

SAYISAL GÖRÜNTÜ İŞLEME · BÖLÜM 3

Yoğunluk Dönüşümleri ve Uzamsal Filtreleme

Intensity Transformations and Spatial Filtering — uzamsal alanda görüntü iyileştirmenin temel araçları: nokta işlemleri, histogram yöntemleri ve komşuluk filtreleri.

İçerik

3.1 Temel kavramlar

Uzamsal alan, komşuluk ve T operatörü

3.3 Histogram işleme

Eşitleme, eşleme (belirtme), yerel yöntemler

3.5 Düzgünleştirme (alçak geçiren)

Kutu, Gauss ve medyan filtreleri

3.7 Filtre türleri

Alçak geçirenden yüksek geçiren / bant filtreleri

3.2 Yoğunluk dönüşüm fonksiyonları

Negatif, log, gama, parçalı doğrusal, bit düzlemi

3.4 Uzamsal filtrelemenin temelleri

Korelasyon, konvolüsyon, ayrılabilir çekirdekler

3.6 Keskinleştirme (yüksek geçiren)

Laplas, keskin olmayan maskeleme, gradyan

3.8 Yöntemlerin birleştirilmesi

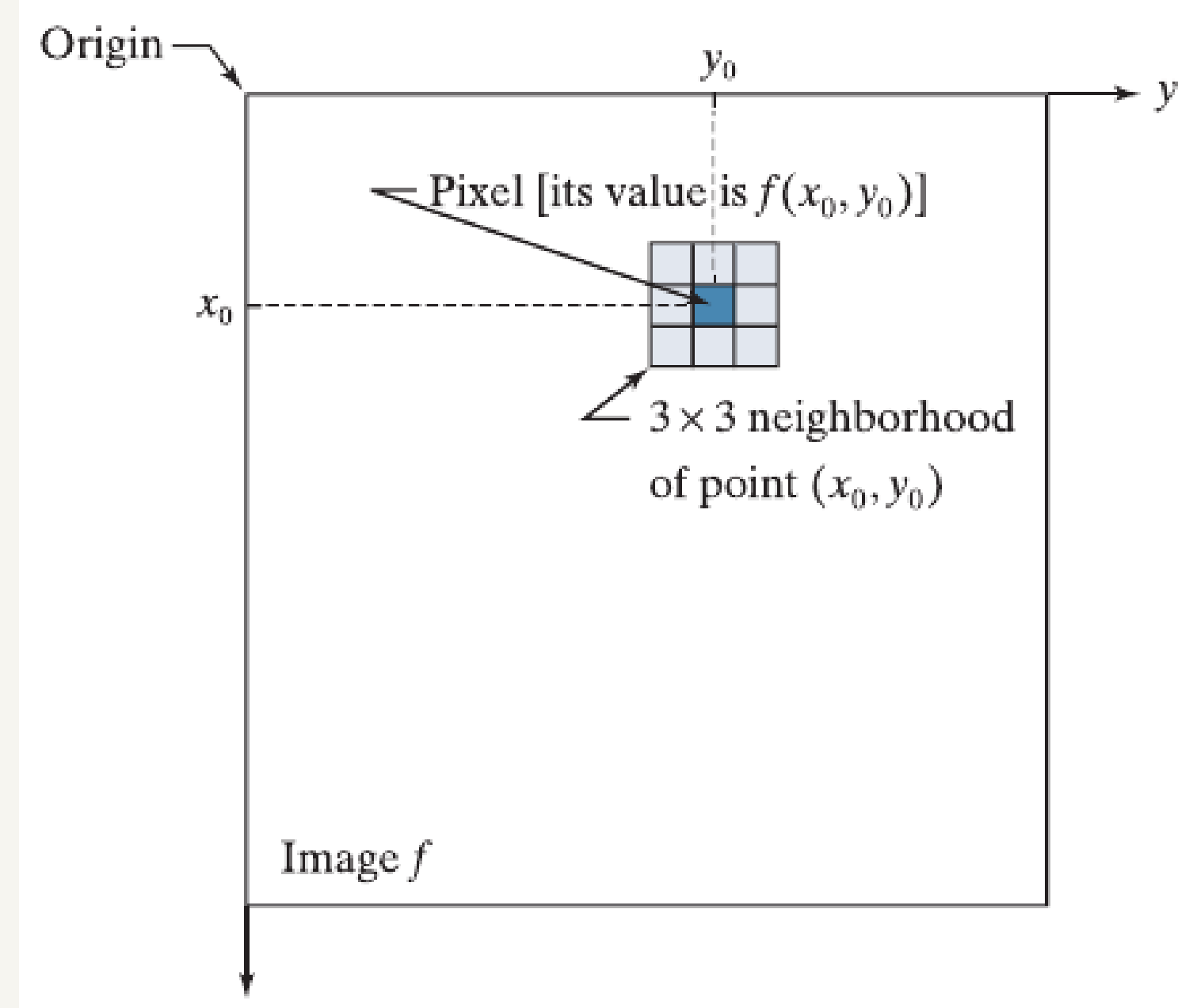
Gerçek bir iyileştirme senaryosu

Uzamsal alanda işleme

Uzamsal alan yöntemleri, görüntü piksellerini **doğrudan** işler. Her çıkış pikseli, giriş görüntüsünde aynı konumdaki pikselin belirli bir **komşuluğuna** uygulanan bir T operatörüyle elde edilir.

