

# Görüntü Bölütleme

# Komşuluk

- $(x, y)$  noktasındaki  $p$  pikseli , 4 adet yatay ve dikey komşuya sahiptir:  
 $(x+1, y), (x-1, y), (x, y+1), (x, y-1)$
- Bu pikseller  $p$ 'nin 4 komşusu olarak bilinir ve  $N_4(p)$  ile gösterilir.
- Her komşu bir birim uzaklıktadır.

# Komşuluk

- $(x, y)$  noktasındaki piksel  $p$ , 4 adet de köşegen komşuya sahiptir:  
 $(x+1, y+1)$ ,  $(x+1, y-1)$ ,  $(x-1, y+1)$ ,  $(x-1, y-1)$
- Bu pikseller  $p$ 'nin D komşusu olarak bilinir ve  $N_D(p)$  ile gösterilir.
- $N_4(p)$  ve  $N_D(p)$  piksel kümelerinin oluşturduğu kümeye ise  $p$ 'nin 8 komşuluğu adı verilir ve  $N_8(p)$  ile gösterilir.

# Komşuluk

	$(x, y-1)$	
$(x-1, y)$	$(x, y)$	$(x+1, y)$
	$(x, y+1)$	

$N_4(p)$

$(x-1, y-1)$		$(x+1, y-1)$
	$(x, y)$	
$(x-1, y+1)$		$(x+1, y+1)$

$N_D(p)$

$(x-1, y-1)$	$(x, y-1)$	$(x+1, y-1)$
$(x-1, y)$	$(x, y)$	$(x+1, y)$
$(x-1, y+1)$	$(x, y+1)$	$(x+1, y+1)$

$N_8(p)$

# Bağlılık ve Bitişiklik

- Pikseller arası bağlılık, görüntü işlemede bölgelerin ve nesnelerin bulunmasında kullanılan önemli bir kavramdır.
- İki pikselin bağlı olup olmadığını anlamak için:
  - İki piksel komşu olmalıdır.
  - Grilik seviyeleri belli bir kriteri sağlamalıdır.

Burada benzerlik tanımını iki pikselin gri seviye değerlerinin de, benzerlik kriteri olarak seçilen bir  $V$  kümesinin elemanı olmalarıdır. İkili görüntü de benzerlik kümesi  $V=\{0, 1\}$  olacaktır.

# Bağlılık ve Bitişiklik

- Gri seviye değerleri  $V$  nin elemanı olan  $p$  ve  $q$  pikselleri
  - $q$ ,  $N_4(p)$  kümesi içindeyse 4-bitişik kabul edilir.
  - $q$ ,  $N_8(p)$  kümesi içindeyse 8-bitişik kabul edilir.
  - $q$ ,  $N_4(p)$  içinde ya da  $q$ ,  $N_D(p)$  içinde ve  $[N_4(p) \cap N_4(q) = \emptyset]$  ise  $m$ -bitişik kabul edilir.

0	1	1
0	1	0
0	0	1

# Bağılılık ve Bitişiklik

0	1	1
0	1	0
0	0	1

0	1	1
0	1	0
0	0	1

8-bitişiklik

0	1	1
0	1	0
0	0	1

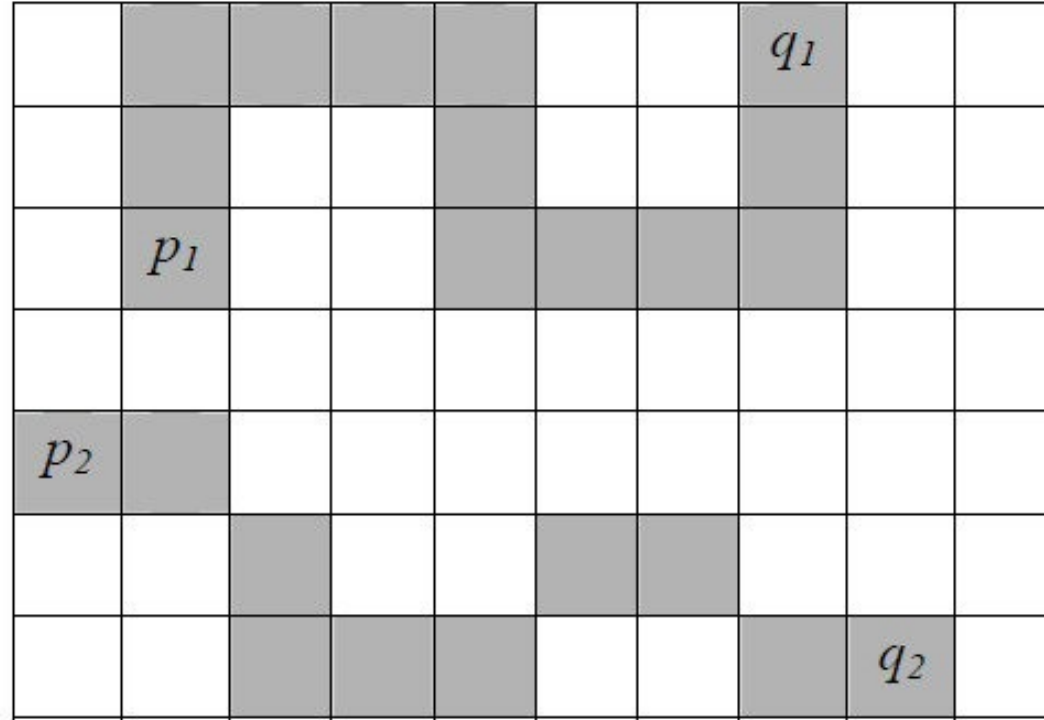
m-bitişiklik

# Yol ve Bölge

- $(x_0, y_0)$  koordinatındaki p pikselinden,  $(x_n, y_n)$  koordinatındaki q pikseline yol  $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$  koordinatlarındaki birbirinden farklı pikseller kümesidir.
- $(x_0, y_0), (x_{i-1}, y_{i-1})$  bitişik piksellerdir.
- Belirlenen bitişiklik türüne göre 4, 8 ya da m-yol tanımlanabilir.



# Yol ve Bölge



4 komşuluk tanımladığında, sadece  $p_1$  pikselinden  $q_1$  pikseline giden beyaz olmayan piksellerden oluşan bir yol vardır. 8 komşuluk tanımlanırsa hem  $p_1$  den  $q_1$  e hem de  $p_2$  den  $q_2$  ye giden beyaz olmayan piksellerden oluşan yollar oluşur.

# Yol ve Bölge

- $p$  ve  $q$  bir görüntünün  $S$  isimli bir alt kümesinin iki pikseli olsun.
- $p$  nin  $q$  ya  $S$  içinde bağlı olma koşulu,  $p$  den  $q$  ya tamamen  $S$  içinde bulunan piksellerden oluşan bir yol bulunmasıdır.
- $S$  kümesi içinde  $p$  pikseline bağlı olan piksellere  $S$  nin bağlı bileşeni denir. Eğer tek bağlı bileşen varsa  $S$  ye bağlantılı küme adı verilir.

