

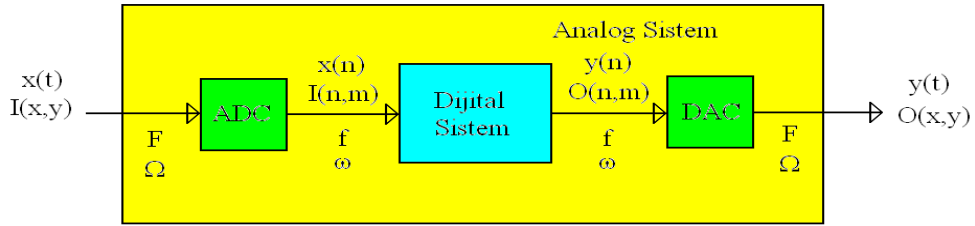
## Görüntü İşleme

### Genel tanım ve kullanım alanları

Görme, duyularımızın en gelişmişidir ve gördüğümüz imgeler çevremizdeki dünyayı algılayışımızda en önemli rolü oynar.

Biyolojik görme sistemi elektromanyetik radyasyonun görünür bölgesindeki frekansları algılayarak, dijital görüntü işleme sistemleri neredeyse tüm elektromanyetik spektrumu kullanır (ultrason, elektron mikroskobu, bilgisayarda üretilmiş görüntüler vb.).

Genel bir işaret veya görüntü işleme sisteminin blok diyagramı aşağıdaki gibidir.



**Görüntü işleme:** Sensörlerden gelen görüntünün bilgisayara aktarılıp üzerinde herhangi bir işlem yapılması ve ardından görüntüleyici çıkışa (Monitöre) iletilmesi sürecidir. [1]

Sayısal olarak elde edilmiş görüntülerin bilgisayar yardımı ile analiz edilmesi ve işlenmesi sürecidir. [2]

### Görüntü İşlemede Genel Tanımlar – Kavramlar

**Piksel (pixel) :** picture element sözcüklerinin birleştirilmesiyle oluşmuştur, görüntünün birim elemanını ifade eder.

**Parlaklık (intensity):** x ve y koordinatlarındaki bir pikselin parlaklık değerini gösterir.

**Ayrıklaştırma (Digitizing):** Analog görüntünün dijital sistemde ifade edilebilmesi için önce uzaysal boyutlarda sonlu sayıda ayrık parçaya bölünmesi (örnekleme, sampling), sonra da her bir parçadaki analog parlaklık değerinin belli sayıda ayrık dijital seviyelerden biri ile ifade edilmesi (kuantalama, quantizing) gerekir.

**Çözünürlük (Resolution):** İnç yada cm başına düşen piksel sayısıdır. (1 inç. = 2.54 cm) Görüntünün kaç piksele bölündüğünü, yani kaç pikselle temsil edildiğini gösterir. Çözünürlük ne kadar yüksekse, görüntü o kadar yüksek frekansta örneklenmiş olur ve görüntüdeki ayrıntılar o kadar belirginleşir.

**Uzaysal Frekanslar (Spatial Frequencies):** Uzaysal boyutlarda belli bir mesafede parlaklık değerinin değişim sıklığını ifade ederler.

**Giriş görüntüsü:** iki boyutlu, MxN uzunluklu bir matris olarak düşünülür ve sol üst köşedeki piksel değeri (1,1) başlangıç noktası olarak numaralandırılır.

**Uzaysal Frekanslar ( $f_x, f_y$ ):** Genellikle örnekleme periyodu 1 piksel kabul edilerek uzaysal frekanslar ve periyotlar piksel cinsinden ifade edilir.

### **Görüntü İşlemenin Amaçları**

- a) Görüntü İyileştirme
- b) Görüntü Onarma
- c) Görüntü Sıkıştırma
- d) Görüntü Analiz Etme
- e) Görüntü Tanıma

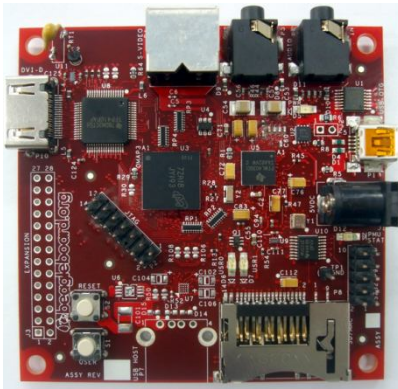
### **Görüntü İşleme Uygulama Alanları**