$$g(x, y) = T[f(x, y)]$$

T : (x, y) merkezli komşuluk üzerinde tanımlı operatör



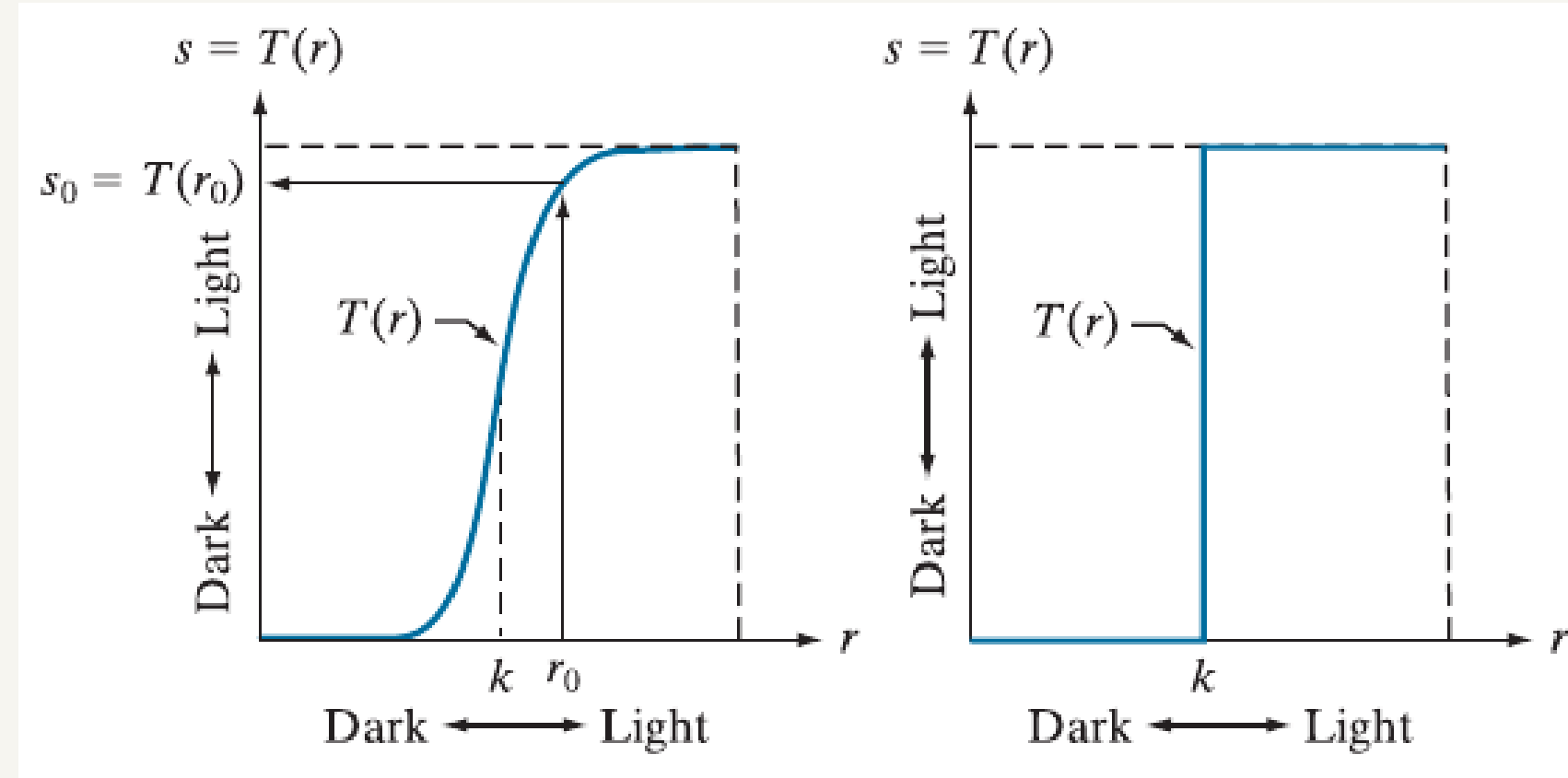
Şekil 3.1 · Bir (x_0, y_0) noktası etrafındaki 3×3 komşuluk

Tek piksellik komşuluk: yoğunluk dönüşümü

Komşuluk tek piksele indiğinde T bir **yoğunluk dönüşüm fonksiyonuna** dönüşür. Eğrinin dikliği kontrastı belirler: dik bölgeler kontrastı artırır, yatay bölgeler bastırır. Sınırdaki hâli eşikleme yapar.

