildiği ışık dalga boyu 4000 angström ultraviyole

bölge ile 7000 angström enfraruj bölge arasındadır. Gözde ki lens ultraviyole ışığının direk retinaya gelmesini engeller hatta lensin çıkarıldığı kişilerde bu ışığın retinaya ulaşması sağlanmış olur.

Göz ve görüş sistemi çok geniş ışık yoğunluğu seviyelerine adapte olabilir. Tecrübelerle bilinir ki; öznel parlaklığın (insan görüş sistemi tarafından algılanan yoğunluk), göze düşen ışık yoğunluğunun logaritmik bir fonksiyonudur.

Herhangi bir özel ışık yoğunluğu değişimlerine adaptasyon ve değişim arasında ki ayırımı yapma yeteneği de önemli bir ilgi alanıdır. İnsan görüş sisteminin parlaklık ayırımı kapasitesini tespit etmek için basit bir test ile ölçülebilir. I aydınlatma şiddeti ile aydınlatılmış alanın merkezinde daire olarak görünen ışıltıya kısa süreli ΔI kadar aydınlatma artışı eklendiğinde eğer I yeterince parlak değilse, algılanabilir değişiklik fark edilmez. Eğer I daha artırılırsa değişiklik algılanabilir seviyeye gelecektir. Sonunda, I yeterince kuvvetli olursa, her zaman fark edilir halde kalır.

I_c zemin aydınlanması olmak üzere, I_c / I büyüklüğüne Weber oranı denir. I 'nin %50 katı ayırt edilebilir aydınlanma artışıdır. Küçük bir I_c / I değeri, aydınlanma şiddeti yoğunluğunda küçük orandaki değişimin ayırt edilebildiği anlamına gelmektedir. Bu “iyi” parlaklık ayırımı gösterir. Tersine, büyük I_c / I değeri yoğunlukta büyük oranda değişim gerektiği anlamına gelmektedir. Bu da “zayıf” parlaklık ayırımı gösterir. İnsan algılama olağanüstülüğünün diğer örnekleri, gözün olmayan bilgileri doldurduğu veya nesnelerin geometrik şekillerinin yanlış algılandığı optik yanılsamalardır.

2.2.2 Sayısal Görüntü

Işık şiddetinin iki boyutlu ifadesi bize bilgisayar ortamında görüntüyü vermiş olur. Çevremiz, gördüklerimiz 3 boyutludur. Çevreden bir görüntü elde edebilmek için 3 boyutlu cisimlerin izdüşümünü alarak 2 boyutla sınırlandırmış oluruz. Bu 2 boyutlu düzleme, resim düzlemi denir. Resim düzleminde ki her nokta, bir renk değerine ve resim düzlemi koordinatına sahiptir. Resim düzleminde bir noktada ki ışık seviyesinin nasıl belirlendiğinin bilinmesiyle, üç boyutlu ortam koordinatlarıyla arada ki seviye farkının bilinmesiyle resmin yapısı anlaşılabilir olur.

Kamera Projeksiyon Modelleri üç boyutlu ortam noktasının, resim düzlemi üzerindeki yerini belirlemek için kullanılır. Perspektif izdüşümünde, resim düzleminin sabit bir mesafe önünde bir küçük delik (pinhole) olduğu ve ışık ışını sadece bu delik den geçerek resim düzlemine

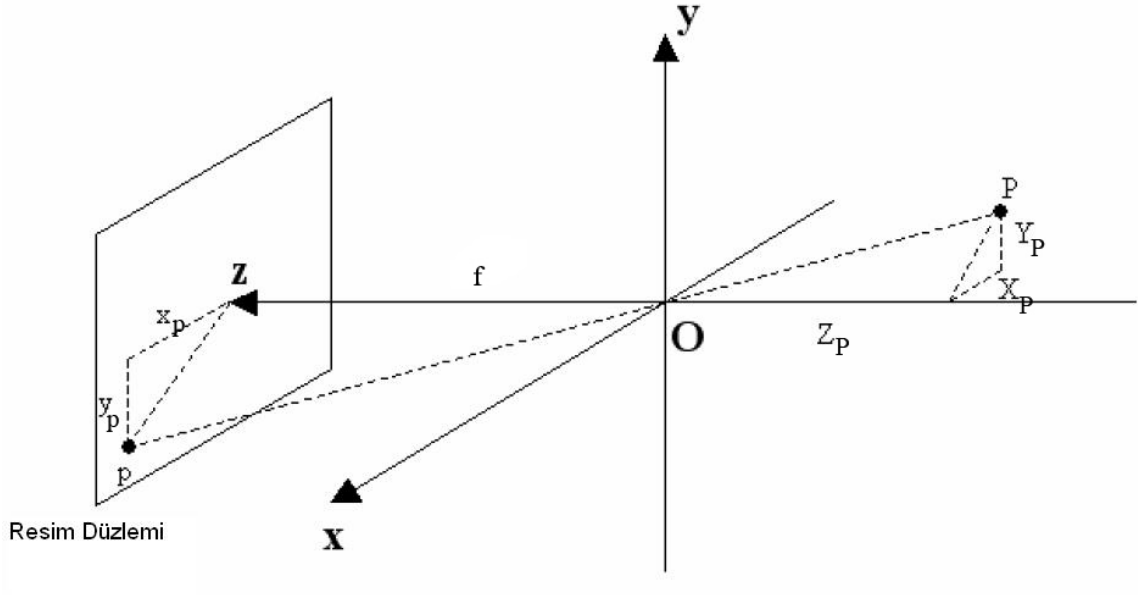
ulaşabileceği kabul edilir. Böylece ışık bu noktadan düz bir çizgi şeklinde geçerek, resim düzlemindeki nokta ile ortam noktası arasında ilişki belirlenir..

Şekil 2-10'de z- ekseni (optik eksen) resim düzlemine diktir. Burada amaç P noktasının resim içindeki yerini bulmaktır. Şekil 2-10'da üçgenlerin benzerliğinden faydalanılarak resimde p noktasının yeri (2-1) ve (2-2) ile hesaplanabilir.

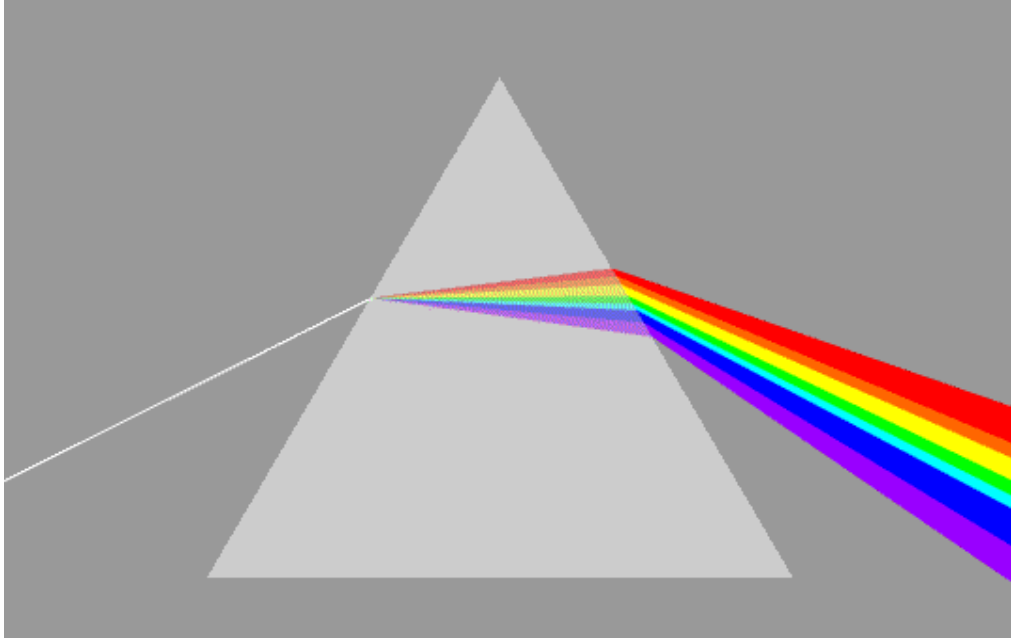
$$\frac{x_p}{f} = \frac{X_p}{Z_p} \quad (2-1)$$

$$\frac{y_p}{f} = \frac{Y_p}{Z_p} \quad (2-2)$$

Burada f resim düzleminin orijinden (pinhole) uzaklığı ve (x_p, y_p) resim düzleminde p noktasının koordinatlarıdır.



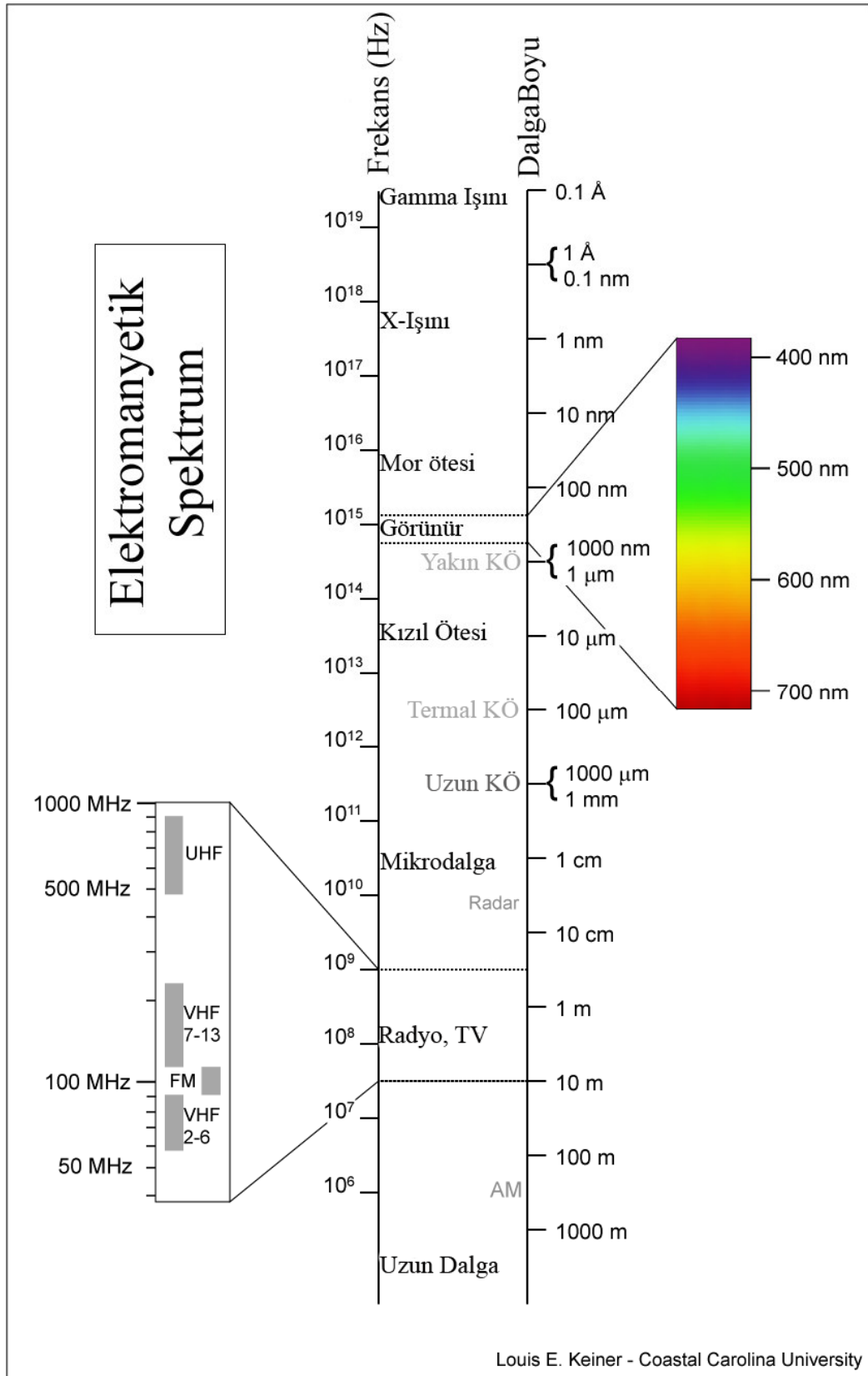
Şekil 2-10 Pinhole kamera projeksiyon modeli



Şekil 2-11 Cam prizma

Şekil 2-11 deki görüldüğü gibi güneş ışını cam prizmadan geçtiğinde görünen ışın beyaz değildir. 1666 yılında Newton, görünen ışının bir ucu mor bir ucu kırmızı olan aralıksız bir spektrum içerdiğini keşfetmiştir. Şekil 2-12’de görüldüğü gibi görünen ışıpta algılanabilen renk aralığı elektromanyetik spektrumun küçük bir kısmını göstermektedir [Gonzelez R.C., 2002]. Kolaylık için, renk spektrumu 6 geniş alana bölünmüştür: mor, mavi, yeşil, sarı, turuncu ve kırmızı.

İnsan gözü yeşil renklerde hassastır, çok küçük renk değişimlerini algılayabilir. Parlaklık fonksiyonu olarak adlandırılan hesaplamalarla da, insan gözünün 5550 angström değerinde en hassas konumunda olduğu hesaplanmıştır. Bu da yeşil renge tekabül eder [4].



Şekil 2-12 Tüm elektromanyetik spektrum ve görünür spektrum [3]

Spektrumda açıkça görülmektedir ki; bir uçta dalga boyu görünen ışıktan milyar kadar daha uzun olan radyo dalgaları, diğer uçta ise görünen ışıktan milyon kadar daha küçük olan gama ışınları vardır. Elektromanyetik spektrum dalga boyu, frekans ve enerji cinsinden ifade edilebilir. Dalga boyu (λ) ve frekans (ν) arasında ilişki aşağıdaki gibidir:

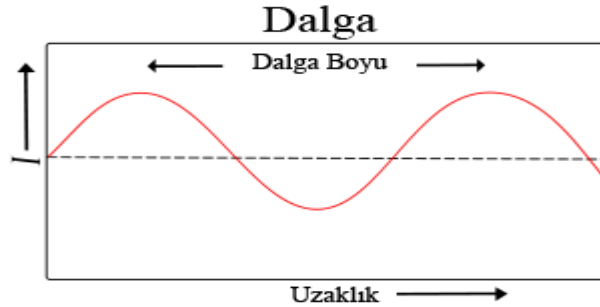
$$\lambda = \frac{c}{\nu} \quad (2-3)$$

Burada c ışık hızını ifade etmektedir (2.998×10^8 m/s). Elektromanyetik spektrumun enerjisi aşağıdaki ifadede verilmiştir:

$$e = h\nu \quad (2-4)$$

Burada h, Planck sabitidir.

Dalga boyu bir uzunluk olduğu için birimi de metre cinsi de kullanılabilir. Mikron ve nanometrede kullanılmaktadır. Frekans Hertz (Hz) cinsinden ölçülür. Şekil 2-13'de görüldüğü gibi dalga boyu λ olan sinüs dalgalar şeklindedir, elektro manyetik dalgalar. Tanım olarak; ışık hızında hareket eden kütsüz parçacıkların akısıdır.



Şekil 2-13 Elektromanyetik dalga formu ve dalga boyu

Kütsüz parçacığın belirli miktarda ki yığın enerjisine foton denir. Eşitlik (2-4) 'den görüldüğü gibi enerji frekans ile orantılıdır. Yani yüksek frekanslı elektromanyetik dalgalar her bir fonda daha fazla enerji taşır anlamına gelmektedir. Yüksek frekans Şekil 2-13 'den de anlaşılacağı gibi kısa dalga boyuna sahiptir. Buradan anlaşılacağı üzere radyo dalgaları düşük enerjili fotonlardır. Mikrodalgalar biraz daha fazla bunu sırasıyla kızılötesi enerjisi takip eder. Daha sonra görülebilir, morötesi, X-ışınları ve son olarak gama ışınları gelir. Gama ışınları hepsinden daha fazla enerjiye sahiptir, yaşayan canlılar için bu nedenle tehlikelidir.

Achromatic yani renksiz ışığın diğer ışıklardan farkı yoğunluğu ve miktarıdır. Bu ışığa tek

renkli (monochromatic) ışık da denmektedir. Gri seviye terimi genelde tek renkli yoğunluğu tanımlamak için kullanılır, çünkü siyahtan griye ve son olarak beyaza değişir. Renkli ışık kaynağının kalitesini tanımlamak için 3 temel büyüklük kullanılır:

Işıma veya Aydınlık (Radiance): ışık kaynağından akan toplam enerji miktarıdır ve genelde watt (W) cinsinden ölçülür.