- a) Tıp
- b) Astronomi
- c) Coğrafya
- d) Biyoloji
- e) Endüstriyel Uygulamalar
- f) Mühendislik
- g) Belge İşleme
- h) Güvenlik, Savunma ve Hukuk
- i) Parmak İzi, Plaka, İris, Yüz Tanıma

### **Görüntü İşlemek İçin Kullanılabilecek Yazılımlar - Donanımlar**

- a) C#
- b) C++
- c) MATLAB
- d) ARM
- e) Beagleboard
- f) FPGA

**Dip NOT:** PIC mikrodeneleyici ailesi ile görüntü işleme direk olarak yapılamamaktadır. Bunun için bir bilgisayar arayüzü kullanılmalıdır. Bu arayüzün PIC mikrodeneleyici kartı ile çeşitli haberleşme metotları (RS-232, USB,  $I^2C$ , v.b.) kullanarak haberleşmesi yapılmaktadır.



**Beagleboard**



**ARM ve Beagleboard**

## **Dijital Görüntü Formatları**

- a) **RGB:** İngilizce Red, Blue ve Green kelimelerinin baş harflerinden oluşan bir kısaltmadır Türkçe'de Red; Kırmızı, Blue; Mavi, Green ise; Yeşil anlamına gelmektedir. Bu üç rengin karışımından oluşan renklerin bulunduğu bir renk uzayıdır. Üç ana rengin değişik oranlarda eklenmesiyle (additive colors) oluşur.

Rgb aynı zamanda renkli resim dosyası uzantısıdır. Sgi tarafından geliştirilmiş resim formatıdır. Güncel birçok resim programı tarafından açılabilir.

- b) **BMP:** En temel resim formatıdır. BMP 16 ya da çok renk kaydedebileceğimiz, herhangi bir sıkıştırma yapmayan oldukça hızlı bir formattır. Bu formatta resmin içindeki renk sayısından çok, resmin büyüklüğü önemlidir.
- c) **GIF:** BMP çok hızlı bir format olmasına karşın, oldukça fazla yer kapladığı için pek tercih edilmez. GIF formatının iyi bir sıkıştırma algoritması vardır (LWZ). Ve görüntüleme de oldukça hızlı bir şekilde gerçekleştirilebiliyor. En fazla 256 renk dışında 8bitdesteklenmektedir. Tek bir GIF dosya içinde birden çok GIF formatlı resim yerleştirilmesine ve anime edilmesine olanak tanır.
- d) **JPEG / JPG:** 256 renkten daha fazlası ve daha az yer kaplaması için JPEG (Joint Photographics Experts Group) daha iyi bir alternatif olabilir. JPEG az renk içeren uygulamalarda hem kaliteyi düşürüyor hem de dosya boyutunda önemli bir değişiklik sağlamıyor. Standart jpg formatında, resmin kalitesinden bir miktar ödün vererek sıkıştırma uygulanır. Özellikle 24 bit true color uygulamalarda resim kalitesinin düştüğünü anlamak mümkün değildir.
- e) **TIFF:** TIFF de yüksek renk derinliği olan görüntülerde kullanılır. Dosya boyutları JPEG' e göre daha büyüktür. TIFF biçimi birden fazla sayfayı desteklediği için, çok sayfalı dokümanlar ayrı ayrı dosyalar yerine tek bir TIFF dosyası olarak kaydedilebilir. Matbaacılıkta kullanılmak üzere planlandığından renkleri hem RGB hem de CMYK şeklinde ayrıştırmaktadır.
- f) **PNG:** Kayıpsız sıkıştırarak görüntü saklamak için kullanılan bir saklama biçimidir. GIF'in gelişmiş versiyonu denebilir. Ancak hareketli resimleri desteklenmez. Çizim ve metinlerde de JPEG' den daha iyidir. TIFF' e göre daha yaygın kullanılmaktadır.
- g) **PSD:** Adobe PhotoShop'un kendi dosya formatıdır. Resim işlemlerinde standart formattır. Sıkıştırma yapmaz. Görüntüye birden fazla katman (layer) eklenebilir. Dosya.psd formatında kayıtlı olduğu sürece tabaka korunur ve asıl görüntüden her an ayrılabilir. Dezavantajı olarak büyük dosya boyutu şeklinde bellekte yer kaplar.