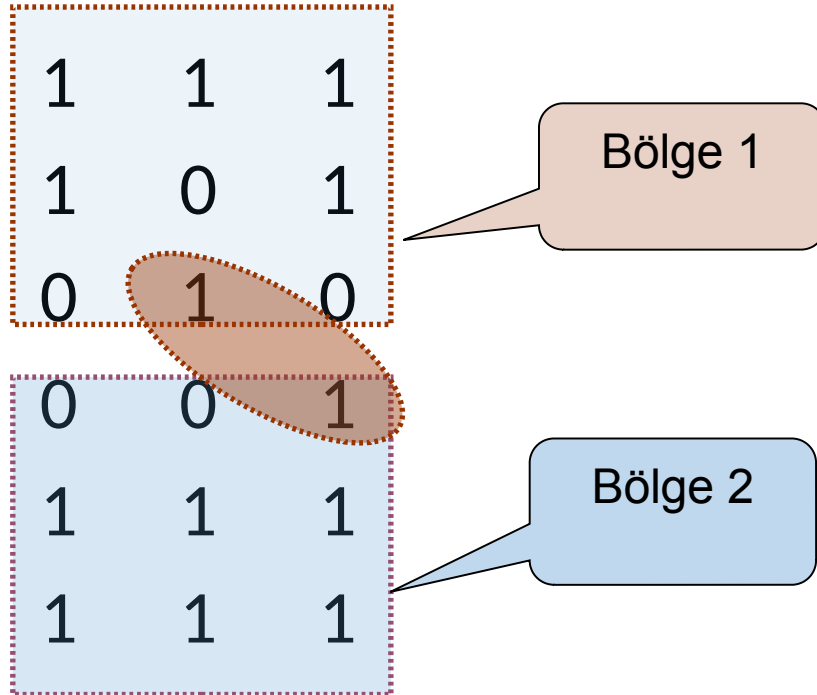
# Yol ve Bölge

$R$  bir görüntüdeki piksellerin alt kümesi olsun.

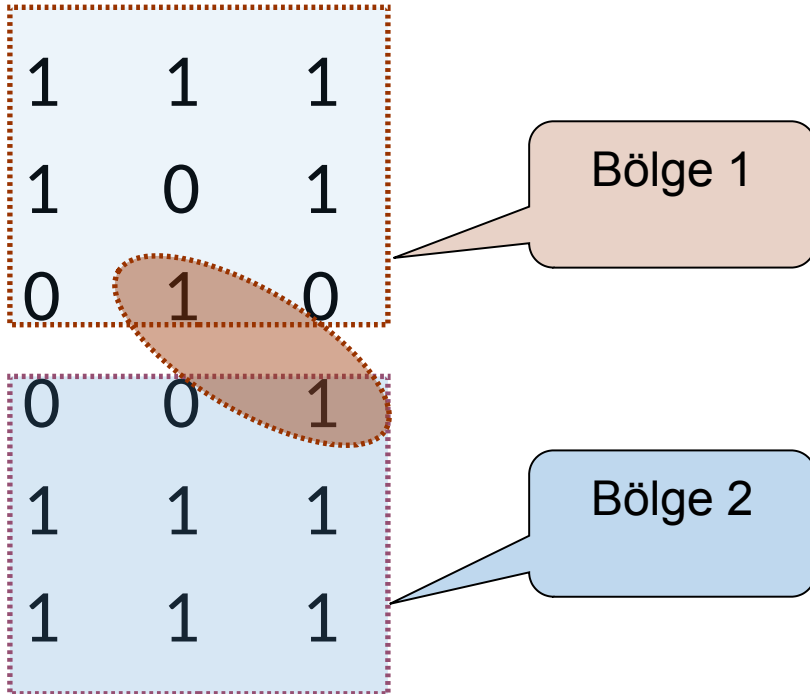
- Eğer  $R$  bağlantılı bir küme ise  $R$  görüntünün bir bölgesi olarak adlandırılır.
- Eğer  $R_i$  ve  $R_j$  bölgelerinin birleşimi bağlı bir küme oluşturursa  $R_i$  ve  $R_j$  bitişik olarak adlandırılır.
- Bitişik olmayan bölgeler de ayrışık olarak adlandırılır.

- Sınır (çevrit)
  - Bir  $R$  bölgesinin sınırı  $R$ 'nin tümleyenindeki noktalarla bitişik olan noktalar kümesidir.
  - Eğer  $R$  tüm bir görüntü olursa, o zaman sınırı görüntünün ilk ve son satır ve sütunlardaki piksellerin kümesi olarak tanımlanır.
- Ön plan ve Arka plan
  - Varsayalım ki bir görüntü, hiçbirisi görüntü sınırına dokunmayan  $K$  tane ayrışık bölge,  $R_k$ ,  $k = 0, 1, 2, \dots, K$  içersin.  $R_u$  tüm  $K$  bölgelerinin birleşimini göstere ve  $(R_u)^c$  de tümleyenini göstere (Bir  $S$  kümesinin tümleyeni  $S$ 'de olmayan noktaların kümesidir).  $R_u$ 'daki tüm noktaları görüntünün ön planı,  $(R_u)^c$ 'deki tüm noktaları ise arka planı olarak adlandırırız.

- Aşağıdaki piksel yerleşiminde, iki bölge (1'ler) bitişik midir? (şayet 8-bitişiklik kullanılırsa)

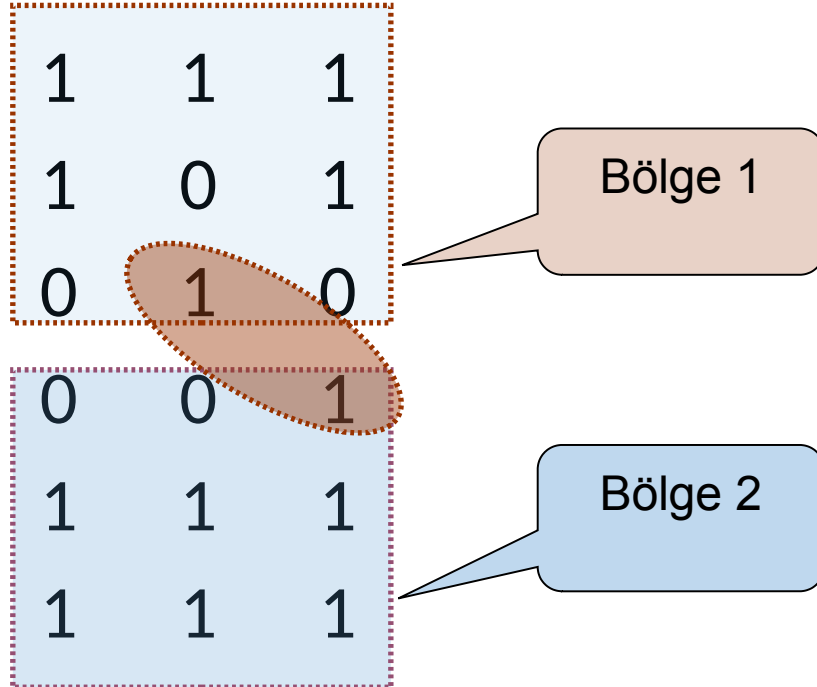


- Aşağıdaki piksel yerleşiminde, iki bölge (1'ler) bitişik midir? (şayet 8-bitişiklik kullanılırsa)

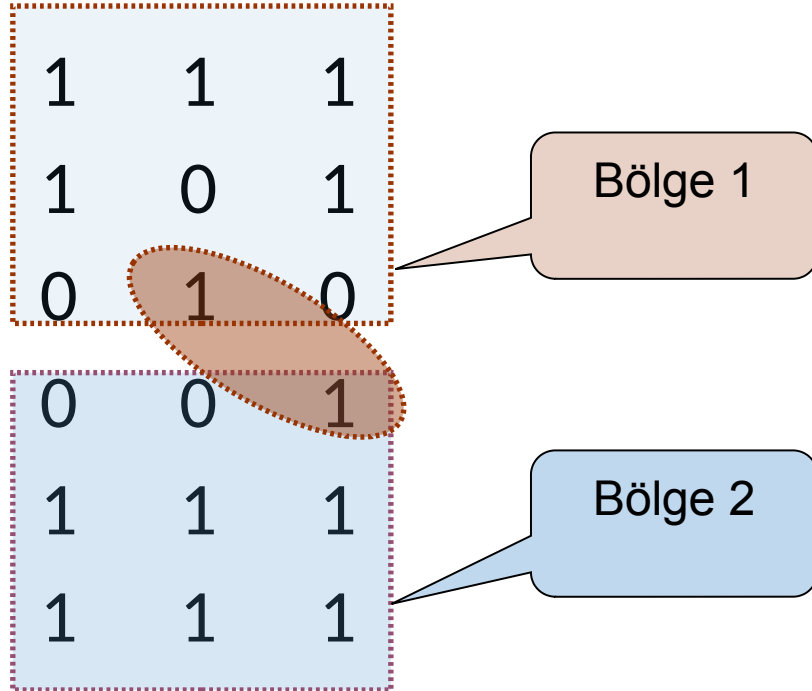


**Cevap: EVET**

- Aşağıdaki piksel yerleşiminde, iki bölge (1'ler) bitişik midir? (şayet 4-bitişiklik kullanılırsa)