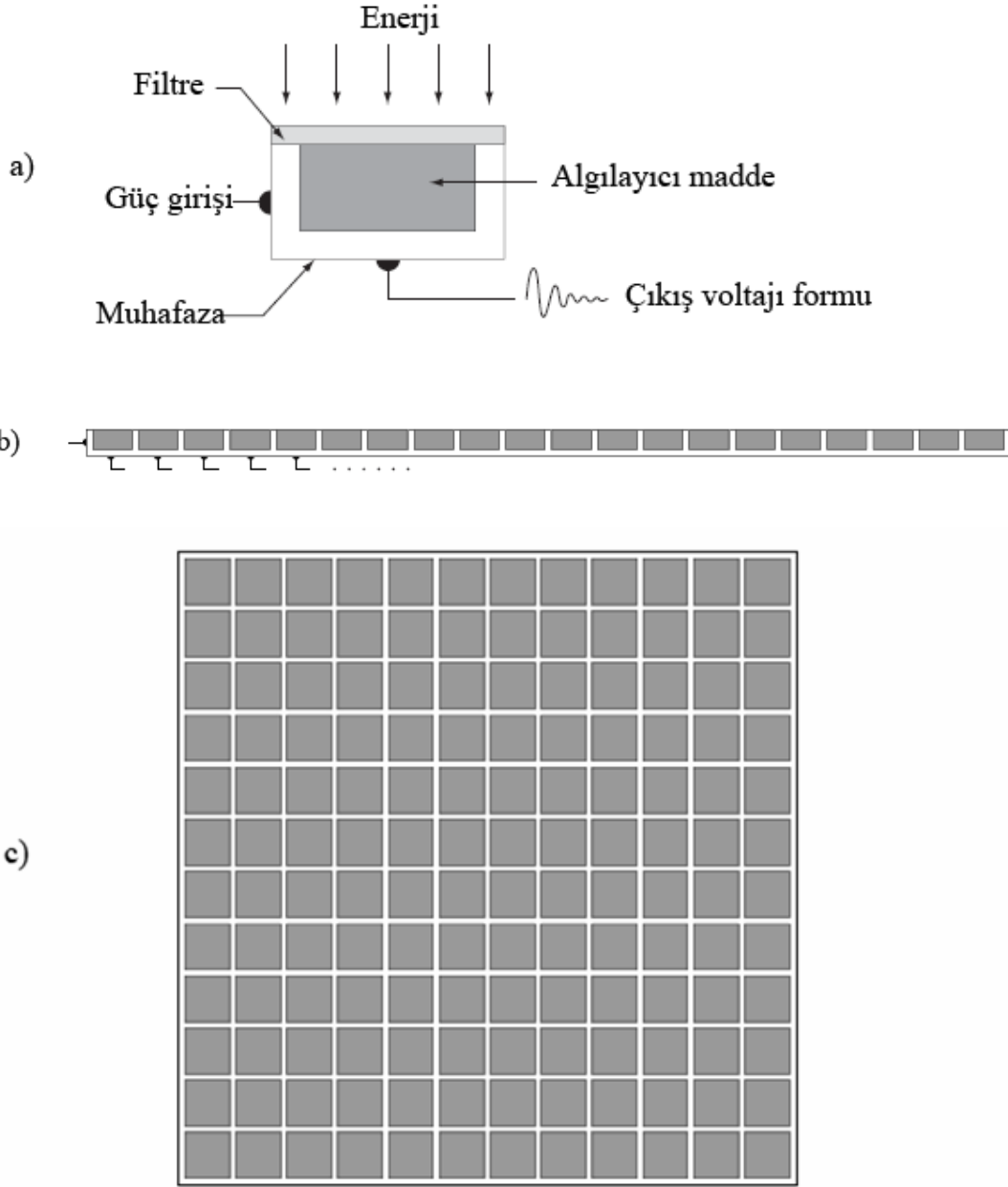
Parıldama (Luminance): lümen (lm) cinsinden ölçülür ve gözlemcinin ışık kaynağından algıladığı toplam enerji miktarının ölçümünü verir. Örneğin kızılötesi bölgesinde ışınım yapan bir kaynaktan yayılan yüksek enerjili bir ışını, gözlemci zor algılar veya algılayamaz. Bu kaynağın ışıma' sı çok yüksek olsa bile gözlemci için parıldama nerdeyse 0 dır.

Parlaklık (Brightness): ölçümünün pratik olarak imkânsız olduğu ışık algılayışının öznel tanımlayıcısıdır. Bu ışık şiddetinin renksiz kavramını belirten ve renk hissini tanımlayan anahtar faktörlerden biridir. Parlaklık için gri seviye değeri, resim ışık şiddeti terimiyle eşdeğeri olarak kullanılır.

Perspektif Projeksiyon Modelinde görüntülenen ortamdaki bir noktadan gelen ışığın, daha fazla küçülemez (pinhole) bir delikten geçerek resim düzlemine ulaştığı kabul edilmişti. Fakat bu durumda resim düzlemi üzerine çok az ışık miktarı düşmüş olur. Dolayısıyla resmi elde etmek için kullanılan ışık algılayıcısına yeterli miktarda ışık gelmemiş olur. Bu durum, perspektif izdüşümün temeliyle çelişse de daha büyük bir açıklık kullanma zorunluluğunu doğurmuştur.

2.2.3 Görüntü Algılama ve Elde Etme

Aydınlatma kaynağından yayılan enerjinin ortamdaki cisimler tarafından yansıtılması veya emilmesi ile gerçekleşmesi görüntüyü oluşturur. Aydınlanma enerjisi nesneden yansıyabileceği gibi nesnenin içinden de iletilebilir. Bunu, şöyle örneklendirebiliriz; tıpta X-ışını filmi oluşturmak için hastaya vücudundan geçebilecek X-ışınları yollar. Bazı uygulamalarda, yansıyan veya iletilen enerji, enerjiyi görülebilir ışığa çeviren foto çevirici üzerine odaklanır (örneğin fosfor ekran). Elektron mikroskop ve gama görüntülemenin bazı uygulamaları bu yaklaşımı kullanır.

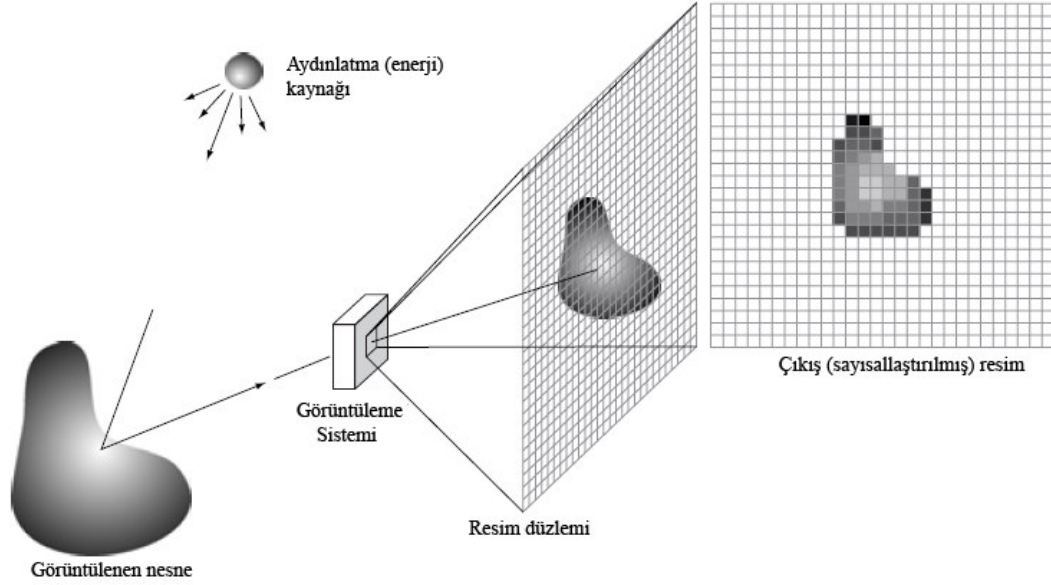


Şekil 2-14 Piksel sensörü ve sensör dizilimleri

Şekil 2-14, aydınlanma enerjisini sayısal görüntüye dönüştürmek için kullanılan üç ana algılayıcı yerleşimini göstermektedir. Basitçe gelen enerji, giriş elektrik gücü ve sezilen özel tip enerjiye duyarlı algılayıcı malzemenin birleşimi voltaja dönüştürülür. Çıkış voltajı dalga şekli algılayıcıların tepkisidir ve tepkisini sayısallaştırarak her bir algılayıcıdan elde edilen bir niceliktir.

2.2.3.1 Algılayıcı dizileri (sensor arrays) kullanarak görüntü elde etme

Şekil 2-14 (c)'de algılayıcıların 2-boyutlu hali verilmektedir. Pek çok elektromanyetik ve bazı ultrasonik algılama donanımları sıklıkla bu şekilde biçimlendirilirler. Günlük hayattan örnek olarak bu dizilim biçimine sahip dijital kameralar vardır. Bu kameralar için tipik algılayıcılar, geniş bir algılama aralığına sahip ve kabaca 4000 X 4000 veya daha fazla algılama elemanından oluşan CCD (Charge Coupled Device) dizilerdir. Algılayıcı yani sensörlerin algıya karşı cevabı sensor elemanının yüzeyine düşen ışık enerjisi toplamı ile orantılıdır.



Şekil 2-15 Algılama elemanı ile sayısal resmin elde edilmesi

Şekil 2-15'de bir nesnenin görüntüsünün CCD elemanı ile elde edilmesi görülmektedir. Şekilde net bir şekilde görülmektedir ki herhangi bir ışık kaynağından yayılan ışık enerjisi nesneden yansıdıktan sonra mercekten geçerek CCD elemanı tarafından bu ışık enerjisi toplanmakta ve resim düzlemi üzerine odaklanmaktadır. Her sensörün cevabı üzerine düşen enerji toplamıyla orantılı olduğunu söylemiştik buna göre doğal olarak sensör üzerine düşen ışık enerjisine göre oransal bir çıktı üretilmektedir. Bu çıktı ile video sinyali oluşturulmakta ve sonrada örnekleme ve miktarlama ile dijital görüntüye ulaşılmaktadır. Bu yapının temel avantajı komple resmin kolayca elde edilebilmesidir.

3. GÖRÜNTÜ OLUŞTURMA VE İŞLEME

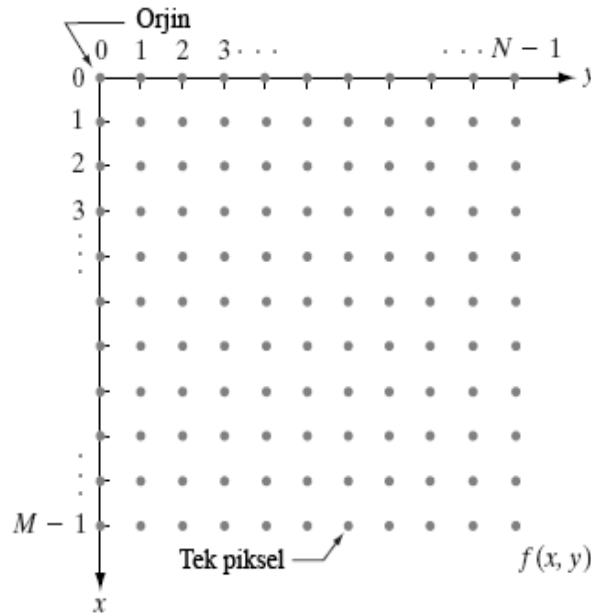
3.1 Yöntemler

3.1.1 Sayısal Görüntünün İfade Edilmesi

Örnekleme ve nicelleştirme işlemleri görüntünün sayısal çözünürlüğü ve resimde ki renk derinliğini gerçek sayılardan oluşan matris yoluyla vermiş olur. Bu değerler herkesin bildiği renk ve görüntü kalitesinin ölçüleri gibi değerlendirilebilir. Örnekleme işlemi sonucu sayısal görüntünün çözünürlüğü belirlenmiş olur (örneğin:1024x768, 720x576, 640x480), nicelleştirme sonucu ise resimdeki renk derinliği belirlenmiş olur (renk derinliği; 8 bit gri seviye için 256, 16 bit gri seviye için 65536).

Görüntü, iki boyutlu ışık şiddeti fonksiyonudur. Bu fonksiyon $f(x,y)$ şeklinde gösterilir. Burada x ve y uzaysal koordinatları, (x,y) noktasındaki f 'nin sayısal değeri ise parlaklık değeri veya görüntünün ilgili noktadaki renk seviye değeridir. Herhangi bir (x,y) koordinatındaki f 'nin genliğine, görüntünün o noktadaki yoğunluğu denir.

Bir sayısal $f(x,y)$ görüntüsünün M satır ve N sütunlu sayısal görüntü oluşturacak şekilde örneklendiğinde koordinatların değerleri (x, y) , ayrık nicelikler halindedir. Notasyon açıklığı ve uygunluğu için, bu ayrık koordinatlar tamsayı değerlerle ifade edilir. Böylece, merkezdeki koordinatların değerleri $(x,y) = (0,0)$ 'dır. Görüntünün sonraki ilk satırı boyunca ki koordinat değerleri $(x,y) = (0,1)$ olarak gösterilir. $(0,1)$ notasyonu ise ilk satır boyundaki ikinci örneği belirtmek için kullanılır. Bu, görüntü örneklendiğinde, fiziksel koordinatların gerçek değerleri olduğu anlamına gelmemektedir. Şekil 3-1'de bu uygulama boyunca kullanılan koordinat tanımını göstermektedir [Gonzalez R.C. 2002].



Şekil 3-1 Sayısal resim koordinat yapısı

Bu notasyon, tüm $M \times N$ boyutlarındaki sayısal görüntüyü aşağıdaki matris formunda yazmayı sağlamaktadır.

$$f(x, y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & \dots & f(1,N-1) \\ \dots & \dots & \dots & \dots \\ f(M-1,0) & f(M-1,1) & \dots & f(M-1,N-1) \end{bmatrix} \quad (3-1)$$

Bu fonksiyonun eşitliği bize sayısal görüntünün tanımını verir. Matris dizisindeki her eleman görüntü elemanı, resim elemanı, piksel olarak adlandırılır. Görüntü ve piksel terimleri, bu tezin kalan bölümleri boyunca sayısal görüntüyü ve elemanlarını belirtmek için kullanılmıştır. Bazı durumlarda, sayısal görüntüyü ve elemanlarını belirtmek için daha geleneksel matris notasyonu kullanmak avantajlı olabilir.

$$A = \begin{bmatrix} a_{0,0} & a_{0,1} & \cdot & \cdot & a_{0,N-1} \\ a_{1,0} & a_{1,1} & \cdot & \cdot & a_{1,N-1} \\ \cdot & \cdot & & & \cdot \\ \cdot & \cdot & & & \cdot \\ a_{M-1,0} & a_{M-1,1} & \cdot & \cdot & a_{M-1,N-1} \end{bmatrix} \quad (3-2)$$

Burada, $a_{ij} = f(x=i, y=j) = f(i, j)$, bundan dolayı $f(x, y)$ ve $A(i, j)$ özdeş matrislerdir. Formüle dayalı bir örnekleme ve nicelleştirme tanımı sayısal görüntüyü tam anlamda manalaştırabilir. Z ve R sırasıyla tamsayılar kümesini ve gerçek sayılar kümesini ifade etmektedir. Örnekleme işlemi x - y düzlemini gridler içinde kısımlara ayırarak görülebilir, burada her bir grid merkezinin koordinatları kartezyen çarpım Z^2 'nin elemanı olan (z_i, z_j) düzenli eleman çiftlerinin kümesidir ve z_i ve z_j Z kümesinin (tamsayılar kümesi) elemanlarıdır. Bu nedenle, eğer $(x, y) \in Z^2$ kümesinin tamsayı elemanları ise $f(x, y)$ sayısal görüntüdür ve f , bir gri seviye değerini gösteren ve gerçek sayılar kümesinin elemanı olan R 'yi, her bir farklı (x, y) koordinat çiftine atayan bir fonksiyondur. Bu fonksiyonel atama açıkça daha önce tanımlanan nicelleştirme işlemidir. Eğer gri seviyeler aynı zamanda tamsayı ise, Z , R 'de yer alır ve sonra sayısal görüntü koordinatları ve genlik değerleri tamsayı olan 2-boyutlu fonksiyon haline gelir.

Sayısallaştırma işlemleri sonrası M , N değerleri hakkında ayrıca her bir piksel için izin verilmiş ayrık gri seviyelerin sayısı L için kararlar gerektirir. M ve N için pozitif tamsayı olmaları gerektiği dışında herhangi bir gereklilik yoktur. Örnekleme donanım faktörlerine bağlı olarak, gri seviyelerin sayısı tipik olarak 2'nin tamsayı kuvvetidir.

$L = 2^k$	(3-3)
-----------	-------

Ayrık seviyeler eş uzaklıkta yerleşmiştir ve $[0, L-1]$ aralığında tamsayıdırlar. Bazen gri skalada uzanan değerler aralığına görüntünün dinamik aralığı denir. Fazla sayıdaki pikseller bu özelliği ortaya koyduğunda, görüntü yüksek kontrasta sahip olacaktır. Tersine düşük dinamik aralığına sahip görüntüler mat bir renge sahiptirler. Sayısallaştırılmış görüntüyü depolamak için gerekli bit sayısı, B ,

$B_s = M \cdot N \cdot k$	(3-4)
---------------------------	-------

şeklinindedir. $M = N$ olduğunda yukarıdaki denklem (3-4) den denklem (3-5) elde edilir.

$B_s = N^2k$	(3-5)
--------------	-------

Bir görüntünün 2^k gri seviyesi olabileceği zaman, görüntüyü k-bit görüntü olarak belirtmek genel bir yaklaşımdır. Örneğin, 256 muhtemel gri-seviye değerleri olan bir görüntüye 8-bit görüntü denir ve $k = 8$ dir. 1024x1024 veya daha yüksek boyuttaki 8-bit görüntülerin depolama için kullanılan miktar 8388608 bit olup bu 8 Megabyte'a eşittir.