## **Görüntü Türleri**

- **İkili Görüntü:**
  - Sadece siyah ve beyaz piksellerden oluşur
  - 1 piksel, 1 bit yer kaplar [0,1]
- **Gri Tonlamalı Görüntü**
  - Sadece grinin tonlarından oluşur.
  - Genelde piksel başına 8 bit ayrılır. [0..255]
- **Renkli Görüntü**
  - RGB renk modeli ile üç renk katmanından oluşur.
  -

## **Görüntü İşlemenin Konuları**

- Görüntü İyileştirme**
- Görüntü Onarma**
- Morfolojik İşlemler**
- Kenar Belirleme**
- Bölütleme**
- Tanıma**
- Nesne izleme, Nesne Takibi**
- Şablon Eşleme**
- Görüntü Sıkıştırma**

## **Görüntü İşlemenin Konularının Açıklamaları**

**Görüntünün elde edilmesi (Image Acquisition):** Kamera, Tarayıcı gibi bir görüntüleme cihazı ile görüntünün elde edilmesi.

**Resim İyileştirme (Image Enhancement):** Görüntünün kalitesini artırma, daha iyi bir görünüme getirme işlemidir. Görüntüyü koyulaştırma, daha açık hale getirme veya kontrastını artırma gibi işlemler yanında daha ileri filtreler de kullanılır.

**Resim Onarma (Image Restoration):** Bozulmuş veya gürültüye maruz kalmış resmi alarak temiz orijinal resmi tahminleme ve elde etme işlemidir.

**Morfolojik İşlemler (Morphological Operations):** Görüntüler üzerinde yapılan genişletme, aşındırma, sınır belirleme, iskelet ve dışbükey kabuk bulma gibi, temeli küme teorisine dayalı işlemlerdir.

**Kenar Belirleme (Edge Detection):** Görüntüdeki süreksizlikleri ve keskin (hızlı) tonlama değişikliklerini bulma işlemidir.

**Bölütleme (Segmentation):** Görüntüleri kendini oluşturan parçalara ve nesnelere ayırma işlemidir. Nesneleri arka plandan ve diğer nesnelerden ayırır.

**Tanıma (Recognition):** Bir nesneye tanımlayıcılarına dayalı olarak etiket atama yani sınıflandırma işlemidir.

**Nesne Takibi (Object Tracking):** Zaman boyunca görüntüdeki nesne ve nesnelerin konumlarını belirleme ve takip etme işlemidir.

**Sablon Eşleme (Template Matching):** Bir resim içerisinde, verilen bir şablon görüntü ile eşleşen parçaların bulunması işlemidir.

**Resim Sıkıştırma (Image Compression):** Resmin boyutunu azaltma işlemidir. Kayıplı veya kayıpsız olabilir. Resmi saklamak için gereken bellek miktarını ve iletmek için gerekli bant genişliğini azaltmayı amaçlar.

### **Görüntü İşleme Algoritmaları**

- a) Görüntü Histogramlarına Bağlı Algoritmalar
- b) Matematik Temelli Algoritmalar
- c) Convolution (Evrişim) Algoritmaları
- d) Filtreler

#### A) Histogram Eşitleme:

Tekrarlı sayılardan oluşan elimizdeki verileri, genelde çetele (ayrıntılı bilgi) tabloları kullanarak oluşturduğumuz veri grupları içine dâhil edip bu gruplardaki verilerin sayılarının kullanılması ile elde edilen sütun grafiklerine histogram adı verilir. Yani çok fazla veri varsa bu verileri aralıklarla yeniden düzenleyerek grafik oluşturma işidir.

Bir resim üzerinde üzerinde birçok değer içeren dağınık bir piksel dağılımı vardır. Histogram eşitleme sayesinde görüntü daha da netleşir.

#### B) Mantık İşlemleri:

- AND / NAND : Giriş olarak 2 adet binary veya integer gri seviye görüntü verilir. Her pikselin sabit bir değerle mantık işlemine tabi tutulması da mümkündür. AND işleminde görüntü içerisinde herhangi bir nesnenin yer değiştirip değiştirmediği taranabilir.
- OR / NOR: Giriş olarak 2 adet binary veya integer gri seviye görüntü verilir. Her pikselin sabit bir değerle mantık işlemine tabi tutulması mümkündür. OR işlemi iki görüntünün birleştirilmesi amacıyla da kullanılabilir.
- XOR / XNOR: Giriş olarak 2 adet binary veya integer gri seviye görüntü verilir. Her pikselin sabit bir değerle mantık işlemine tabi tutulması mümkündür. İki veya daha çok görüntü taranarak sabit tutulan resim ile değişmiş koşuldaki resim birleştirilerek sabit olan cisimler silinip hareketli cisimlerin tespiti için kullanılır.
- NOT (TERSLEME) : TERSLEME işlemi ikili seviyede “1” lerin “0” ve “0” ların “1” olması şeklindedir.