- Aşağıdaki piksel yerleşiminde, iki bölge (1'ler) bitişik midir? (şayet 4-bitişiklik kullanılırsa)



**Cevap: HAYIR**



## Soru 2

- Aşağıdaki piksel yerleşiminde, iki bölge (1'ler) ayrışık mıdır? (şayet 4-bitişiklik kullanılırsa)

1	1	1
1	0	1

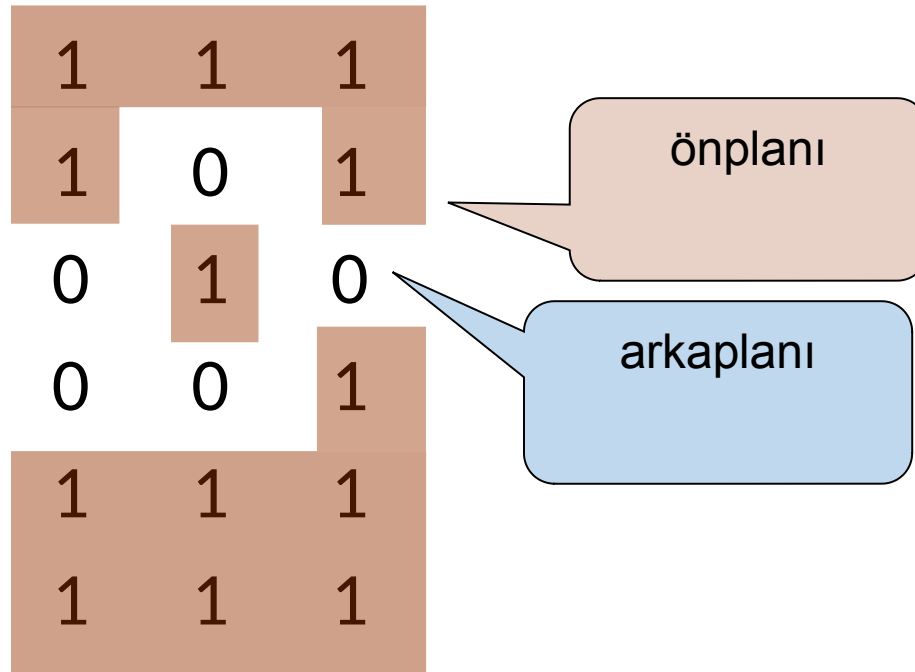
0 1 0

0	0	1
1	1	1
1	1	1

Bölge 1

Bölge 2

- Aşağıdaki piksel yerleşiminde, iki bölge (1'ler) ayrıdır (şayet 4-bitişiklik kullanılırsa)



- ▶ Aşağıdaki piksel yerleşiminde, işaretlenmiş nokta 1-değerli piksellerin sınırının bir parçasıdır (şayet 8-bitişiklik kullanılırsa doğru mu yanlış mı?)

0	0	0	0	0
0	1	1	0	0
0	1	1	0	0
0	1	1	1	0
0	1	1	1	0
0	0	0	0	0

- Aşağıdaki piksel yerleşiminde, işaretlenmiş nokta 1-değerli piksellerin sınırının bir parçasıdır (şayet 8-bitliklik kullanılırsa doğru mu yanlış mı?)

0	0	0	0	0
0	1	1	0	0
0	1	1	0	0
0	1	1	1	0
0	1	1	1	0
0	0	0	0	0

**Cevap: EVET**

- ▶ Aşağıdaki piksel yerleşiminde, işaretlenmiş nokta 1-değerli piksellerin sınırının bir parçasıdır (şayet 4-bitişiklik kullanılırsa doğru mu yanlış mı?)

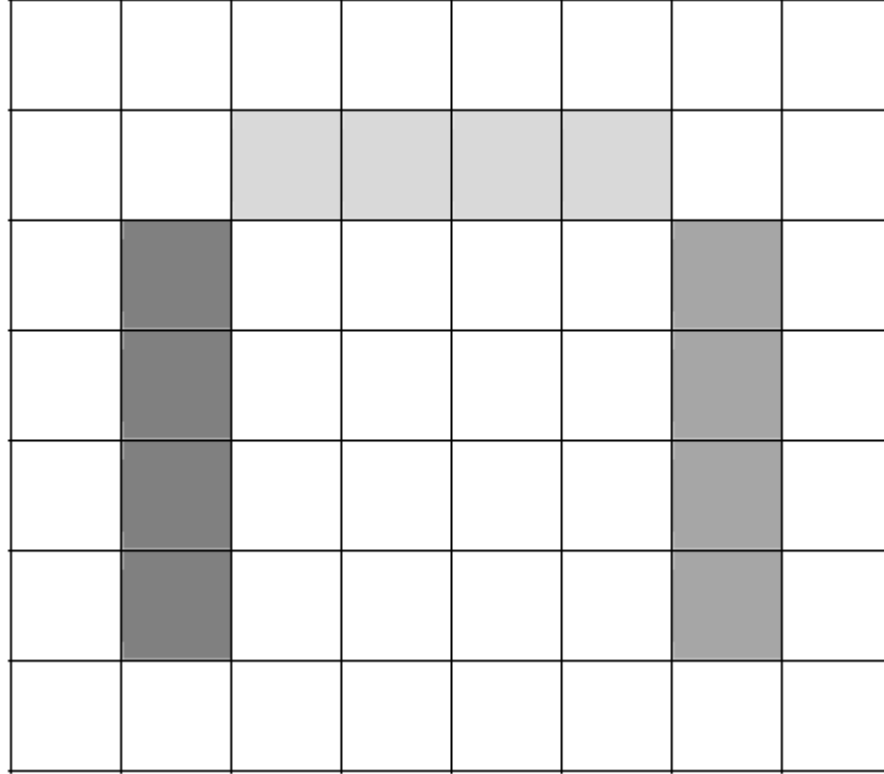
0	0	0	0	0
0	1	1	0	0
0	1	1	0	0
0	1	1	1	0
0	1	1	1	0
0	0	0	0	0

- Aşağıdaki piksel yerleşiminde, işaretlenmiş nokta 1-değerli piksellerin sınırının bir parçasıdır (şayet 4-bitişiklik kullanılırsa doğru mu yanlış mı?)

