

GÖRÜNTÜ İŞLEME - (1.Hafta)

TEMEL KAVRAMLAR

Tanım

Görüntü İşleme (Gi) **askeri, endüstri, robotik, reklamcılık, astronomi, tıp, coğrafya, trafik** gibi günlük yaşamın pek çok alanında kullanılan, alınan görüntünün işlenmesinden sonra devamında **yapay sinir ağları (YSA), bulanık mantık** gibi pek çok algoritma ile değerlendirilen bir teknolojidir. Günümüzde artık her işlemin insan yerine otonom sistemlere yaptırılması, görüntü işlemenin önemini bir kat daha artırmıştır.

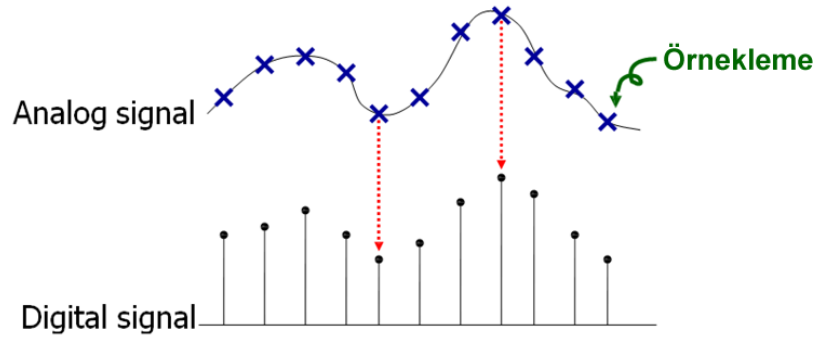
Kavramlar

Görüntü: 3 Boyutlu nesnelerin 2 Boyutlu yüzey üzerine düşürülmüş haritası olarak tanımlanabilir. Bu haritalamada her noktanın **konum bilgisi** $f(x,y)$ ve **renk bilgisi** tutulur. Bu resimlerin insan gözünün görebileceği şekilde gösterilmesine ise **Resim** diyebiliriz. Görüntü henüz daha sinyal şeklinde ise, insan gözünün görebileceği şekilde değilse resim haline gelmemiş demektir.

Görüntü **Analog** ve **Dijital** görüntü olarak iki kısımda ele alınabilir.

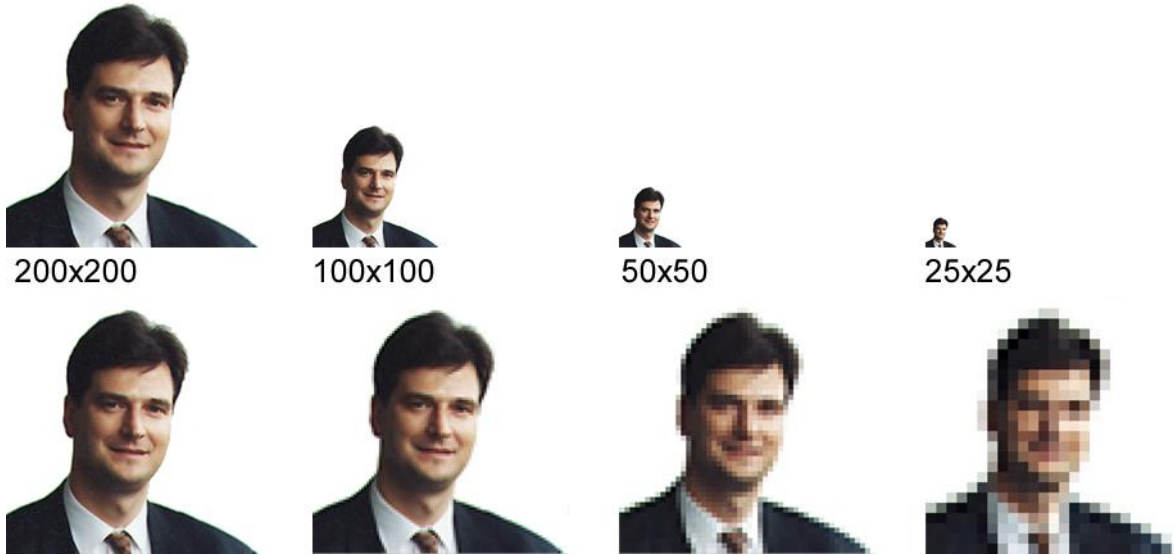
Analog Görüntü: Analog görüntüde, görüntüyü oluşturan fonksiyonun- $f(x,y)$ değişkenleri reel değerler alıyorsa (yani tüm sayıları kapsıyorsa) bu görüntü Analog görüntüdür. Analog bir görüntüde bir resme ne kadar yakından bakarsak bakalım (örneğin mikroskopla) orada hala görüntüyü oluşturan renkler bulunur. Sayısal bilgisayarlar, sürekli fonksiyonları/parametreleri işleyemezler. Bu fonksiyonların sayısallaştırılması gerekir.

Dijital (Sayısal) Görüntü: $f(x,y)$ şeklinde temsil edilen sürekli görüntüyü (analog görüntü) **ayrık örnekler** cinsinden (discrete) ifade edilmesidir ve gösterimi $f[x, y]$ şeklindedir.



Şekil. Analog sinyalin, dijital sinyale dönüşümünde örneklerin alınması.

Piksel: Dijital bir görüntünün 2-boyutlu dizi şeklindeki **her bir elemanı**'na bir piksel denir. Aşağıdaki şekilde aynı resmin farklı piksellerle gösterilmiştir.



Şekil. Aynı resmin farklı piksel boyutlarında gösterimi.

Görüntü İşlemenin kullanıldığı Uygulamalar

Resim video çekme uygulamaları: Doğru diyafram ayarları ile uygun renk ve ışık kalitesini ayarlamak için kullanılır.



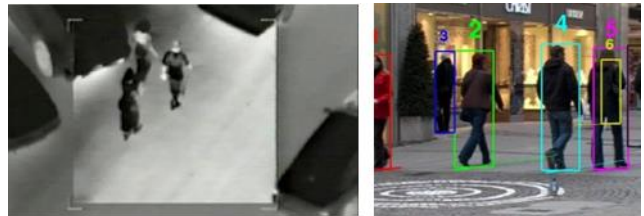
Ekran görüntülemeleri ve Yazdırma işlemleri: Bu uygulamalarda görüntü boyutu, renk, tonlama ayarları için kullanılır.



Resim saklama ve iletimini kolaylaştırmak: Görüntüyü verimli bir şekilde depolama ve aktarmak için kullanılır.



Güvenlik uygulamaları: Güvenlik taramaları için renkleri zenginleştirme, netliği artırma vs uygulamaları için kullanılır.



Bilgi çıkarma: Barkod okuma, plaka okuma, yazı karakterini metne dönüştürme gibi uygulamalar için kullanılır.



Mosaik Resim oluşturma: Yüksek çözünürlüklü (GByte resimler) resimler, panoramik (geniş açılı) resimler yada 360 derece (her yönde resimler) oluşturmak için bir çok görüntünün çakıştırılarak büyük resimler haline getirilmesi gerekir. Bu tür işlemler için görüntü işlemeye ihtiyaç vardır.



Yüz tanıma: Gerek güvenlik için gerekse insan sayma, otomatik yüz gizleme yada diyafram ayarı gibi hususlar için yüz tanımaya ihtiyaç vardır.



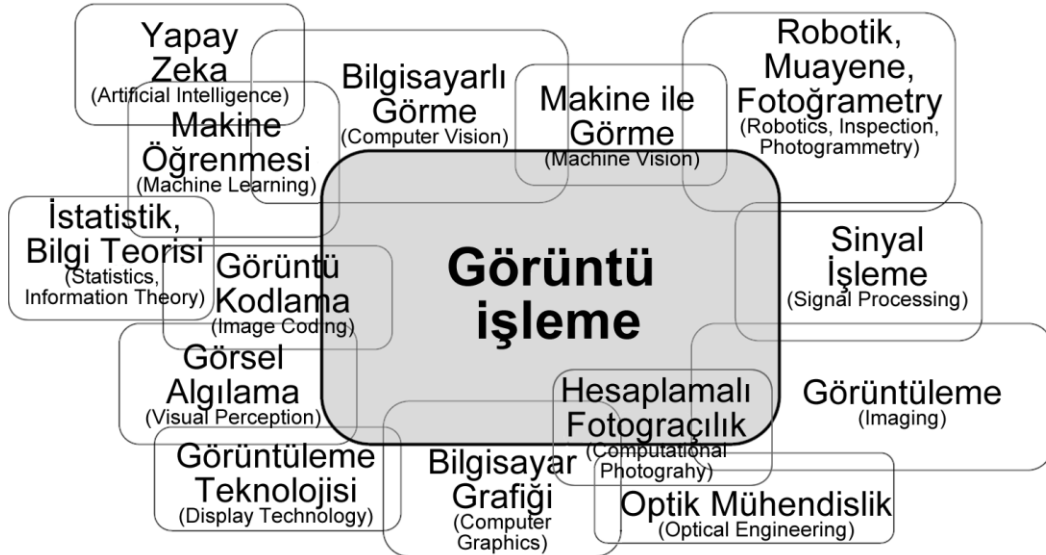
Otonom görüş yeteneği: Robotların ve otonom araçların etrafındaki dünyayı tanıyabilmesi için görüntü işlemeye ihtiyaç vardır.



Sanal gerçeklik uygulamaları: 3 boyutlu görüntülerin hazırlanması, gerçek dünya ile sanal dünya görüntülerinin birleştirilmesi gibi bir çok uygulama görüntü işleme ile yapılmaktadır.



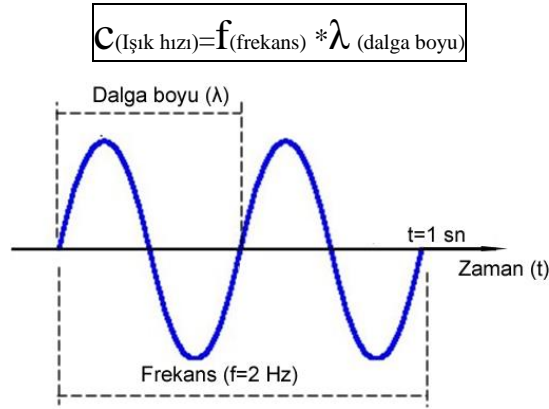
Görüntü işlemede bütün bu uygulamalar yapılırken bir çok disiplin ile ilişkisi vardır. Bu disiplinler ve aralarındaki kesişim aşağıdaki şekilde gösterilebilir.



