



# BSM 409 GÖRÜNTÜ İŞLEME

Hafta - 6

Grileştirme Yöntemleri – Eşikleme Yöntemleri – İki Görüntü İşlemleri

Doç. Dr. Eftal ŞEHİRLİ

Bilgisayar Mühendisliği Bölümü



# İçerik

- RGB – Gri seviye dönüşümü
- Gri seviye – ikili (binary) görüntü dönüşümü
- İki görüntü işlemleri
  - Toplama
  - Çıkarma
  - Kesişim
  - Birleşim
  - Fark



# Gri Seviye Görüntüye Dönüşüm

- Elde edilen görüntüler içerisinde aranan nesnelerin performanslı olarak bulunabilmesi için görüntünün olabildiğince basite indirgenmesi gerekir.
- İkili görüntüye ulaşmadan önce renkli görüntü gri görüntüye çevrilerek diğer işlemler için hazırlık yapılır.
- Gri seviye dönüşüm yöntemleri temelde aynı mantığa dayansa bile, işlem aşamaları ve işlemler sonrası oluşan çıktı farklıdır.

# Ortalama Değer Yöntemi

- Ortalama (average) değer  
$$T = ( R + G + B ) / 3$$
- Ortalama değer yönteminde görüntüyü oluşturan her bir piksel için RGB değerleri ayrı ayrı elde edilerek toplanır ve üçe bölünerek ortalamaları alınır.
- Bu yöntem en basit yöntemlerden birisidir.
- Her bir piksel için bulunan bu ortalama değeri, pikselin gri değeri olarak belirlenir.
- Görüntü üzerindeki tüm pikseller üzerinde bu işlem uygulanarak renkli görüntü gri görüntüye çevrilmiş olur.
- Bu yöntem tüm renklerden aynı katsayida veri aldığı için küçük detayların kaybolmasına sebep olabilir. Bu yüzden belirli bir rengin daha çok ortaya çıkarılması gereken çözümlerde tercih edilmemelidir.



# BT-709 ve Luma Algoritması

- Ortalama yönteme benzer olarak piksel renk değerlerinin ortalaması alınır ancak insan gözünün algılamasına bağlı olarak ağırlıklı bir ortalama oluşturulur.
- İnsan gözü yeşile daha duyarlı olduğu için yeşil değeri ağırlıklı olarak hesaplanır.
- BT-709 Algoritması  
 $T=(R*0.2125)+(G*0.7154)+(B*0.072)$



- Parlaklık (Luminosity) Yöntemi  
 $T = (R * 0.3 + G * 0.59 + B * 0.11)$



OpenCV, MATLAB kütüphanelerinde varsayılan olarak bu uygulanır.