3.1.2 Temel İkili Düzey (Binary) Görüntü İşlemleri

İkili resim yapısında, resim renk değerleri 1 bit ile ifade edilir. Dolayısıyla $2^1 = 2$ renk değeri (0 siyah 1 beyazı ifade eder) sunar. Genellikle '0' arka plan piksellerini, 1 de nesne piksellerini oluşturur. İkili resim yapısı resimdeki bazı özellikleri çıkarmak için kullanılabilir. 8 bit gri düzey resim ikili resim yapısına dönüştürüldükten sonra değişik işlemlere tabi tutularak istenen alan, kütle merkezi vb. nesneye ait özellikler resimden çıkartılabilir. Şekil 3-2 'de ikili resim üzerinde uygulanan mantıksal işlemler aşağıdaki gibidir ve resim üzerinde gösterimi Şekil 3-2 da görülmektedir.

$$A \text{ or } B = A \cup B$$

$$A \text{ and } B = A \cap B$$

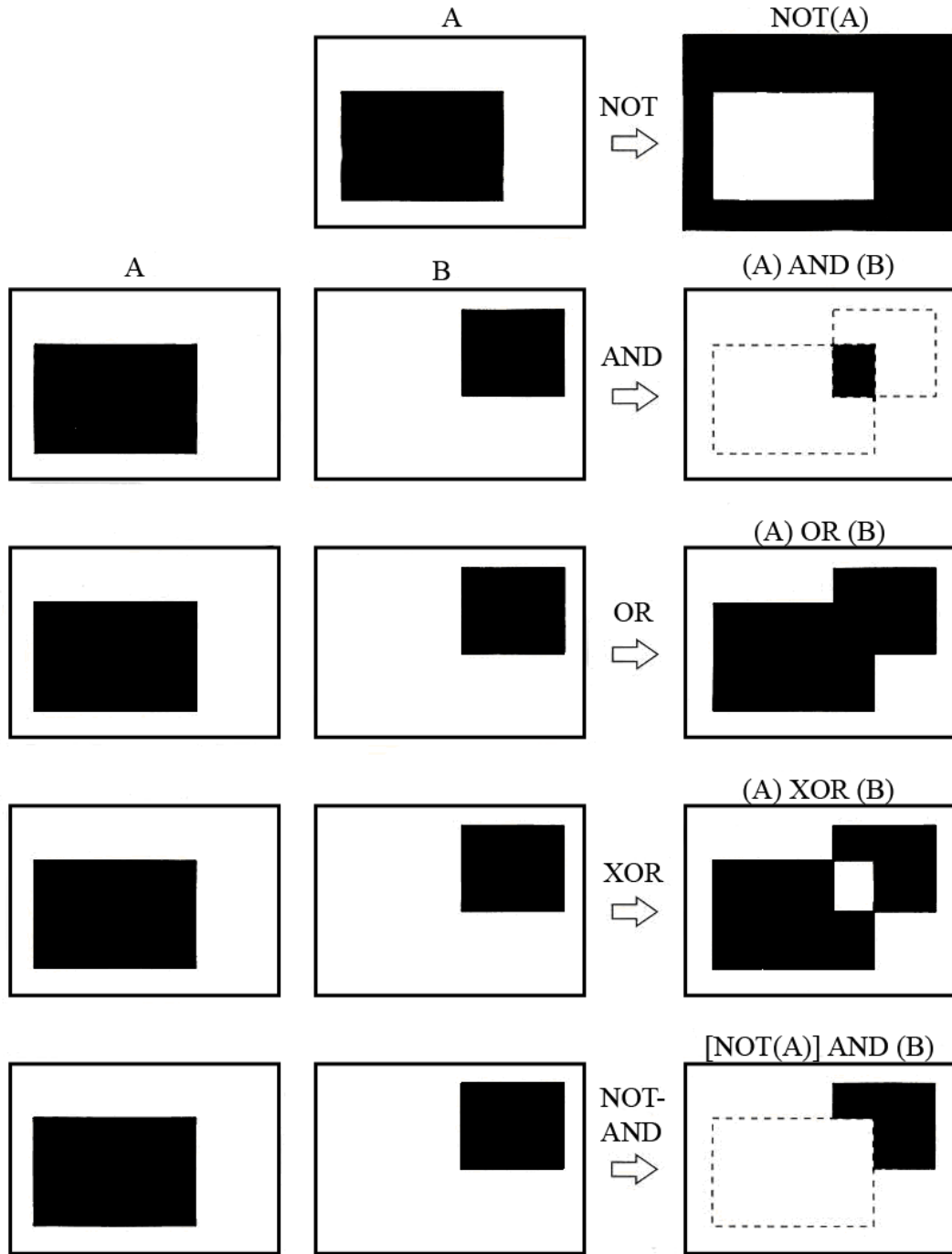
$$\text{not } A = A^c$$

$$A \text{ and } (\text{not } B) = A - B$$

(3-6)

İkili düzey resimde her iki piksel ve belirtilen komşuluk ilişkisine göre komşuları, yapısal eleman ile yukarıda belirtilen mantıksal işlemlere tabii tutularak Şekil 3-2'de hazırlanmıştır. Şekilde siyah kısımlar 1'leri, beyaz kısımlar 0'ları temsil eder.

İçindeki nesnelere ilgili bilgilerin edinilmesinde daha doğru sonuçlar elde edilmesini sağlar.



Şekil 3-2 Bazı ikili düzey işlemler ve sonuçları [Gonzalez R.C. 2002]

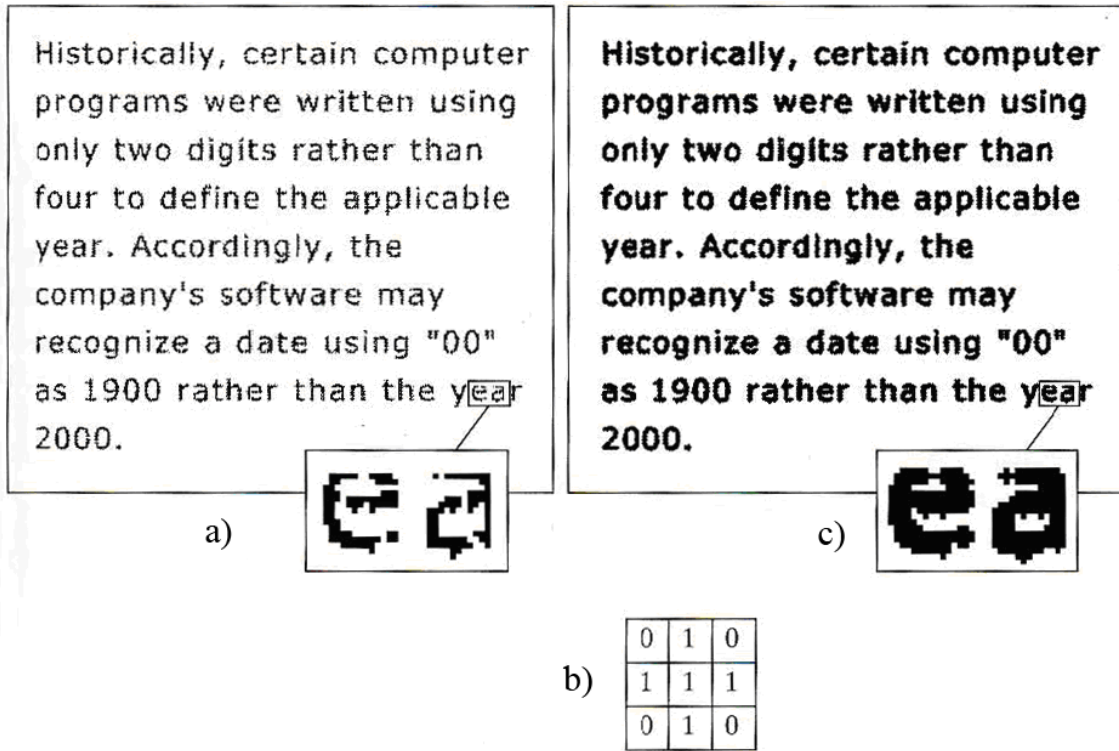
3.1.3 Genişleme (dilation) Operatörü

Genişleme işleminin kullanım amacı görüntü kalınlaştırma, büyütme yapmaktır.

A ve B , Z^2 üzerindeki kümeler olma üzere Genişleme operatörü şu şekilde tanımlanır.

$$A \oplus B = \left\{ z \mid \left(\hat{B} \right)_z \cap A \neq \emptyset \right\} \quad (3-7)$$

B yapılandırma elemanı olarak adlandırılır. Yapılandırma elemanı, baklava dilimi, disk, çizgi, kare, dikdörtgen gibi şekillerde tanımlanabilir veya bizim istediğimiz özel bir şekle sahip olabilir. Böylece nesne kenarları istenilen forma sokulabilir. Şekil 3-3 (a)'da verilen resim bölümü için, (b)'deki yapılandırma elemanın (structuring element) nasıl uygulandığı ve (c)'de genişleme işlem sonucu görülmektedir.



Şekil 3-3 Genişleme İşlemi örneği

3.1.4 Aşınma (erosion) Operatörü

Aşınma işlemi görüntüde istenmeyen gürültüleri filtreleme, görüntüde düzeltme yapmak amacıyla kullanılabilir.

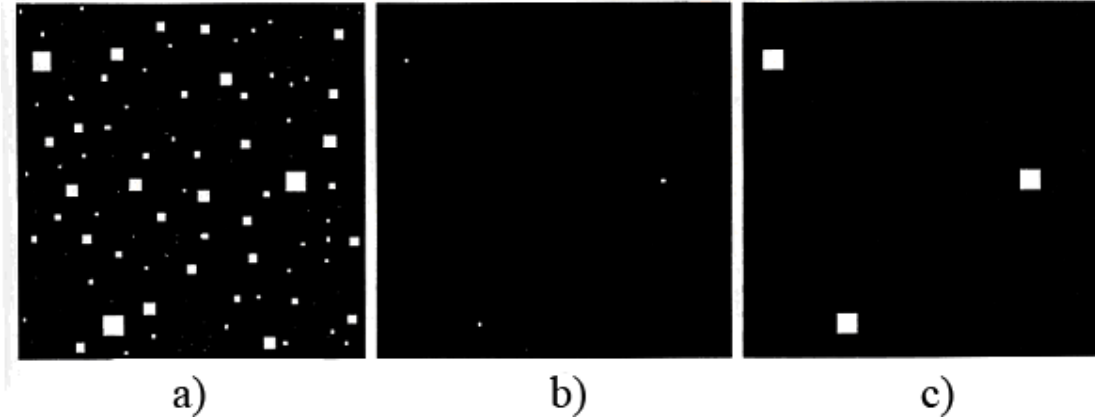
A ve B , Z^2 üzerindeki kümeler olma üzere Aşınma operatörü şu şekilde tanımlanır.

$$A \odot B = \{z | (B)_z \subseteq A\} \quad (3-8)$$

Aşınma işlemi genişlemenin tersidir. Aşınmada da genişlemede olduğu gibi aşınmanın şeklini yapılandırma elemanı belirler. Aşınma ile Genişleme operatörünün arasında şu ilişki mevcuttur.

$$(A \odot B)^c = A^c \oplus \hat{B} \quad (3-9)$$

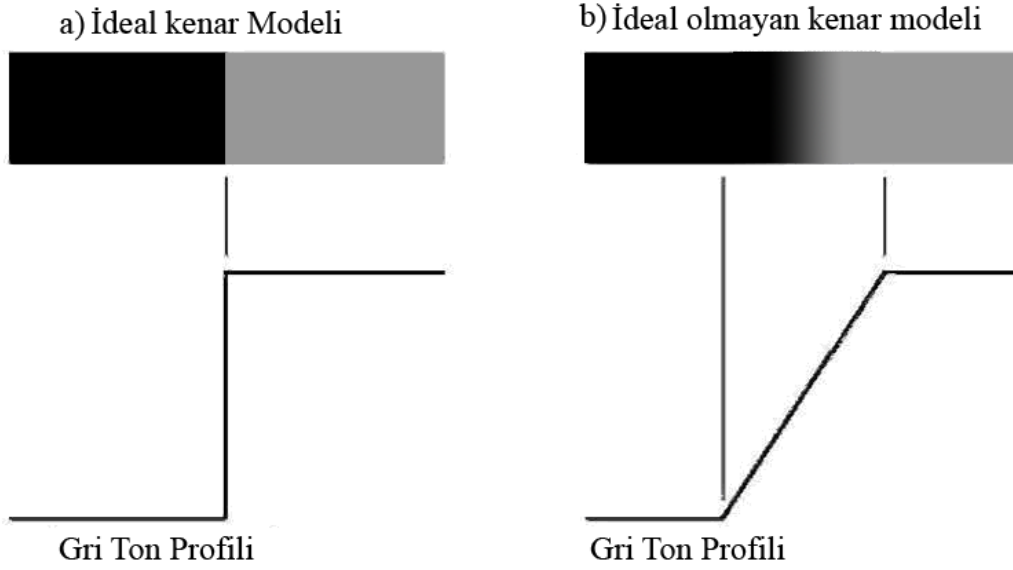
Şekil 3-4 ile gösterilen örnekte ise, (a) şekilde 11'den oluşan düzlem üzerinde 1,3,5,7,9 ve 15 lik kareler görülmektedir. (b) şekli, (a) şeklinin 13'lük yapılandırma elemanı ile aşındırılmış halidir. (c) şekli ise, (b) şeklinin 15'lik kare yapılandırma elemanı ile genişletilmiş halidir. Bu sayede (a) şeklinde istenmeyen gürültüler (c) şeklinde temizlenmiş, sadece aranan 15'lik kare cisimler elde edilmiştir.



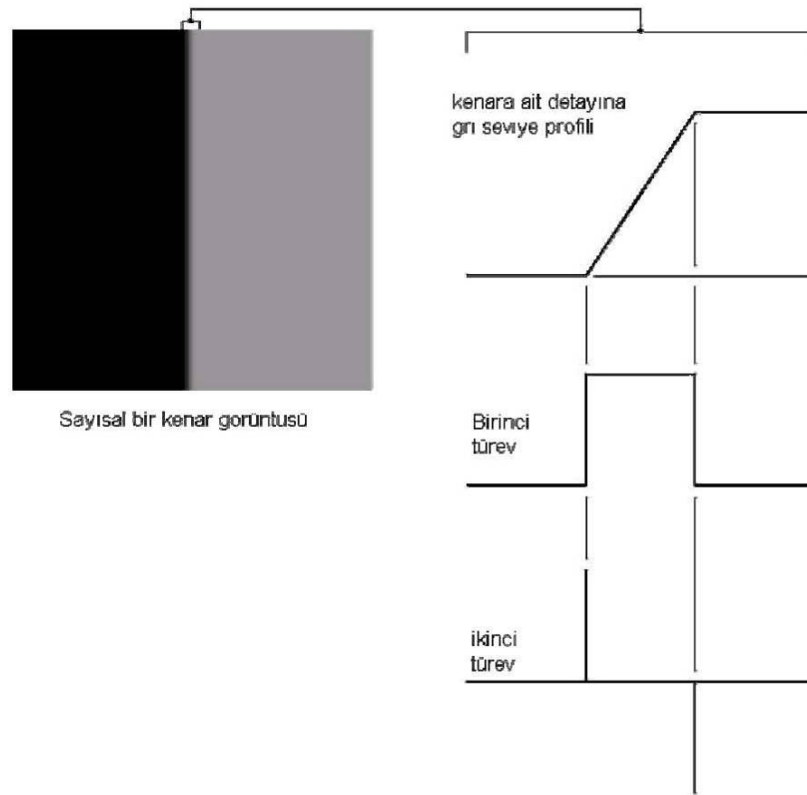
Şekil 3-4 Aşınma ve Genişleme örneği [Gonzalez R.C. 2002]

3.1.5 Kenar Belirleme

Kenar belirleme işlemi için dijital bir görüntüde kenarın nasıl görüldüğüne yakından bakılırsa bir sayısal resimde ideal bir kenarla, normal bir kenarın detayı Şekil 3-5'deki gibidir. İdeal olmayan kenardaki rampanın eğimi görüntünün bulanıklık (blur) seviyesi ile orantılıdır. Pratikte elde edilen kenar görüntüsünün bulanık ve dağınık olması, örnekleme, yetersiz aydınlatma, optik ve kullanılan diğer donanımların ideal olmaması nedenlerinden ve hatalarından kaynaklanmaktadır. Bunun sonucunda nesne kenarlarındaki gri seviye değerinde rampa şeklinde bir değişim görülür. Şekil 3-6'de ideal olmayan nesne kenarının birinci ve ikinci türevleri görülmektedir .



Şekil 3-5 Kenar profil modeli



Şekil 3-6 Sayısal kenarın birinci ve ikinci türevi

Bu işlemde şu sonuç çıkmaktadır. Bir görüntünün birinci türevinin genliği (gradient), görüntü içerisindeki o noktada herhangi bir kenarın var olup olmadığını kontrol etmek için

kullanılabilir. İkinci türev ise kenar pikselinin siyah bölgede mi yoksa beyazda mı olduğunu anlatmak için kullanılabilir. Ayrıca ikinci türev sonucu iki ek özellik daha oluşmaktadır. Bunlar; kenar bölgesinde iki değer üretilmektedir ki bu istenmeyen bir durumdur. Diğeri ise; kenarın tam orta noktasında ikinci türev sıfır olmaktadır ki bu duruma “zero-crossing” özelliği denir ve sınırı tek ince bir çizgi olarak ifade etmek istenildiğinde kolaylık sağlar.

3.1.6 Geometrik Dönüşümler

Bir f resminin üzerindeki (x, y) koordinatına sahip bir piksel alalım. Bu resim herhangi geometrik bozulmaya tabi tutulsun ve (x, y) noktamız (x', y') noktasına gitsin. Bu dönüşümü aşağıdaki gibi gösterebiliriz.

$$x' = r(x, y) \quad (3-10)$$

$$y' = s(x, y) \quad (3-11)$$

$r(x, y)$ ve $s(x, y)$ fonksiyonları resmi x ve y yönünde dönüştürecek geometrik fonksiyonun bileşenleridir.