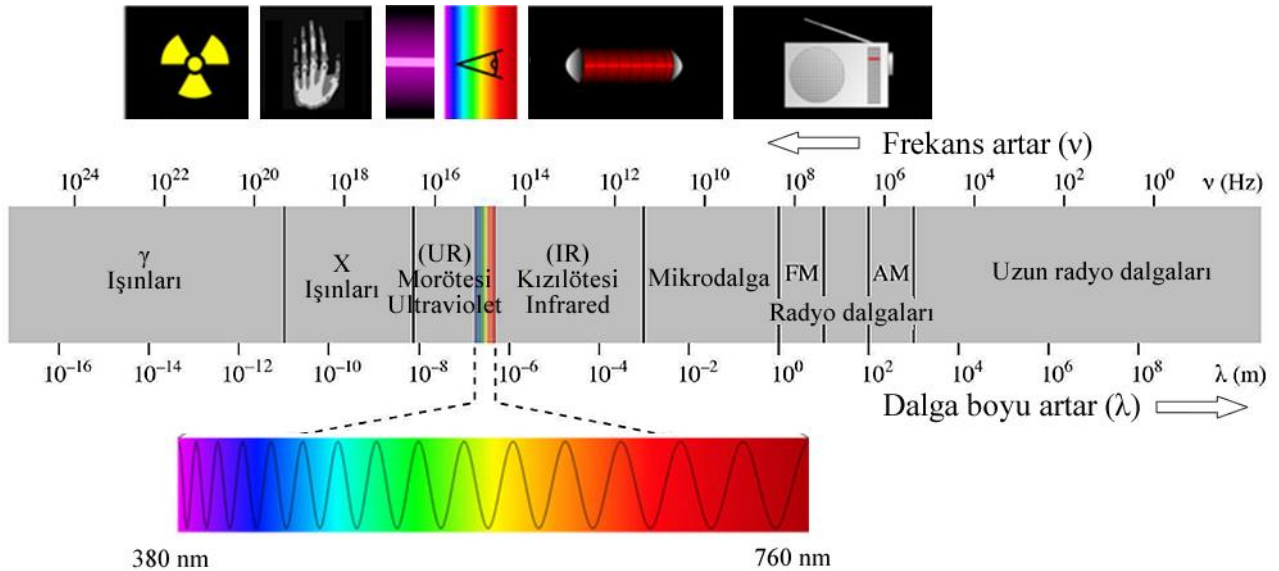
Işık Nedir?

Işık bir enerjidir. Işık hakkında çeşitli teoriler vardır. Işığın hem madde (kütle), hem de enerji (dalga) olduğu ispatlanmıştır. Işık, ışık kaynağı tarafından yayılan küçük dalgacıklardır. Bu dalgacıklar radyasyon enerjisi şeklindedir (elektromanyetik radyasyon). Etrafa düz çizgiler halinde yayılır. Görünen ışık elektro manyetik spektrumda çok küçük bir aralıkta bulunur (%2 civarı). Bu aralığın dışındaki ışık dalgalarını çıplak gözle göremeyiz.

Görünen ışık bölgesindeki her rengin bir dalga boyu vardır. Genel olarak gözlerimizle gördüğümüz ışığın dalga boyu 0,380-0,760 mikron (370 -760 nanometre) aralığındadır. Işığın hızını saniyede 300 bin km alırsak aşağıdaki bağlantıdan frekansını da bulabiliriz. Bu bağlantıya göre ışık, hem frekans (ν) hem de dalga boyu (λ) ile ifade edilebilir. Işık hızı boşlukta 299.792.458 m/s dir (300.000 km/s). Dalga boyu (λ) birimi metre (m) , frekans (f) birimi ise Hertz (Hz) dir.



Beyaz ışık, kendi renklerine ayrıştırılırsa, bir renk bandından (spektrumundan) oluştuğu görülür. Işığın dalga boyu küçük değerden büyük değere doğru ilerledikçe rengi de maviden (mor) kırmızıya (kızıl) doğru dönüşüm yapar.



Şekil. Elektromanyetik spektrum.

Tamamen siyah bir cisim, ultraviyole (mor ötesi) ve enfraruj (kızıl ötesi-infrared) ışınlar gibi bütünüyle görünmez. Üstüne düşen bütün ışığı emer. Bununla birlikte **cisim ısıtıldığı zaman** yalnızca sınırlı ve belirli renklerde ve frekanslarda ışınım verir. Önce kızılötesi ışınlar saçar. Sonra ısındıkça kırmızı ve sonuç olarak beyaz olur. Eğer yeterli derecede ısıtılabilseydi mavi renge dönüşüp, en sıcak yıldızlar gibi ultraviyole ışınlar saçardı.

Işığın frekansını değiştirmenin ya da gözle görülmeyen frekansı görünür hale çevirmenin bazı yararları vardır. Flüoresan maddeler (**fosforlu**), değişik frekanslarda ışık alırlar ve bunları daha farklı frekanslarda yayarlar. Kendisine gönderilen bizim gözle göremediğimiz ışıkları da görünür hale getirdiğinden yansımaları daha parlak olur. Bazı reklamlarda kullanılan boya ve mürekkeplerin esası budur. Bazı temizleme tozlarında, görünmez ultraviyole ışınları mavi ışığa çeviren, böylece çamaşırı daha parlak gösteren optik parlaticılar bulunur. Paraların kontrolünde kullanılan ışık da, paranın üzerine yansıtılan ışık gözle gözükmemektedir fakat paranın üzerindeki bu tip özel boyalara geri yansıma verirken parlamaktadır.

Ayrıca **gece görüş kameralarında** bulunan sistem gözle göremediğimiz kızıl ötesi (sıcak) ışınları gözle görebildiğimiz dalga boyuna getirerek gözükmesini sağlar.

Gözün Yapısı ve Görme

Gözümüz çevremizde gördüğümüz cisimlerin görüntüsünü işleyen ve bunu beynin anlayacağı sinyallere dönüştüren, çevremizden **en fazla bilgi** edinmemizi sağlayan duyu organımızdır.

Gözün yapısı aşağıdaki tabakalardan oluşur.

a. **Sert Tabaka:** Göz yuvarlağını dıştan saran beyaz bağ dokudan oluşmuş sert bir tabakadır. Sert tabaka göz yuvarlağının ön tarafında saydam bir yapı kazanır. Burası kornea adını alır. Işığı kırıcı etkiye sahiptir.

b. **Damar Tabaka:** Sert tabakanın altında damarlarca zengin bir tabakadır. Göz içinde siyah karanlık bir odanın oluşmasını sağlar ve göz içi yansımalarını önler.

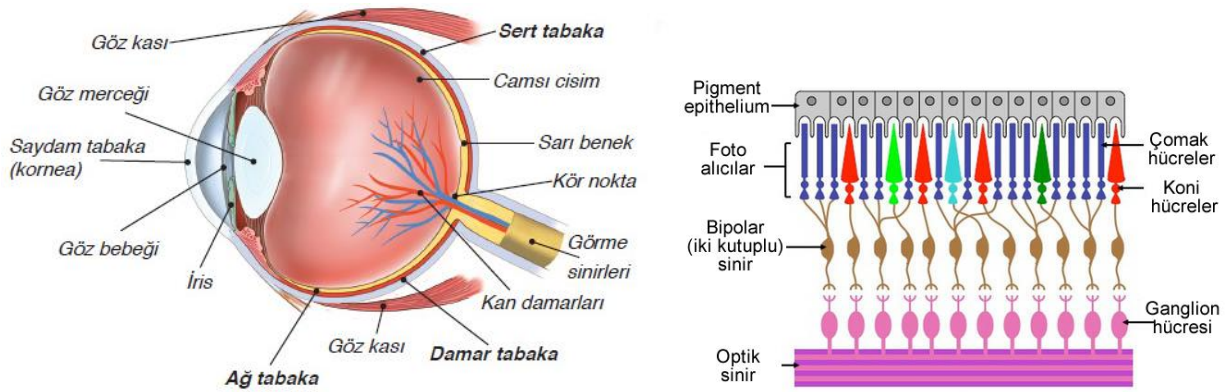
Damar tabaka gözün ön kısmında iris adı verilen, gözümüzün renkli kısmını oluşturur. İrisin yapısında bulunan kaslar göz bebeğinin genişlemesini ya da daralmasını sağlarlar. İrisin ortasında göz bebeği açıklığı bulunur. Göz bebeğinin daralıp genişlemesi ile göze gelen ışık miktarı ayarlanır. İrisin hemen arkasında göz merceği yer alır. Mercek, cisimden gelen ışınları kırarak ağ tabaka üzerine düşmesini sağlar.

c. **Ağ Tabaka (Retina) :** Işığa duyarlı reseptör (alıcı) hücrelerinin ve sinirlerin bulunduğu tabakadır. Bu bölgede çomak ve koni reseptörleri bulunur.

Duyu sinirleri gözün arka tarafında bir noktada birleşerek göz yuvarlağından dışarı çıkar. Görme duyusunun alınmadığı bu yere **kör nokta** denir.

Göz merceğinin merkezi ile aynı hat üzerinde bulunan retina merkezi, görme işleminin en fazla olduğu bölgedir (**sarı benek**). Bu bölgede bir cismin ayrıntılarını seçmekle sorumlu ışığa duyarlı **koni hücreleri** kümelenmiştir. Bu koni hücreleri **renklere karşı duyarlıdır** ve üç tip koni hücresi bulunur. Kırmızı, yeşil ve mavidir.

Retinanın her yerine dağılmış, cisimlerin **şekillerini** algılamaya yarayan **çomak hücreleri** bulunur. Bu hücreler ışığa karşı duyarlıdır. Ancak renklere karşı duyarsızdır.



Şekil. Gözün bölümleri ve temel yapıları

İnsan algılama sistemi görüntü işleme açısından bakıldığında, görüntü yakalama, gruplama ve analiz konusunda bilinen **en karmaşık sistemdir**. İnsan görme sistemi gözlerle başlar. Göz bir fotoğraf makinesi gibi ve beynin görme bölümleri de analog temele dayanan bir görüntü işleme sistemi olarak düşünülebilir.

Göz Kusurları

Görüntüyü bizler onu nasıl görüyorsak öyle algılarız. Bu nedenle görüntüyü gözümüzün nasıl algılayacağı ile ilgili kusurlar hakkında bilgi sahibi olmakta görüntüyü yorumlamak için gereklidir.

a. **Miyopluk:** Göz küresi optik eksen doğrultusunda **uzamışsa**, mercekten çıkan görüntü retinanın önünde oluşur. Buda netliğin kaybolmasına yol açar. Miyop fertler yakını iyi görür, uzağı iyi göremezler. Kalın kenarlı merceklerden yapılmış gözlüklerle bu kusur giderilebilir.

b. **Hipermetropluk:** Göz küresi optik eksen doğrultusunda **kısalırsa**, mercekten çıkan görüntü retinanın gerisine düşer ve netlik sağlanamaz.

Böyle kişiler, uzağı iyi gördükleri halde, yakını iyi göremezler. Görüntüyü netleştirmek için ince kenarlı merceklerden yapılmış gözlükler kullanılır.

c. **Astigmatizm:** Astigmatizm korneanın her yönde **aynı eğiklikte** (düzgün küre kapağı şeklinde) olmamasından kaynaklanan bir göz kusurudur. Basitçe gözün top gibi değil yumurta gibi olması şeklinde tarif edilebilir. Buna bağlı olarak retinada (görme tabakasında) bulanık bir görüntü oluşur. Astigmatlar hem uzağı, hem yakını net göremezler. Aynı zamanda düzgün kırınım oluşmadığından şekilleri çarpık görürler. Astigmatlarda baş ve göz ağrısı şikayetlerine sık rastlanır.