$$s = T(r)$$

r : giriş yoğunluğu · s : çıkış yoğunluğu



Şekil 3.2 · (a) Kontrast germe (b) Eşikleme dönüşümü

BÖLÜM 3.2

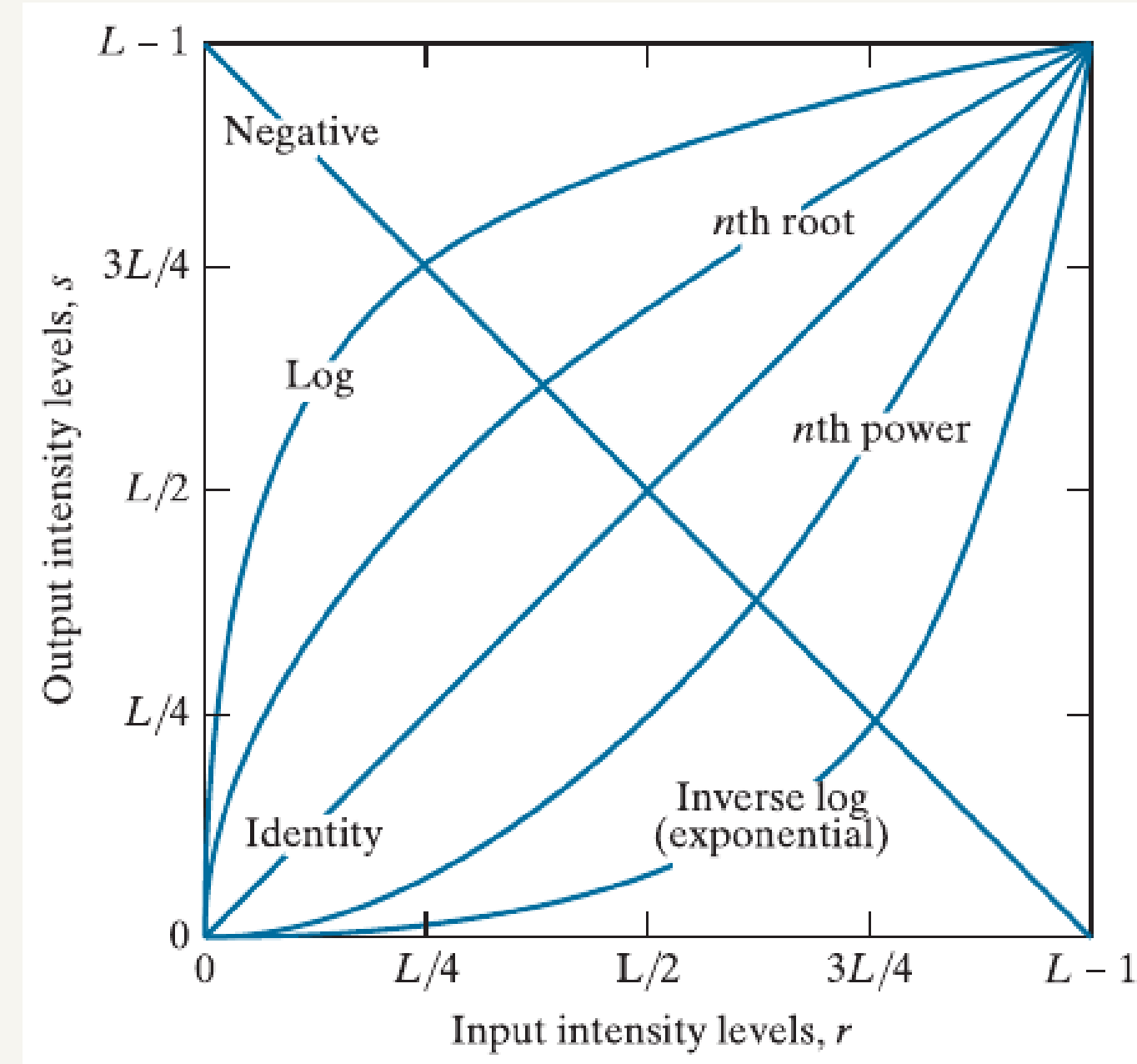
Yoğunluk dönüşüm fonksiyonları

Tek piksel üzerinde çalışan en basit ve en yaygın iyileştirme ailesi: negatif, logaritmik, kuvvet yasası (gama) ve parçalı doğrusal dönüşümler.

Üç temel dönüşüm ailesi

Bütün nokta işlemleri tek bir grafikte özetlenebilir. Eğri **köşegenin üstünde** kalıyorsa görüntü aydınlanır, **altında** kalıyorsa kararır.

- **Doğrusal:** özdeşlik ve negatif
- **Logaritmik:** log ve ters log
- **Kuvvet yasası:** n. kuvvet ve n. kök
- **Parçalı doğrusal:** germe, dilimleme, bit düzlemi



Dört aile, dört farklı iş

DOĞRUSAL

$$s = r$$

$$s = L - 1 - r$$

Eğim sabittir. Özdeşlik hiçbir şeyi değiştirmez; **negatif** ölçeği baştan sona ters çevirir ve koyu bir zemine gömülü açık ayrıntıyı okunur kılar. Tersinirdir — bilgi kaybı yoktur.

LOGARİTMİK

$$s = c \log(1 + r)$$

$$s = e^{r/c} - 1$$

Karanlık uçtaki **dar** bir aralığı **geniş** bir aralığa yayar, parlak ucu sıkıştırır. Dinamik aralığı ekranın gösterebileceğinden çok geniş olan verilerde tek pratik çözüm. Ters log tam tersini yapar.

KUVVET YASASI

$$s = c r^\gamma$$

n. kök ($\gamma < 1$) karanlığı açar, **n. kuvvet** ($\gamma > 1$) parlaklığı bastırır. Log'dan farkı: tek bir γ ile eğrinin sertliği sürekli ayarlanır. Ekran ve kamera gama düzeltmesinin temeli budur.

PARÇALI DOĞRUSAL

$$s = a_i r + b_i, \quad r \in [r_i, r_{i+1})$$

Birkaç doğru parçasının birleşimi; istenildiği kadar karmaşık yapılabilir. Bedeli: her kırılma noktası **elle** belirlenmelidir. Kontrast germe, yoğunluk dilimleme ve bit düzlemi dilimleme bu aileden.