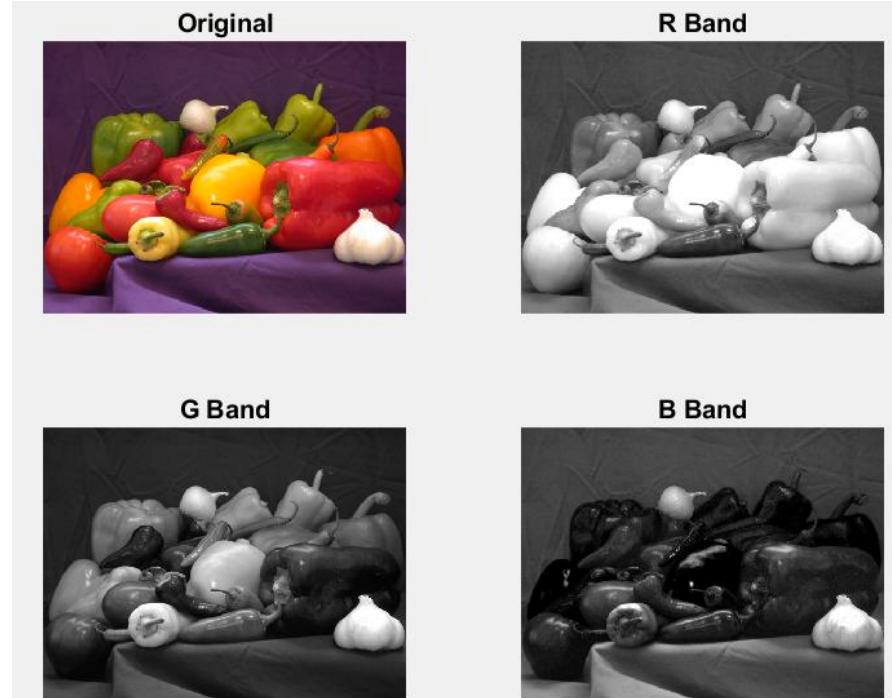
# Açıklık (Lightness) Yöntemi

- En belirgin ve en az belirgin renklerin ortalamalarının alınarak gri değeri bulunur.
- Aşağıdaki eşitlikte olduğu gibi pikselin R,G,B değerlerinden en büyüğü ve en küçüğünün ortalaması alınır.
- $T=(\max(R,G,B) + \min(R,G,B)) / 2$



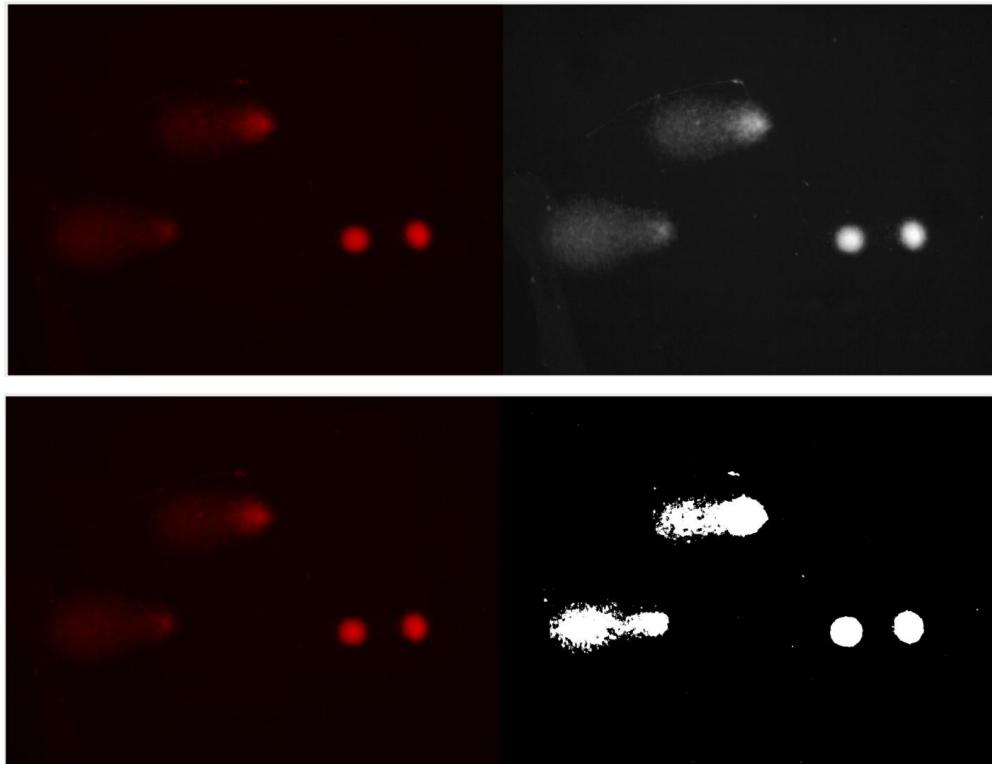
# Renk Kanalı Yöntemi

- Tek renk kanalındaki veriyi kullanarak grileştirme yapılan yöntemdir. Kırmızı, Yeşil ya da Mavi değerlerinden birisi seçilerek gri değeri bu değerlerden birisi olur.
- Diğer yöntemlere göre hesaplama olmadığı için daha hızlıdır.
- Coğu dijital fotoğraf makinesinin kullandığı gri滤resi bu yöntem ile sağlanmaktadır.