0	0	0	0	0
0	1	1	0	0
0	1	1	0	0
0	1	1	1	0
0	1	1	1	0
0	0	0	0	0

**Cevap: HAYIR**

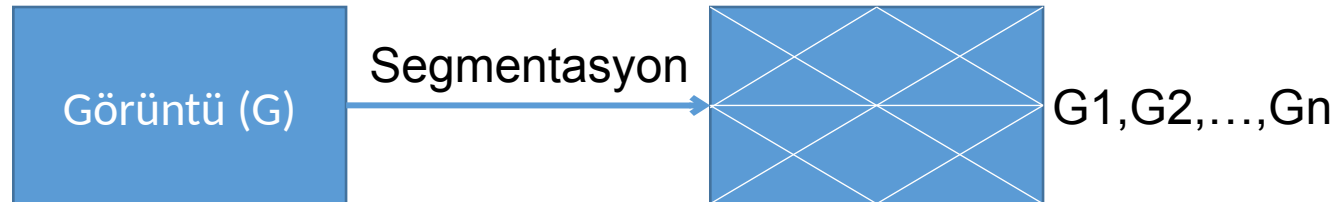
# Yol ve Bölge



- 4 komşuluk tanımlanmışsa farklı gri tonlarla ayrılmış ve beyaz olmayan 3 bölge, 8 komşuluk tanımlanmışsa beyaz olmayan piksellerin oluşturduğu bir bölge vardır.

# Görüntü Bölütleme

- Görüntüyü her biri kendi içerisinde benzer fakat çevresiyle farklı özellikler içeren bölgelere (segment) ayırmak olarak tanımlanabilir.
- **Özellik:** Görüntü içerisinde tanımlanabilen ve ölçülebilen her türlü değer özellik olarak kullanılabilir. (Örneğin: Piksellerin parlaklık değerleri.)





# Görüntü Bölütlemenin Amacı

- Benzer Pikselleri Bir Araya Toplamak
  - Görüntüyü işleme sokmadan önce ön bir işlem ile benzer pikseller bir araya toplanarak daha etkin sonuçlar elde etmek.
- Süper pikseller oluşturmak
  - Tek bir piksel çoğu zaman anlam ifade etmezken birden fazla pikselin bir araya toplanması sonucu anlamlı süper pikseller oluşturmak
- İşlem Yükünü Azaltmak
  - İşlenmesi oldukça maliyetli olabilecek herhangi bir görüntünün sadeleştirilerek işleme maliyetini azaltmak.
- Görüntü Sıkıştırma, Nesne Takibi

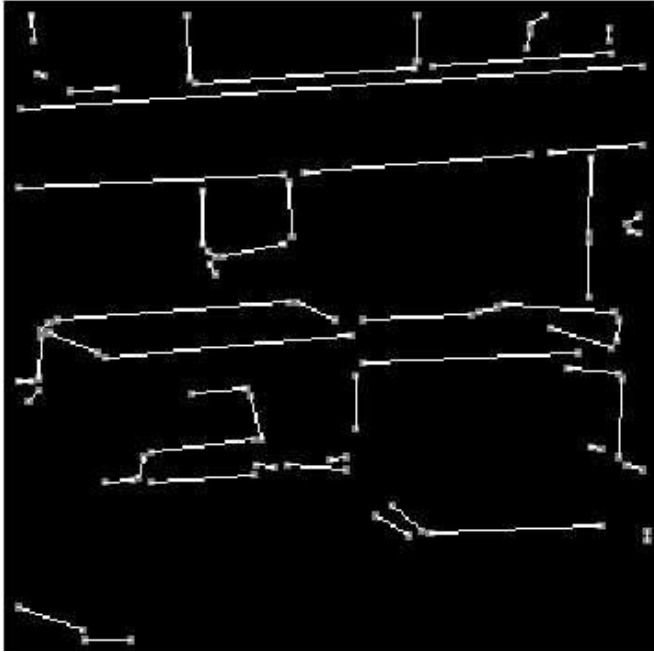
# Görüntü Bölütleme

- Bölütleme algoritmaları pikseller arasındaki gri seviye değerlerinin iki temel özelliğine dayanır:
  - Süreksizlik
  - Benzerlik
- Bu iki özelliğe göre bölütleme algoritmaları temel olarak iki kategoriye ayrılmıştır.

# Görüntü Bölütleme

- Kenar tabanlı bölütleme denebilecek birinci kategoride, gri seviyelerdeki ani değişimler göz önünde bulundurulur. Böylece görüntüdeki çizgiler ve kenarlar elde edilerek birbirine bağlanır ve alanlar kenar çerçevelerini (contour) kapsayacak şekilde tanımlanır.
- İkinci kategoride, bir bölümdeki pikseller arasındaki benzerlikten yola çıkılır. Bir görüntüyü bölütlerken, piksellerin çeşitli yerel özellikleri kullanılır. Böylece alan içerisinde bulunan pikseller alanı tanımlamış ve bölütleme işlemi parçalanmış bütün bölümleri sınırlı sayıda olan alanları kapsamış olur.

# Görüntü Bölütleme



# Görüntü Bölütleme Yöntemleri

- Histogram Tabanlı Eşikleme
- Bölge Büyütme (Region Growing)
- Bölge Ayırma ve Birleştirme
- Kümeleme / Sınıflandırma (Clustering / Classification)
- Graf Teorik Yaklaşım (Graph Theoric Approach)
- Kural tabanlı veya Bilgi Güdümlü Yaklaşım (Rule Based or Knowledge-Driven Approach)

# Region Growing Bölütleme

- Bölge büyümesi, bölge tabanlı görüntüyü parçalara ayırma metodudur. Aynı zamanda başlangıç noktalarının seçimini de kapsadığı için bu metot piksel tabanlı görüntü parçalama metodu olarak da sınıflandırabilir.
- Bu metot başlangıç noktası(seed points) seçilen noktanın etrafındaki pikselleri inceler ve bu piksellerin bölgeye katılıp katılmayacağına karar verir. Bu işlem genel veri kümeleme algoritmalarında olduğu gibi tekrarlanır.

# Region Growing Bölütleme

- Görüntü parçalama işleminin sonucunda toplu olarak resmi kaplayan bir dizi görüntü bölgesi veya görüntüden çıkarılan bir dizi şekil oluşur. Bölgedeki her piksel renk, yoğunluk ve doku gibi bazı özellikler bakımından benzerdirler. Komşu bölgeler ise bu özellikler bakımından oldukça farklıdır.
- Thresholding gibi bazı segmentasyon metotları bölgelere bölme işlemini, bölgeler arasındaki sınırlara, gri düzeylerdeki veya renk özelliklerinde süreksizliklerine göre bakarak gerçekleştirir.

# Region Growing Bölütleme

- Bölge büyümesi tekniği potansiyel bölge içinde bir pikselle başlar ve komşu piksellerin benzerliği test edilerek büyütülür.
- İlk piksel veya belli sayıda (seed) başlangıç pikseli manuel yada otomatik olarak görüntü üzerinden seçilir.
- Başlangıç pikseli ile küme pikseller arasında benzerlik ölçüsü tanımlanmalıdır ki büyümenin ana kriteri budur.



# Region Growing Bölütleme

- Genellikle, hangi pikselin bölgeye ekleneceğine bir istatistiksel test ile karar verilir.
  - Bölge benzer istatistiğe sahip bir popülasyondur.
  - İstatistiksel test ile komşu yada kenar pikselin bölgedeki popülasyona uyup uymadığı test edilir.
- $r$ ,  $N$  pikseli bir bölge,  $p$  ise  $y$  gri değerli ve bu bölgeye komşu piksel olsun.
- Ortalama  $X$  ve örnek varyans  $S^2$  ile tanımlansın.

$$\bar{X} = \frac{1}{N} \sum_{(r,c) \in R} I(r,c) \quad S^2 = \frac{1}{N} \sum_{(r,c) \in R} (I(r,c) - \bar{X})^2$$

# Region Growing Bölütleme

- T istatistik değeri aşağıdaki formül ile tanımlansın.

$$T = \left( \frac{(N-1)N}{(N+1)} (p - \bar{X})^2 / S^2 \right)^{1/2}$$

- Eğer r bölgesi ve p pikseli bağımsız ve Gauss dağılımına uygunsa,  $T_{N-1}$  dağılımına uygundurlar denilebilir.