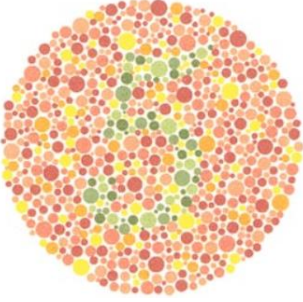
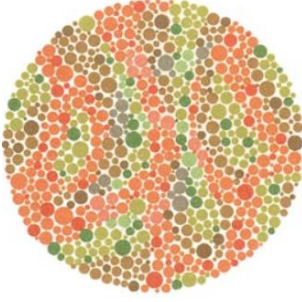
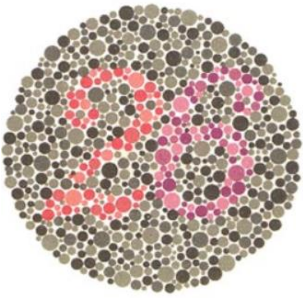
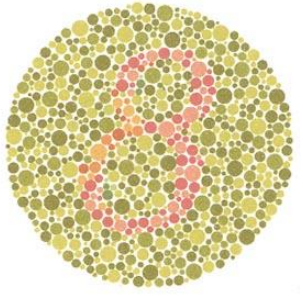
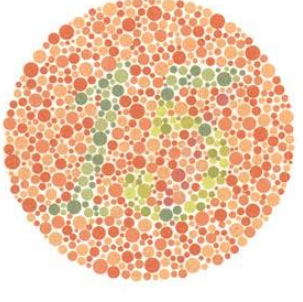
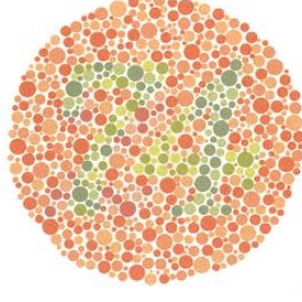
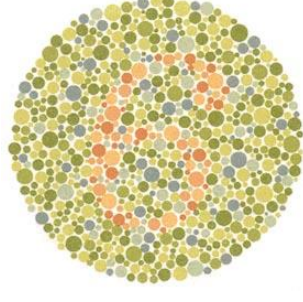

d. **Presbitlik:** Yaşlandıkça **merceğin esnekliğinin** kaybolmasıyla ortaya çıkar. 40 cm den daha yakını göremezler. İnce kenarlı merceklerle düzeltilir.

e. **Renk körlüğü:** Renkli görmeyi sağlayan **3 tip koni hücreden** (Kırmızı, yeşil, mavi) bir veya ikisinin genetik bozukluk sonucu bulunmamasından ortaya çıkar. Kalıtsaldır, düzeltilemez. Bu kişiler genellikle kırmızı ve yeşil renkleri ayırt edemezler. Erkeklerde yüzde 8, kadınlarda yüzde 0.4 oranında görülmektedir. Renk körleri dünyayı siyah-beyaz görmezler. Genellikle kırmızı ve yeşilin tonlarını ayırt edemezler. Birçok kişi renk körü olduğunu kendiliğinden fark etmez.

f. **Şaşılık:** Göz kaslarının uzun veya kısa olması sonucu göz bebeğinin yana kaymasıdır. Ameliyatla düzeltiler.

g. **Katarak:** Katarakt göz merceğinin saydamlığını kaybetmesidir. Bu durumu buğulanmış cama benzetebiliriz.

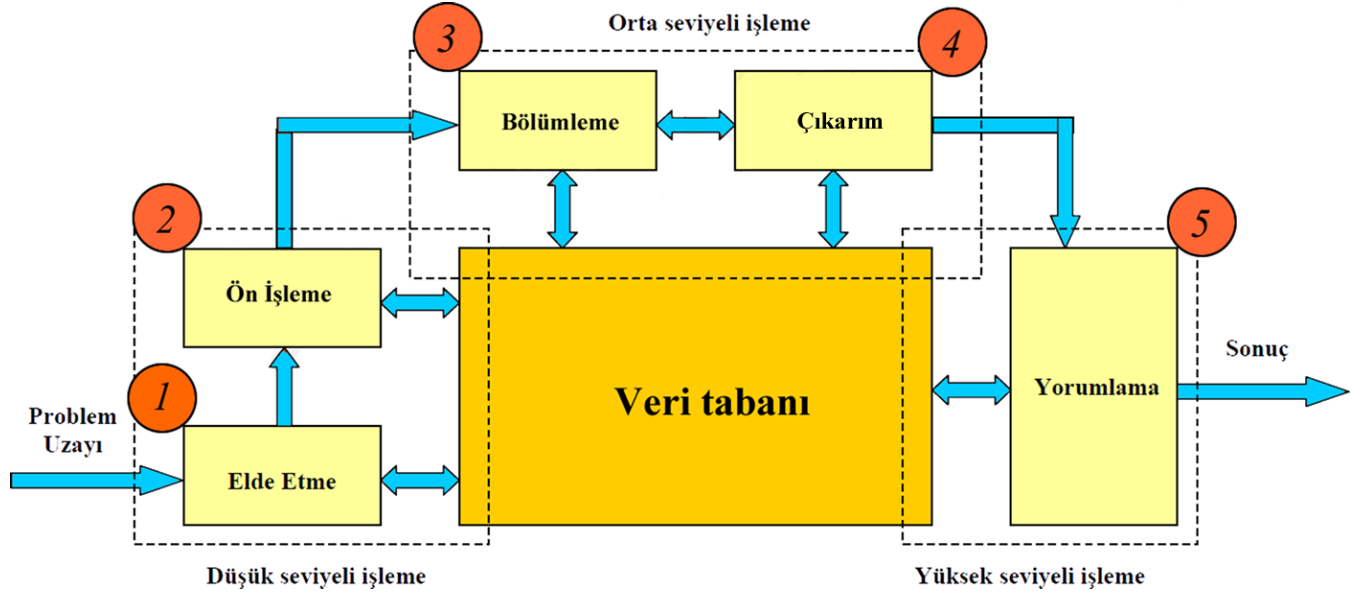
Tablo. Renk körlüğü testi örnek resimleri.

			
Normal görenler 5 olarak görür. Tüm renklere karşı renk körü olanlar hiçbir şey göremez.	Normal görenler hiç bir şey göremez. Kırmızı ve yeşil renk körü olanlar 5 olarak görür.	Normal görenler 26 olarak görür. Renk körü olanlar 2 veya 6 olarak görürler.	Normal görenler 8 olarak görür. Kırmızı yada Yeşil renk körü olanlar 3 olarak görür.
			
Normal görenler 15, Kırmızı-yeşil renk körü olanlar 17, Tüm renklere karşı renk körü olanlar ise hiç bir şey göremezler.	Normal görenler 74, Kırmızı-yeşil renk körü olanlar 21, tüm renklere karşı renk körü olanlar hiç bir şey göremezler.	Normal görenler 6 olarak, tüm renklere karşı renk körü olanlar hiç bir şey göremezler.	Normal görenler ve Renk körü olanların her ikisinde 12 olarak okur.

Sayısal görüntü işleme

Sayısal görüntü işleme, analog bir görüntünün sayısal biçime dönüştürülmesi ve daha sonra çeşitli amaçlar için (iyileştirme, onarma, sınıflandırma, sıkıştırma, anlama ve yorumlama, v.b.) sayısal bilgisayarlarla işlenmesi ile ilgilenir.

Temel basamakları aşağıdaki şekilde özetlenmiştir.



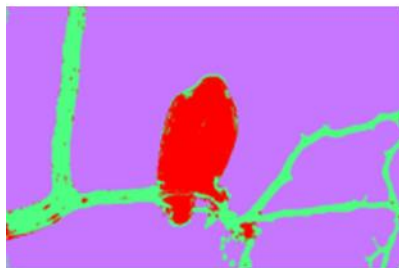
Şekil. Sayısal görüntü işlemenin temel basamakları.



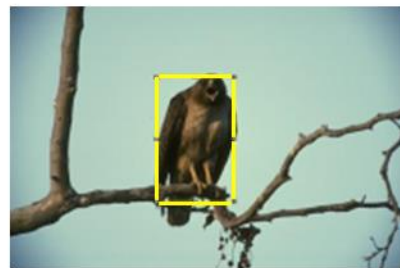
Orjinal Resim



Sınır Bulma



Anlamsal Bölümleme



Cisim Tespiti

Elde Etme: Görüntü işlemedeki ilk adım, sayısal görüntünün sayısal kamera ile **elde etme** işlemidir.

Ön İşleme: Sayısal resim elde edildikten sonraki aşama ise **ön-işleme**'dir. Ön-işleme, elde edilen sayısal görüntüyü kullanmadan önce daha başarılı bir sonuç elde edebilmek amacıyla görüntünün bazı ön işlemlerden geçirilmesidir. Bu işlemler temel olarak;

- görüntü **iyileştirme** (image enhancement),
- görüntü **onarma** (image restoration) ve
- görüntü **sıkıştırma** (image compression)

alt başlıkları altında toplanabilir. Görüntünün elde edilmesi ve ön işlemeye tabi tutulması düşük seviyeli görüntü işleme olarak adlandırılır.

Bölümleme: Ön-işlemeden sonra bölümleme (bölütleme, parçalara ayırma, segmentasyon denir) aşamasına geçilir. Bölümleme, görüntü işlemenin **en zor** uygulaması olup bir görüntüdeki nesne ve arka planın veya görüntü içerisindeki ilgilenilen değişik özelliklere sahip **bölgelerin birbirinden ayrıştırılması** işlemidir. Bölümleme; bir görüntüdeki nesnenin **sınırları** ve **alanlarını** tespit ederek **şekli** üzerinde ham bilgiler üretir. Eğer nesnelerin şekilleriyle ilgileniyorsak, bölümleme bize o nesnenin kenarları, köşeleri ve sınırları hakkında bilgiler verir. Diğer taraftan görüntü içerisindeki nesnelerin yüzey kaplaması, alanı, renkleri, iskeleti gibi iç özellikleriyle ilgileniliyorsa bölgesel bölümleme yapılmalıdır. Karakter veya genel olarak örüntü (pattern) tanıma gibi oldukça karmaşık problemlerin çözümü için her iki bölümleme yönteminin (sınırlar ve alanlar) bir arada kullanılması gerekebilir.

Çıkarım: Görüntüden elde edilen ham bilgilerin, ilgilenilen ayrıntıların ön plana çıkarılmasıdır. Yani aranan özellikli alanların arka plandan ve birbirinden ayrıştırılmasıdır.

Yorumlama: Yüksek seviyeli görüntü işleme grubuna giren bu aşamada, çeşitli karar verme mekanizmaları (yapay zeka algoritmaları gibi) ile görüntüdeki arka plandan çıkarılmış nesnelerin veya bölgelerin etiketlendirilmesi, sınıflandırılması yapılır.