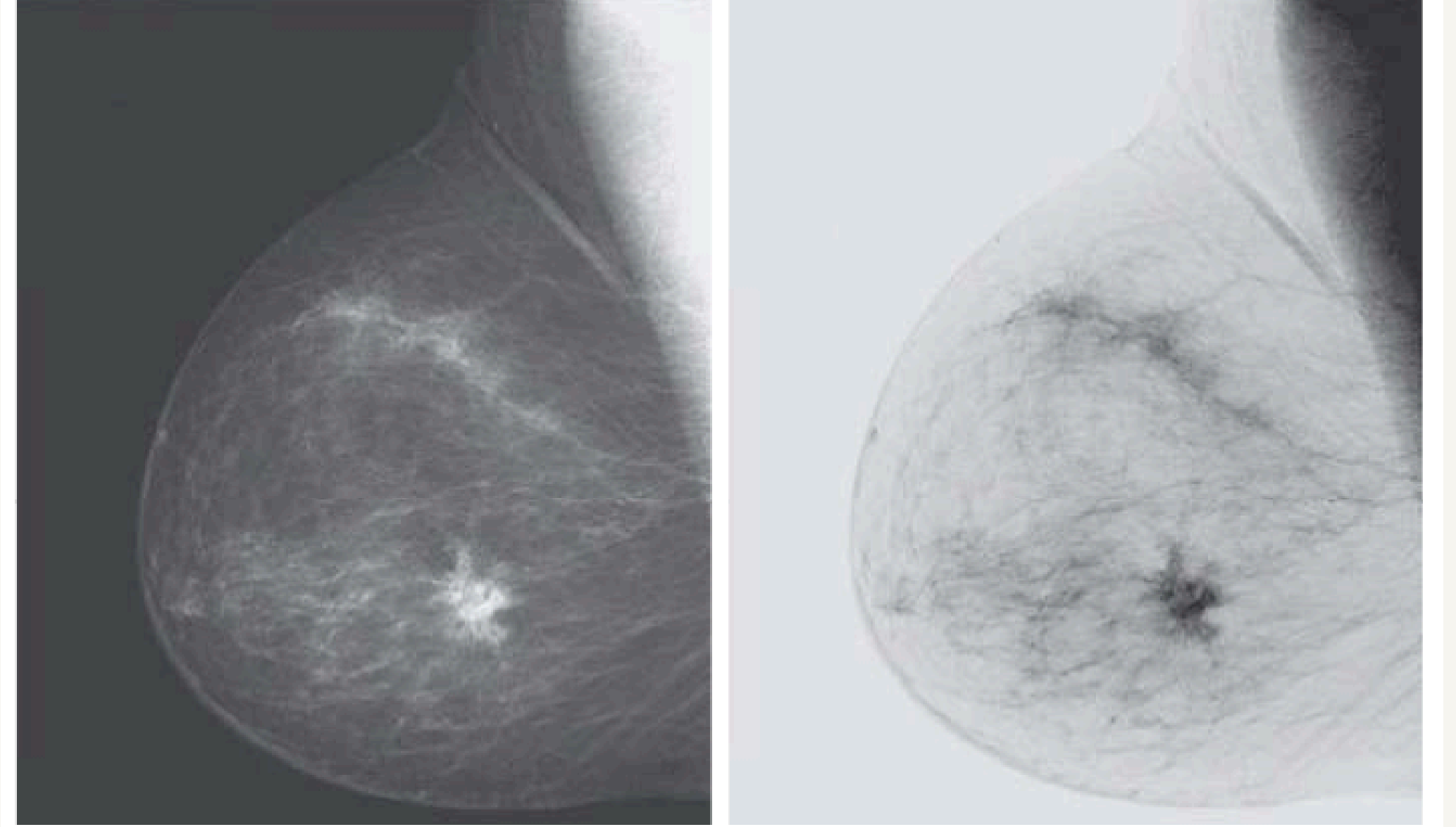
Görüntü negatifi

Yoğunluk ölçeğini ters çevirir. Boyutça küçük olan **beyaz veya gri ayrıntıların** baskın koyu bir alan içinde gömülü olduğu görüntülerde çok etkilidir — tıbbi görüntülemede standart bir inceleme adımıdır.

$$s = L - 1 - r$$

$$s = 255 - r \quad (L = 256)$$

Kendi tersidir: negatfin negatifi özgün görüntüdür.
Bilgi kaybı yoktur.