# Region Growing Bölütleme

- T dağılım için, istatistik tabloda olasılık değeri belli güven aralığı ve serbestlik derecesi için  $P_r(T \leq t)$ 'dir. Buradan, pik değeri t olarak alınır.
- Eğer  $T \leq t$  ise p pikseli R bölgesine eklenir, ortalama ve varyans yeniden hesaplanır.
- Eğer T büyük ise, p ilgili R bölgesine dahil edilmez, yeni bir bölge seçilerek işlem yenilenir.

# Region Growing Bölütleme

- Segmentasyon işlemi tamamlanmalı ve tüm pikseller bölgenin içinde olmalı

$$(a) \bigcup_{i=1}^n R_i = R.$$

- Bölge içindeki noktalar önceden tanımlanmış bir mantıkla birbiriyle ilişkilendirilmeli

$$(b) R_i \text{ is a connected region, } i = 1, 2, \dots, n$$

- Bölgeler birbirinden ayrı olmalı.

$$(c) R_i \cap R_j = \emptyset \text{ for all } i = 1, 2, \dots, n.$$

- Bölünmüş kısımdaki piksellerin özelliklerinin bazı şartları taşıması gerekir. Örnekte verildiği gibi  $R_i$  alanındaki tüm piksellerin aynı gri seviyede olması gerektiği gibi.

$$(d) P(R_i) = TRUE \text{ for } i = 1, 2, \dots, n.$$

- Herhangi komşu  $R_i$  ve  $R_j$  parçaları için seçilen  $P$  özelliğinde farklı olması

$$(e) P(R_i \cup R_j) = FALSE \text{ for any adjacent region } R_i \text{ and } R_j.$$

# Region Growing Bölütleme

- Bölge büyümede ilk adım başlangıç noktalarını (seed points) seçmektir. Bu noktaların seçimi kullanıcıların seçtiği bazı kriterlere (pikselin gri seviyesi) göre yapılır. Başlangıç alanı bu noktaların konumu olarak belirlenir. Sonrasında alan bu seçilen noktalardan komşu noktalara doğru seçilen kriterler göre büyümeye başlar. Bu kriterler piksel yoğunluğu, gri seviye dokusu veya renk olabilir.
- Alan seçilen kriterler üzerinden büyümeye başlamasıyla görüntü bilgisi önemli hale gelir. Örneğin eğer kriter piksel yoğunluğu eşik değeri ise görüntünün histogram bilgisi kullanışlı olabilir. Çünkü kullanıcı histogram bilgisini alan üyelik kriterleri için uygun eşik değeri olarak belirleyebilir.

# Bazı Önemli Noktalar

- **1.Uygun başlangıç noktası seçimi**
- Başlangıç noktasının seçimi kullanıcının tercihiine göre değişir. Örneğin gri seviyeli şimşek görüntüsünde, bu görüntüyü arka plandan ayırmak isteyebiliriz. Bu durumda histogramı inceleyip en yüksek dizisinden başlangıç noktasını seçebiliriz.
- **2. Daha fazla görüntü bilgisi**
- Bağlanabilirlik veya komşu piksel bilgisi eşik değeri ve başlangıç noktasını belirlemek için bize yardımcı olur.

# Bazı Önemli Noktalar

- **3.En küçük bölge eşik değeri**
- Bölge büyüme metodunda parçalara ayrılmış görüntüde hiçbir alan seçilen eşikten daha küçük olamaz.
- **4.Benzerlik Eşik değeri**
- Eğer piksel değeri farklılığı veya bir dizi pikselin ortalama gri seviye farklılık değeri benzerlik eşik değerinden az ise alan aynı bölge düşünülür.
- Seçtiğimiz benzerlik ve homojenlik kriteri aynı zamanda önemlidir. Genellikle istediğimiz bölütleme sonuçlarına ve orijinal görüntüye göre değişir.
- Sıklıkla kullanılan bazı kriterler : Gri seviye (ortalama yoğunluk ve varyans), renk, doku ve şekil.

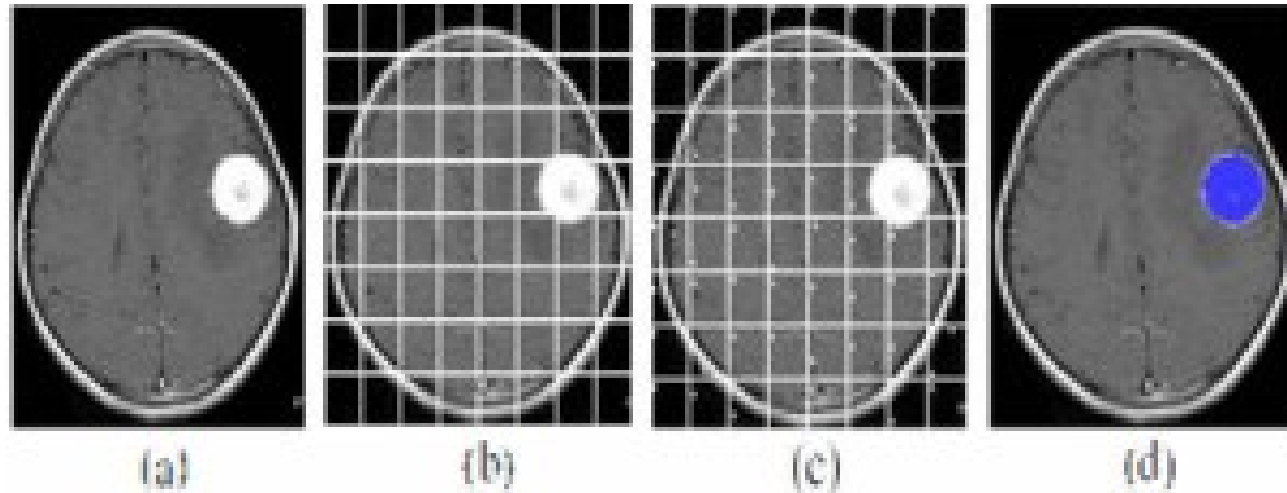
# Avantaj ve Dezavantajları

- Bölge büyüme metodu tanımladığımız özellikleri aynı olan bölgeleri düzgün bir şekilde ayırabilir.
- Bölge büyüme metodu iyi ayırma sonuçları ile düzgün kenarlara sahip orjinal görüntüler sağlayabilir.
- İhtiyaç olunan şey istediğimiz özellikleri belirtmemiz için az sayıda başlangıç noktası ve sonrasında bölgeyi büyütmeektir.
- Başlangıç noktaları ve kriterleri belirlenebilir.
- Aynı anda birden fazla kriter seçilebilir.
- Gürültüye göre göreceli olarak iyi sonuçlar verebilir.
- Gürültü problemini boşlukları filtrelemek için bazı maskeler kullanılarak aşılabilir. Bu yüzden gerçek manada bir gürültü probleminden bahsedilemez. Sonuç olarak bölge büyümedeki en önemli problem güç ve zaman harcamasıdır.