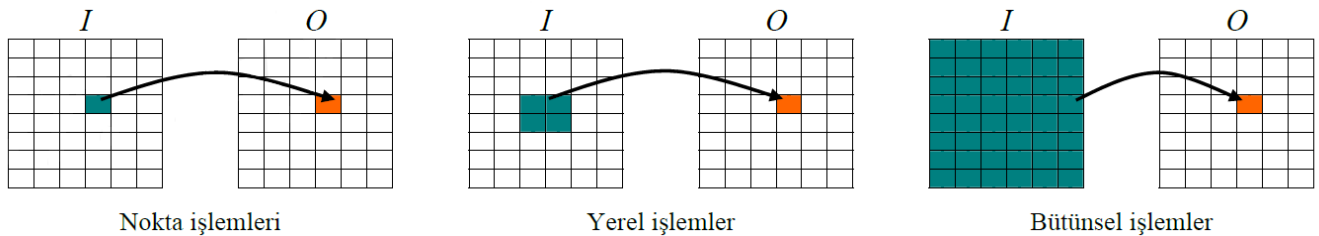
Piksel üzerinde yapılan işlemler

Görüntü işleme aşamasında yapılacak işlemlerin tümü görüntüyü oluşturan pikseller üzerinde yani bu piksellerin sahip olduğu renk bilgisi üzerinde gerçekleştirilir. Bu işlemler; nokta işlemleri, yerel (bölgesel) işlemler ve bütünsel işlemler olmak üzere üç grupta toplanabilir.

Nokta İşlemleri: Çıkış resminin bir pikselini oluşturmak için, giriş resminin bir pikseli üzerinde yapılan işlemlerdir. Kendisine komşu olan piksellerden bağımsız olarak pikselin kendi renk bilgisinin değiştirilmesiyle yapılır. Yani girdi görüntüsündeki her piksel üzerinde yapılan işlem çıktı görüntüsündeki karşılığı olan pikseli oluşturur.

Yerel (Local) İşlemler: Burada çıkış resmini oluşturacak bir noktanın rengi, giriş resminde bu noktanın komşularının renk özelliklerine de bağlıdır. Hangi komşuların rengine bağlı olacağı belirlenen maskenin büyüklüğüne bağlıdır. Bu maskeler, görüntüdeki tüm pikseller üzerinde kaydırılarak görüntünün filtrelenmesi sağlanır. Bu anlamda; görüntüdeki bulanıklaşmanın yok edilmesi, gürültünün temizlenmesi, kenar ve bölge özelliklerinin saptanması yerel işlemlere birer örnek olarak verilebilir. Özetle bu işlemde çıktı görüntüsündeki bir pikselin değeri girdi görüntüsündeki birçok pikselin değerine bağlıdır.

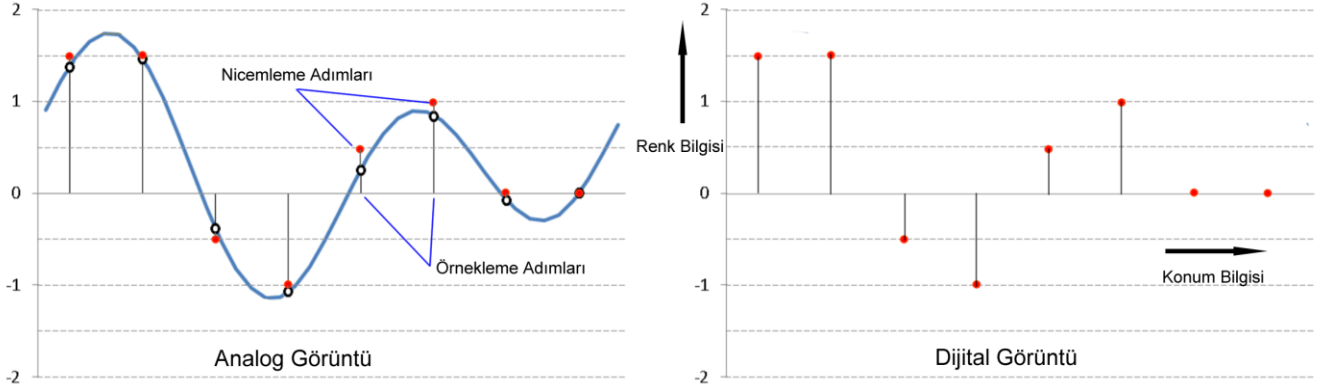
Bütünsel (global) İşlemler: Bütünsel (global) işlemlerde resmin tüm piksellerinin renk özellikleri çıkış resminin oluşturulacak pikselini etkiler. Yani bu işlemde çıktı görüntüsündeki bir pikselin değeri girdi görüntüsündeki tüm piksellerin değerlerine bağlıdır.



Şekil. Sayısal görüntü üzerinde yapılan işlemlerin gösterimi.

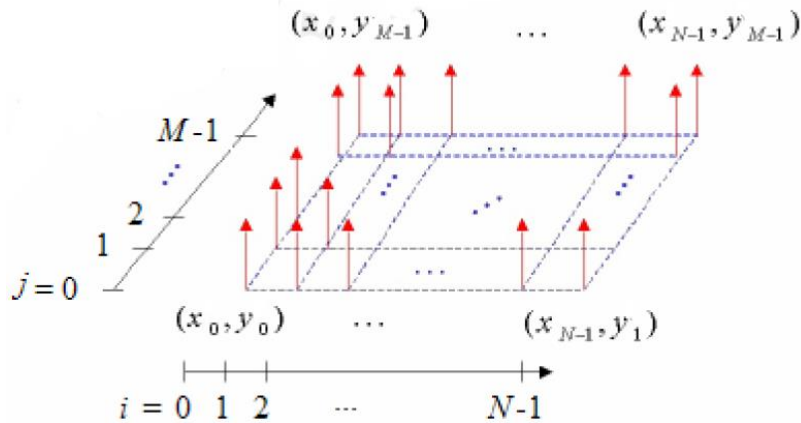
Analog Görüntünün Sayısallaştırılması (dijitale dönüştürülmesi)

Analog görüntülerin bilgisayar ortamında işlenebilmesi için sayısallaştırılmaları gerekir. Sayısallaştırmada resim üzerinde konumsal olarak **örnekleme** (sampling) ve renk derinliği için ise **nicemleme** (quantization/kuantalama) yapılır. Görüntü fonksiyonuna ilişkin koordinatlarının sayısallaştırılması, görüntü örnekleme olarak adlandırılırken; genlik değerlerinin sayısallaştırılmasına görüntü nicemleme adı verilir. Yani aşağıdaki grafikte anlatacak olursak, x ekseninde alınan her adımdaki değer örnekleme temsili eder, y ekseninde ise alabileceğim en yakın değer ise (genlik değerini en yakın sayıya yuvarlatmak gibi) nicemleme olarak adlandırılır.



Görüntü Örnekleme (Sampling):

Sayısal bir görüntü, sürekli bir görüntü fonksiyonu üzerinden eşit aralıklarla x-ekseni boyunca N adet örnek ve y-ekseni boyunca M adet örnek alınarak oluşturulabilir. Böylece, sürekli-zamanlı görüntü fonksiyonundan ayrık-zamanlı görüntü fonksiyonuna geçiş gerçekleşmiş olur. 2-B ayrık-zamanda yatayda N ve dikeyde M örnekten oluşan toplam $N \times M$ sonlu örnek değeri ile analog bir görüntü yaklaşık olarak ifade edilebilir. Bu işlemde, analog görüntü fonksiyonu düzgün örneklenmiş olur. Yani düzgün örnekleme, analog görüntüden hem yatay hem de dikey yönde eşit aralıklarla örnek alınarak oluşturulur. Oluşan dijital (sayısal) görüntü aslında N satır ve M sütundan oluşan bir matristir. Bu işlemde bilgi kaybı vardır.



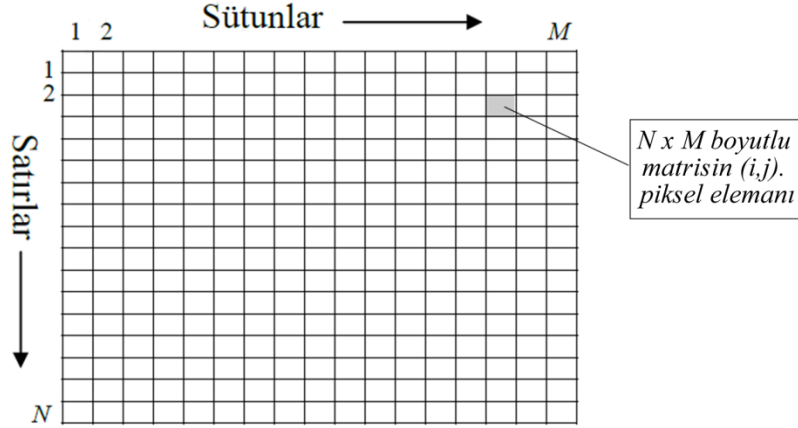
Şekil. Analog görüntünün düzgün örnekler alınarak sayısallaştırılması.

Görüntü Nicemleme (Quantization):

Görüntünün her bir elemanın (pikselin) **parlaklık şiddetini** gösteren pozitif tamsayı değeri, nicemleme ile belirlenir. Görüntü elemanının en küçük ve en büyük genlik değerleri aralığı **basamaklara** ayrılarak, ilgili basamak değerine en yakın olan görüntü değerini almasıdır. Yani analog renk değerinin, geçerli en yakın dijital renk değerine dönüştürülmesidir.

Bu iki işlem sonucunda, bilgisayarlar tarafından işlenebilen sayısal bir görüntü elde edilmiş olur.

Örnekleme ve niceme işleminden sonra elde edilen sayısal görüntü, bileşenleri pozitif tamsayı değerlerinden oluşmuş **iki-boyutlu matris** yapısındadır. Sayısal görüntüyü temsil eden matrisin her bir elemanı piksel olarak adlandırılır. Piksel, sayısal bir görüntüyü oluşturan en küçük eleman olup bir pikselin sahip olduğu değer ilgili görüntü elemanının parlaklık şiddetini belirtir. Parlaklık şiddeti ile ilgili olan pozitif tamsayı değeri, niceme ile belirlenir.



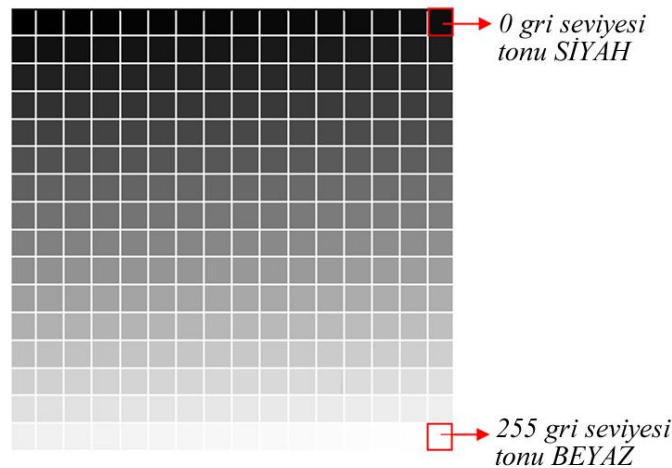
Şekil. $N \times M$ büyüklüğüne sahip 2-B bir sayısal görüntünün temel yapısı.

Gri Görüntü:

Sayısallaştırma işleminde, görüntü boyutlarının ve her bir pikselin sahip olabileceği **parlaklık değeri** belirlenmesi gerekir. Sayısal görüntünün her bir pikselinin sahip olduğu parlaklık değeri **gri seviyeler** olarak adlandırılır. Her bir pikseldeki parlaklık değerinin kodlandığı **bit sayısına göre** gri seviye aralığı belirlenir.

Gri seviye sınırlarında iki renk vardır, **siyah** ve **beyaz**. Bu ikisi arasında kodlanan görüntülere ise gri-ton (gray scale, monochromatic) görüntüler adı verilir.