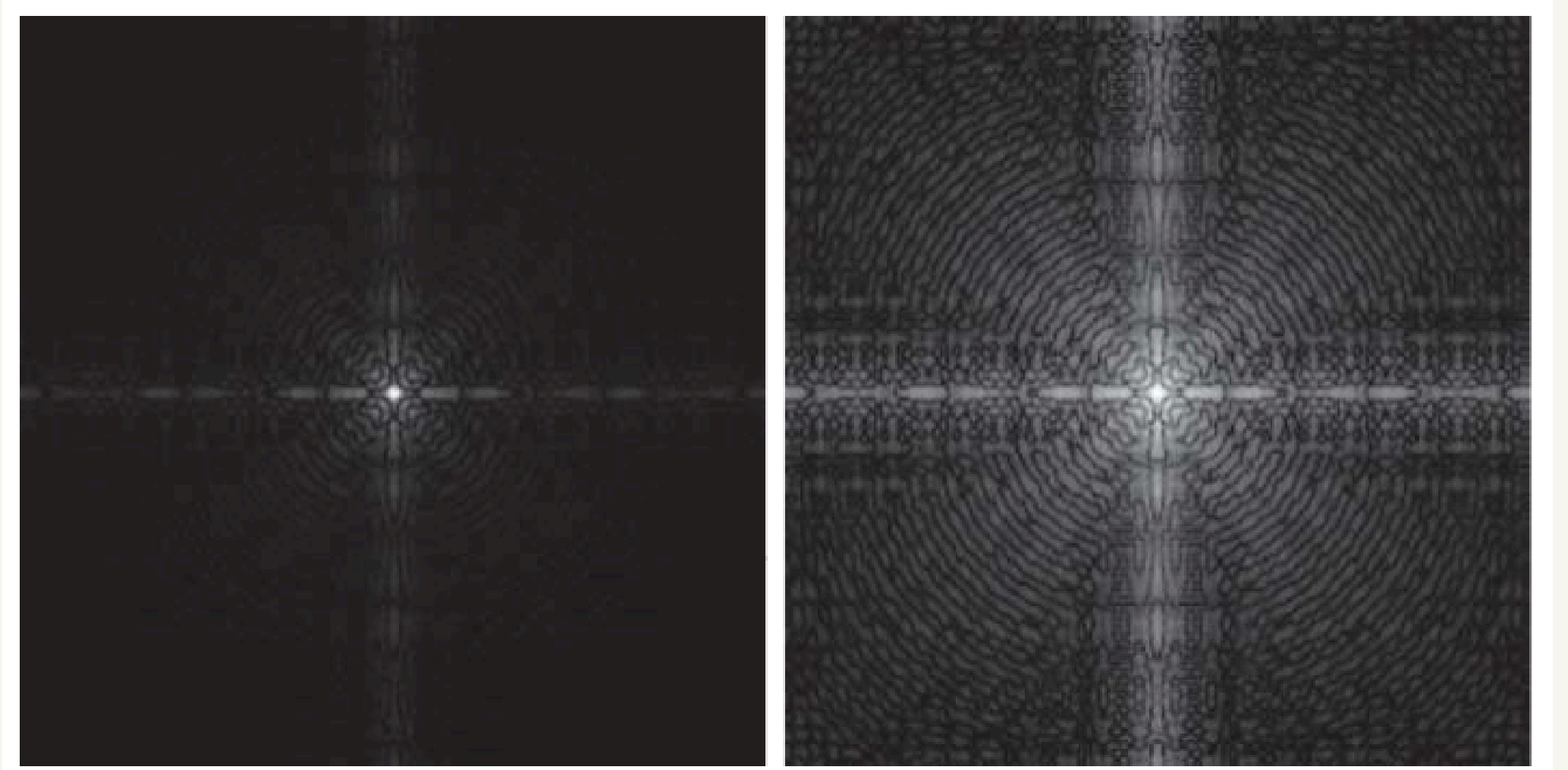


Logaritmik dönüşüm

Girişteki **dar bir düşük yoğunluk aralığını** çıkışta geniş bir aralığa yayar; yüksek yoğunlukları ise sıkıştırır. Dinamik aralığı ekranın gösterebileceğinden çok geniş olan verilerde vazgeçilmezdir.

$$s = c \log(1 + r)$$

c: ölçek sabiti — bir sonraki slaytta



Ölçek sabiti c neyi çözüyor?

SORUN

$$0 \leq r \leq R_{\max} \Rightarrow 0 \leq \log(1 + r) \leq \log(1 + R_{\max})$$

ÇÖZÜM

$$c = \frac{L - 1}{\log(1 + R_{\max})} \quad s = c \log(1 + r)$$

Ters log dönüşümü ($s = e^{r/c} - 1$) tam tersini yapar: parlak uçtaki değerleri yayar, karanlığı sıkıştırır.

Neden gerekli? $\log(1 + r)$ en fazla $\log(1 + R_{\max})$ olur. Bu sayının ekranın yoğunluk ölçeğiyle hiçbir ilgisi yoktur — 8 bit bir ekrana 0 ile 14 arasında değerler göndermiş olursunuz ve görüntü neredeyse tamamen siyah kalır.

c ne yapıyor? Tek işi ölçeklemek. Paydadaki $\log(1 + R_{\max})$ en büyük çıkışı 1'e indirger, $L - 1$ ile çarpmak da onu ekranın tavanına taşır. Yani $r = R_{\max}$ iken s tam olarak $L - 1$ çıkar; $r = 0$ iken $s = 0$.

Sayısal örnek. Fourier spektrumunda $R_{\max} \approx 1.5 \times 10^6$, ekran 8 bit ($L = 256$):