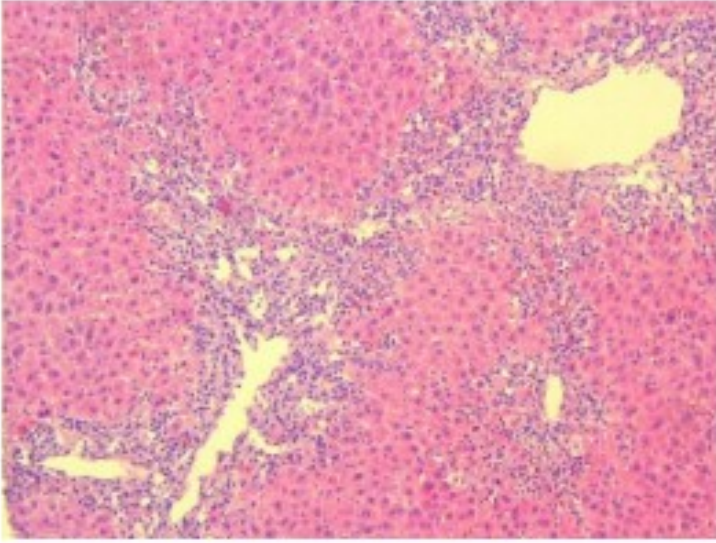
# Region Growing Bölütleme

- Beyin tümörü tespiti

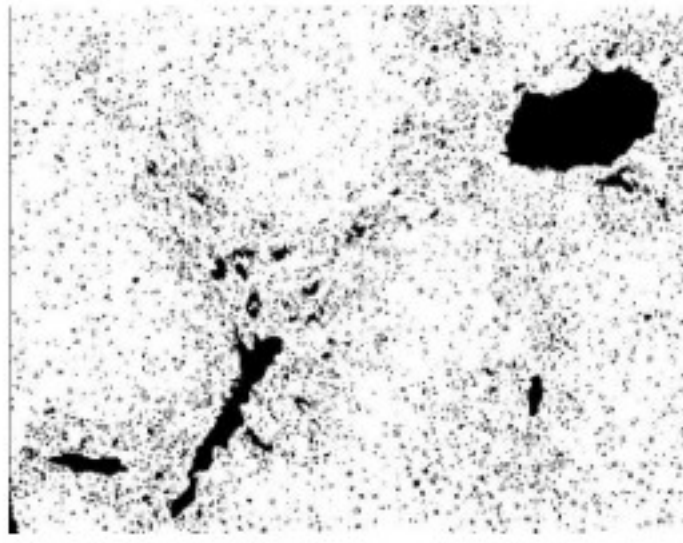


**Fig 4. Results.** (a) MRI image with tumour, (b) gridded image, (c) seed point selected image, (d) segmented MRI image

# Region Growing Bölütleme



Şekil 2. Örnek bir karaciğer dokusu



Şekil 3. Örnek karaciğer dokusu için bölütleme sonucu



Şekil 4. Örnek karaciğer dokusu için nekrozlu bölge tespit sonucu

```

function J=regiongrowing(I,x,y,reg_maxdist)
if(exist('reg_maxdist','var')==0), reg_maxdist=0.2; end
if(exist('y','var')==0), figure, imshow(I,[]); [y,x]=getpts; y=round(y(1));
x=round(x(1)); end
J = zeros(size(I)); % Output
Isizes = size(I); % Dimensions of input image
reg_mean = I(x,y); % The mean of the segmented region
reg_size = 1; % Number of pixels in region
% Free memory to store neighbours of the (segmented) region
neg_free = 10000; neg_pos=0;
neg_list = zeros(neg_free,3);
pixdist=0; % Distance of the region newest pixel to the regio mean
% Neighbor locations (footprint)
neighb=[-1 0; 1 0; 0 -1;0 1];
% Start regiogrowing until distance between regio and posible new
pixels become
% higher than a certain treshold
while(pixdist<reg_maxdist&&reg_size<numel(I))
% Add new neighbors pixels
for j=1:4,
    % Calculate the neighbour coordinate
    xn = x +neighb(j,1); yn = y +neighb(j,2);
    % Check if neighbour is inside or outside the image
    ins=(xn>=1)&&(yn>=1)&&(xn<=Isizes(1))&&(yn<=Isizes(2));

```

```

% Add neighbor if inside and not already part of the segmented area
if(ins&&(J(xn,yn)==0))
    neg_pos = neg_pos+1;
    neg_list(neg_pos,:) = [xn yn I(xn,yn)]; J(xn,yn)=1;
end
end
% Add a new block of free memory
if(neg_pos+10>neg_free), neg_free=neg_free+10000;
neg_list((neg_pos+1):neg_free,:)=0; end
% Add pixel with intensity nearest to the mean of the region, to the region
dist = abs(neg_list(1:neg_pos,3)-reg_mean);
[pixdist, index] = min(dist);
J(x,y)=2; reg_size=reg_size+1;
% Calculate the new mean of the region
reg_mean= (reg_mean*reg_size + neg_list(index,3))/(reg_size+1);
% Save the x and y coordinates of the pixel (for the neighbour add proccess)
x = neg_list(index,1); y = neg_list(index,2);
% Remove the pixel from the neighbour (check) list
neg_list(index,:)=neg_list(neg_pos,:); neg_pos=neg_pos-1;
end
% Return the segmented area as logical matrix
J=J>1;
-----
I = im2double(imread('rice.png'));
x=198; y=177;
J = regiongrowing(I,x,y,0.2);
figure, imshow(J);

```

# Region Splitting Bölütleme

- Bölge tabanlı bölütleme yöntemleri, bir bölge içindeki komşu piksellerin yakın değerlere sahip olduğu yaklaşımına dayanır. Bir piksel komşularıyla karşılaştırılır ve belirli bir kriteri sağlarsa bölgeye dahil edilir.
- Basit olarak bölge tabanlı bölütleme, görüntünün piksel yoğunluğu vb. özelliklere bağlı olarak alt parçalara ayrılmasıdır.
- Region splitting bölütleme de bu tür bir bölütleme yöntemidir.

# Region Splitting Bölütleme

1- Bölge ayrımı için ortalama, varyans, histogram, doku vb. bir kriter seçilir.

2- Görüntü 4 adet alt görüntüye ayrılır.

3- Her alt görüntü kritere göre kontrol edilir.

Eğer homojen değilse tekrar 4 alt görüntüye bölünür.

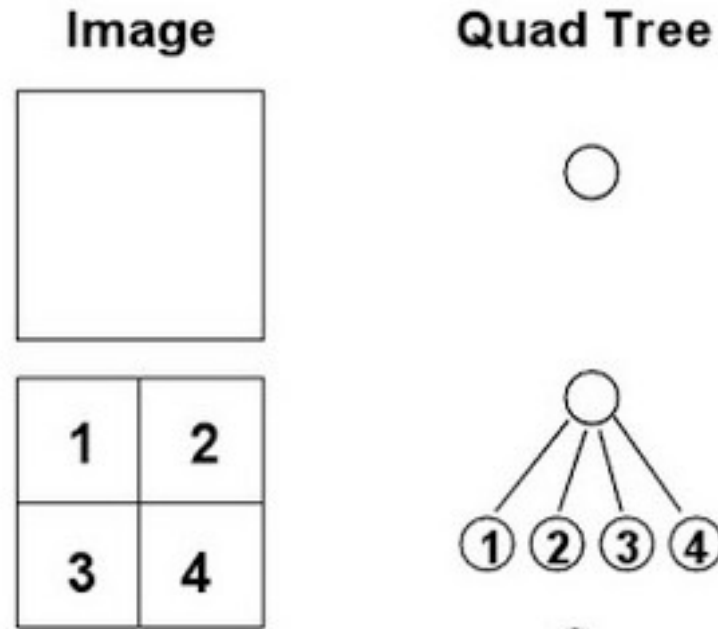
4- Madde 3'e göre bölünmeler devam eder.

5- Alt görüntüler komşu kısımlarına bakılarak karşılaştırılır ve eğer benzerlerse birleştirilir.

6- Madde 5 başka bir birleştirme kalmayana kadar devam eder.