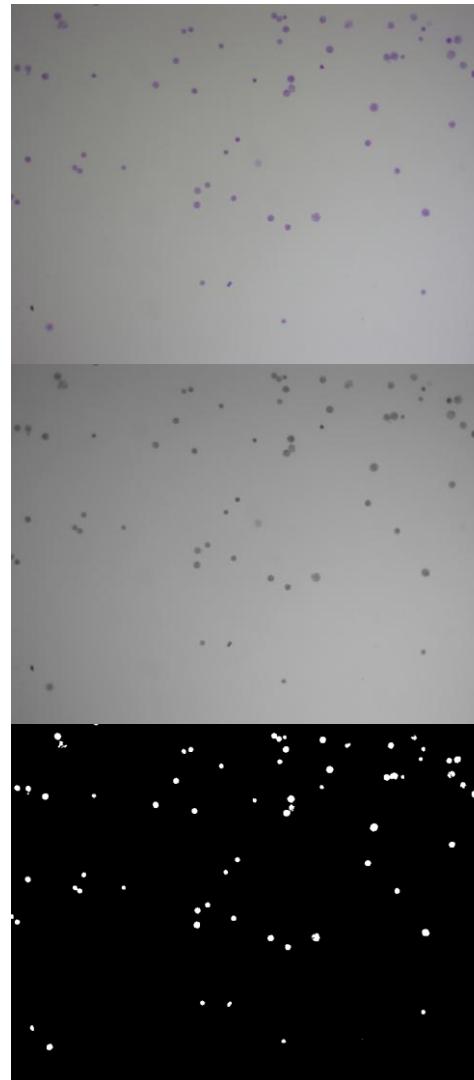
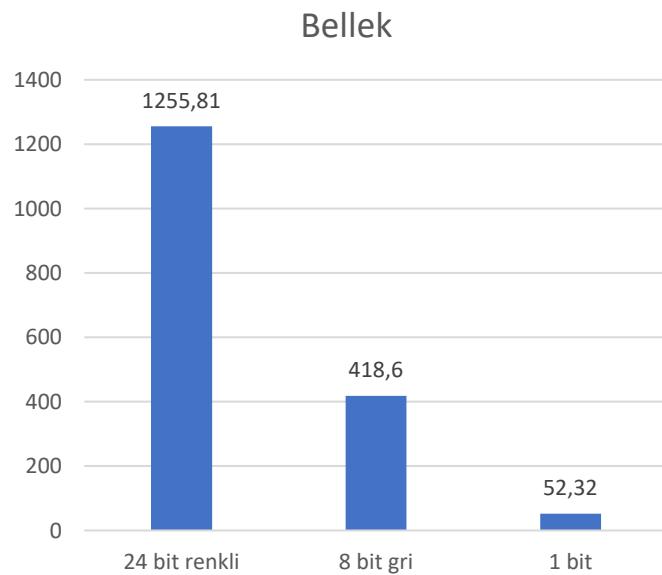
# Eşikleme (Threshold) Yöntemi

- Görüntüler binary (ikili) hale getirilir.
- Eşikleme, görüntüdeki ön planda olan nesneleri (obje) arka plandan ayıran bir yöntemdir.
- Ön tanımlı veya adaptif eşik değerleri seçilebilir.
- Global ve lokal olarak çalışabilirler.
- RGB to Grayscale -> Grayscale to Binary
- Formül:  $T = \begin{cases} 0, & f(x, y) < T \\ 1, & f(x, y) \geq T \end{cases}$