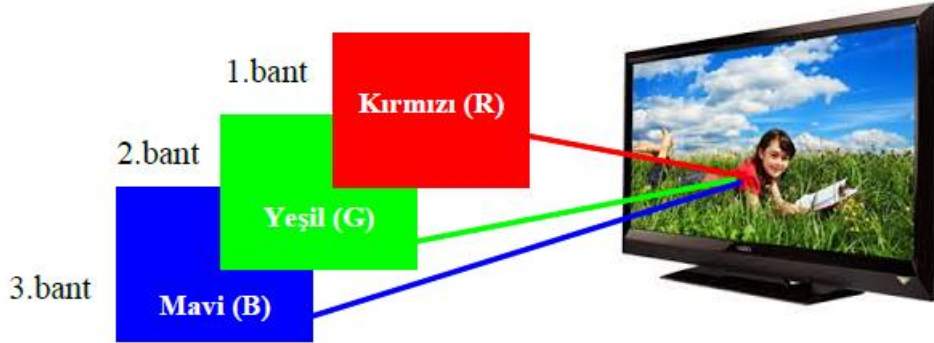
Uygulamada yaygın olarak kullanılan her bir piksel **8 bit** ile kodlanmıştır. Bu tip görüntülerde her bir piksel $2^8 = 256$ farklı gri ton karşılığı (parlaklık seviyesi) değerlerinden oluşur ve gri değer aralığı $G = \{0, 1, 2, \dots, 255\}$ biçiminde ifade edilir. Kural olarak; 0 gri seviyesi siyah renge, 255 gri seviyesi ise beyaz renge ve bu değerler arasındaki gri seviyeler ise gri tonlara karşılık gelir. Şekil de $N \times M = 16 \times 16$ 'lık bir ızgara üzerinde 256 farklı gri seviyenin gösterimi verilmiştir.



Şekil. 16×16 'lık bir ızgara üzerinde 256 farklı gri seviyenin gösterimi.

Renkli Görüntü:

Renkli görüntüler, R(Kırmızı), G(Yeşil), B (Mavi) kodlanmış aynı cisme ait üç adet gri düzeyli görüntünün üst üste ekranda gösterilmesi ile oluşur. Renkli görüntüyü oluşturan bu üç renk bant olarak isimlendirilir.



Şekil. Renkli görüntünün bilgisayar ekranındaki oluşumu.

Renkli görüntülerin her bir pikseli bilgisayar ekranlarında 24 bit'lik veri olarak görüntülenir. Şöyle ki, her bir renk 8 bit ($2 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \cdot 2 = 2^8 = 256$) ile kodlanacağına göre üç renk (RGB) $3 \times 8 = 24$ bit ile kodlanacaktır. Bu durumda, RGB görüntülerin her bir pikseli $2^8 \cdot 2^8 \cdot 2^8 = 2^{24} = 16.777.216$ (yaklaşık 17 milyon) farklı renk değerine sahip olabilir ve bu üç rengin birleşiminin değer aralığı RGB = (0, 0, 0),..., (255, 255, 255) biçiminde gösterilir. Aşağıdaki tabloda bazı örnek renkler ve değerleri verilmiştir.

Tablo. Bazı temel renklerin RGB değerleri

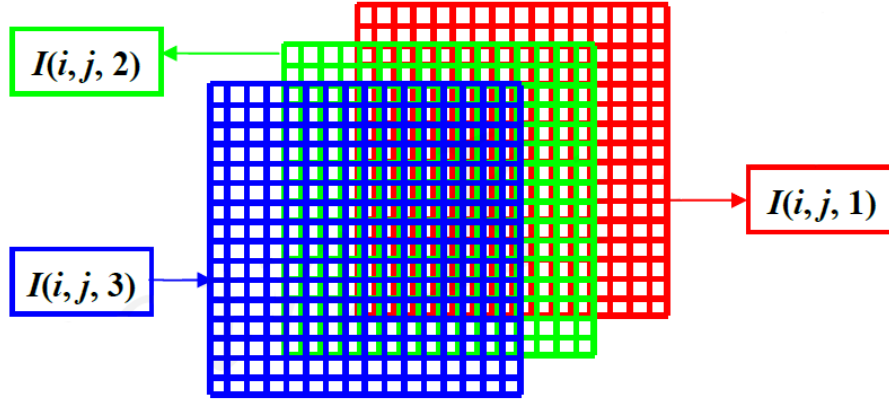
Renk	R	G	B	Görünüm
Kırmızı	255	0	0	
Yeşil	0	255	0	
Mavi	0	0	255	
Beyaz	255	255	255	
Siyah	0	0	0	
Açık Gri	200	200	200	
Koyu Gri	100	100	100	
Sarı	255	255	0	
Turkuaz	0	255	255	
Eflatun	255	0	255	

Matris anlamında düşünüldüğünde doğal renkli 2 Boyutlu bir RGB görüntü, her biri ($N \times M$) büyüklüğünde üç matrisin (ızgaranın) bileşiminden oluşur ve matris gösteriminde $\{I(i, j, k) \mid i = 1, 2, \dots, N; j = 1, 2, \dots, M; k = 1, 2, 3\}$ biçiminde temsil edilir. Genel olarak bu matrislerin her biri, pikselleri 256 seviyeden birine sahip görüntüyü temsil eder.

$I(i, j, 1); i = 1, 2, \dots, N, j = 1, 2, \dots, M \rightarrow$ Kırmızı banda ilişkin matris

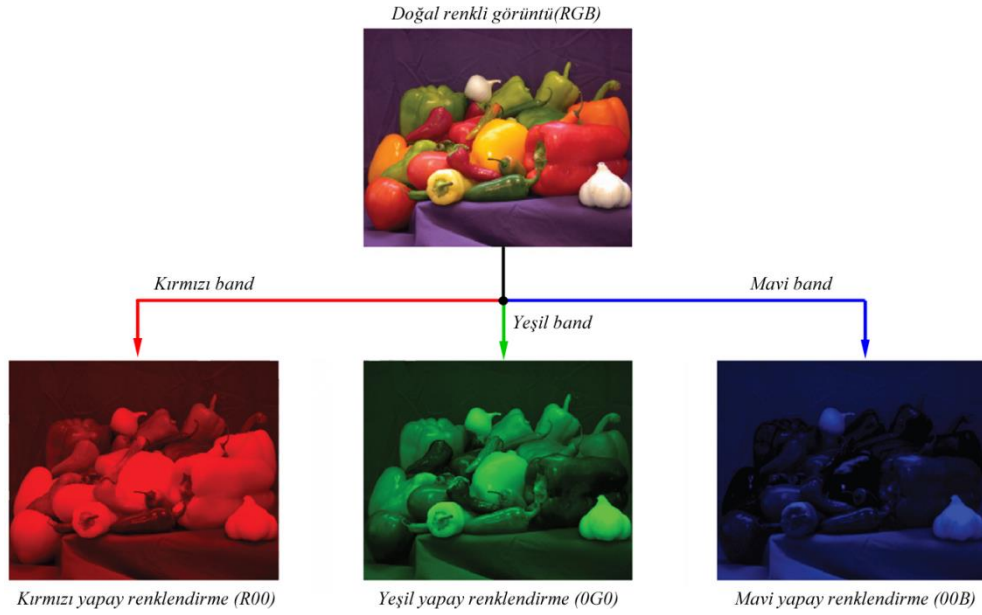
$I(i, j, 2); i = 1, 2, \dots, N, j = 1, 2, \dots, M \rightarrow$ Yeşil banda ilişkin matris

$I(i, j, 3); i = 1, 2, \dots, N, j = 1, 2, \dots, M \rightarrow$ Mavi banda ilişkin matris



Şekil. RGB görüntüsünü oluşturan NxM boyutlarındaki 3 bandın matris gösterimi

Renkli görüntünün yeşil ve mavi bantlarına ilişkin görüntülerin sıfır alınmasıyla sadece kırmızı bandın görüntüsü elde edilir. Böylece, RGB gösterimi kırmızı yapay renklendirilmiş görüntüsüne ulaşır. Bu durumda kodlama $RGB = R00 = (0, 0, 0), \dots, (255, 0, 0)$ şeklinde olacaktır. Benzer şekilde, yeşil ve mavi yapay renklendirilmiş görüntüler de sırasıyla $RGB = 0G0 = (0, 0, 0), \dots, (0, 255, 0)$ ve $RGB = 00B = (0, 0, 0), \dots, (0, 0, 255)$ olacaktır. Bu şekilde oluşan yapay renklendirme görüntüleri aşağıdaki gibi olacaktır.



Şekil. Doğal renkli RGB bir görüntünün ana renklerle (Kırmızı-yeşil-mavi) yapay renklendirilmesi

RGB formatında doğal renklerden oluşmuş renkli bir görüntü için **bantların doğru birleşimi 1-2-3 (Red, Green, Blue)** sırasıyla olmalıdır. Aşağıdaki şekilde üç rengin bir araya geldiği doğal rengin gri-ton görüntüsü ve **her bir bandın** ayrı ayrı **gri-ton görüntüsü** verilmiştir.

RGB görüntü (doğal renk)



Kırmızı (R) bandın (1. bant) gri-ton görüntüsü



Yeşil (G) bandın (2. bant) gri-ton görüntüsü



Mavi (B) bandın (3. bant) gri-ton görüntüsü



Şekil. Doğal renkli bir görüntü ve bu görüntüye ait RGB bantlarının gri-ton görüntüler

Eğer RGB gösterimde bant birleşimlerinin **yerleri değiştirilecek olursa** renkler de değişecektir. Bu şekilde oluşan görüntülere yapay renkli görüntüler adı verilir. Aşağıdaki resimler bununla ilgili örnekleri göstermektedir.

Doğal renkli görüntü;
Bant birleşimi (1-2-3)Yapay renkli görüntü;
Bant birleşimi (2-1-3)Yapay renkli görüntü;
Bant birleşimi (1-3-2)Yapay renkli görüntü;
Bant birleşimi (2-3-1)

Şekil. Doğal renkli RGB bir görüntü ile bu görüntüden türetilen yapay renkli görüntüler

Sayısal Görüntünün Kapladığı Bellek Alanı

N satır ve M sütundan oluşan ve her bir pikseli m bit ile temsil edilen ($N \times M$) büyüklüklü sayısal bir görüntüyü depolamak için gerekli bit sayısı aşağıdaki gibi hesaplanır:

$$\text{Bit sayısı} = N \times M \times m$$

8 bit = 1byte, 1024 byte = 1Kbyte, 1024 Kbyte = 1Mbyte olarak hesaplanır.