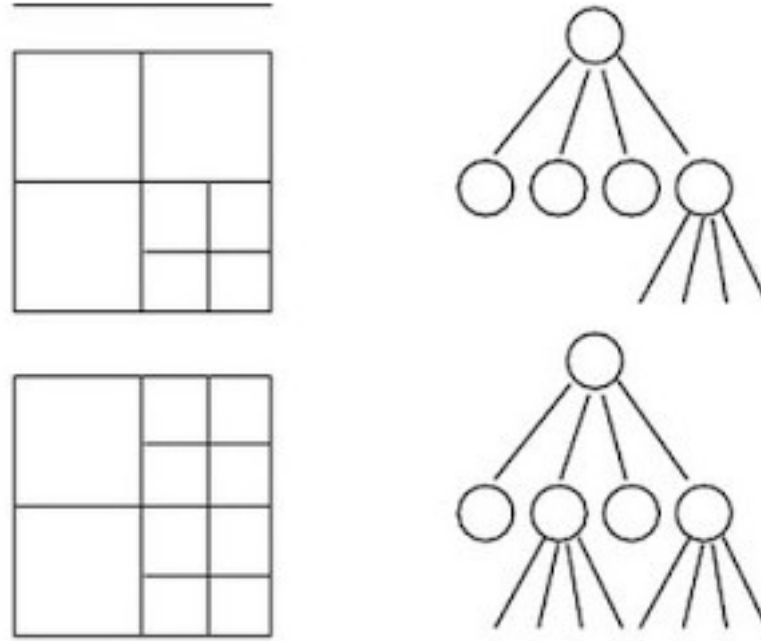
# Region Splitting Bölütleme

- Görüntünün 4 alt görüntüye ayrılmasından dolayı bu yöntem quad-tree division adı da verilir.



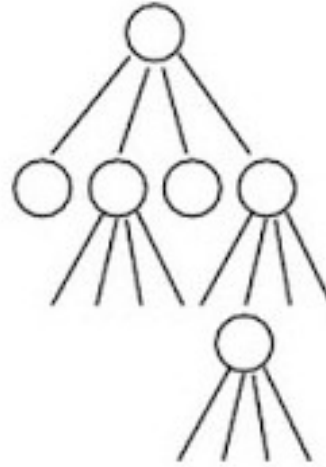
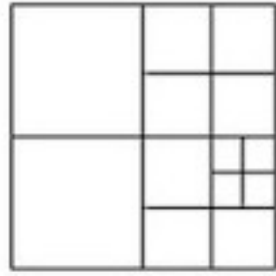
# Region Splitting Bölütleme

- Bazı bölgeler homojen değilse bölünme devam eder.



# Region Splitting Bölütleme

- Homojen bölge kalmayana kadar işlem devam eder.





# Region Splitting Bölütleme

- Region splitting, belli bir aralığa sahip değerler daha homojen bölgeler oluşturduğu için en iyi ikili veri ile çalışmaktadır.
- Işık etkileri gibi çeşitli sebeplerden dolayı bölgeler her zaman homojen olmayabilir.

# Region Splitting Bölütleme

- MATLAB içerisinde *qtdecomp* isimli fonksiyon, quad-tree decomposition işlemini gerçekleştirmektedir. MATLAB help' e bakılabilir.

```
I = imread('liftingbody.png');
S = qtdecomp(I,.27);
blocks = repmat(uint8(0),size(S));

for dim = [512 256 128 64 32 16 8 4 2 1];
    numblocks = length(find(S==dim));
    if (numblocks > 0)
        values = repmat(uint8(1),[dim dim numblocks]);
        values(2:dim,2:dim,:) = 0;
        blocks = qtsetblk(blocks,S,dim,values);
    end
end

blocks(end,1:end) = 1;
blocks(1:end,end) = 1;

imshow(I), figure, imshow(blocks,[])
```