# Neden İhtiyaç Duyulur?

- Daha az kaynak tüketmek
  - Zaman
  - Bellek
  - İşlemci



756x567 24 bit renkli görüntü:  
1255,81 KB

756x567 8 bit gri görüntü:  
418,60 KB

756x567 1 bit binary görüntü:  
52,32 KB

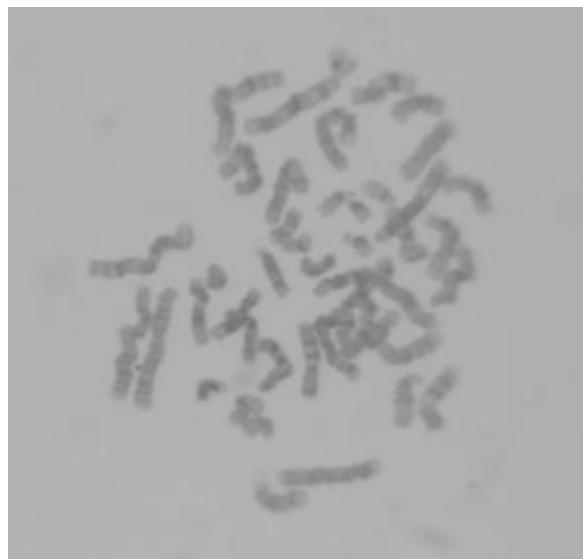


# Ön Tanımlı Threshold Değeri

- Gerçekleştirilme basitliği ve hızı nedeniyle ön tanımlı eşikleme görüntü işleme uygulamalarında önemli bir yere sahiptir.
- Bir kullanıcı tarafından keyfi ya da duruma en uygun eşikleme değeri tanımlanır.
- Ancak otomotize sistemler için çözüm değildir.

# Ön Tanımlı Threshold Değeri

- Orijinal görüntü (solda),  $T=155$  olarak eşiklenen görüntü (sağda).



# Ön Tanımlı Threshold Değeri

- Orijinal görüntü (solda),  $T=128$  olarak eşiklenen görüntü (sağda).





# Otsu Threshold

- Görüntü işleme alanında kullanılan **otomatik** bir eşik (threshold) değeri belirleme tekniğidir.
- Alanda gold standarttır.
- Her olası eşik değeri  $t$  için, sınıflar arası ayrimı ölçer ve sınıflar arasındaki varyansı maksimum yapan  $t$  değerini seçer. Bu da nesne ile arka planı en iyi ayıran eşik değeridir.



# Otsu Threshold

- Her sınıf ( $w$ ) için olasılık ve ortalama hesaplanır.
- Sınıf 1 olasılığı:  $w_1(t) = \sum_{i=0}^t p(i)$
- Sınıf 2 olasılığı:  $w_2(t) = \sum_{i=t+1}^{L-1} p(i)$
- Ortalama
- Sınıf 1 için ortalama:  $\mu_1 = \frac{\sum_{i=0}^t i \cdot p(i)}{w_1(t)}$
- Sınıf 2 için ortalama:  $\mu_2 = \frac{\sum_{i=t+1}^{L-1} i \cdot p(i)}{w_2(t)}$



# Otsu Threshold

- Sınıflar arası Varyans hesaplanır.
- $\sigma_b^2(t) = w_1(t)w_2(t)[\mu_1(t) - \mu_2(t)]^2$
- En iyi eşik (threshold) değer:
- $t = \max \sigma_b^2(t)$



# Otsu Threshold

```
clc; clear; close all;
```

```
% Görüntüyü oku (gri seviyeli)
```

```
I = imread('ornek_resim.png');  
if size(I,3) == 3  
    I = rgb2gray(I);  
end
```

```
% Histogram ve normalize etme
```

```
counts = imhist(I);  
p = counts / sum(counts); % Olasılık dağılımı  
L = 256; % Gri seviye aralığı
```

```
% Genel ortalama
```

```
mu_total = sum((0:L-1).*p);
```

```
sigma_b2 = zeros(L,1); % sınıflar arası varyans için dizi
```

```
% Tüm olası eşikler için hesaplama
```

```
for t = 1:L  
    % Sınıf 1 (0..t)  
    w1 = sum(p(1:t));  
    if w1 == 0  
        continue;  
    end  
    mu1 = sum((0:t-1).*p(1:t)) / w1;  
  
    % Sınıf 2 (t+1..L-1)  
    w2 = sum(p(t+1:end));  
    if w2 == 0  
        continue;  
    end  
    mu2 = sum((t:L-1).*p(t+1:end)) / w2;  
  
    % Sınıflar arası varyans  
    sigma_b2(t) = w1 * w2 * (mu1 - mu2)^2;  
end  
  
% Maksimum varyansı veren t  
[~, T] = max(sigma_b2);  
fprintf('Otsu threshold değeri: %d\n', T-1);  
  
% Eşiklenmiş görüntü  
BW = imbinarize(I, (T-1)/255);
```

# Otsu Threshold

- Orijinal görüntü (solda), *otsu ile* eşiklenen görüntü (sağda).