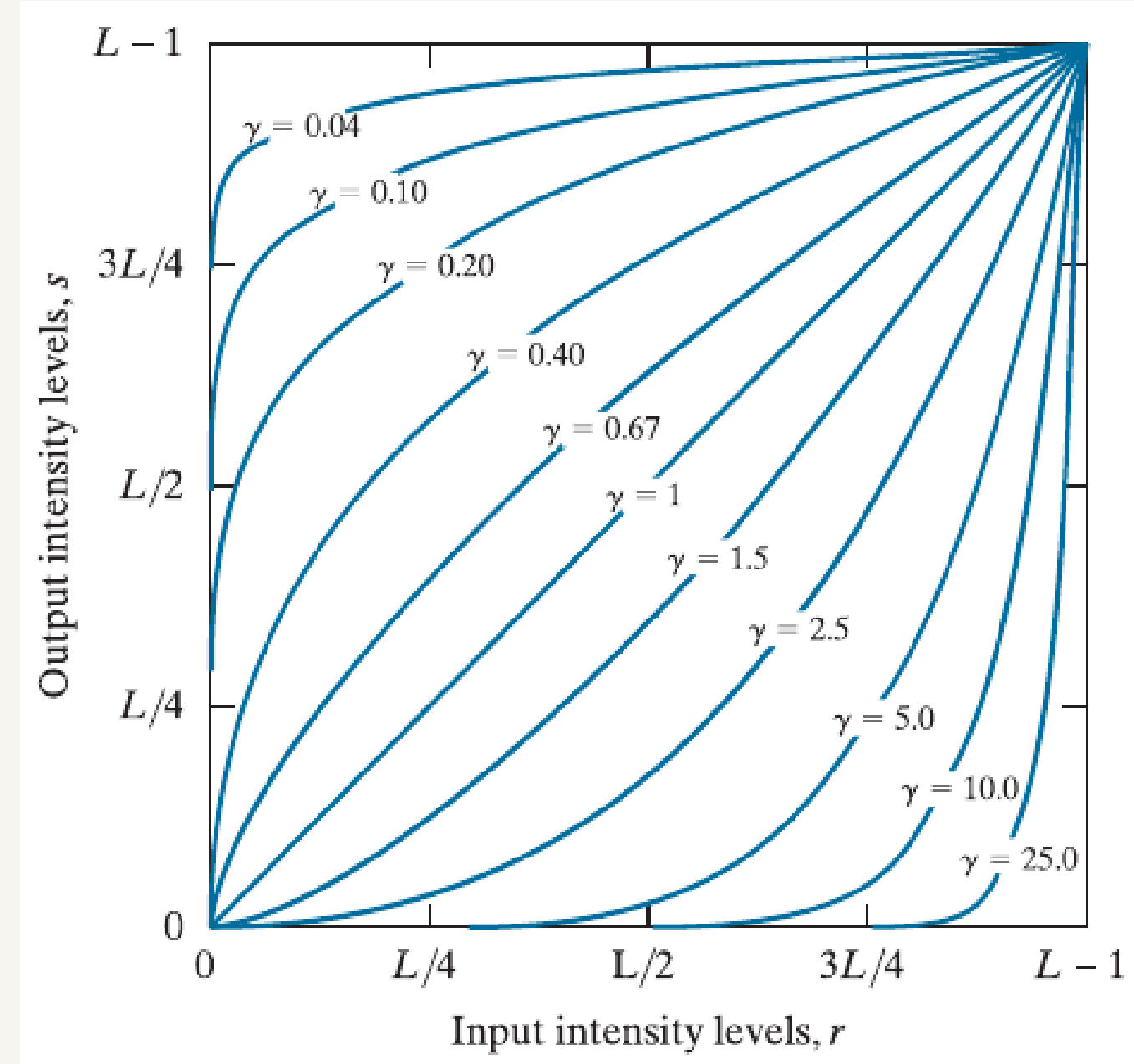
$$c = 255 / \log(1 + 1.5 \times 10^6) \approx 255 / 14.2 \approx \mathbf{18}$$

Kuvvet yasası (gama) dönüşümü

Tek bir γ parametresiyle çok geniş bir dönüşüm ailesi elde edilir. $\gamma < 1$ karanlık bölgeleri açar, $\gamma > 1$ parlak bölgeleri bastırır. $\gamma = 1$ özdeşlik dönüşümüdür.

$$s = c r^\gamma$$

$\gamma < 1$ (n. kök): aydınlatır · $\gamma > 1$ (n. kuvvet): karartır ·
 $\gamma = 1$: özdeşlik



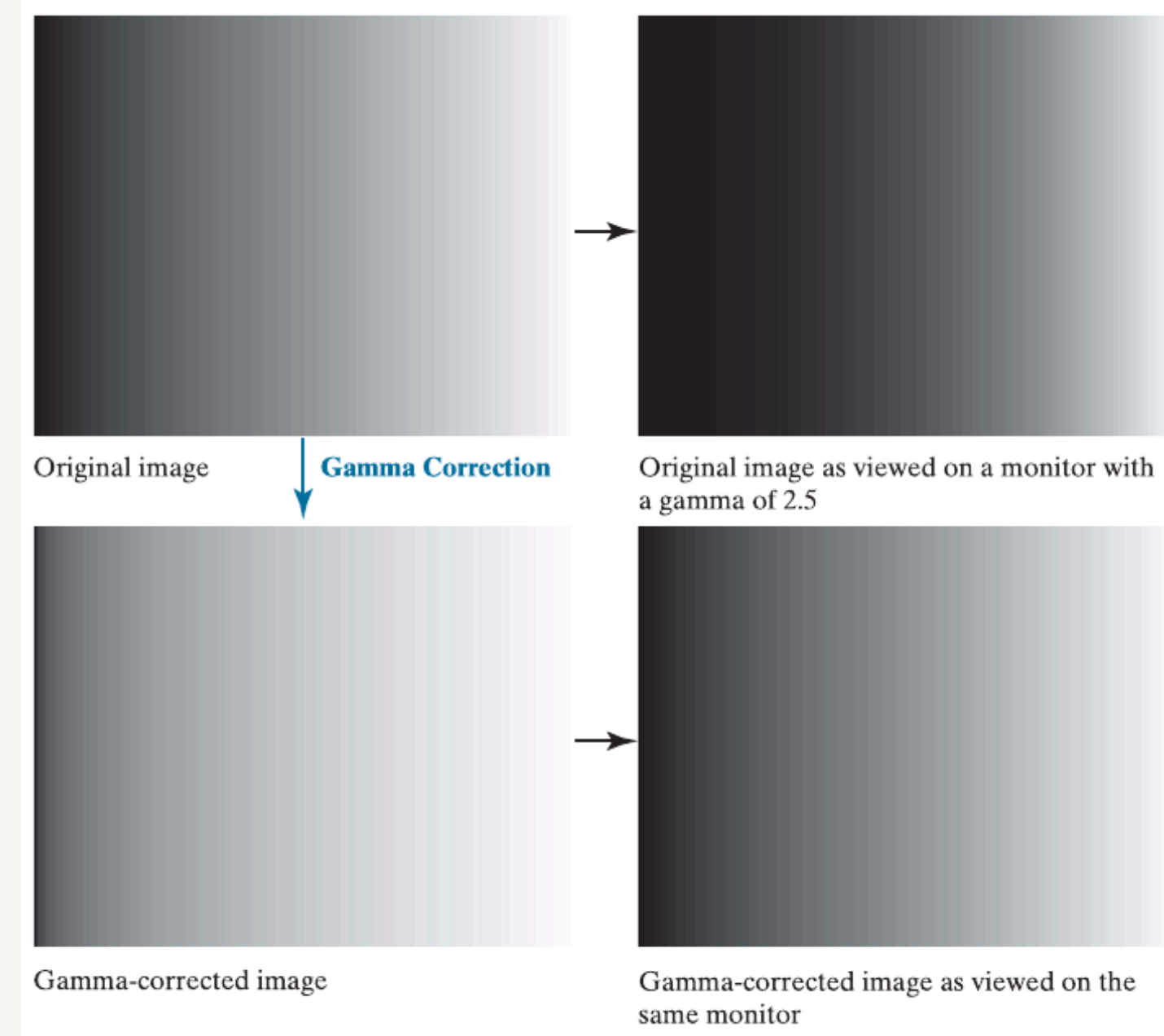
Şekil 3.6 · Çeşitli γ değerleri için kuvvet yasası eğrileri