Sıkça kullanılan geometrik dönüş işlemlerinden biride boyutlandırmadır. $r(x, y) = x/2$ ve $s(x, y) = y/2$ ile resmin boyunu yarıya indirebilir. Bu durumda r ve s fonksiyonları

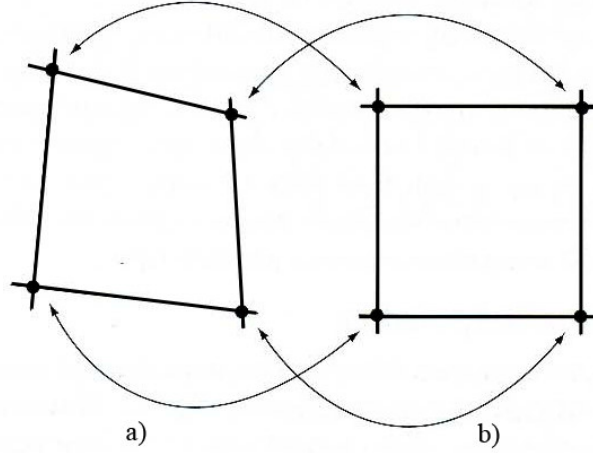
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1/2 & 0 \\ 0 & 1/2 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} \quad (3-12)$$

şeklinde olur. Bir resim üzerinde θ açısı kadar döndürme yapmak için ise döndürme noktasını orijin olmak üzere

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} \quad (3-13)$$

fonksiyonunu kullanabiliriz.

Geometrik olarak bozulmuş resimlerde genellikle kullanılan yöntem ise düğüm metodudur. Daha önceden referans olarak belirlenen düğüm noktaları resimde lens problemi yüzünden (fıçı etkisi) kaymaya uğraya bilir.



Şekil 3-7 (a) lens yüzünden bozulmaya uğramış resim, (b) düzgün resim.

Bu tür bozulmalar ikili doğrusal (bilinear) denklemler şeklinde modellenebilir.

$$r(x, y) = c_{00}xy + c_{01}x + c_{02}y + c_{03} \quad (3-14)$$

$$s(x, y) = c_{10}xy + c_{11}x + c_{12}y + c_{13} \quad (3-15)$$

Fonksiyonda dört düğüm noktası ile c_{00} , c_{01} , c_{02} , c_{03} , c_{10} , c_{11} , c_{12} , c_{13} bilinmeyen katsayılarını bulunmaktadır. Bu dört nokta, her biri x ve y olmak üzere iki bileşene sahip olduğundan toplam sekiz bilinmeyen ve sekiz tane de nokta bulunmaktadır. Bu sekiz bilinmeyen ve sekiz adet noktayı kullanarak aşağıdaki matris elde edilebilir.

$$\begin{bmatrix} x'_1 \\ y'_1 \\ x'_2 \\ y'_2 \\ x'_3 \\ y'_3 \\ x'_4 \\ y'_4 \end{bmatrix} = \begin{bmatrix} x_1y_1 & x_1 & y_1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & x_1y_1 & x_1 & y_1 & 1 \\ x_2y_2 & x_2 & y_2 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & x_2y_2 & x_2 & y_2 & 1 \\ x_3y_3 & x_3 & y_3 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & x_3y_3 & x_3 & y_3 & 1 \\ x_4y_4 & x_4 & y_4 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & x_4y_4 & x_4 & y_4 & 1 \end{bmatrix} \cdot \begin{bmatrix} c_{00} \\ c_{01} \\ c_{02} \\ c_{03} \\ c_{10} \\ c_{11} \\ c_{12} \\ c_{13} \end{bmatrix} \quad (3-16)$$

3.1.7 Renk Sistemleri

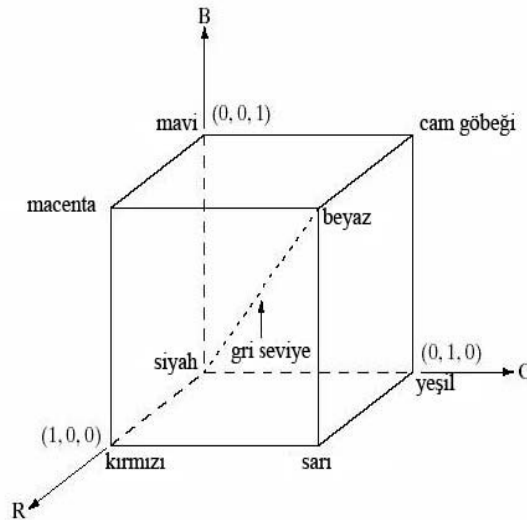
Bir renk sistemlerinin amacı, renklerin bir standartta belirlenmesini kolaylaştırmaktır. Esasında bir renk sistemi, bir koordinat sisteminin ve sistem içindeki her rengin tek bir nokta ile gösterildiği bir alt başlığın belirtilmesidir.

Monitör ve yazıcılar gibi donanımlar bir renk sistemi kullanırlar. Renk sistemindeki renklerin belirli oranda karışımlarından yeni renkler elde ederler. Dijital resim işleme yoluyla pratikte en yaygın olarak kullanılmakta olan donanıma yönelik modelleri RGB (Red, Green, Blue – Kırmızı, Yeşil, Mavi) sistemidir. Renkli monitörler ve geniş bir sınıf video kameralar için yaygın sistem CMY (Cyan, Magenta, Yellow – Camgöbeği, Eflatun, Sarı) sistemdir. CMYK (Cyan, Magenta, Yellow, Black – Camgöbeği, Eflatun, Sarı, Siyah) modelleri; renkli yazım için HSI (Hue, Saturation, Intensity – Renk özü, Doymunluk, Şiddet) modeli ki bu model insanın rengi tanımlama ve yorumlama yoluna yakındır. Ayrıca HSI sisteminin bir avantajı da resimdeki renk bilgisini ve gri-seviye bilgisini ayırır. Bu sayede bu sistemin çoğu gri-seviye teknikleri için uygun hale getirir.

3.1.7.1 RGB renk sistemi

Kartezyen koordinat sistemlerinde yaygın olarak kullanılan model RGB sistemdir. Her renk, kendi kırmızı, yeşil ve mavisinin ana spektrumsal bileşeni olarak görünür.

Bahsedilen renk alt boşluğu Şekil 3-8’de gösterilen küptür. Burada RGB değeri üç köşededir; camgöbeği, eflatun ve sarı diğer üç köşede; siyah orijinde ve beyaz orijinden en uzak köşededir. Bu modelde gri skala (eşit RGB değerleri noktaları) siyahtan beyaza kadar, bu iki noktayı birleştiren çizgi boyunca uzanır. Bu modeldeki farklı renkler küpün üstünde veya içindeki noktalardır ve orijinden çıkan vektörlerle tanımlanır.



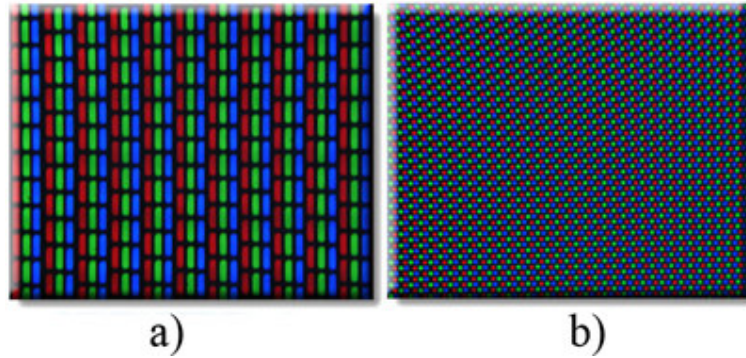
Şekil 3-8 Renk sistemi küpü

Resimler bu sistemde, kırmızı, yeşil ve mavi için üç ayrı bileşen resminden oluşur; her ana

renk için bir tane. Bir RGB monitörde bu üç resim birleşik bir renkli resim oluşturmak için fosfor ekran üzerinde birleşir.

Şekil 3-9'de bir görüntü oluşturmak için bir arada bulunan renk hücreleri görünmektedir. (a) 21" Katot Işın Tüplü (CRT) Televizyon ekranı , (b) 17" Katot Işın Tüplü (CRT) bilgisayar ekranıdır.

RGB sisteminde her pikseli temsil etmek için kullanılan bit sayısına "piksel derinliği" denir. Her kırmızı, yeşil ve mavi resmin bir 8 bit resim olduğu bir RGB resmi düşünün. Bu durumda her RGB renk pikseli (yeni bir RGB değerleri üçlüsü) 24 bit'lik bir derinliğe sahiptir (her yüzey için üç resim yüzeyi). "Gerçek renk" terimi genellikle bir 24 bit RGB renkli resmi göstermek için kullanılır. Bir 24 bit RGB resminde ki toplam renk sayısı $(2^8)^3 = 16.777.216$ 'dır.



Şekil 3-9 (a) Katot Işın Tüplü (CRT) televizyon ve bilgisayar ekranı

3.1.7.2 CMY ve CMYK renk modelleri

Camgöbeği, eflatun, sarı ışığın ara renkleridir veya başka bir deyişle pigmentlerin ana renkleridir. Örneğin camgöbeği pigmenti ile kaplanmış bir yüzey beyaz ışık ile aydınlatılınca, yüzeyden kırmızı ışık yansımaz. Yani camgöbeği, kırmızı ışığı, kırmızı, yeşil ve mavinin eşit miktarıyla oluşan yansıyan beyaz ışıktan çıkarır.

Yazıcılar ve fotokopiler gibi renkli pigmentleri kağıt üzerinde toplayan çoğu aygıtlar, içlerinde RGB' den CMY' ye dönüşüm için CMY data girişine ihtiyaç duyarlar. Bu dönüşüm şu basit işlem kullanılarak yapılır;

$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix}$	(3-17)
---	--------

Renkleri [0, 1] aralığında olduğu varsayılarak, Eşitlik (3-17)'den görülüyor ki saf camgöbeği ile hesaplanmış bir yüzeyden yansıyan ışık kırmızıyı içermez (yani denklemde $C = 1-R$): benzer şekilde, saf eflatun yeşili yansıtmaz ve saf sarı da maviyi yansıtmaz. Ayrıca eşitlik şunu ortaya çıkarıyor ki RGB değerleri, bir grup CMY değerleri kullanılarak kolayca elde edilebilir; her CMY değeri 1'den çıkarılır.

Genel olarak resim işlemede bu renk modeli, donanımsal çıktı elde etmek ile ilgili olarak kullanılmaktadır, dolayısıyla, RGB'nin tersi olan CMY dönüşüm işlemi pratikte pek kullanılmaz. Pigment analarının (camgöbeği, eflatun ve sarı) eşit miktarda karışımı siyahı oluşturur. Pratikte yazdırma işlemi için bunları birleştirirsek çamurumsu bir siyah ortaya çıkar. Dolayısıyla, gerçek siyahı (yazdırmada baskın renktir) elde etmek için CMYK renk modelini ortaya çıkaran, dördüncü bir renk, siyah, eklenir. Bu yüzden, üretici firmalar “dört-renkli yazımdan” bahsederken CMY'nin üç rengi artı siyahı kastetmektedirler.

3.1.8 HSI renk sistemi

Pratikte insanın algıladığı renkleri tanımlamada RGB ve CMY sistemleri tam uygun değildirler. Çünkü insanlar bir nesnenin rengini belirtirken, onun rengini oluşturan ana renklerin yüzdelerini vermeleri zordur. Bunun yerine renkleri nesnenin renk özü, doygunluk ve parlaklığı ile tanımlarız.

Renk özü (hue), saf rengi (saf sarı, turuncu veya kırmızı) tanımlayan bir renk özelliğidir. Doygunluk ise bir saf rengin beyaz ışıkla sulandırılma derecesinin ölçüsünü verir. Parlaklık pratikte ölçülmesi imkânsız olan subjektif bir tanımlayıcıdır. Işık şiddetinin renksiz ölçüsünü oluşturur ve renk hissini tamamlayan anahtar faktörlerden biridir. Işık şiddeti (gri seviyesi) tek renkli resimlerin en kullanışlı tanımlayıcılarından biridir. Bu özellik tam olarak ölçülebilir ve yorumlanabilir.

HSI renk sisteminde, bir renkli resimdeki şiddet bileşenini, rengi taşıyan bilgidan (renk özü ve doygunluk) ayırır. Sonuç olarak HSI modeli insanlar için doğal ve sezgisel olan renk tanımlamaları üzerine kurulu resim işleme algoritmaları geliştirmek için ideal bir araçtır. Şu şekilde özetlenebilir; RGB resim rengi üretmek için idealdir (renkli kamerayla resim

yakalama veya renkli monitör ekranında resim görüntülemeye ki gibi) fakat rengi tanımlamak için kullanımı daha sınırlıdır.

3.1.9 Renkleri RGB'den HSI'ya çevirme

RGB renk formatında verilen bir resimde, her RGB pikselinin H bileşenini elde etmek için Eşitlik (3-18) kullanılır;

$$H = \begin{cases} \theta & \text{eğer } B \leq G \\ 360 - \theta & \text{eğer } B > G \end{cases} \quad (3-18)$$

Burada θ Eşitlik (3-19) den elde edilir.

$$\theta = \cos^{-1} \left\{ \frac{\frac{1}{2}[(R-G) + (R-B)]}{\left[(R-G)^2 + (R-G)(G-B) \right]^{1/2}} \right\} \quad (3-19)$$

Doygunluk

$$S = 1 - \frac{3}{(R+G+B)} [\min(R, G, B)] \quad (3-20)$$

Işık şiddeti,

$$I = \frac{1}{3} [R+G+B] \quad (3-21)$$

dır. RGB değerlerinin [0,1] aralığında ve θ açısının HSI olduğu kırmızı eksenine bağlı olarak ölçüldüğü varsayılmıştır. Renk özü, Eşitlik (3-18)'dan çıkan bütün değerlerin 360° ile bölünmesi ile [0, 1] aralığında oluşturulabilir. Diğer iki HSI bileşeni, eğer verilen RGB değerleri [0, 1] aralığında ise, zaten bu aralık içindedirler.

3.1.10 Renkleri HSI 'den RGB 'ye çevirme

HSI 'nın [0, 1] aralığında verilen değerleri için, aynı aralığa karşılık gelen RGB değerlerini bulmak amacıyla uygulanabilir denklemler H 'nin değerlerine dayanır. Ana renklerin ayrımındaki 120° lik aralıkları gösteren, üç sektör vardır. Öncelikle H 360 ile çarpılarak renk özü orijinal aralığa olan [0°, 360°] aralığına döndürülür [Gonzalez R.C. 2002].

RG Bölgesi ($0^\circ \leq H < 120^\circ$): H bu bölgede iken RGB bileşenleri ,

$$B = I(1 - S) \quad (3-22)$$

$$R = I \left[1 + \frac{S \cos H}{\cos(60^\circ - H)} \right] \quad (3-23)$$

$$G = 3I - (R + B) \quad (3-24)$$

GB Bölgesi ($120^\circ \leq H < 240^\circ$): H bu bölgede ise;

$$H = H - 120^\circ \quad (3-25)$$

dır. Bu durumda RGB bileşenleri Es. (3-26) - Es. (3-28) ile elde edilir.

$$R = I(1 - S) \quad (3-26)$$

$$G = I \left[1 + \frac{S \cos H}{\cos(60^\circ - H)} \right] \quad (3-27)$$

$$B = 3I - (R + G) \quad (3-28)$$

BR Bölgesi ($240^\circ \leq H < 360^\circ$): H bu bölgede ise;

$$H = H - 240^\circ \quad (3-29)$$

dır. Bu durumda RGB bileşenleri Eşitlik (3-30) - Eşitlik (3-32) ile elde edilir.

$$G = I(1 - S) \quad (3-30)$$

$$B = I \left[1 + \frac{S \cos H}{\cos(60^\circ - H)} \right] \quad (3-31)$$

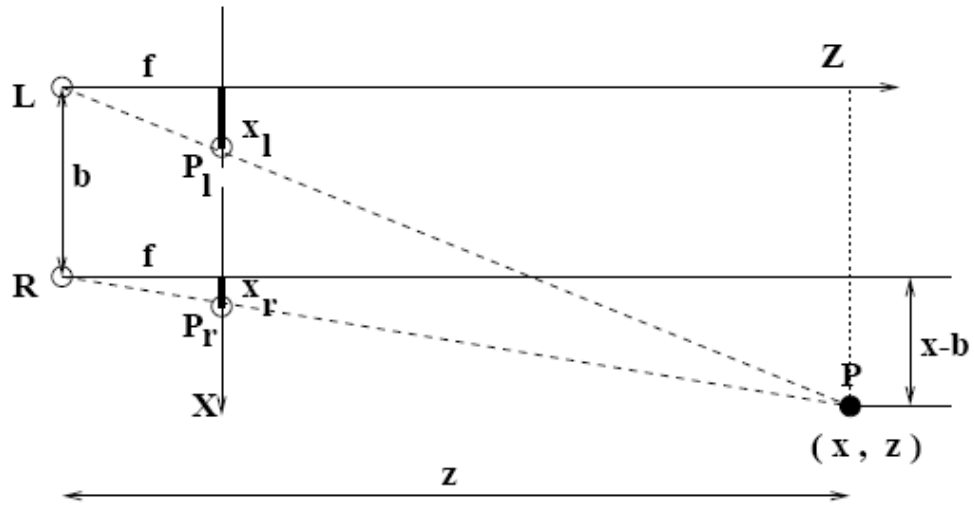
$$R = 3I - (G + B) \quad (3-32)$$

4. STEREO GÖRME

4.1 Mesafe Tahmini

Yalnızca tek bir resimdeki cisimler arasındaki mesafenin algılanması zordur. Resimde ki cisimler arasındaki mesafeyi algılamak için birden çok görüntüye ihtiyaç vardır.