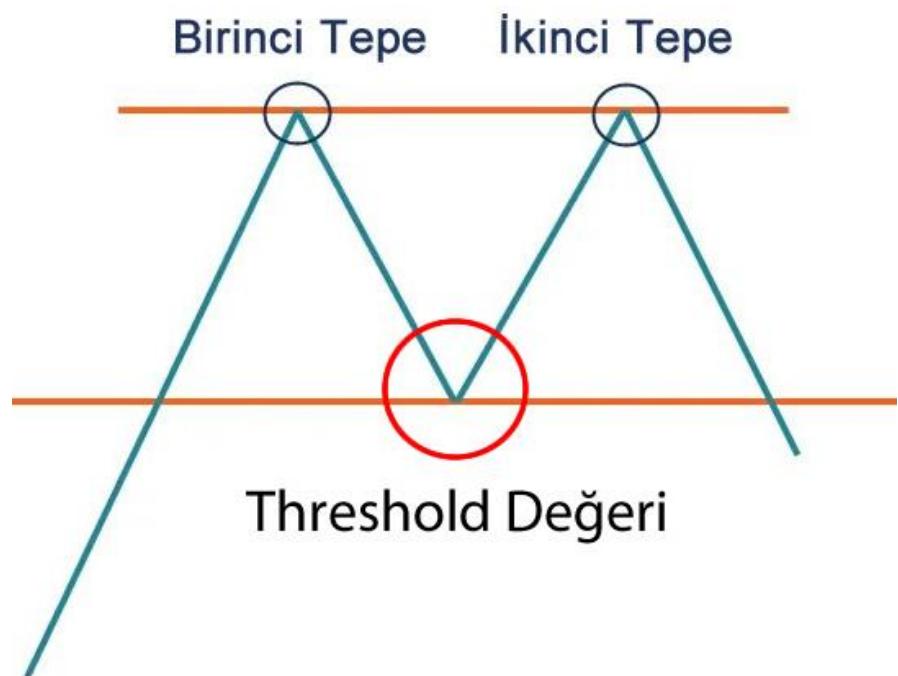
# Randomized Binomial Cascade Mean Thresholding

- 1- Vektör boyutu seçilir. Örnek: 1x4
- 2- İterasyon sayısı seçilir. Örnek: (width x height)/250
- 3- Random dikey ya da yatay yön seçilir. 0: Dikey 1:Yatay
- 4- Görüntü sınırlarını aşmayan random bir piksel seçilir.
- 5- Bu piksel vektörün ilk indeksi olacak şekilde değerlerin binomial olarak ortalaması alınır ve bu değer ortamlar vektörüne atanır.
- 6- 3-5 adımları iterasyon sayısı kadar tekrarlanır.
- 7- İterasyon sayısı boyutundaki ortamlar vektörünün değerleri binomial olarak ortalaması alınır. Bulunan değer threshold değeridir.

10	20	0	80
15	10	40	
12.5 25			
18.75			

# Intermodes Threshold

- Sık kullanılan genel eşikleme yöntemlerindendir ve histogram analizine dayanır.
- İki durumlu bir histogram varsayırlar.
- Bir görüntünün histogramına bakıldığından, eğer histogram **iki belirgin tepeye (bimodal)** sahipse, ön plan ve arka plan bu iki tepeye karşılık gelir.
- Histogram yalnızca iki yerel maksimum değer elde edilinceye kadar üç noktalı ortalamafiltresi kullanılarak tekrarlanır.
- Histogramda iki tepe noktası kaldığında tepe noktaların ortası veya tepe noktalar arasındaki minimum yerel eşik değeri seçilerek eşikleme yapılır.
- Otsu gibi tüm olasılıkları denemez, daha sezgisel çalışır.
- Histogram net iki tepeye sahip değilse (örneğin çok gürültülü veya multimodal histogram), işe yaramaz.
- Tepeleri bulmak için histogramın düzgün şekilde yumusatılması şarttır.