#### D) Filtre işlemleri:

Filtreler görüntü zenginleştirme amacı ile de uygulanan, adından da anlaşılacağı gibi görüntüde yer alan gürültülerin atılması, belirli ayrıntıların ayıklanması ya da daha belirgin hale getirilmesi vb. gibi operasyonları gerçekleştiren operatörlerdir.

Farklı amaçlar için farklı filtreleme operatörleri vardır. Bunlar:

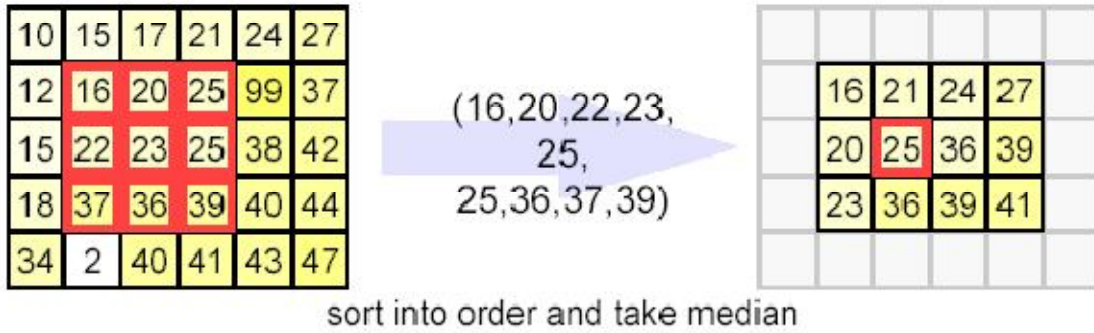
- Gürültülerin giderilmesi
- Kenar keskinleştirme ve yakalama
- Görüntü yumuşatma ve bunun gibi daha bir çok amaçla kullanılan filtreler örnek verilebilir.

#### E) Medyan Filtreleme

Medyan filtreleme, lineer bir işlem değildir. Filtreleme merkez piksel etrafında yer alan komşu piksellerin medyanlarının alınması esasına dayanarak gerçekleştirilir.

Komşu pikseller sıralanarak ortada yer alan piksel değeri merkez pikselin yerini alır.

e.g. 3×3 median filter



Yararları

Gürültü eklenmiş bir resim eski haline bu methodla döndürülebilir.

Trishot, kenar bulma kaldı..

## Örnek Matlab Yazılımları

### 1. Yazılım.

Yazılım Açıklaması: Herhangi bir renkli resim dosyasını matlab kodları ile incelenerek;

1. İlk önce gri seviyeye getireceiz,
2. Gri seviyesinin imaj üzerinde grafiksel dağılımına bakacağız,
3. Resim üzerindeki parlaklık seviyesini her yerde aynı düzeye getireceiz,
4. Oluşturulan son süreçteki vektörümüzde tüm değerleri 0-255 arasından 0-1 değerlerine dönüştüreceğiz.

### 1. Yazılımın Matlab kodu:

```
%Proje RoboTek 2013 =====>>> DeltaAST <<<=====%
```

```
DeltAST=imread('c:\resim.jpg'); %işlenecek dosyanın yerini gösterdik.
```

```
imshow(DeltAST);          %gelen matrisi ekranda gösterir.
```

```
%-----%
```

```
AST=rgb2gray(DeltAST);    %matrisi vektöre çeviriyor.
```

```
Yani 0-255 arasında gri değere dönüştürüyor.
```

```
imshow(AST);
```

```
%-----%
```

```
imhist(AST);              %Griye çevrilmiş imaj üzerindeki değerlerin grafiksel dağılımını  
gösterir.
```

%-----%

DEL=imadjust(AST);                   %resim üzerindeki parlaklıkların hepsini aynı seviyeye getirir.

imshow(DEL);

%-----%

A=im2bw(DEL);                   %vektördeki tüm değerleri 0 ve 1 lere çevirir.

imshow(A);



İlk Aşamada →



Son Aşamada →

## 2. Yazılım.

Yazılım Açıklaması: Mevcut bulunan şekerlerden kırık olanlarının tespiti.