Stereodan bir resimden uzaklık bulmak için üçgen benzerliği kuralını kullanırız. Kameraların X eksenine paralel ve aynı doğrultuda ve Y ve Z eksenlerine paralel olması gerekir. Aşağıdaki resmi inceleyelim:



Şekil 4-1 Stereo sistemin geometrik yapısı

Yukarıdaki geometrik yapıda Y eksenine sayfa dik. Orijini L olarak kabul edelim P noktasını ise (x,z) koordinatında olsun. Sistem P noktasını P_l ve P_r noktalarında algılıyor olsun. Geometrik olarak diyebiliriz ki LP_l doğrultusundan çıkan bir ışın ile RP_r doğrultusundan çıkan ışınlar P noktasında kesişirler.

Üçgen benzerliği kuralından diyebiliriz ki:

$$z = \frac{x}{x_l} f \quad (4-1)$$

$$z = \frac{(x-b)}{x_r} f \quad (4-2)$$

$$\frac{z}{f} = \frac{y}{y_l} = \frac{y}{y_r} \quad (4-3)$$

Geometrik yapı gereği y_l ve y_r eşitlerdir. Bu ifadelerden P noktasının koordinatlarını oluşturmak istersek:

$x = \frac{x_l z}{f} = b + \frac{x_r z}{f}$	(4-4)
$y = \frac{y_l z}{f} = \frac{y_r z}{f}$	(4-5)
$z = \frac{fb}{(x_l - x_r)} = \frac{fb}{d}$	(4-6)

Yukarıda kullanılan d , fark (disparity) olarak isimlendirilir ve x_l ile x_r farkını temsil eder. Fark (disparity), stereo kameralarda 3Boyutlu bir noktanın görüntü düzleminde olduğu yerlerin arasındaki mesafe farkıdır. P noktasının uzaklığı arttıkça d azalır, uzaklık azaldıkça d artar. Bu durumda diyebiliriz ki yakındaki cisimlerin yer değiştirmesi fazla uzaktakilerin azdır.

Sadece bir P noktası bulunduğundan P noktasının oluşturduğu P_l ve P_r noktalarının yerlerini bilebiliyoruz ve aralarındaki farkı (disparity) rahatlıkla söyleyebiliyoruz. Gerçek bir resimde ise P gibi birçok nokta bulunur. Bu durumda hangi noktanın diğer resimdeki nokta ile eşleşeceğini bilmemiz gerekir. Çok dikkatli ayarlanmış bir stereo kamerada genelde ilişkili noktalar aynı satır üzerinde bulunurlar. Burada karşılaşılabilecek problemlerden biri bir karakteristik noktanın diğer resimde karşılığının olmaması, diğer bir problem ise bir nokta için diğer resimde birden çok nokta bulunması. Belirli bir doku (texture) kaplı objelerde bulunan karakteristik nokta sayısı yüksekken, düz dokulu objelerde daha az bulunmaktadır.

4.1.1 Karakteristik (Feature) Noktaları Bulma

Birçok karakteristik nokta bulma metodu mevcuttur. Bu metotlar 3 kategoride gruplanabilir: kontur-tabanlı (contour-based), sinyal-tabanlı(signal-based), şablon-uydurma(template-fitting) metodu.

Kontur-tabanlıda eğriler veya bükümler araştırılır. Alternatif olarak bir çokgeni oluşturan çizgilerin kesişim noktası kabul edilebilir. Harris Operatörü bu sınıfa girer.

Sinyal-tabanlı metodun da frekans analizinden bulunur.

Şablon-uydurma metodunda tepe gibi parametrik modelleri aranır.

4.1.2 Harris Operatörü

Karakteristik nokta bulma işleminde resmin dönmüş olması, parlaklığının artmış/azalmış olması durumlarında dahi doğru sonuçlar vermesi gerekir. Harris detektörü bu durumlara uygun doğruluğu veren metotlardan biridir.

Harris karakteristik nokta detektörü Moravac detektörünün geliştirilmesiyle oluşturulmuştur. Moravac, $(x, y) \in Z^2$ uzaysal (spatial) koordinat da I şiddet (intensity) olmak üzere bir bölgenin özelliğini Eşitlik (4-7) formülü ile değerlendirmiştir:

$$E(x, y) = \sum_{(u,v) \in P} w(u, v) (I(u+x, v+y) - I(u, v))^2 \quad (4-7)$$

Formülde ki $w(u, v)$ penceresinin ağırlığıdır.

Harris daha genel bir pencere kullanmıştır ve Taylor açılımını $I(u+x, v+y)$ ye uygulamıştır. Sonuç olarak:

$$E(x, y) \approx \sum_{u,v} w(u, v) (xI_x(u, v) + yI_y(u, v))^2 \quad (4-8)$$

Eşitliğin sağ tarafı $[x \ y]M[x \ y]^T$ şeklinde yazalım. M,

$$M = \begin{pmatrix} A & B \\ B & C \end{pmatrix} \quad (4-9)$$

Ve

$$A = \sum_{u,v} w(u,v) I_x^2(u,v), \quad (4-10)$$

$$B = \sum_{u,v} w(u,v) I_x(u,v) I_y(u,v), \quad (4-11)$$

$$C = \sum_{u,v} w(u,v) I_y^2(u,v), \quad (4-12)$$

dır. I_x ve I_y , I 'nın türevleridir. Türev almak için,

$$I \text{ resmini, sırasıyla } \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \text{ ve } \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} \text{ ile konvolüsyonu alınır}$$

$E(x,y)$ sayesinde ilgili (x,y) koordinatının karakteristik nokta olup olmadığına karar verilebilir.

Farz edelim ki, M matrisinin özdeğerleri λ_i , $i=1,2$ olsun. Şu üç durumu dikkate alalım:

1. λ_1 ve λ_2 küçük ise, uzaklıktan bağımsız olarak E muhakkak küçük olmalıdır. Bu durumda bölge tekdüze (uniform).
2. Özdeğerlerden biri küçük, diğeri büyük ise, yer değiştirmeye bağlı olarak büyük özdeğer, büyük E oluşmasına neden olur. Bu durumda bölgede kenar algılanmıştır.
3. Özdeğerlerden ikisi de büyük ise, E muhakkak büyük olmalıdır. Bu durumda bölge de köşe (karakteristik nokta) algılanmış olur.

Bu üç durumu ayırt etmek için Eşitlik (4-13) 'ı M ye uyguluyoruz:

$R = \text{Det}(M) - k(\text{Trace}(M))^2 = \lambda_1 \lambda_2 - k(\lambda_1 + \lambda_2)^2$	(4-13)
---	--------

k pozitif çok küçük bir sabittir. Denemeler sonucunda 0.04 ile 0.06 arası alınması uygun görünmüştür.

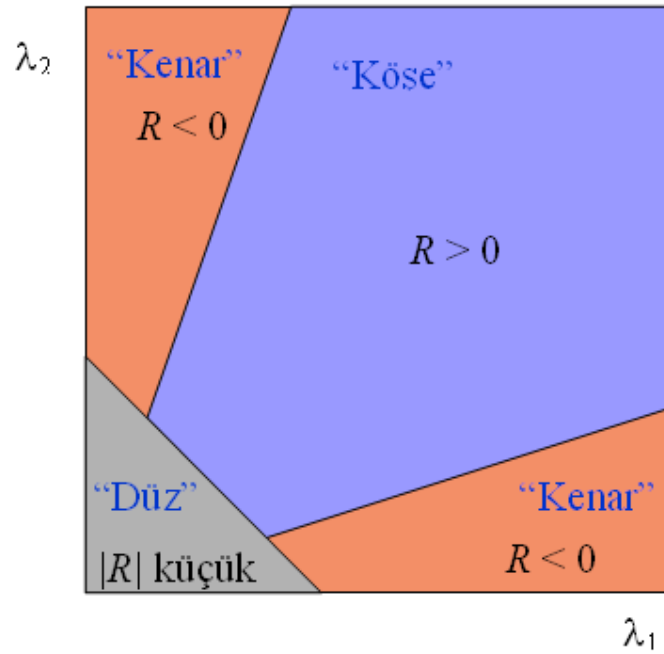
Küçük özdeğerler için (birörnek) R nin sıfıra yaklaştığı görülüyor, biri büyük biri küçük özdeğerler için ise (kenar) R büyük negatif değer alır, büyük özdeğerler için ise (karakteristik nokta) R büyük pozitif değer alır.

Bu tezin yazılım kısmında özdeğerlerini bulmaya gerek kalmadan da

$Trace(M) = \lambda_1 + \lambda_2 = A + C$ ve $Det(M) = \lambda_1 \lambda_2 = AC - B^2$ eşitliklerini kullanarak algoritmamızı daha basitleştirilmiştir.

Karakteristik noktaları bulmak için R nin büyük değerlerini bulmamız yeterli olacaktır. Bunun için aşağıdaki işlemleri sırayla gerçekleştiririz.

1. Tüm $R(x, y)$ leri $I(x, y)$ içerisinde bulunur.
2. Bir bölge seçilir. Bu en büyük $R(x, y)$ lerini maksimum filtre yardımıyla bulunur. Bulunan maksimum a R_{\max} denir. Belirli bir eşik değerini θR_{\max} şeklinde gösterilir. Eşik değerindeki θ pozitif bir sabit olmalıdır. θR_{\max} 'dan büyük R_{\max} belirlenir.
3. 3x3 lük bölgelerdeki R_{\max} lar bulunur. Bulunan R_{\max} , θR_{\max} dan büyük ise karakteristik nokta kabul edilir.



Şekil 4-2 R nin aldığı değerler göre bölgenin sınıflandırılması

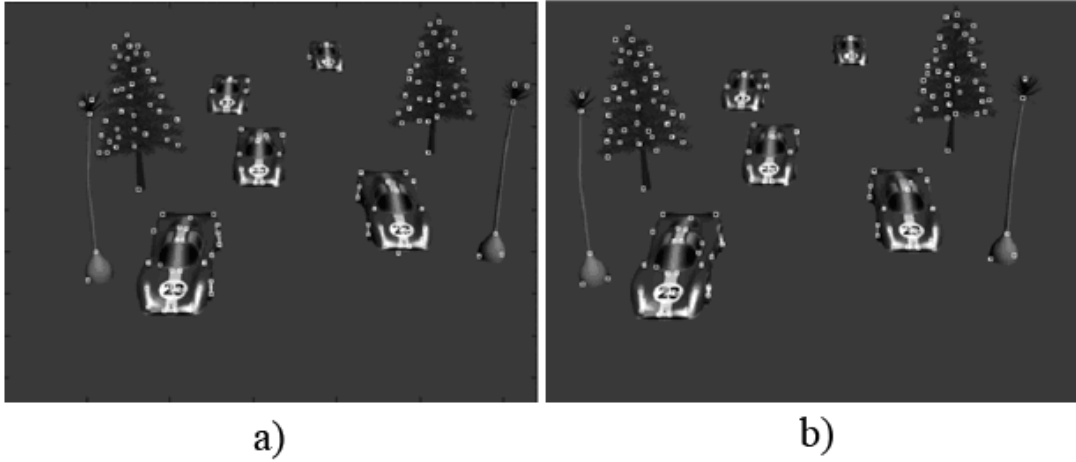
Pencere fonksiyonu olarak genellikle Gaussian formu kullanılır.

$w(u, v) = e^{-\frac{u^2+v^2}{2\sigma^2}}$	(4-14)
--	----------

Projenin yazılım kısmında k sabitini 0.06, ve eşik değeri $\theta R_{\max} = 0.001$ alındı. Bulunan

karakteristik noktalar dokular ve obje kenarlarında oluşmuştur. Eşik noktası bulunan karakteristik nokta sayısı etkiler. Büyük eşik değerinde daha az karakteristik nokta bulunur ve daha doğru sonuçlar elde edilir, düşük de karakteristik nokta sayısı artar.

Sağ ve sol resimlerin karakteristik noktalarının bulunmuş hali aşağıdaki resimlerdeki gibidir:



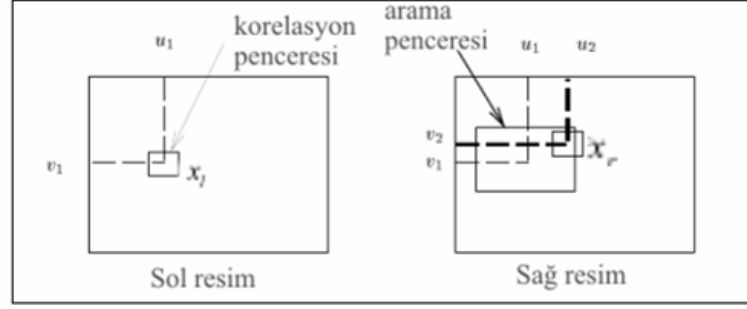
Şekil 4-3 Karakteristik noktaları bulunmuş sol ve sağ resimler

4.1.3 Harris Operatörünün Özellikleri

Harris operatörü ile köşe bulurken resmin dönmüş olması durumunda dahi köşe noktalarını doğru bir şekilde bulabilir. Ayrıca şiddetin (Intensity) türevi üzerinden işlemler yapıldığı için şiddetteki küçük farklarda sonuç değişmemektedir.

4.2 Karakteristik Noktaların Eşleştirilmesi

Sağ ve sol resimlerdeki karakteristik noktaları eşleştirmek için geniş bir arama yapılır. Sağ resimde bulunan her bir karakteristik nokta sol resimdeki karakteristik nokta ile benzerliği araştırılır. Benzerlik seviyesi uygun ise iki nokta eşleştirilir. Bir nokta için birden fazla eşleştirme olması durumunda en büyük skorlu eşleştirme kabul edilir.



Şekil 4-4 Korelasyon penceresi ve arama penceresi

$x_l = (x_l, y_l)$ ve $x_r = (x_r, y_r)$ koordinatları sırasıyla sağ ve sol resimlerdeki karakteristik noktalar için arama yapılır. x_l noktası için $(2n+1) \times (2m+1)$ boyutlarında merkezi x_l noktası olan korelasyon penceresi kullanılır. Daha sonra ikinci resimde $(2d_u + 1) \times (2d_v + 1)$ boyutlarında bir arama penceresi içerisinden sağ ve sol resimlerdeki karakteristik noktalar örtüştürülür.

Arama penceresi, arama bölgesini küçültür bu sayede daha az işlem yükü getirmiş olur. Aynı zamanda arama penceresi örtüşen noktaların birbirine belli bir yakınlıkta olmasını sağlar. Buda örtüşmenin doğruluğunu artırır.

Örtüşmeni doğruluğu Eşitlik (4-15) deki skor formülle hesaplanır:

$$Skor(x_l, x_r) = \frac{\sum_{i=-n}^n \sum_{j=-m}^m [I_1(u_1 + i, v_1 + j) - \overline{I_1(u_1, v_1)}] \times [I_2(u_2 + i, v_2 + j) - \overline{I_2(u_2, v_2)}]}{(2n+1)(2m+1) \sqrt{\sigma^2(I_1) \times \sigma^2(I_2)}} \quad (4-15)$$

Formüldeki noktasındaki $\overline{I_k(u, v)}$ ortalamadır ve ,

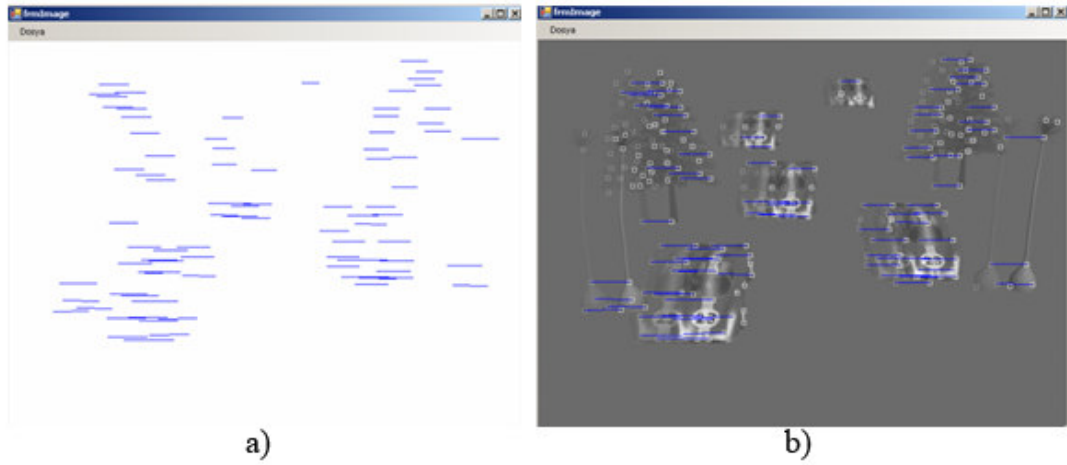
$$\overline{I_k(u, v)} = \frac{\sum_{i=-n}^n \sum_{j=-m}^m I_k(u + i, v + j)}{[(2n+1)(2m+1)]} \quad (4-16)$$

Eşitlik (4-16) formülüyle bulunur. $\sigma(I_k)$ korelasyon penceresinin standart sapmasıdır ve,

$$\sigma(I_k) = \sqrt{\frac{\sum_{i=-n}^n \sum_{j=-m}^m I_k(u+i, v+j)}{[(2n+1)(2m+1)]} - I_k(u, v)} \quad (4-17)$$

dır. Skor -1 ile 1 arasında değişen bir sonuç verir. Eğer sağ ve sol resimlerdeki korelasyon penceresi aynı ise skor 1 çıkar. 2 noktayı eşleştirmek için belli bir eşik değerinin üstündeki skorları kabul ederiz.