# Global vs Local Threshold

- Eşikleme, yoğunluk değerleri bir eşliğin üzerinde olan tüm pikselleri bir ön plan değerine ve kalan tüm pikselleri bir arka plan değerine ayarlayarak bir görüntüyü bölgelere ayırmak için kullanılır.
- Global yöntemler tüm pikseller için genel bir eşik kullanırken, lokal yöntemler, eşiği görüntü üzerinde dinamik olarak değiştirir.
- Lokal yöntemler daha karmaşık yaklaşım görüntüdeki değişen aydınlatma koşullarına en iyi çözümü getirmeyi amaçlar.

# Bazı Threshold Metotları

- **Global** Thresholding Methods
  - Otsu Threshold Method
  - Iterative Threshold Method
  - Difference Threshold
  - Mean Threshold
  - Minimum Method
  - Maximum Entropy Thresholding
  - Kittler-Illingworth
  - Li's Minimum Cross-Entropy
  - Huang's Fuzzy Method
  - Moments Preserving Method
- **Adaptive** Thresholding Methods
  - Niblack's Method
  - Sauvola's Method
  - Wolf-Jolion Method
  - Bernsen's Method
  - Adaptive Mean Thresholding
  - Adaptive Gaussian Thresholding
  - Bradley Local Thresholding
- **Hybrid** Methods
  - Fuzzy Thresholding
  - Clustering-based (k-means, Otsu + k-means)
  - Neural Network-based Thresholding
  - Histogram Shape-based
  - Edge-based Thresholding



# İki Görüntü İşlemleri

- Add (toplama işlemi)
- Subtract (çıkarma işlemi)
- Intersect (kesişim işlemi)
- Merge (birleştirme işlemi)
- Difference (fark işlemi)



# Ön Şart

- Aynı boyutlarda olmalı (300x400 vb.)
- Aynı renk uzayında olmalı (rgb, binary vb.)
- Aynı renk bitinde olmalı (int, uint vb.)

# Toplama İşlemi

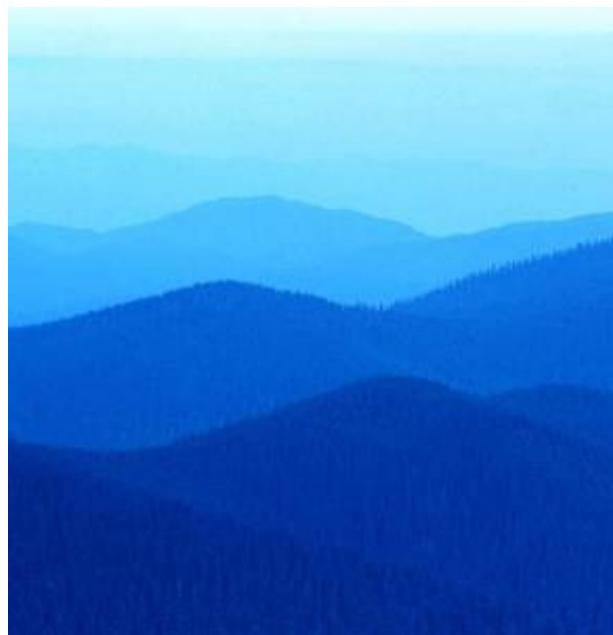
- Her pikseller sırasıyla toplanır.
- max değeri geçenler indirgenir.
- Yeni görüntü oluşturulur.

	1. Resimden			2. Resimden			Sonuç Resim		
	R1	G1	B1	R2	G2	B2	R=R1+R2	G=G1+G2	B=B1+B2
1	119	43	107	225	253	255	344	296	362
2	117	41	105	223	253	255	340	294	360
3	116	40	104	221	252	255	337	292	359
4	118	42	106	218	252	254	336	294	360
5	119	43	107	217	252	255	336	295	362
6	118	42	106	214	254	255	332	296	361
7	119	43	107	211	252	255	330	295	362
8	121	45	109	209	250	254	330	295	363
9	128	46	110	204	249	252	332	295	362
10	122	40	104	204	249	252	326	289	356