Örneğin her pikseli 8 bitle temsil edilen Gri bir resim $N \times M = 1000 \times 600$ piksel boyutlarında ise

$$b = 1000 \times 600 \times 8 = 4.800.000 \text{ bit} / 8 = 600.000 \text{ byte} = 600 \text{ Kbyte} = 0,6 \text{ Mbyte olur.}$$

Aynı resmi 24 bitlik bir kodlama ile (renkli resim) depolarsak (her bir pikseli $RGB = 2^8 + 2^8 + 2^8 = 2^{24} = 16.777.216$ milyon renkle gösterilir)

$$b = 1000 \times 600 \times 24 = 14.400.000 \text{ bit} / 8 = 1.800.000 \text{ byte} = 1.800 \text{ Kbyte} = 1,8 \text{ Mbyte olur.}$$

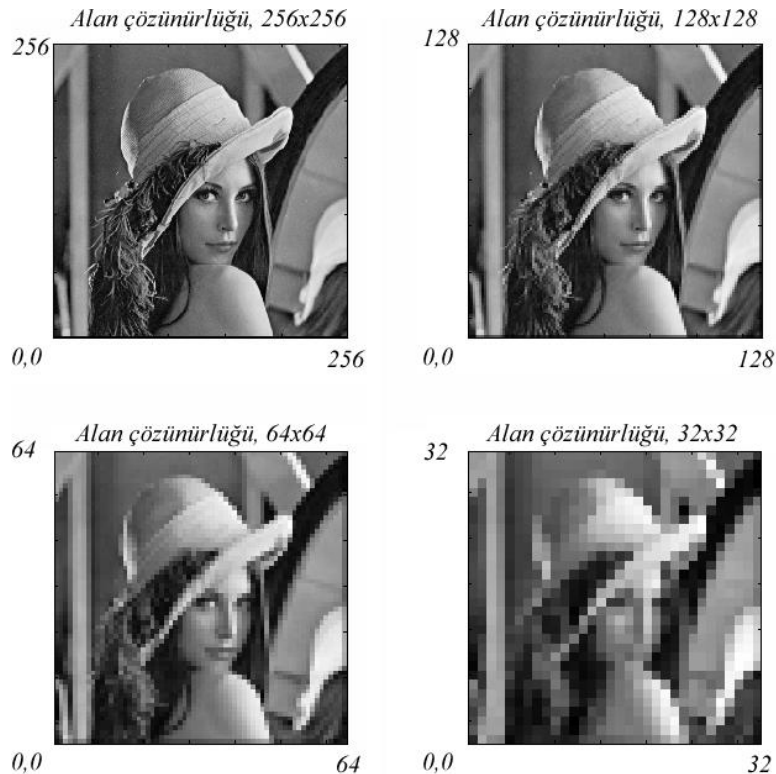
Her bir pikseli kodlamak için gerekli bit sayısı arttıkça görüntüyü depolamak için gerekli olan bellek miktarı da artmaktadır. Bunun bir sonucu olarak, **renkli** görüntüler, **gri-ton** görüntülere göre bellek anlamında bilgisayarlarda daha fazla yer kaplarlar. Örneğin; 8 bitlik bir resim 3Mbyte olurken, gri-ton bir görüntü için bu değer 1Mbyte'dır.

Çözünürlük

Bir görüntünün çözünürlüğü, görüntü içerisindeki ayrıntıların fark edilebilme derecesidir. Bu kavram hem Alan çözünürlüğünü hem de Parlaklık çözünürlüğünü içinde barındırır. Alan çözünürlüğü yüzeyden alınan örnek sayısı ($N \times M$) ile Parlaklık çözünürlüğü ise piksel üzerindeki rengin gri seviye karşılığı olarak parlaklığını belirtir. Bu parametrelerin değerlerindeki artış ne kadar fazla olursa sayısallaştırılmış görüntü orijinaline o oranda yaklaşıp. Bununla birlikte, görüntünün kapladığı bellek alanı hızlı bir şekilde artar.

Alan Hassaslığı (Piksel çözünürlüğü)

Alan hassaslığı, sayısal görüntüyü elde etmek amacıyla analog görüntünün yatay/dikey taranması esnasında toplanan örnek sayısı ile ilgilidir. Sayısal görüntüyü oluşturan piksellerin sayısı ($N \times M$) ne kadar fazla olursa orijinal (analog) görüntüye o kadar fazla yaklaşıp. Diğer taraftan, piksellerin sayısı azaltılırsa bu durumda görüntünün uzaysal çözünürlüğü düşer ve görüntü içerisindeki ayrıntılar kaybolmaya başlar. Buna dama tahtası etkisi (checkerboard effect) adı verilir. Bu etki, görüntü içerisinde yapay karelerin oluşması ile sonuçlanır. Aşağıdaki şekilde çeşitli alan çözünürlüğüne sahip 8 bit gri-ton görüntüler verilmiştir. Görüldüğü gibi, piksel sayısı azaldıkça, dama tahtası etkisi artmaktadır.



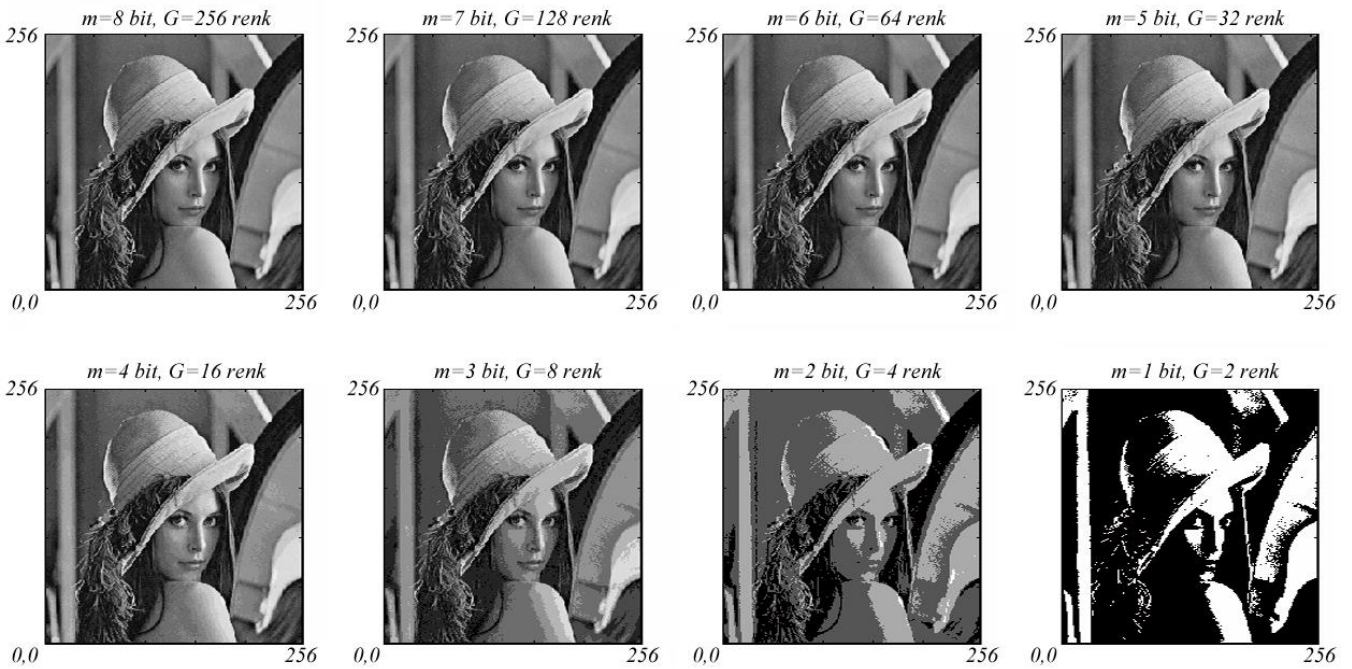
Şekil. Her pikseli 8 bit le temsil edilen, farklı alan çözünürlüklerine sahip örnek resimler.

Parlaklık hassaslığı (gri-seviye renk çözünürlüğü)

Parlaklık hassaslığı, her bir pikselin parlaklığının karşılığı olan gri seviye skalasındaki sayısını gösterir. Yani gri seviye skalasında siyahtan beyaza doğru renk açılırken 256 tane adım (8 bit) kullanılmıştır. Bu skaladaki gri renk yerine, herhangi bir rengin karşılığı o rengin parlaklığını gösterecektir.

Daha önce de ifade edildiği gibi, uygulamada yaygın olarak kullanılan gri-ton görüntülerin gri seviye anlamında tam parlaklık değeri 255 olup, bu değer beyaz renge karşı düşer. Yani, bu tip görüntülerin pikselleri $G = \{0, 1, 2, \dots, 255\}$ gri seviye aralığında olmak üzere 256 farklı değer alabilir ve sonuç olarak her bir piksel $m = 8$ bit ile temsil edilir. Gri seviyelerin sayısı azaldığında, görüntü içerisinde yapay çizgisel hatlar ortaya çıkmaya başlar. Aşağıda şekilde 256 renk (8 bit), 128 renk (7 bit), 64 renk (6 bit), 32 renk (5 bit), 16 renk (4 bit), 8 renk (3 bit), 4 renk (2 bit) ve 2 renk (1 bit) olan aynı alan çözünürlüğüne sahip ($N \times M = 256 \times 256$) gri-ton görüntüler verilmiştir.

Görüntüler incelendiğinde 8 bit ile 5 bit arasındaki farklar çıplak gözle tam anlaşılamamaktadır (renkli resim olsa biraz daha ortaya çıkar). Fakat 4 bit ile 1 bit arasında kalitenin düştüğü bariz şekilde görülebilmekte ve yapay sınırlar farkedilmektedir.



Şekil. Çeşitli gri-seviye renk çözünürlüklerine sahip aynı alan çözünürlüklü gri-ton görüntüler.