Yazılımın Matlab kodu:

close all;

clear all;

rgb = imread('c:\objeler.jpg'); % görüntünün okunması sağlandı.

imshow(rgb);

I = rgb2gray(rgb); % resmin gri tonlamalı haline getirildi.



```

imshow(I);

background = imclose(I, strel('disk',15)); % şekerleri resimden çıkardık. beyaz bölgelerin
içindeki siyahlıklar gideriliyor.
imshow(background);

I2 = imsubtract(background,I);

imshow(I2,[]);
%biraz önce oluşturduğumuz görüntüden (8. Satır) ilk gri seviyesine dönüştürülmüş resmi
çıkartıyoruz.Bu adımdan sonra arka plandaki ışıklandırma kalkmış oluyor.

%-----%
%buradan sonra objelerin bölümlenmesi kısmı başlıyor.

BW = im2bw(I2,graythresh(I2));
%Şekerleri daha rahat ayırt edebilmek için ilk önce resmi binary resim
%hale getiriyoruz.

delta = imfill(BW,'holes');
imshow(delta);
%komutu ile resimde çukur diye nitelendirilen yerleri dolduruyoruz

%-----%
%Ayırt edici özelliklerin bulunması kısmı

[labeled,numObjects] = bwlabel(delta,4);
stats = regionprops(labeled,'Eccentricity','Area','BoundingBox');
areas = [stats.Area];
eccentricities = [stats.Eccentricity];

%“labeled” değişkeninde şekerlerin bilgileri tutuluyor. “numObjects” de ise kaç tane şeker
bulunduğu tutuluyor. Örneğin burada numObjects değeri 8’dir.

%-----%
%Kırık Objelerin Bulunması için Özelliklerinin Analizi

minSize = mean(areas) - 0.25 * std(areas);
idxOfDefects = find(areas < minSize & eccentricities > .5);
statsDefects = stats(idxOfDefects);

%Önceki adımda bulduğumuz alanları kullanarak tüm şekerler için ortalama bir alan
buluyoruz. ( minSize ) . Alanı “minSize”dan küçük olan yuvarlak şekilleri “idxOfDefects”
değişkenine atadık.

%-----%
%Kırık Objelerin İşaretlenmesi Son adımımız kırık şekerlerin işaretlenmesidir.

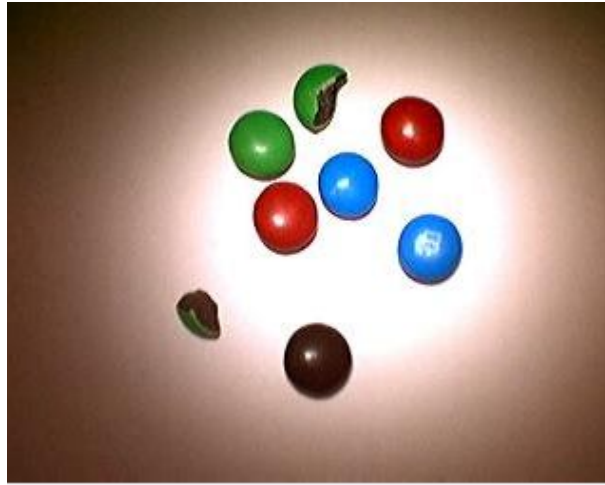
figure;
imshow(rgb);

```

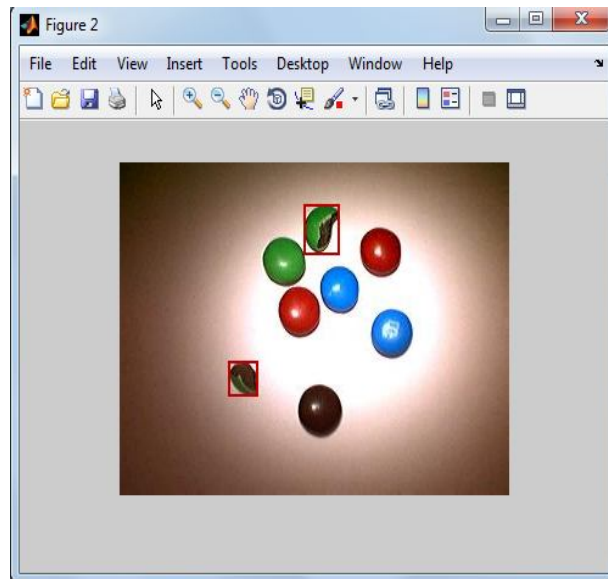
```

hold on;
for idx = 1 : length(statsDefects)
    h = rectangle('Position',statsDefects(idx).BoundingBox,'LineWidth',2);
    set(h,'EdgeColor',[.75 0 0]);
    hold on;
end
hold off;

```



İlk Aşamada →



Son Aşamada →