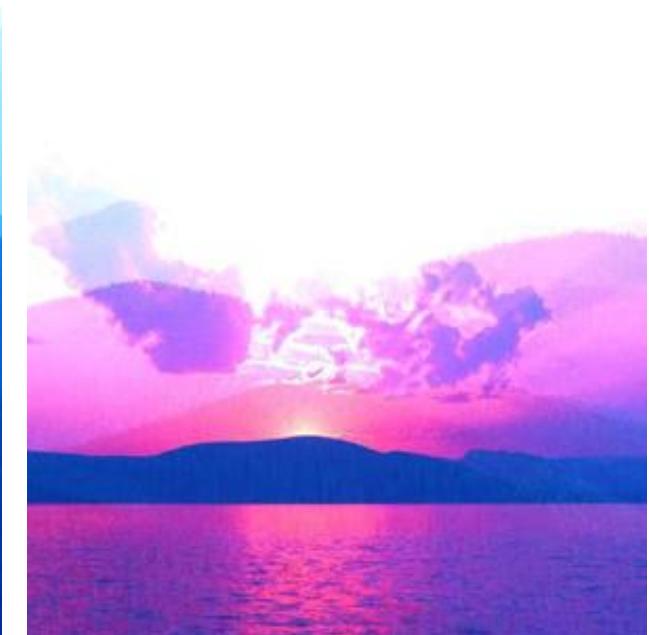
# Toplama İşlemi



Görüntü1



Görüntü2



Sonuç Görüntü



# Çıkarma İşlemi

- Her pikseller sırasıyla çıkarılır.
- min değerden düşenler yükseltilir.
- Yeni görüntü oluşturulur.

	1.Resimden			2.Resimden			Sonuç Resim		
	R1	G1	B1	R1	G1	B1	R=R1-R2	G=G1-G2	B=B1-B2
1	119	43	107	225	253	255	-106	-210	-148
2	117	41	105	223	253	255	-106	-212	-150
3	116	40	104	221	252	255	-105	-212	-151
4	118	42	106	218	252	254	-100	-210	-148
5	119	43	107	217	252	255	-98	-209	-148
6	118	42	106	214	254	255	-96	-212	-149
7	119	43	107	211	252	255	-92	-209	-148
8	121	45	109	209	250	254	-88	-205	-145
9	128	46	110	204	249	252	-76	-203	-142
10	122	40	104	204	249	252	-82	-209	-148

# Çıkarma İşlemi



# Kesişim İşlemi (Intersection)

- Her piksel sırasıyla iki görüntüde karşılaştırılır.
- Küçük piksel değeri alınır.
- Yeni görüntü oluşturulur.



# Birleştirme İşlemi (Merge)

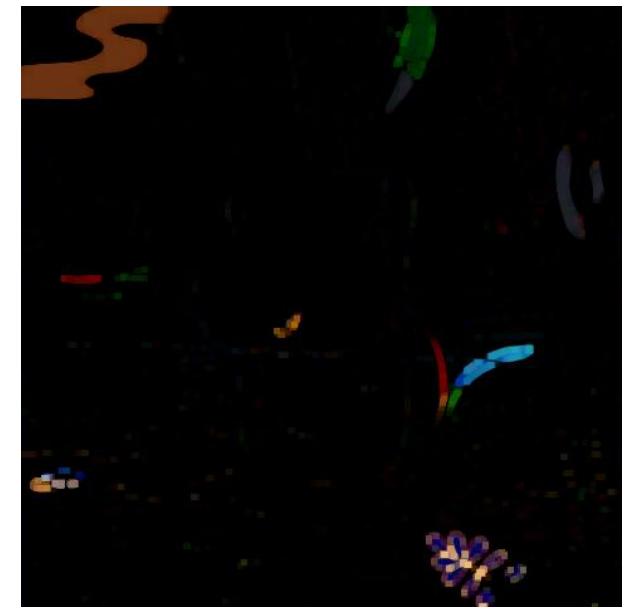
- Her piksel sırasıyla iki görüntüde karşılaştırılır.
- Büyük piksel değeri alınır.
- Yeni görüntü oluşturulur.



# Fark İşlemi (Difference)

- Pikseller birbirinden çıkarılır.
- Oluşan sonucun mutlak değeri alınır.
- Yeni görüntü oluşturulur.





## İki Resimdeki 7 Fark Oyunu

Hangi Yöntem?



- Ders Sonu