Uygun Çözünürlük Değerlerinin Belirlenmesi

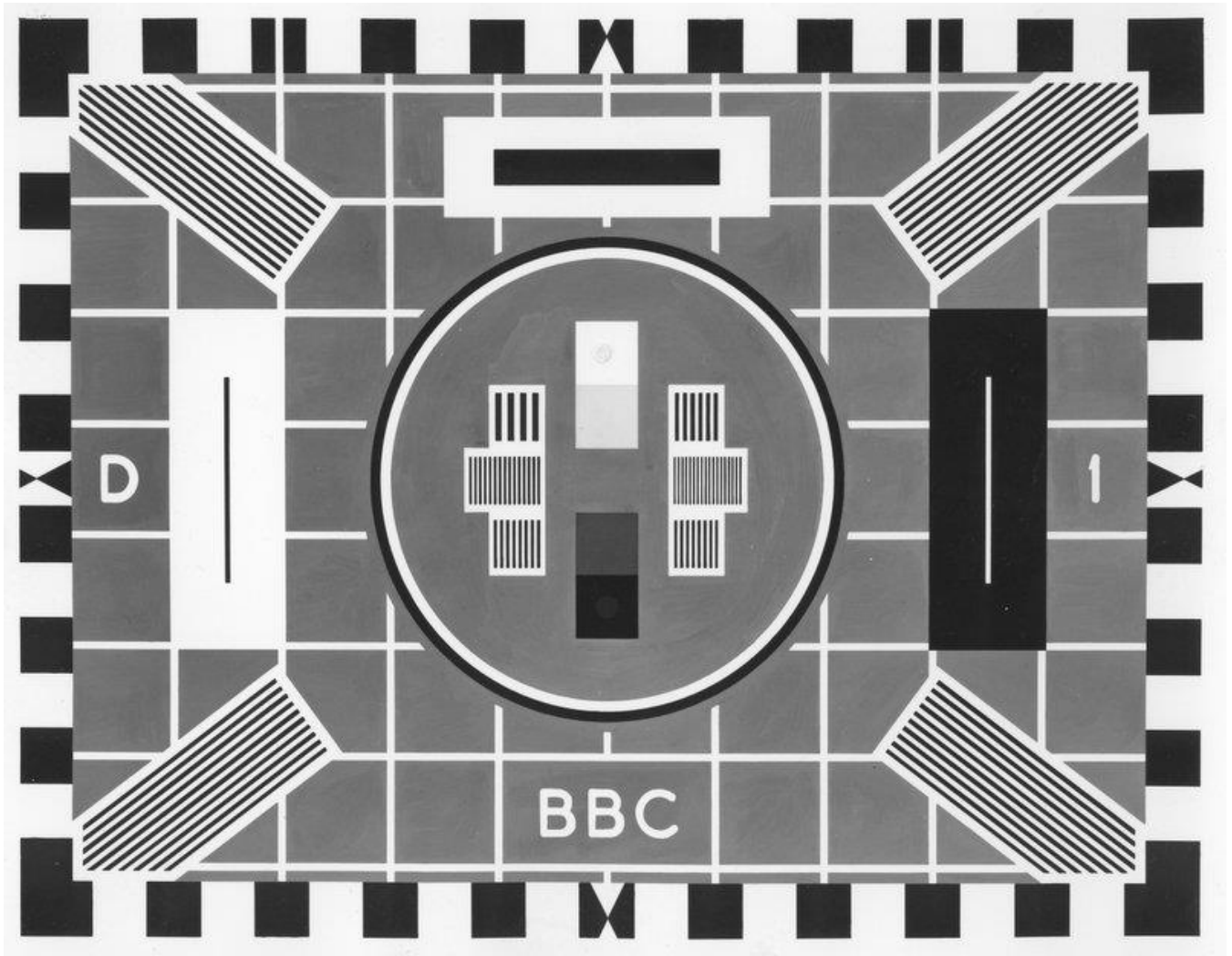
Dijital bir cihazda, görüntülerin belli sayıda örnek ile temsil edildiğini öğrendik. Görüntünün boyutu, alınan örnek sayısı ile belirlenir (örnekleme/sampling). Herhangi bir örnek üzerinde rengin gri ton seviyesi ise yine sınırlı bir sayı ile temsil edilir (nicemleme/quantization).

Analog bir görüntüye (sürekli görüntü) en yakın dijital görüntüye dönüştürmek için örnekleme nokta sayısı (konum nokta sayısı) ve nicemleme seviyesi (renk gri ton seviyesi) ne olmalıdır? Bu sayılar arttıkça çözünürlük artar fakat depolama alanı ve işleme zamanı uzar. Dolayısı ile uygun optimal değerlerde bu işlemi gerçekleştirmeliyiz.

Görüntüleri sınırsız sinyaller olarak modellersek, Fourier analizi bize, görüntü bant genişliklerinin iki katından daha yüksek frekanslarda örnekleme hatası olarak kurtarmamıza olanak verdiğini söyler (örneklenen görüntüyü düşük geçişli filtreleyerek). Ancak, örnekleme frekansları bu **Nyquist sınırının** altındaysa, spektrum bozulur ve bilgi kaybolur. Bu olay aliasing (bilgi kaybolması) olarak bilinir. Nyquist sınırı, örnekleme frekansının, analog frekanstan en iki kat daha fazla olmasını söyler (bilgi: <https://www.youtube.com/watch?v=QUqj5paVCjM>).

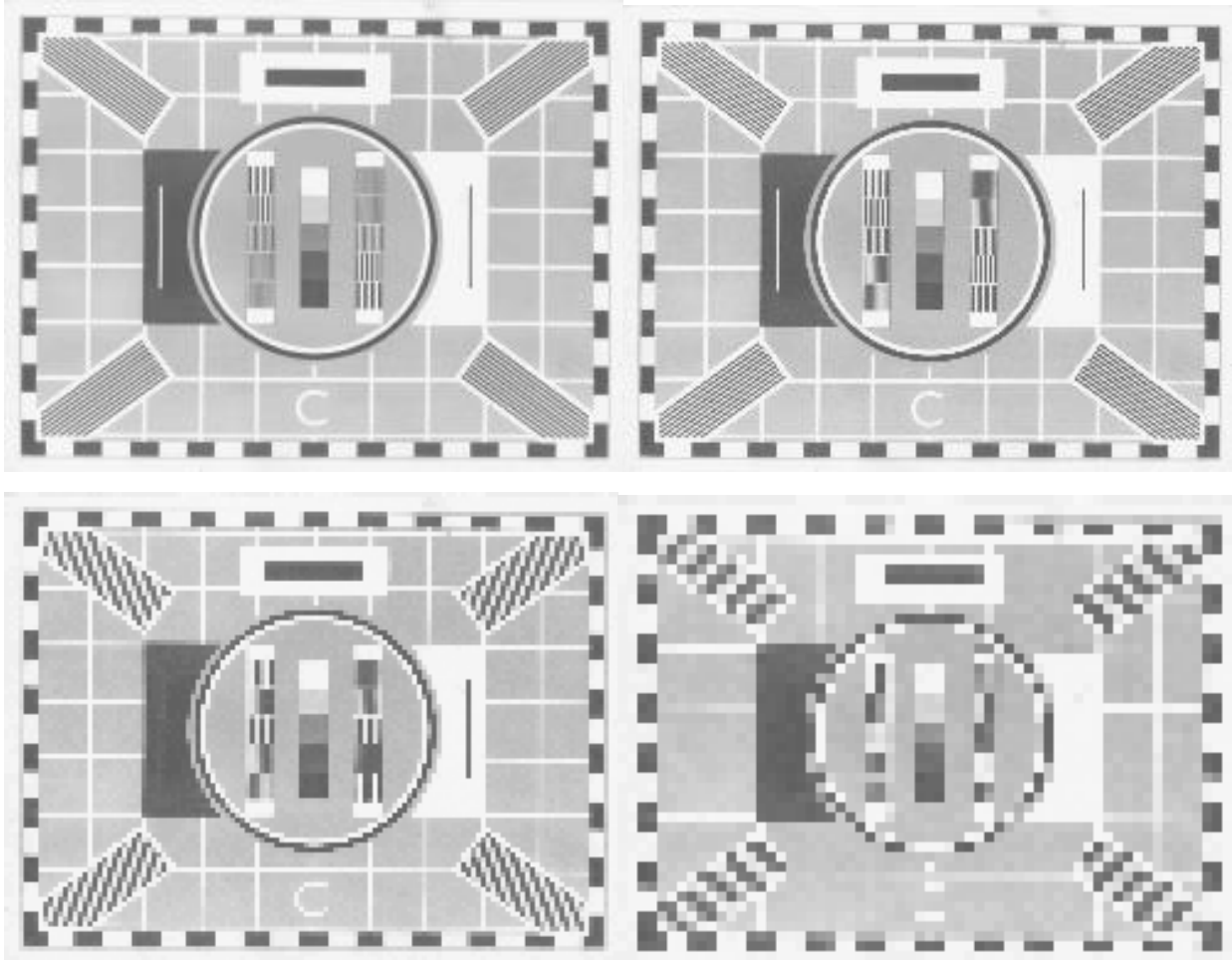
Uygulama

Aşağıdaki test ekranını kullanarak çeşitli örnekleme test sonuçlarını görmeye çalışın.



Örnekleme ızgarası boyutunu yarıya indirmenin etkisi ilk resimde verilmiştir.

Maksimum örnekleme seviyesi veya nicemleme seviyesi bu örnekte sabit tutulmuştur. Ayrıca, her örnekte kullanılan görüntüleme alanı aynı olduğundan, görüntüleme alanını doldurmak için düşük çözünürlüklü görüntülerdeki pikseller çoğaltılır. Çözünürlüğün 4, 8 ve 16 faktörü ile azaltılmasının etkileri aşağıdaki resimlerde sırasıyla verilmiştir.



2-Boyutlu sayısal görüntülerin matris olarak gösterimi

Daha önce de ifade edildiği gibi $(N \times M)$ piksele sahip 2-B sayısal bir görüntü, N satır ve M sütundan oluşan bir matris gibi düşünülebilir. Matrisin bileşenleri, satır ve sütunların kesiştiği her bir noktada –ki bu noktalar sayısal görüntünün en küçük parçası olan pikseli temsil eder – pozitif tamsayı değerlerine sahiptir. Örneğin, 256 adet gri seviye içeren görüntüler için bu değer 0 – 255 aralığındaki tamsayılardan oluşur.

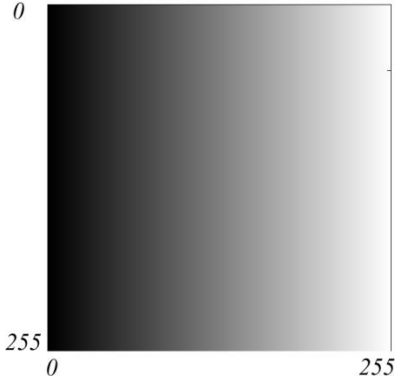
Sayısal bir görüntüyü temsil eden matrisin bileşenleri hiçbir zaman negatif ve tamsayı dışındaki değerlerden oluşamaz. Ancak, görüntü üzerinde işlemler yapıldıktan sonra elde edilen yeni görüntü içerisinde bu gibi sonuçlarla karşılaşmak olasıdır. Böyle durumlarda, yeni görüntüyü bilgisayar ekranında doğru bir biçimde görüntüleyebilmek için matris değerleri üzerinde uygun ölçeklendirme ve yuvarlatma işlemleri yapılmalıdır.

$$A = \begin{matrix} \xrightarrow{256 \text{ sütun}} \\ \begin{bmatrix} 0 & 1 & \dots & 255 \\ 0 & 1 & \dots & 255 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 1 & \dots & 255 \end{bmatrix} \downarrow 256 \text{ satır} \end{matrix}$$

UYGULAMALAR

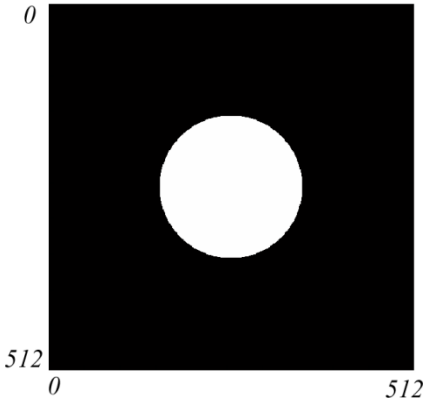
Ödev 1:

$A = \begin{bmatrix} 0 & \dots & 255 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 255 \end{bmatrix}$ şeklinde verilen matrisin görünümünü C# programı kullanarak oluşturunuz. 255 sayısını değişkene bağlı olarak oluşturun.



Ödev 2:

$A(i,j) = \begin{cases} 1; & \text{eğer } \sqrt{(i-256)^2 + (j-256)^2} < 100 \\ 0; & \text{diğer} \end{cases}$ matrisi ile verilen $i=1,2,\dots,512$ ve $j=1,2,\dots,512$ değerleri için oluşan şekli programlayarak gösteriniz. 512 sayısını değişkene bağlı olarak oluşturun.