Tezin yazılım kısmında eşik değeri 0.95 alınmıştır. Korelasyon penceresinin parametreler $n=m=7$ alınmıştır. Arama penceresi ise orijinal resmin enin ve boyunun çeyreği büyüklüğünde alınmıştır.

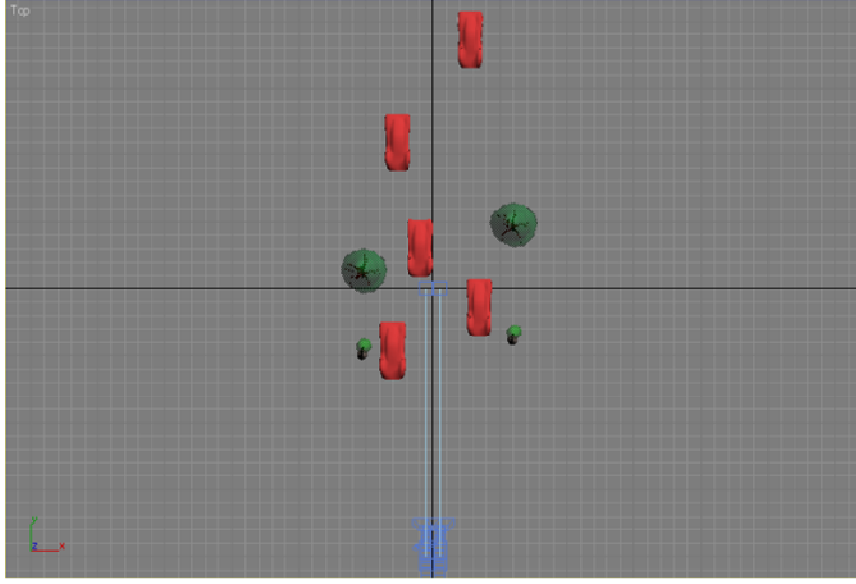


Şekil 4-5 Eşleşen noktaların gösterimleri

4.2.1 Eşleştirmenin Özellikleri

Harris operatörü ile karakteristik noktaları bulduk daha sonra iki resimde bunları eşleştirdik. Sol ve sağ resimleri eşleştirerek, objelerin kameraya uzaklığı hakkında bilgi edinebiliriz.

Uzaklık bilgisinin doğru sonuç vermesi için çok hassas davranılmalıdır. Doğruluğun yüksek olması için, sağ ve sol resimlerini çeken kameraların aynı x ekseninde ve birbirlerine yakın olması gerekir. 3DSMax programında hazırlanan yarisL.pgm ve yarisR.pgm resimlerini üstten görünüşü aşağıdaki resimde verilmektedir. Kameraların bir birine yakınlığı ve paralellığı görülebilmektedir.



Şekil 4-6 3D Studio Max ile hazırlanmış projenin üstten görünüşü

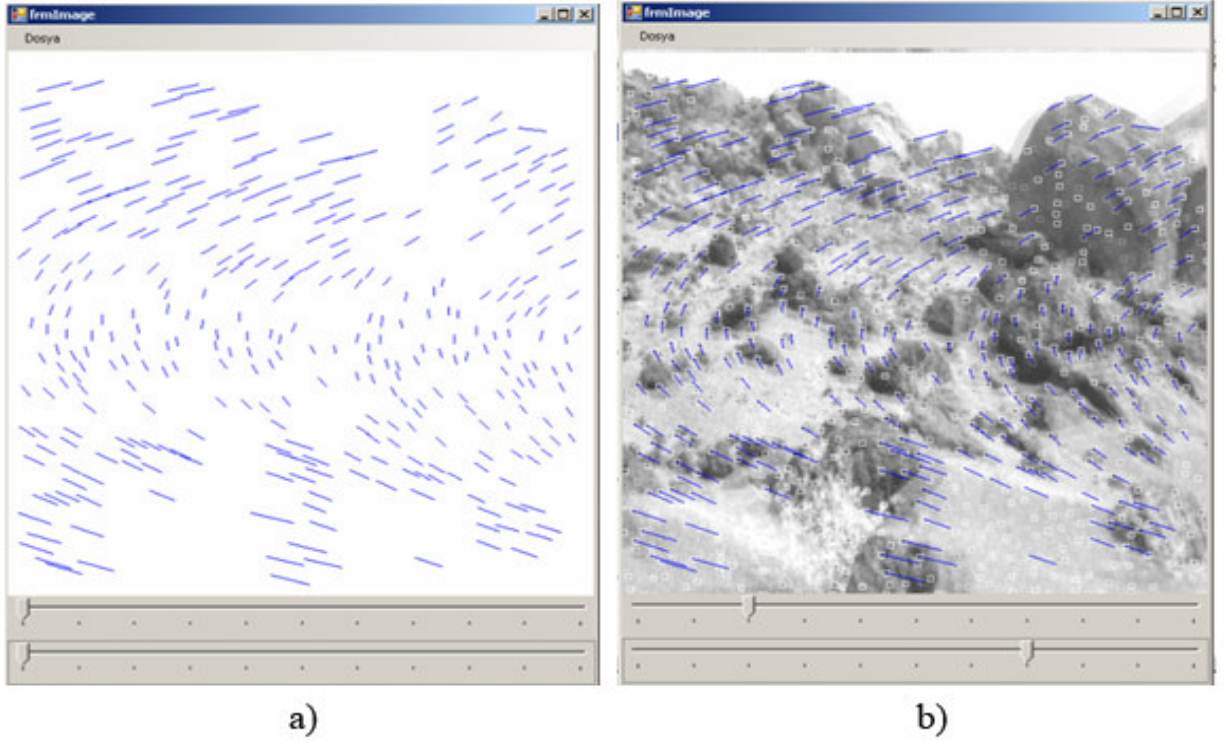
Aşağıdaki resimde kameralardan biri diğerine göre bir miktar dönmüş durumda. Karakteristik noktaları eşleştirerek döndürme miktarı hakkında bir fikir edine biliriz.



a)

b)

Şekil 4-7 Sol ve Sağ resimler

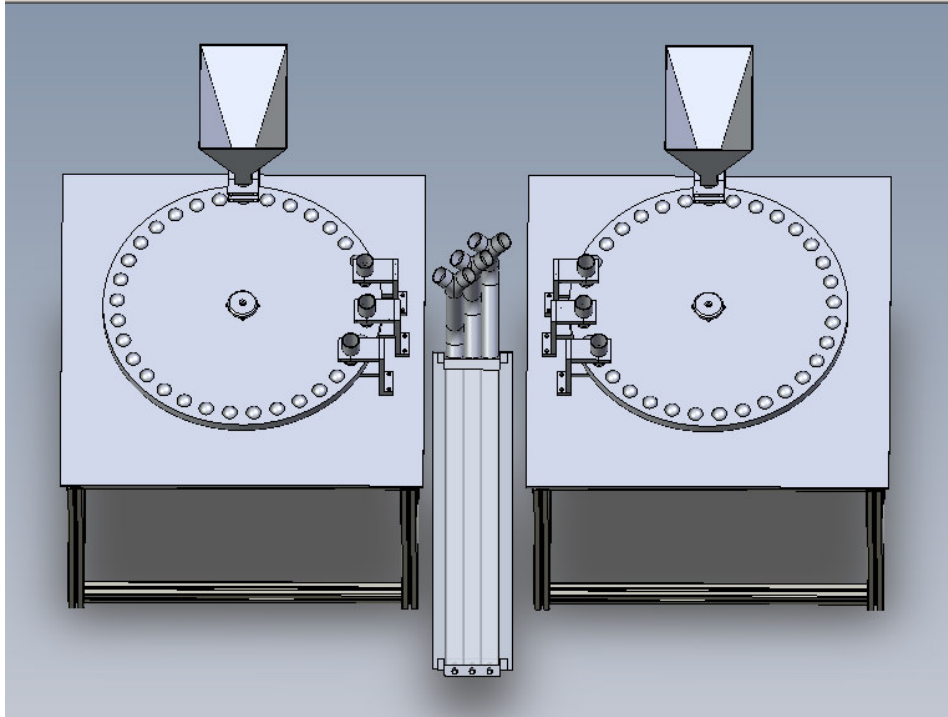


Şekil 4-8 Eşleştirilmiş noktaların görünümü

5. UYGULAMA DÜZENEGİ

5.1 Mekanik Aksamı

Uygulama düzeneğinin mekanik aksamı CNC tezgâhında hazırlandı. Birbirine yapışık iki konveyör bandı bulunmaktadır. Bir konveyör bandı üzerinde; dışarıdan topların eklenebileceği bir depo, 33 adet pinpon topunun sığabileceği plaka, topları dış ortama çıkışını sağlayacak üçlü boru yapısı, kamera ayağı, motorun ve pistonların bağlanacağı düzenekler, yüksekte durması için gereken ayakları bulunmaktadır. Mekanik aksamın 3 Boyutlu çizimi Şekil 5-1 de verilmiştir. Pinpon topların konduğu plakalar 80 cm çapındadır.



Şekil 5-1 3B uygulama düzeneği çizimi

5.2 Elektrik Aksamı

5.2.1 Doğru Akım Servo Motoru

Pinpon topların dizildiği 80 cm çapındaki plakaları döndürmek ve hassas bir şekilde durdurmak için Doğru Akım Servo Motoru tercih edilmiştir.

D.A. Servo Motorları çok küçük güçlerden çok büyük güçlere imal edilirler (0.05 HP den 10000 HP arasında) çeşitli Bu motorlar klasik D.C. motorlar gibi imal edilirler. Proje

kapsamında kullanılan Servo motorlar Phenix Contact markadır. Endüvileri (yükseklik . uzunluk / Çap oranıyla) kutup atalet momentini minimum yapacak şekilde tasarlanırlar. Küçük çaplı ve genellikle içerisinde kompanzasyon sargısı olan, kuvvetli manyetik alanı boyu uzun doğru akım motorlarına da servo motor denir.

D.A. servo motor çalışma prensibi açısından aslında, Statoru Daimi Mıknatıs bir D.C. motordur. Manyetik alan ile içinden akım geçirilen iletkenler arasındaki etkileşim nedeniyle bir döndürme momenti meydana gelir. Bu döndürme momenti manyetik alan vektörü ile sargı akım vektörü arasındaki açı 90° olduğunda maksimum değer alır.

Bazı D.C. Servomotorların çok küçük zaman sabitleri vardır. Düşük güçlü D.C. servomotorlar piyasada genellikle bilgisayar kontrollü cihazlarda (disket sürücüler, teyp sürücüler, yazıcılar, kelime işlemciler, tarayıcılar vs.) kullanılırlar. Orta ve büyük güçlü servomotorlar ise sanayide genellikle robot sistemleri ile sayısal denetimli hassas diş açma tezgâhlarında kullanılır.

Servo motor sürücüler, genellikle DIN normlarına uygun, kontrol akımı ve motor durumu belirten göstergelere sahip, otomatik ve elle kumanda olanağı sağlayan tamamen katı hal (Solid state) tekniği ile tasarlanmış elektronik cihazlardır.

5.2.2 PLC

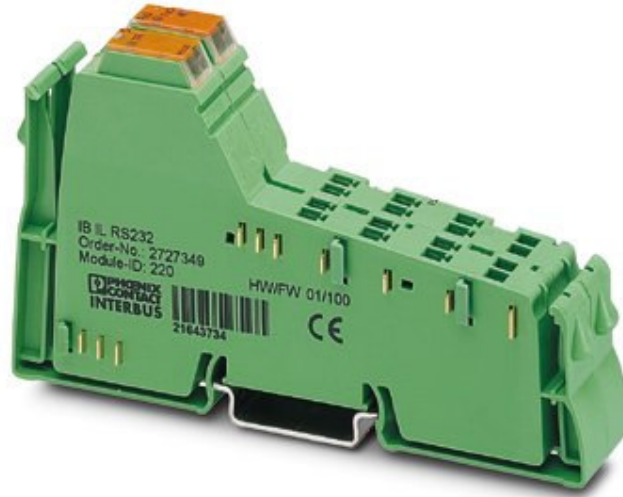
Programlanabilir Lojik kontrolör olarak Phoenix Contact firmasının ürettiği ILC 150 ETH tercih edilmiştir. Üretimi yeni başlayan bu ürünün, Ethernet entegre edilmiş kompakt PLC 'dir. PC WorX yazılımı ile konfigüre edilir ve programlanır. Entegre edilmiş iki portlu switch ile Ethernet ve TCP/IP tabanlı iletişim sağlanmıştır. AX OPC Server kullanılarak her türlü Scada (Veri toplama ve Denetimsel Kontrol, Supervisory Control And Data Acquisition) programları ile Ethernet üzerinden hızlı ve kolay bağlantı yapmak mümkündür.



Şekil 5-2 ILC 150 ETH

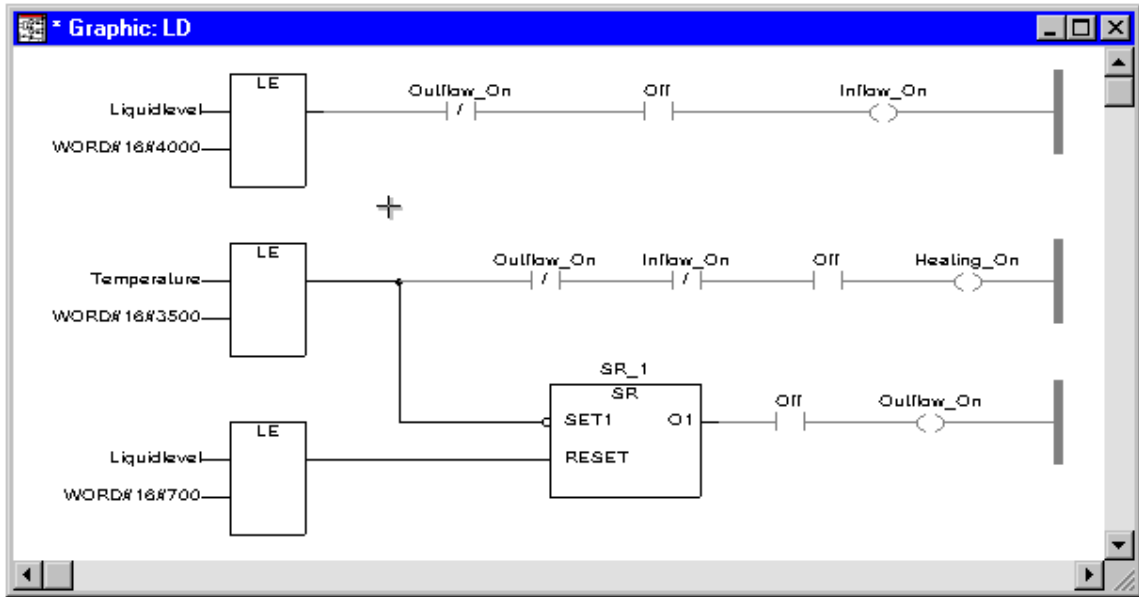
Üzerine entegre 8DI ve 4DO yanı sıra 63 modüle kadar genişleyebilen ILC 150 ETH ile dağıtılmış kontrol noktaları oluşturulabilir. 128 kbyte programlama hafızası ve 2.1ms den daha az bir sürede 1 K komutu işleme kapasitesine sahip olan ILC 150 ETH kendi sınıfında en düşük fiyat performans oranına sahiptir.

ILC 150 ETH ürünü entegre edilen IB IL RS 232 kablolu haberleşme modülü ile beraber kullanılmıştır. Bilgisayara ve PLC ye bağlanan RS-232 kablosu sayesinde; PLC den servo motor durdu komutu alındı, PLC ye topun rengi bilgisi gönderildi.



Şekil 5-3 IB IL RS 232

PLC yazılımını PC WORX ile ladder (merdiven) diagramında hazırlandı.



Şekil 5-4 PC Worx Merdiven Diagramı

6. DENEYSEL SONUÇLAR

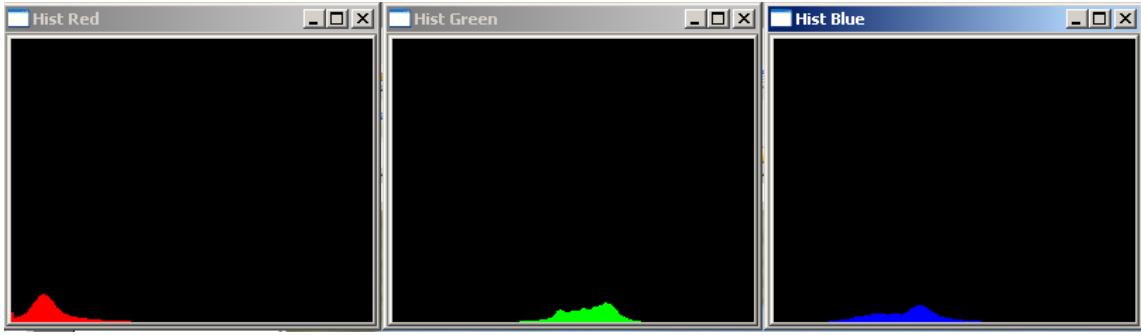
Hazırlanan düzenekte 3 ana renk ve 3 ara renk olmak üzere 6 ayrı renk pinpon topu kullanılmıştır.

Yeşil renk pinpon topunun bilgisayar kamerasından görünümü Şekil 6-1 Yeşil topun görüntüsü Şekil 6-2 de görülmektedir. Topun rengini bilgisayar ortamına aktarmak için yazılama fare yardımı ile topun bulunduğu alan çerçeve içine alındı.