# Region Splitting Bölütleme

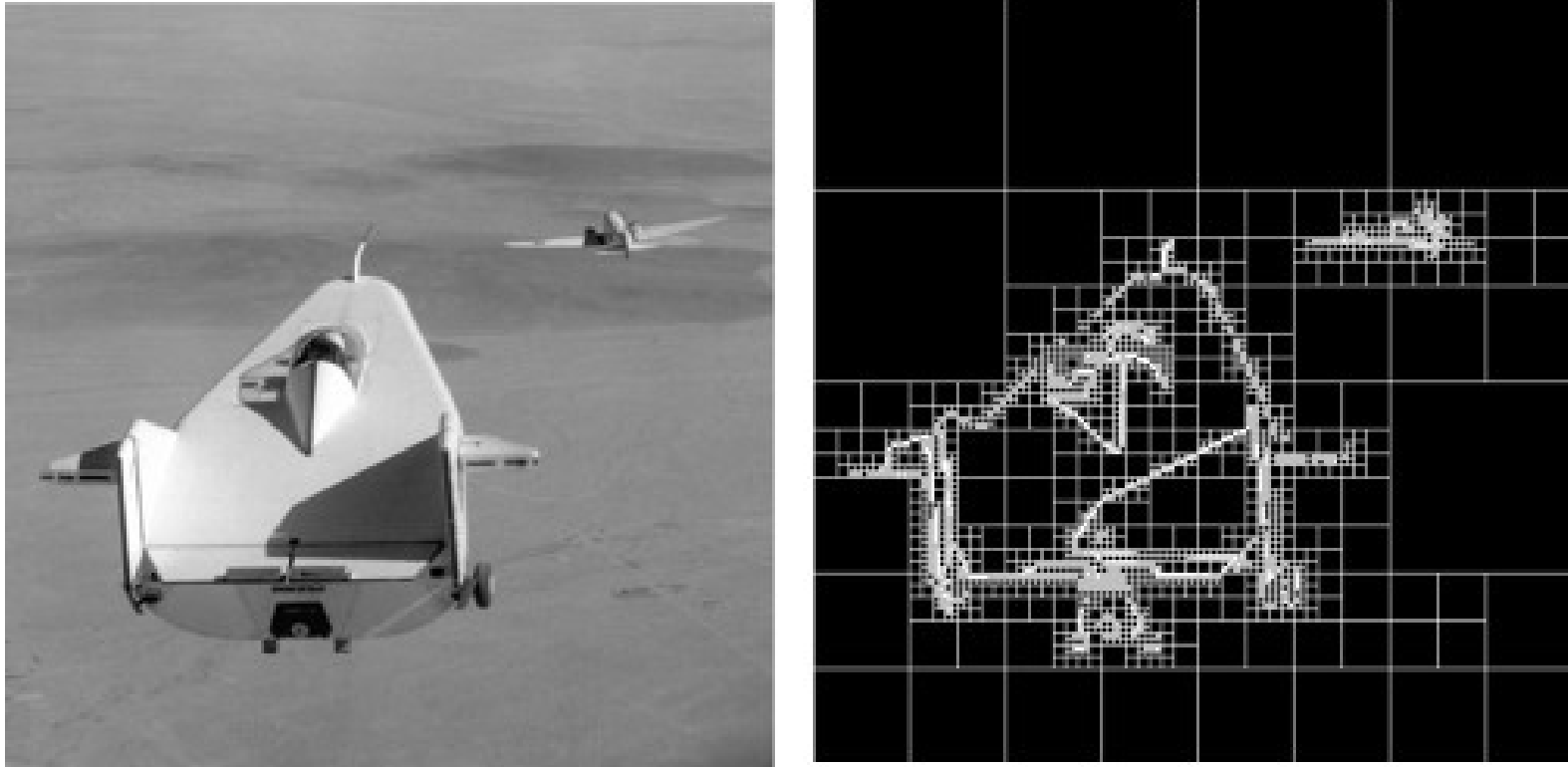


Image courtesy of NASA

# Region Splitting Bölütleme

*qtdecomp* fonksiyonun işlevini  
gerçekleştiren kod yazılırsa

```
function [segm, depth] = split(image, depth)

[rows, columns, dimension] = size(image);

    if ( not( predicate(image) ) && rows > 1 && columns > 1)

        row_split = fix(rows/2);
        column_split = fix(columns/2);

        [seg1, depth] = split(image(1:row_split, 1:column_split), depth);
        [seg2, depth] = split(image(1:row_split, (column_split+1):columns), depth);
        [seg3, depth] = split(image((row_split+1):rows, 1:column_split), depth);
        [seg4, depth] = split(image((row_split+1):rows, (column_split+1):columns), depth);

        segm = [seg1 seg2 ; seg3 seg4];

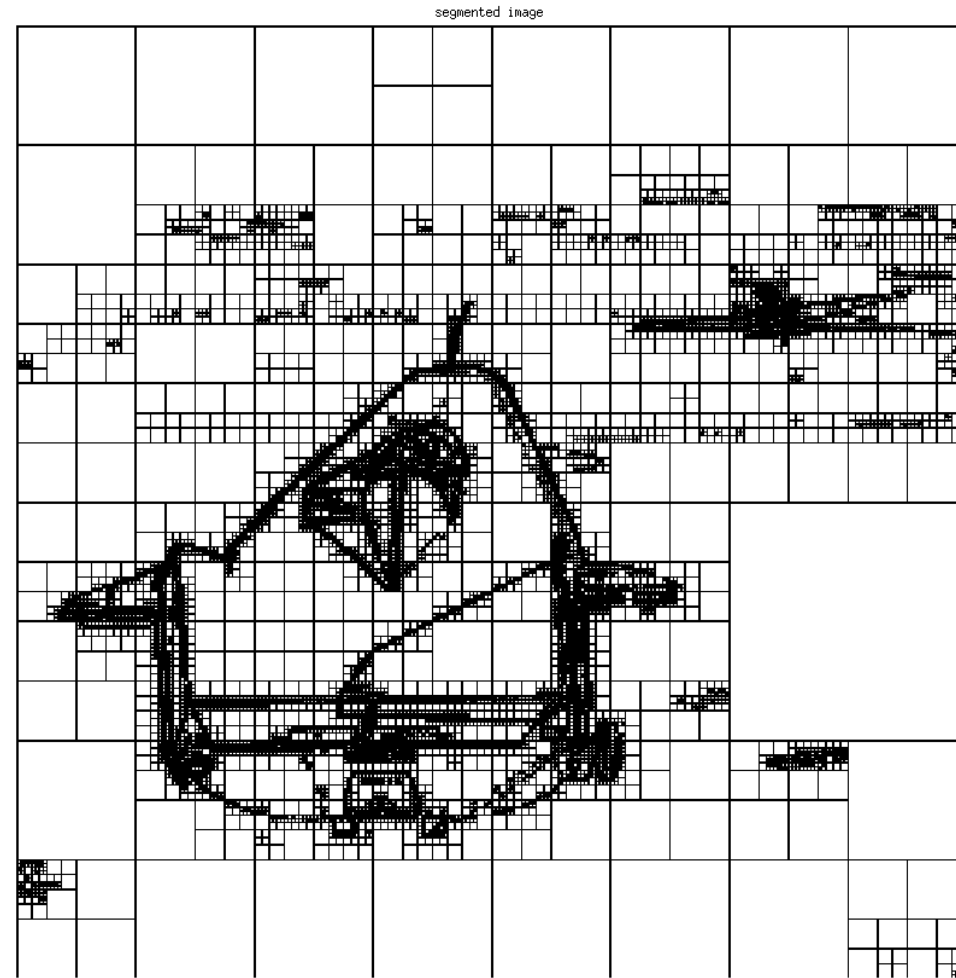
    else
        segm = zeros(rows, columns);
        segm(2:rows, 2:columns) = 255;

        depth = depth + 1;
    end

end

function [flag] = predicate(region)
    sigma1 = std2(region);
    flag = sigma1 <= 5;
end
```

# Region Splitting Bölütleme



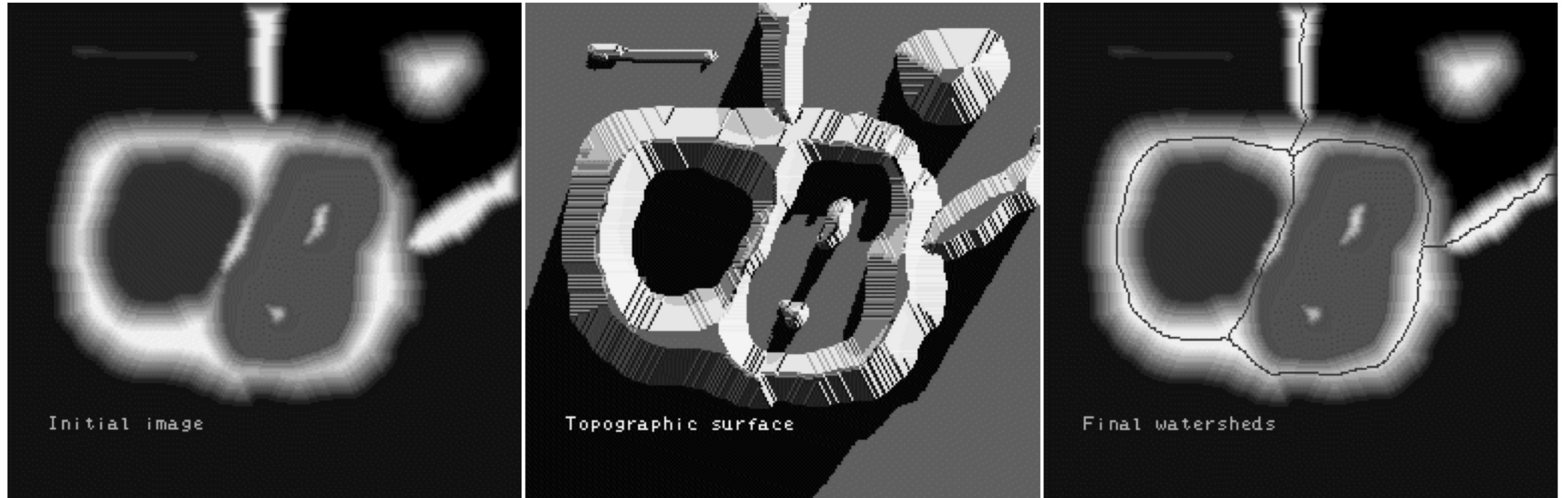
# Watershed Bölütleme

- Bölge çıkarımı temeline dayanan ve topografya biliminde yer alan su bölümü çizgileri ve su kaynağı kavramlarını temel almaktadır.
- Watershed segmentasyon gri düzeydeki görüntüler için iyi bir yöntemdir.
- Watershed dönüşümünde, her bir gri seviyeli görüntü, gri seviye değerleri yükseklik ifade eden bir topolojik yüzey olarak düşünülebilir.
- Eğer bu topolojik yüzeyin minimum noktalarından su bastırmaya başlanırsa ve farklı kaynaklardan gelen suların birleşmesi engellenirse, görüntü iki farklı kümeye bölünür.

# Watershed Bölütleme

- Orijinal görüntüye Watershed dönüşümü uygulandıktan sonra mozaik görüntü elde edilir.
- Eğer mozaik görüntü çok küçük bölgelerden oluşuyorsa bu duruma aşırı bölütleme denir ve bu yüzden gürültüye karşı çok duyarlıdır.
- Bu durumdan kaçınmak için ilk önce görüntü yumuşatılır. Yumuşatma işleminden sonra Watershed dönüşümü uygulanırsa, mozaik görüntü içinde mümkün mertebe homojen olan birçok bölge oluşacaktır.

# Watershed Bölütleme





# Watershed Bölütleme