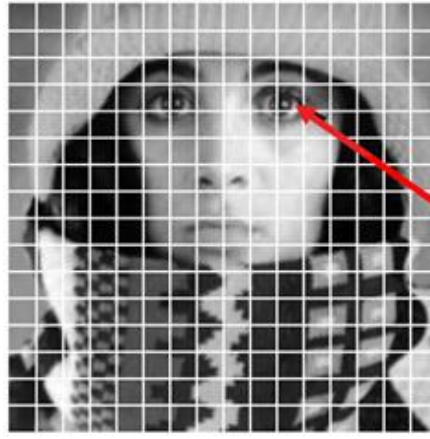


Ödev 3:

1000x1000 piksel büyüklüğündeki bir resmi 2,5,10,25,50, piksel aralıklarla örnekleyerek yeni oluşan resimlerin durumunu gösteriniz. Aynı zamanda 0-255 aralığındaki her renk kodu en yakın 10 katları değerine yuvarlatılarak örnekleyin. Örneğin R=63 değeri R=60 olsun. Renk nicemlemesi yaparken (10, 50 ve 100) katları şeklinde en yakın renge yuvarlayın. Değişimleri gözlemleyin.

Ödev 4:

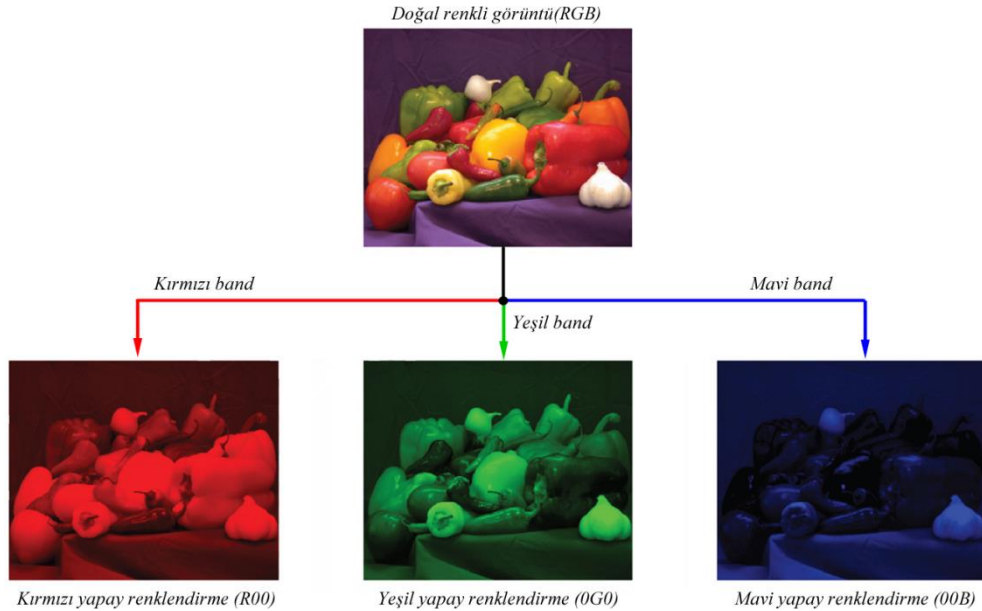
Aşağıdaki gibi bir resmi önce renkli resimden gri renge dönüştürün. Daha sonra bu resmi belli aralıklarla mozaik haline getirin. Bu mozaik alanların ortalama rengini bularak bunlardan yeni bir resim oluşturun. (Daha küçük boyutta olacaktır). Burada mozaik boyutlarını 2, 5, 10,25 pikseller şeklinde alarak ortaya çıkan yeni resimlerin nasıl görüldüğünü gösterin. Şekil görüntünün $(i, j) = (4, 11)$ koordinatlarındaki mozayığın gri seviye değerinin 110 olduğu görülmektedir.



Piksel gri
seviye değeri
 $f_{gray}(4,11)=110$

Ödev 5:

Renkli görüntünün yeşil ve mavi bantlarına ilişkin görüntülerin sıfır alınmasıyla sadece kırmızı bantın görüntüsü elde edilir. Böylece, RGB gösterimi kırmızı yapay renklendirilmiş görüntüsüne ulaşır. Bu durumda kodlama $RGB = R00 = (0, 0, 0), \dots, (255, 0, 0)$ şeklinde olacaktır. Benzer şekilde, yeşil ve mavi yapay renklendirilmiş görüntüler de sırasıyla $RGB = 0G0 = (0, 0, 0), \dots, (0, 255, 0)$ ve $RGB = 00B = (0, 0, 0), \dots, (0, 0, 255)$ olacaktır. Bu şekilde oluşan yapay renklendirme görüntüleri aşağıdaki gibi olacaktır. Bu görüntüleri verecek programı yazınız.



Ödev 6:

Aşağıdaki gibi renkli bir görüntünün her bir bant renginin gri tonlarda gösterimini elde ediniz. Kırmızı bantı oluştururken diğer iki bantın değerini de kırmızı ile aynı yapınız. Benzer işlemleri Yeşil ve Mavi bantlarda da deneyiniz.

RGB görüntü (doğal renk)*Kırmızı (R) bandın (1. bant) gri-ton görüntüsü**Yeşil (G) bandın (2. bant) gri-ton görüntüsü**Mavi (B) bandın (3. bant) gri-ton görüntüsü***Ödev 7:**

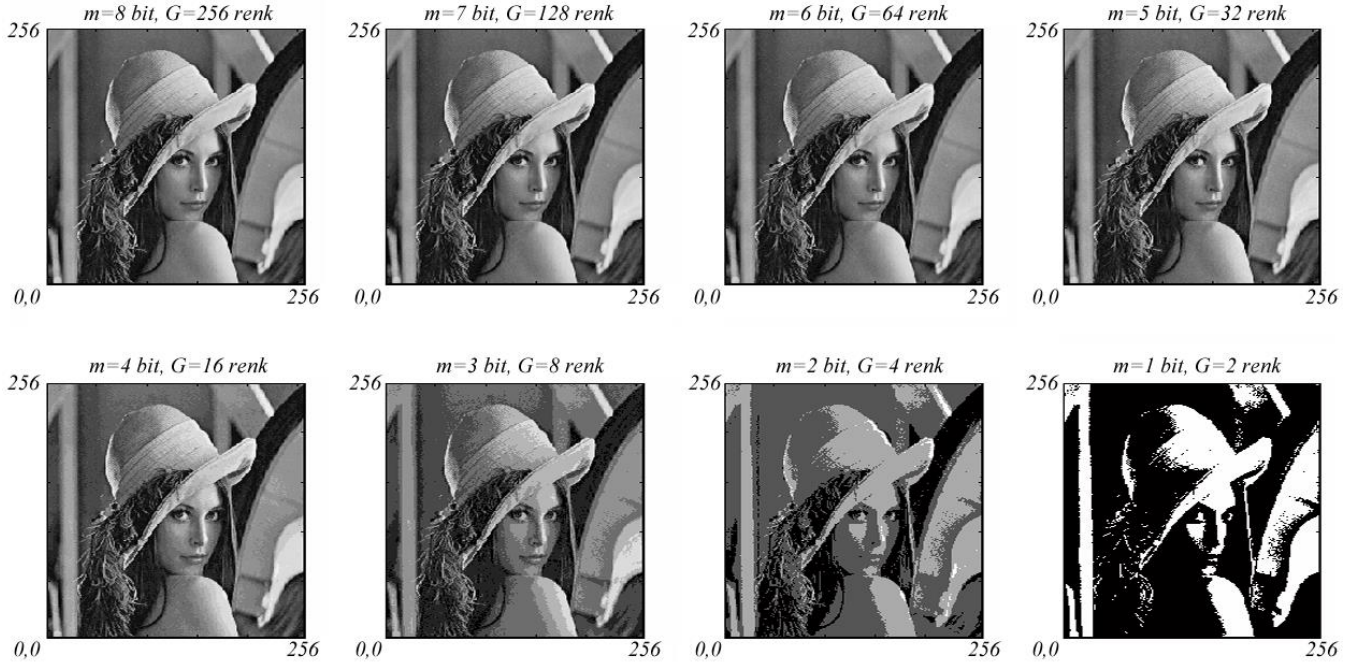
Eğer RGB gösterimde bant birleşimlerinin yerleri değiştirilecek olursa renkler de değişecektir. Bu şekilde oluşan görüntülere yapay renkli görüntüler adı verilir. Aşağıdakine benzer yapay renkli görüntüleri oluşturacak programı yazınız. Bunun için orjinal rengin RGB değerlerini aldıktan sonra bu değerlerin sıralamasını değiştirin ve yeni oluşturulan RGB sırasıyla tekrar görüntüleyin. Burada RGB, GRB yapılırsa yeni renk kodunda Yeşilin rengi kırmızının yerine, kırmızı ise yeşilin yerine yazılmalıdır.

*Doğal renkli görüntü;
Bant birleşimi (1-2-3)**Yapay renkli görüntü;
Bant birleşimi (2-1-3)**Yapay renkli görüntü;
Bant birleşimi (1-3-2)**Yapay renkli görüntü;
Bant birleşimi (2-3-1)***Ödev 8:**

Uygulamada yaygın olarak kullanılan gri-ton görüntülerin gri seviye anlamında tam parlaklık değeri 255 olup, bu değer beyaz renge karşı düşer. Yani, bu tip görüntülerin pikselleri $G = \{0, 1, 2, \dots, 255\}$ gri seviye aralığında olmak üzere 256 farklı değer alabilir ve sonuç olarak her bir piksel $m = 8$ bit ile temsil edilir. Gri seviyelerin sayısı azaldığında, görüntü içerisinde yapay çizgisel hatlar ortaya çıkmaya başlar. Aşağıda şekilde 256 renk (8 bit), 128

renk (7 bit), 64 renk (6 bit), 32 renk (5 bit), 16 renk (4 bit), 8 renk (3 bit), 4 renk (2 bit) ve 2 renk (1 bit) olan aynı alan çözünürlüğüne sahip ($N \times M = 256 \times 256$) gri-ton görüntüler verilmiştir.

Aynı bit değerleri kullanılarak RGB şeklindeki renkli bir görüntüdeki değişimi de gösterin.



Okunan resmi istenen Gri skala renk değerinde görüntülemek için aşağıdaki kodları kullanarak deneyin.

```
private void bitToolStripMenuItem_Click(object sender, EventArgs e)
{
    Bitmap bmp = new Bitmap(pictureBox_Gelen.Image, 256, 256);
    pictureBox_Montajli.Image = bmp;
}

private void bitToolStripMenuItem1_Click(object sender, EventArgs e)
{
    Bitmap bmp = new Bitmap(pictureBox_Gelen.Image, 64, 64);
    pictureBox_Montajli.Image = bmp;
}

private void bitToolStripMenuItem2_Click(object sender, EventArgs e)
{
    Bitmap bmp = new Bitmap(pictureBox_Gelen.Image, 16, 16);
    pictureBox_Montajli.Image = bmp;
}

private void bitToolStripMenuItem3_Click(object sender, EventArgs e)
{
    Bitmap bmp = new Bitmap(pictureBox_Gelen.Image, 4, 4);
    pictureBox_Montajli.Image = bmp;
}

private void bitToolStripMenuItem4_Click(object sender, EventArgs e)
{
    Bitmap bmp = new Bitmap(pictureBox_Gelen.Image, 2, 2);
    pictureBox_Montajli.Image = bmp;
}
```