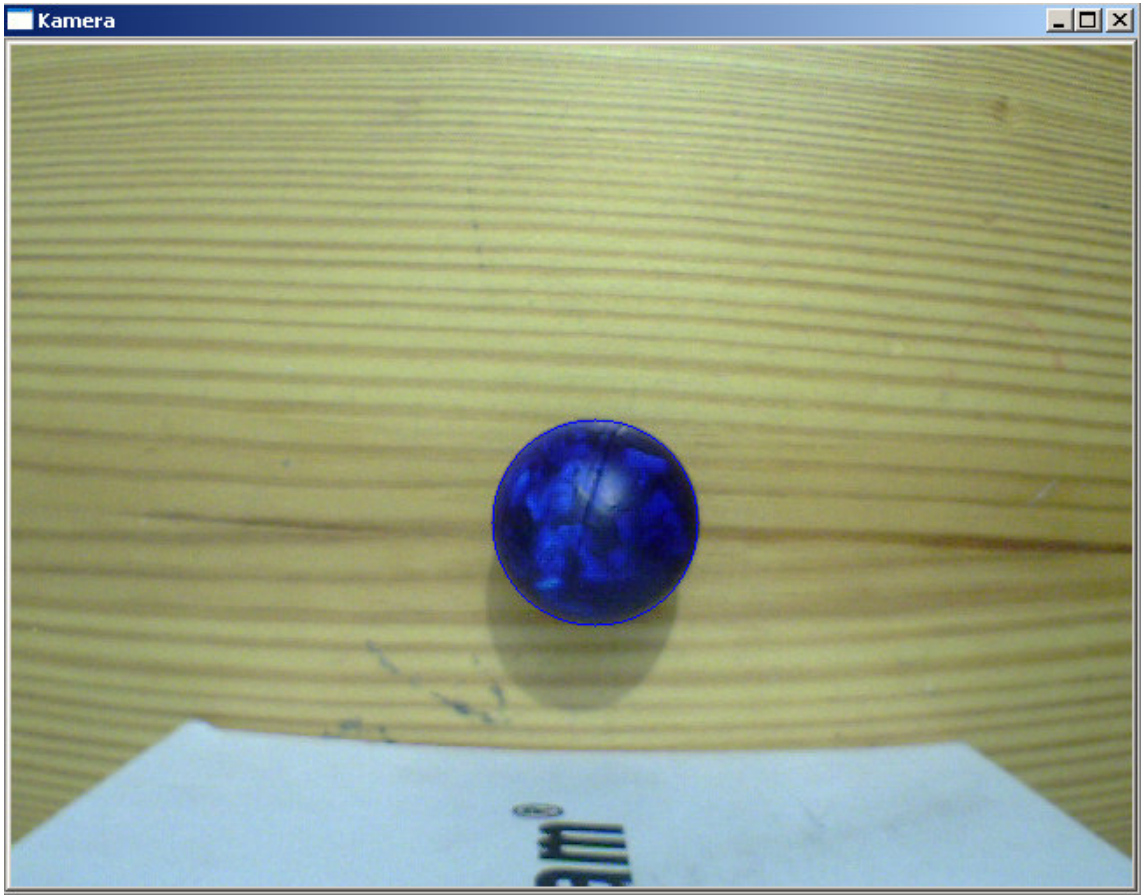
Şekil 6-1 Yeşil topun görüntüsü

Mavi çerçeve bölgesinin histogramına ayrılmış görüntüsü Şekil 6-2 görülmektedir. Şekilden yeşil histogramın diğer renklere nazaran daha yoğun olduğu görülmektedir. Kırmızı renk çok az içerirken, mavi renk bir miktar görülmektedir. Buradan çıplak gözle yeşil kabul ettiğimiz rengin içerisinde mavi renkleri de barındırdığı görülmektedir.



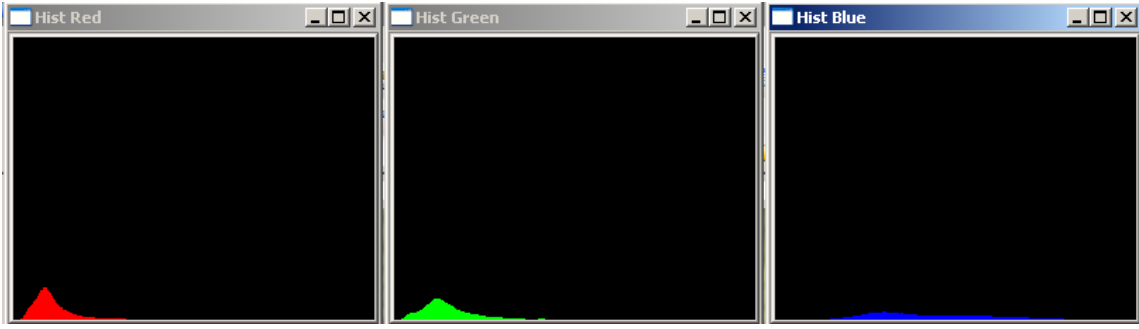
Şekil 6-2 Yeşil topun histogram görüntüsü

Mavi renk için ise Şekil 6-3 de görünmektedir.



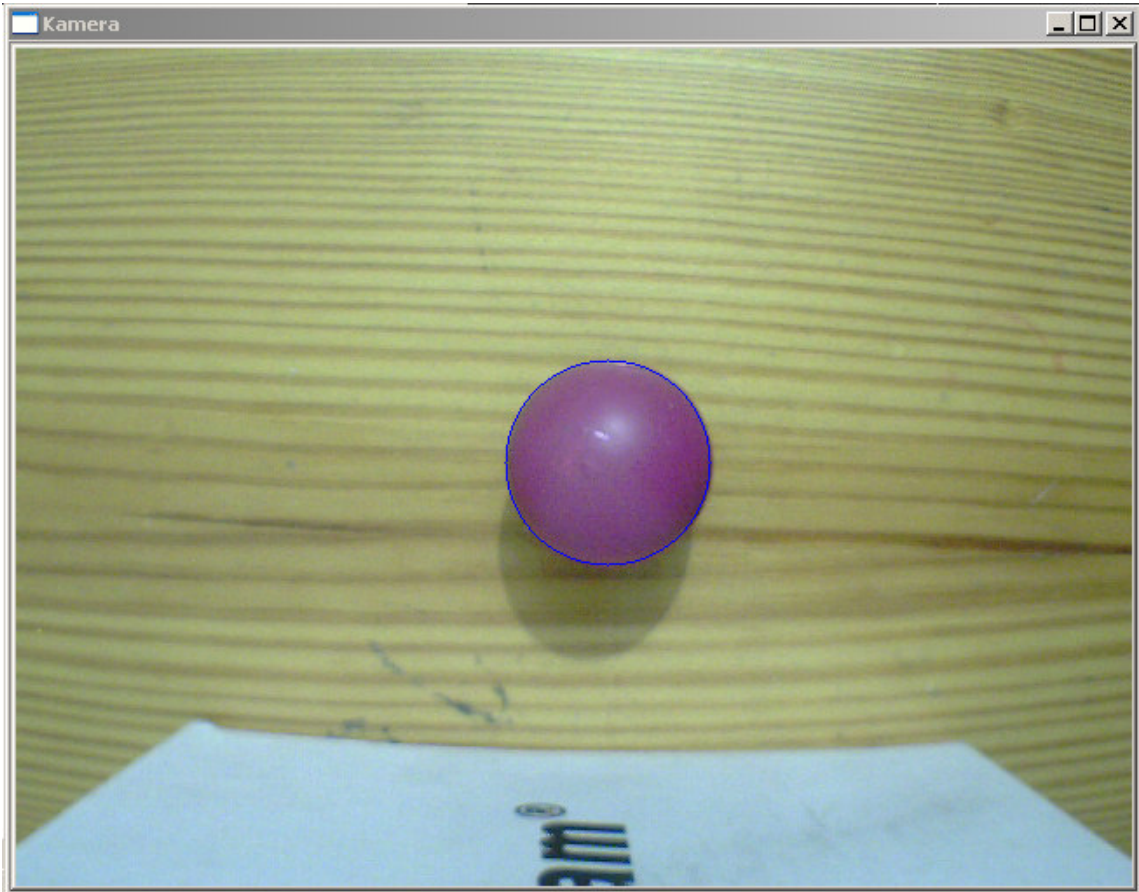
Şekil 6-3 Mavi topun yazılımdaki görüntüsü

Mavi renk için çıkan histogram değerlerinden kırmızı ve yeşil rengin çok az bulunduğu görülmektedir. Mavi renk ise geniş bir skalada bulunmaktadır. Bunun nedeni ise pinpon topunun boyasının homojen bir dağılım gösterememesidir.



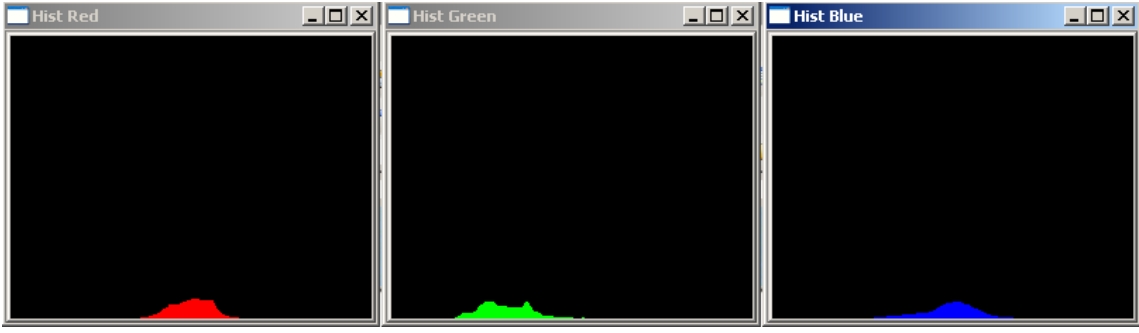
Şekil 6-4 Mavi topun histogram değerleri

Magenta renk topun görüntüsü Şekil 6-5 görülmektedir.



Şekil 6-5 Magenta topun görüntüsü

Magenta ara bir renktir; kırmızı ile mavinin karışımından oluşmaktadır. Bu yüzden yeşil renk biraz daha az olmak üzere yoğun olarak kırmızı ve mavi renk histogramdan görülebilmektedir. Yeşil rengin hafif yoğun bir miktarda bulunması pinpon topunun renginin açık magenta olduğunu göstermektedir.



Şekil 6-6 Megenta topun histogramı

7. SONUÇLAR VE ÖNERİLER

Bu çalışmada görüntü sistemi donanımları ile elde edilen görüntülerden, hazırlanan yazılım ile görüntü işleme teknikleri kullanılarak ilk önce stereo kamera ile mesafe algılama işlemi, sonrasında CMUCAM V2+ ile renk algılama işlemi yapıldı.

Uygulamanın ilk bölümünde kurulan düzenekle, tıpkı insanın iki gözü ile mesafeyi algılaması gibi iki adet sayısal kamera kullanılarak mesafe algılanmıştır. Bilgisayar ortamına aktarılan ikili görüntüler üzerinde aranan belirli renkteki cismin üzerinde özellik noktası (feature point) yardımıyla elde ettiğimiz iki resimdeki cismin kayma miktarı bize kameranın cisme olan uzaklığı ile orantılı bir değer elde ederiz. Cisim hızlı akan bir düzlem üzerinde bulunduğundan işlemlerin hızlı yapılmasını sağlayan akıllı algoritmalar kullanılmıştır. Ayrıca programlama dili olarak C++ kullanılmıştır.

Renkli görüntü üzerinde aranan cisim bulduktan sonra görüntüler 256 gri düzeye dönüştürülmüştür. 256 gri düzey kullanımında ana nedeni daha yüksek renk seviyesi kullanmanın sonuçlar üzerinde önemli bir değişiklik yapmamasıdır. Bu çalışmada renk seviyesini (miktarlama) yükseltmek yerine çözünürlüğü (örnekleme) artırmak sonuçların daha doğru olmasını sağladığı özellikle birinci bölüm uygulamada gözlenmiştir. Yapılacak uygulamaya (havacılık-uzay, tıp, malzeme, nesne tanıma) göre seçilecek renk derinliği ve çözünürlük, görüntü içerisindeki detaylara bağlı olarak farklılık göstermektedir.

Daha etkin bir sonuç alabilmek için aydınlatma çok büyük bir önemi vardır. Her ne kadar görüntülerin karşılaştırılmasında kullanılan algoritma ışığa az duyarlı olsa da, gereğinden fazla aydınlık veya karanlık görüntünün düzgün işlenememesine neden olabilmektedir.

Kameraların aynı model olmasına dikkat etmek de çok önemlidir. Farklı model kameraların CCD entegre boyutu farkı, lens odaksal uzunluğu farkı, renklere hassasiyeti farkı gibi farklılıklar taşıyabilir. Bu farklılıklar görüntü işleme sırasında hatalı hesaplamalara neden olur. Bunun için çalışmalarımızda aynı marka ve model kamera ile lensler kullanmaya özen gösterilmiştir.

İkinci bölüm uygulamada uçan bir maket helikopterin önceden belirlenmiş renkteki nesneyi algılamak ve merkez bilgisayara koordinat bilgisini göndermek. Bu sayede merkez bilgisayardan hareketlenen yer aracının aynı noktaya ulaşip, helikopterin yer aracı üzerine inişini sağlamak. Bu amaç için kullanılan CMUCAM v2+ kamerasının en önemli özelliği hafif olmasıdır. Helikopter de denemeleri PC ortamında simülasyonu için yazılım hazırlanmıştır. Bu yazılımda aranan renk aralığı PC kullanıcısı tarafından girilebilmektedir.

Karşılığında ise cismin kameranın tam ortasında bulunabilmesi için helikopterin gitmesi gereken yön sonuç olarak kullanıcıya verilmiştir. CMUCAM v2+ kullanımı için hazırlanmış yazılımlar Visual Basic 6.0 yazılım platformunda hazırlanmıştır.

İlk bölüm uygulama yazılımında daha kullanıcı dostu ve daha esnek bir yazılım haline getirerek her türlü nesneyi algılaya bilen, mesafeyi ölçebilen, kullanıma hazır bir ürün haline getirilebilir. Bunun yanında gelişen teknoloji sayesinde yazılımın daha hızlı koşması ve sonuçların saniyede 25 çıkması sağlanabilir. Böyle bir hız hata payını çok düşürecektir. Ayrıca hatalı ürün kontrolü yaparak yaklaşan cismin hatası algılanır algılanmaz dışarı atılması sağlanabilir. İkinci bölüm uygulama yazılımında daha güçlü bir işlemci ile helikopterin özel cisimlerin bulunması sağlanabilir. Fakat helikopterin sınırlı bir taşıma kapasitesi olduğundan kullanılacak ürünün çok ağır olmamasına dikkat edilmelidir.

Robotik uygulamaları için görüntü ve sinyal işleme algoritmaları geniş bir alanda hâlihazırda kullanılmaktadır. Görüntü işleme teknikleri ile nesne tanımanın yanında nesne konumlandırma, nesne takibi, esnek üretim hatlarının takibi gibi uygulamalar yapılabilir. Ayrıca tekstil, tıp, askeri alanlarda, desen tanıma, doku inceleme, hedef takibi gibi uygulamalar için görüntü işlem teknikleri kullanılabilir.

KAYNAKLAR

Darya Frolova, Debis Simakov, “Matching with Invariant Features”, The Weizmann Institute Mart 2004

Filiz Soykan, “Nesne tanıma ve aralarındaki topolojik ilişkilerin semantik düzeyde incelenmesi”, Trakya Üniversitesi 2005

Gonzalez R.C., Woods R.E., “Digital Image Processing, Second Edition”, Prentice-Hall, New Jersey, 25-344,519-732 ,(2002).

Hikmet Bal, “Kamera ile görüntü işleme teknikleriyle malzeme tane büyüklüğü analizi”, Gazi Üniversitesi 2006

Janusz Konrad ve Zhong-Dan Lan, “Dense Disparity Estimation From Feature Correspondences”, SPIE Ocak 2000

Volkan Verim, “Görüntü işleme yöntemleri ile doku sınırlarının belirlenmesi”, Gazi Üniversitesi 2006

Zhengyou Zhang, Rachid Deriche, Oliver Faugeras, Quang-Tuan Luong, “A robust Technique for Matching Two Uncalibrated Images Through the Recovery of the Unknown Epipolar Geometry”, INRIA Mayıs 1994

INTERNET KAYNAKLARI

[1] <http://www.bilesim.com.tr/mistportal/showmakale.nsf?xd=35.xml>

[2] http://en.wikipedia.org/wiki/Computer_vision

[3] http://en.wikipedia.org/wiki/Electromagnetic_spectrum

[4] http://en.wikipedia.org/wiki/Luminosity_function

EKLER

Ek 1 Yazılım Kaynak kodu

Ek 1 Yazılım Kaynak Kodu

```

// stdafx.h başlık dosyası
#pragma once
#define WIN32_LEAN_AND_MEAN // Exclude rarely-used stuff from Windows headers
#include <stdio.h>
#define STRICT
#include <windows.h>
// Open CV başlık dosyası
#include "cxcv.h"
// haberleşme başlık dosyası
#include "Serial.h"

typedef struct Mask
{
    float *dblGauss;
    int intGaussSize; //Karesi değil bir kenarı
    float dblSumGauss;
} _Mask;

// fonksiyonlar
void detect_and_draw( IplImage* img ,int intCamNo);
void GaussMaskOluştur(int intSize,float dblStdSap);
void on_mouse( int event, int x, int y, int flags, void* param );
void ilişkiliNoktalarıBul( );
void cemberRenkHesapli( IplImage* imgBoya, CvPoint pointMerkez, int intYaricap);
void cemberRenkHesap20li( IplImage* imgBoya, CvPoint pointMerkez, int intYaricap);
void initialize ( );
// haberleşme fonksiyonları
char* Haberlesme_Verioku();
void Haberlesme_Basla();
void Haberlesme_VeriGonder (uchar chrGonder);

// main.cpp dosyası

// standart başlık (header) dosyaları
#include "stdafx.h"
#include <uchar.h>
#include <iostream>
#include <fstream>
#include "time.h"
#include <math.h>
#include <float.h>
#include <time.h>
// open cv başlık dosyası
#include "cv.h"
#include "cvcam.h"
#include "highgui.h"

#define PI 3.14159265F

#ifdef _EiC
#define WIN32
#endif

// Çerçeveler (frame)
IplImage *image = 0;
IplImage *histRed = 0;
IplImage *histGreen = 0;
IplImage *histBlue = 0;
////////////////////////////////////
// Çember hesabı için değişkenler
bool booCemberBasla = false;
bool booCemberSurukle = false;
CvPoint pointCemberMerkez = cvPoint (-1,-1);
CvPoint pointCemberYaricap = cvPoint (-1,-1);
int intCemberYaricap = 0;
CvLineIterator iteratorCember;
////////////////////////////////////
// Histogram işlemleri
// 256 ton renk, 3 kanal, 6 top rengi
long lngPtrHist[256][3][7];
long lngTotalPx1 [7];
int intframeSayisi=0;
int intRenkHazirlanan = 0;
// Çıktı dosyaları
char DosyaRed[][20] =
{"c:\\top1_R.txt", "c:\\top2_R.txt", "c:\\top3_R.txt", "c:\\top4_R.txt", "c:\\top5_R.txt", "c:\\top6_R.txt"};
char DosyaGrn[][20] =
{"c:\\top1_G.txt", "c:\\top2_G.txt", "c:\\top3_G.txt", "c:\\top4_G.txt", "c:\\top5_G.txt", "c:\\top6_G.txt"};
char DosyaBlu[][20] =
{"c:\\top1_B.txt", "c:\\top2_B.txt", "c:\\top3_B.txt", "c:\\top4_B.txt", "c:\\top5_B.txt", "c:\\top6_B.txt"};
// genel değişkenler
bool booBulundu[7];
uchar chrOrtalamaRed[7];
uchar chrOrtalamaGreen[7];

```

```

uchar chrOrtalamaBlue[7];
uchar chrOlculenRed;
uchar chrOlculenGreen;
uchar chrOlculenBlue;

//Haberleşme
CSerial serial;
LONG lLastError = ERROR_SUCCESS;

using namespace std;

// mouse hareketleri
void on_mouse( int event, int x, int y, int flags, void* param )
{
    if( !image )
        return;
    if( image->origin )
        y = image->height - y;
    if (booCemberBasla)
    {
        if( event == CV_EVENT_LBUTTONDOWN )
        {
            pointCemberYaricap = cvPoint(x,y);
            intCemberYaricap = cvInitLineIterator( image, pointCemberMerkez,
pointCemberYaricap,&iteratorCember, 8, 0 );
            booCemberBasla = false;
        }
        else if( event == CV_EVENT_LBUTTONDOWN )
        {
            pointCemberMerkez = cvPoint(x,y);
            pointCemberYaricap = cvPoint(x,y);
        }
        else if( event == CV_EVENT_MOUSEMOVE )
            if (pointCemberMerkez.x != -1)
            {
                pointCemberYaricap = cvPoint(x,y);
                intCemberYaricap = cvInitLineIterator( image, pointCemberMerkez,
pointCemberYaricap,&iteratorCember, 8, 0 );
            }
        }
    }
    else if (pointCemberMerkez.x != -1)
    {
        if( event == CV_EVENT_LBUTTONDOWN )
        {
            pointCemberMerkez = cvPoint(x,y);
            booCemberSurukle = false;
        }
        else if( event == CV_EVENT_LBUTTONDOWN )
        {
            pointCemberMerkez = cvPoint(x,y);
            booCemberSurukle = true;
        }
        else if( event == CV_EVENT_MOUSEMOVE && booCemberSurukle)
            pointCemberMerkez = cvPoint(x,y);
    }
}

int main( int argc, char** argv )
{
    bool booRenkBulunamadi = false;
    CvCapture* capture = 0;
    int c;
    capture = cvCaptureFromCAM( 0 );

    if( !capture )
    {
        fprintf(stderr,"Görüntü yakalama aykıtı bulunamadı...\n");
        return -1;
    }

    cvNamedWindow( "Kamera", 1 );
    cvNamedWindow( "Hist Red", 1 );
    cvNamedWindow( "Hist Green", 1 );
    cvNamedWindow( "Hist Blue", 1 );
    cvSetMouseCallback( "Kamera", on_mouse, 0 );

    Haberlesme_Basla ();
    initialize();
    for(;;)
    {
        IplImage* frame = 0;
        frame = cvQueryFrame( capture );
        if( !frame )
            break;
        if( !image )
        {
            // gerekli belleğin ayrılması
            image = cvCreateImage( cvGetSize(frame), 8, 3 );
            histRed = cvCreateImage( cvSize (256,200), 8, 3 );

```

```

        histGreen = cvCreateImage( cvSize (256,200), 8, 3 );
        histBlue = cvCreateImage( cvSize (256,200), 8, 3 );
        image->origin = frame->origin;
    }

    cvCopy( frame, image, 0 );
    // Cemberi Ciz
    if (pointCemberMerkez.x != -1)
    {
        cemberRenkHesap201i (image,pointCemberMerkez,intCemberYaricap );
        cvCircle( image, pointCemberMerkez , intCemberYaricap, CV_RGB(0,0,255), 1, 8, 0 );
    }
    // Belleği ekranda göster
    cvShowImage( "Kamera", image );
    cvShowImage ( "Hist Red" ,histRed);
    cvShowImage ( "Hist Green" ,histGreen);
    cvShowImage ( "Hist Blue" ,histBlue);
    char* Okunan;
    uchar Gonder;

    Okunan = Haberlesme_Verioku() ;
    if (71 == (uchar) Okunan[0])
        fprintf(stderr,"Konveyör durdu\n");

    int intFark = 40;
    if (booBulundu [1] ||booBulundu [2] ||booBulundu [3]
        ||booBulundu [4] ||booBulundu [5] ||booBulundu [6])
    {
        booRenkBulunamadi = TRUE;
        cemberRenkHesap11i (image,pointCemberMerkez,intCemberYaricap );
        for ( int fori = 1 ; fori < 7; fori ++ )
        {
            if (booBulundu [fori])
            {
                if ((chrOlculenRed > chrOrtalamaRed[fori] - intFark
                    && chrOlculenRed < chrOrtalamaRed[fori] +
intFark) &&
                    (chrOlculenGreen > chrOrtalamaGreen[fori] - intFark
                    && chrOlculenGreen < chrOrtalamaGreen[fori] +
intFark) &&
                    (chrOlculenBlue > chrOrtalamaBlue[fori] - intFark
                    && chrOlculenBlue < chrOrtalamaBlue[fori] +
intFark))
                {
                    fprintf(stderr,"Bulunan Renk :%d ... \n", fori);
                    booRenkBulunamadi = FALSE;
                    /* PLC Haberleşme komutları
                    //2lik Renk Hex 10luk
                    //01101000 R 68 104
                    //01010100 G 54 84
                    //01001010 B 4A 74
                    //01000101 C 45 69
                    //01100010 M 62 98
                    //01010001 Y 51 81
                    //01000000 Boş 40 64
                    //
                    //Durdu kodu
                    //01000111 1 47 71
                    //01111000 2 78 120
                    */
                    switch (fori)
                    {
                        case 1:
                            Gonder = 104;
                            break;
                        case 2:
                            Gonder = 84;
                            break;
                        case 3:
                            Gonder = 74;
                            break;
                        case 4:
                            Gonder = 69;
                            break;
                        case 5:
                            Gonder = 98;
                            break;
                        case 6:
                            Gonder = 81;
                            break;
                    }
                    Haberlesme_VeriGonder ( Gonder );
                }
            }
        }
    }
    if (booRenkBulunamadi)

```

```

        fprintf(stderr, "Bulunan Renk :YOK ...\\n" );

    c = cvWaitKey(20);
    switch( (char) c )
    {
        case 27:
            break;
        case 'c':
            // çember çizme komutu
            pointCemberMerkez = cvPoint (-1,-1);
            booCemberBasla = true;
            break;
        // Renkler
        case '1':
            intRenkHazirlanan = 1;
            intframeSayisi = 0;
            break;
        case '2':
            intRenkHazirlanan = 2;
            intframeSayisi = 0;
            break;
        case '3':
            intRenkHazirlanan = 3;
            intframeSayisi = 0;
            break;
        case '4':
            intRenkHazirlanan = 4;
            intframeSayisi = 0;
            break;
        case '5':
            intRenkHazirlanan = 5;
            intframeSayisi = 0;
            break;
        case '6':
            intRenkHazirlanan = 6;
            intframeSayisi = 0;
            break;
        default:
            ;
    }
}

cvReleaseCapture( &capture );
cvDestroyWindow("Kamera");

return 0;
}

void cemberRenkHesap20li (IplImage* imgBoya, CvPoint pointMerkez, int intYaricap)
{
    // 20 çerçevedeki çemberin renk hesabı
    int intCemberdekiY =0;
    CvScalar scalarGetRenk ;
    if (!intRenkHazirlanan)
        return;
    if (!intframeSayisi)
    {
        for (int i = 0; i < 256 ; i++)
            for (int j = 0; j < 3 ; j++)
                lngPtrHist[i][j][intRenkHazirlanan] = 0L;

        cvZero(histRed);
        cvZero(histGreen);
        cvZero(histBlue);
        lngTotalPxl[intRenkHazirlanan];
    }

    if (intframeSayisi < 20)
    {
        intframeSayisi++;
        // çember denklemi (x-a)^2 + (y-b)^2 = r^2
        // a,b merkez, x,y aranan nokta olmak üzere
        for (int forX = pointMerkez.x - intYaricap; forX <= pointMerkez.x + intYaricap; forX++)
        {
            intCemberdekiY = (int)(sqrt((double) pow((double) intYaricap,2) - pow((double) forX -
            pointMerkez.x ,2) ) + pointMerkez.y );
            for (int forY = pointMerkez.y -(intCemberdekiY -pointMerkez.y ); forY <= pointMerkez.y +
            (intCemberdekiY -pointMerkez.y) ; forY ++)
            {
                scalarGetRenk = cvGet2D( imgBoya,forY, forX);
                if((long)scalarGetRenk.val [0]<253 || (long)scalarGetRenk.val
                [1]<253|| (long)scalarGetRenk.val [2]<253)
                {
                    lngPtrHist[(long)scalarGetRenk.val [0]][0][intRenkHazirlanan] ++;
                    lngPtrHist[(long)scalarGetRenk.val [1]][1][intRenkHazirlanan] ++;
                    lngPtrHist[(long)scalarGetRenk.val [2]][2][intRenkHazirlanan] ++;
                    lngTotalPxl[intRenkHazirlanan]++;
                }
                //lngPtrHist[(int)scalarRenk.val [2]]++;
            }
        }
    }
}

```

```

    }
else
{
    int bin_w = histRed->width / 256;
    cvZero(histRed);
    cvZero(histGreen);
    cvZero(histBlue);
    booBulundu [intRenkHazirlanan] = true;
    long lngToplamRed = 0;
    long lngToplamGreen = 0;
    long lngToplamBlue = 0;
    for (long fori = 0 ; fori < 256; fori ++ )
    {
        lngToplamRed += lngPtrHist[fori][2][intRenkHazirlanan] *fori ;
        lngToplamGreen += lngPtrHist[fori][1][intRenkHazirlanan]*fori ;
        lngToplamBlue += lngPtrHist[fori][0][intRenkHazirlanan]*fori;
    }
    chrOrtalamaRed[intRenkHazirlanan] = lngToplamRed / lngTotalPxl[intRenkHazirlanan];
    chrOrtalamaGreen[intRenkHazirlanan] = lngToplamGreen / lngTotalPxl[intRenkHazirlanan];
    chrOrtalamaBlue[intRenkHazirlanan] = lngToplamBlue / lngTotalPxl[intRenkHazirlanan];

    for(int i = 0; i < 256; i++ )
    {
        cvRectangle( histRed, cvPoint(i*bin_w,histRed->height),
                    cvPoint((i+1)*bin_w,histRed->height -
lngPtrHist[i][2][intRenkHazirlanan]/intframeSayisi/20),
                    CV_RGB(255,0,0), -1, 8, 0 );
        cvRectangle( histGreen, cvPoint(i*bin_w,histRed->height),
                    cvPoint((i+1)*bin_w,histRed->height -
lngPtrHist[i][1][intRenkHazirlanan]/intframeSayisi/20),
                    CV_RGB(0,255,0), -1, 8, 0 );
        cvRectangle( histBlue, cvPoint(i*bin_w,histRed->height),
                    cvPoint((i+1)*bin_w,histRed->height -
lngPtrHist[i][0][intRenkHazirlanan]/intframeSayisi/20),
                    CV_RGB(0,0,255), -1, 8, 0 );
    }
    //dosya işlemleri
    ofstream fileRed;
    ofstream fileGreen;
    ofstream fileBlue;

    fileRed.open (DosyaRed [intRenkHazirlanan -1]);
    fileGreen.open (DosyaGrn[intRenkHazirlanan -1]);
    fileBlue.open (DosyaBlu [intRenkHazirlanan -1]);

    for(int i = 0; i < 256; i++ )
    {
        fileRed << lngPtrHist[i][2][intRenkHazirlanan] << "\n";
        fileGreen << lngPtrHist[i][1][intRenkHazirlanan] << "\n";
        fileBlue << lngPtrHist[i][0][intRenkHazirlanan] << "\n";
    }
    fileRed.close();
    fileGreen.close();
    fileBlue.close();
    intframeSayisi = 0;
    intRenkHazirlanan =0;
}
}

void cemberRenkHesaplli (IplImage* imgBoya, CvPoint pointMerkez, int intYaricap)
{
    int intCemberdekiY =0;
    CvScalar scalarGetRenk ;

    // çember denklemi (x-a)^2 + (y-b)^2 = r^2
    // a,b merkez, x,y aranan nokta olmak üzere
    long lngToplamRed = 0;
    long lngToplamGreen = 0;
    long lngToplamBlue = 0;
    long lngToplamPixel = 0;
    for (int forX = pointMerkez.x - intYaricap; forX <= pointMerkez.x + intYaricap; forX++)
    {
        intCemberdekiY = (int)(sqrt((double) pow((double) intYaricap,2) - pow((double) forX -
pointMerkez.x ,2) ) + pointMerkez.y ) ;
        for (int forY = pointMerkez.y -(intCemberdekiY -pointMerkez.y ); forY <= pointMerkez.y +
(intCemberdekiY -pointMerkez.y) ; forY ++ )
        {
            scalarGetRenk = cvGet2D( imgBoya,forY, forX);
            if( (long)scalarGetRenk.val [2]<253|| (long)scalarGetRenk.val
[1]<253|| (long)scalarGetRenk.val [0]<253)
            {
                lngToplamRed += (long)scalarGetRenk.val [2];
                lngToplamGreen += (long)scalarGetRenk.val [1];
                lngToplamBlue += (long)scalarGetRenk.val [0];
                lngToplamPixel++;
            }
        }
    }
}

```

```

        }
        //lngPtrHist[(int)scalarRenk.val [2]]++;
    }
}

chrOlculenRed= lngToplamRed / lngToplamPixel;
chrOlculenGreen = lngToplamGreen / lngToplamPixel;
chrOlculenBlue = lngToplamBlue / lngToplamPixel;
}

void initialize ()
{
    for (int fori = 0 ; fori <7 ;fori++)
    {
        booBulundu[fori]=false;
        chrOrtalamaRed[fori]=0;
        chrOrtalamaGreen[fori]=0;
        chrOrtalamaBlue[fori]=0;
    }
}

void Haberlesme_Basla()
{
    lLastError = serial.Open (_T("COM7"),0,0,false);
    lLastError = serial.Setup(CSerial::EBaud9600,CSerial::EData8,CSerial::EParNone,CSerial::EStop1);
    lLastError = serial.SetupHandshaking(CSerial::EHandshakeHardware);
}

char* Haberlesme_Verioku()
{
    DWORD dwBytesRead = 0;
    char szBuffer[101];
    lLastError = serial.Read(szBuffer,sizeof(szBuffer)-1,&dwBytesRead);
    if (dwBytesRead > 0)
    {
        szBuffer[dwBytesRead] = '\0';
        printf("%s", szBuffer);
    }
    return szBuffer;
}

void Haberlesme_VeriGonder (uchar chrGonder)
{
    char* str ="";
    str = (char*)&chrGonder;
    lLastError = serial.Write(str);
}

```


ÖZGEÇMİŞ

Doğum tarihi	23.10.1982	
Doğum yeri	İstanbul	
Lise	1996–2000	Yahya Kemal Beyatlı Lisesi (Bakırköy, İstanbul)
Lisans	2000–2004	Sakarya Üniversitesi Mühendislik Fakültesi, Elektrik ve Elektronik Mühendisliği
Yüksek Lisans	2005–2008	Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü Elektrik Müh. Anabilim Dalı, Kontrol ve Otomasyon Programı

Çalıştığı kurumlar

2004–2005	SBS Bilişim Sistemleri, Bilgi İşlem Sorumlusu
2005–2007	Ergin Elektrik A.Ş., Yazılım Mühendisi
2007-Devam	Koroza Ambalaj A.Ş., Yazılım Mühendisi