Gama düzeltmesi

Ekranlar, yazıcılar ve kameralar doğal olarak bir kuvvet yasası tepkisine sahiptir. Görüntüyü gösterimden önce **ters gama** ile ön işlemek, cihazın bozucu etkisini iptal eder.

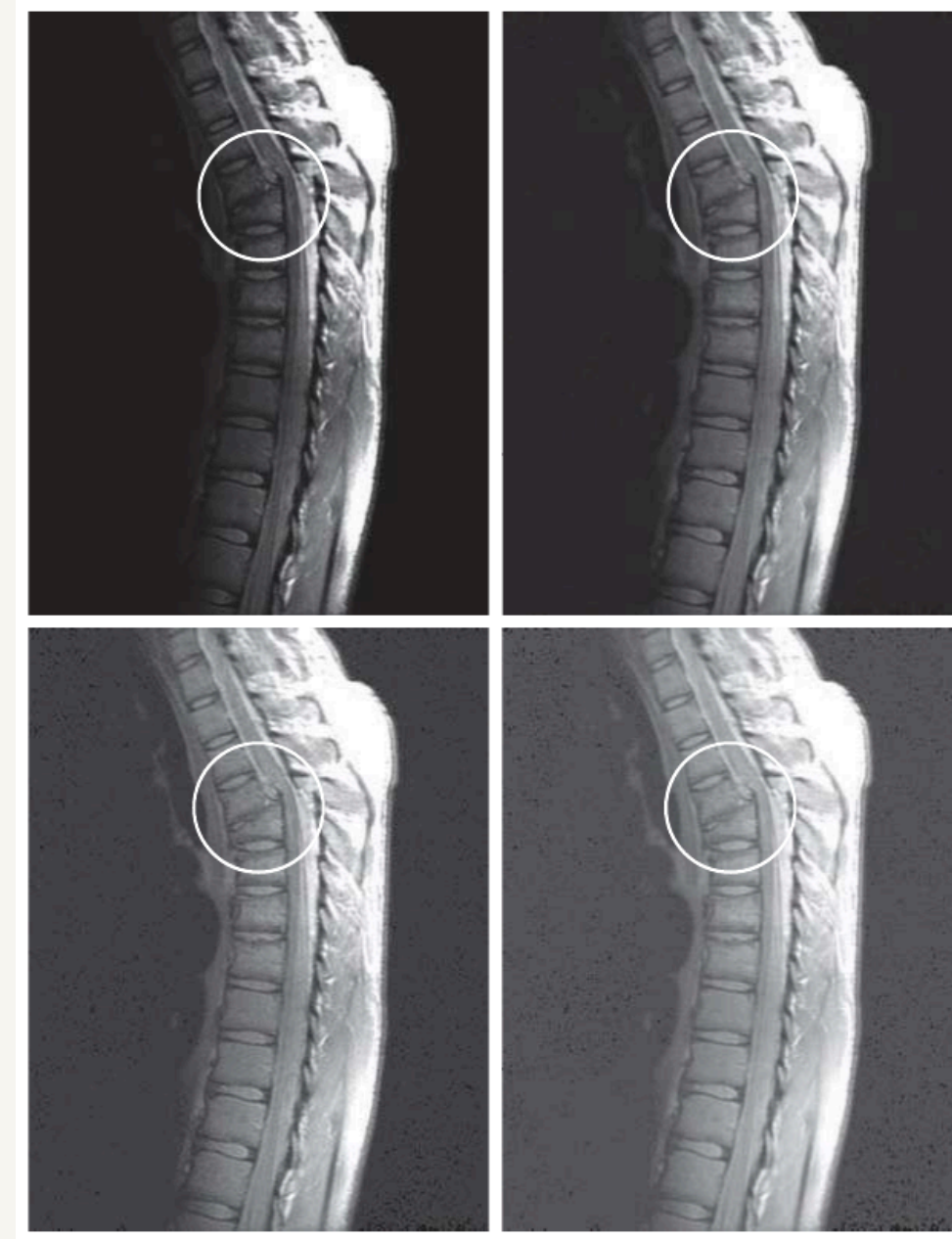
$$s = r^{1/2.5} = r^{0.4}$$

Ekran gaması 2.5 ise uygulanacak düzeltme



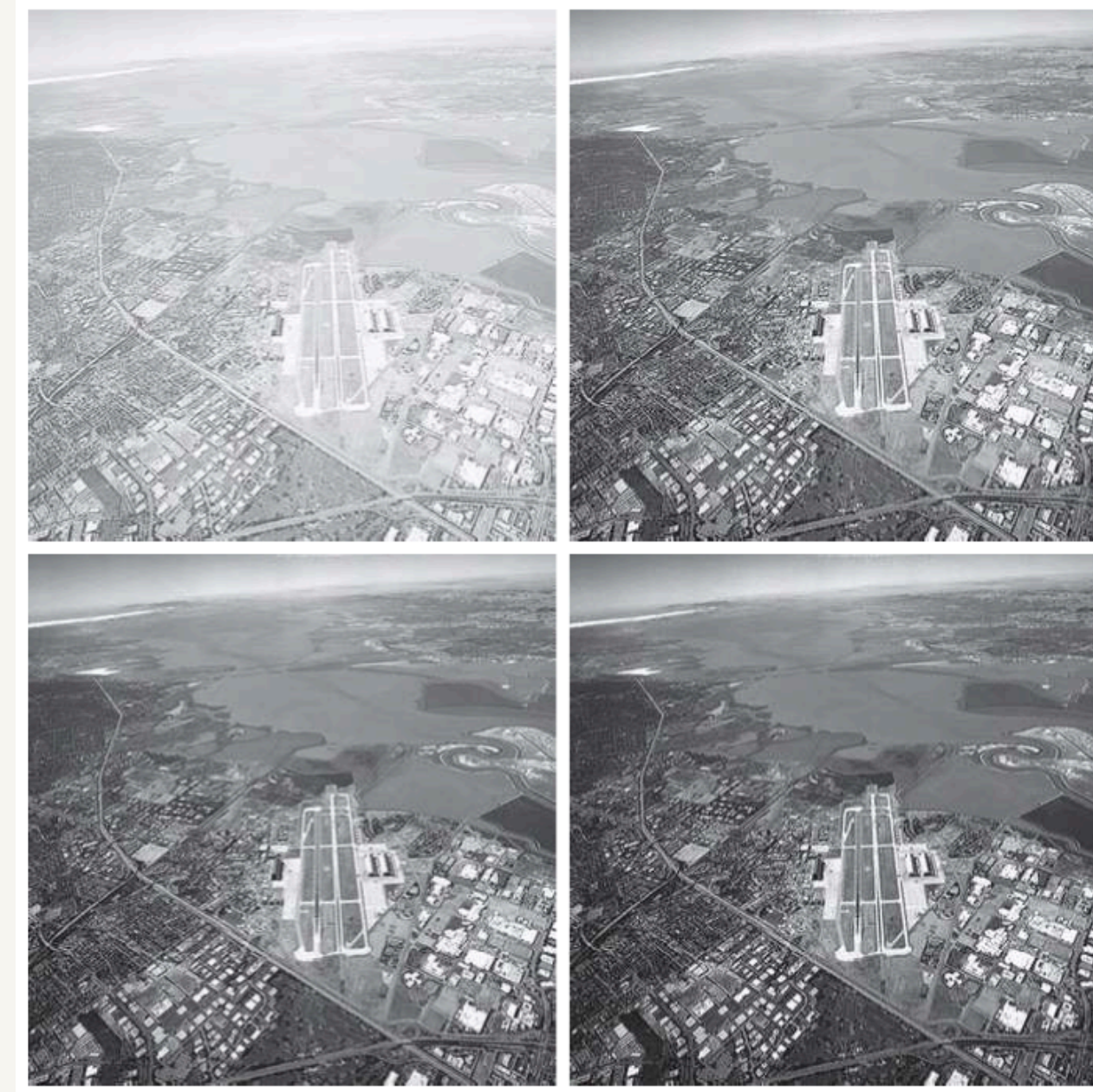
Örnek: küçük γ ile karanlık ayrıntıyı açmak

Kırık omurganın MR görüntüsü fazla karanlıktır. γ değeri $0.6 \rightarrow 0.4 \rightarrow 0.3$ olarak düşürüldükçe kırık bölgesi belirginleşir; ancak **aşırı düşük γ** kontrastı yıkar ve görüntü yıkanmış görünür.



Örnek: büyük γ ile yıkanmış görüntüyü toparlamak

Havadan çekilmiş görüntü ise tersine **fazla açıktır**. $\gamma = 3.0$, 4.0 ve 5.0 ile kontrast geri kazanılır; $\gamma = 4.0$ bu görüntü için en dengeli sonucu verir.



Parçalı doğrusal: kontrast germe

Düşük kontrastlı bir görüntünün yoğunluk aralığını, ekranın tüm dinamik aralığına yayar. (r_1, s_1) ve (r_2, s_2) kontrol noktaları fonksiyonun şeklini belirler.

ÜÇ DOĞRU PARÇASI

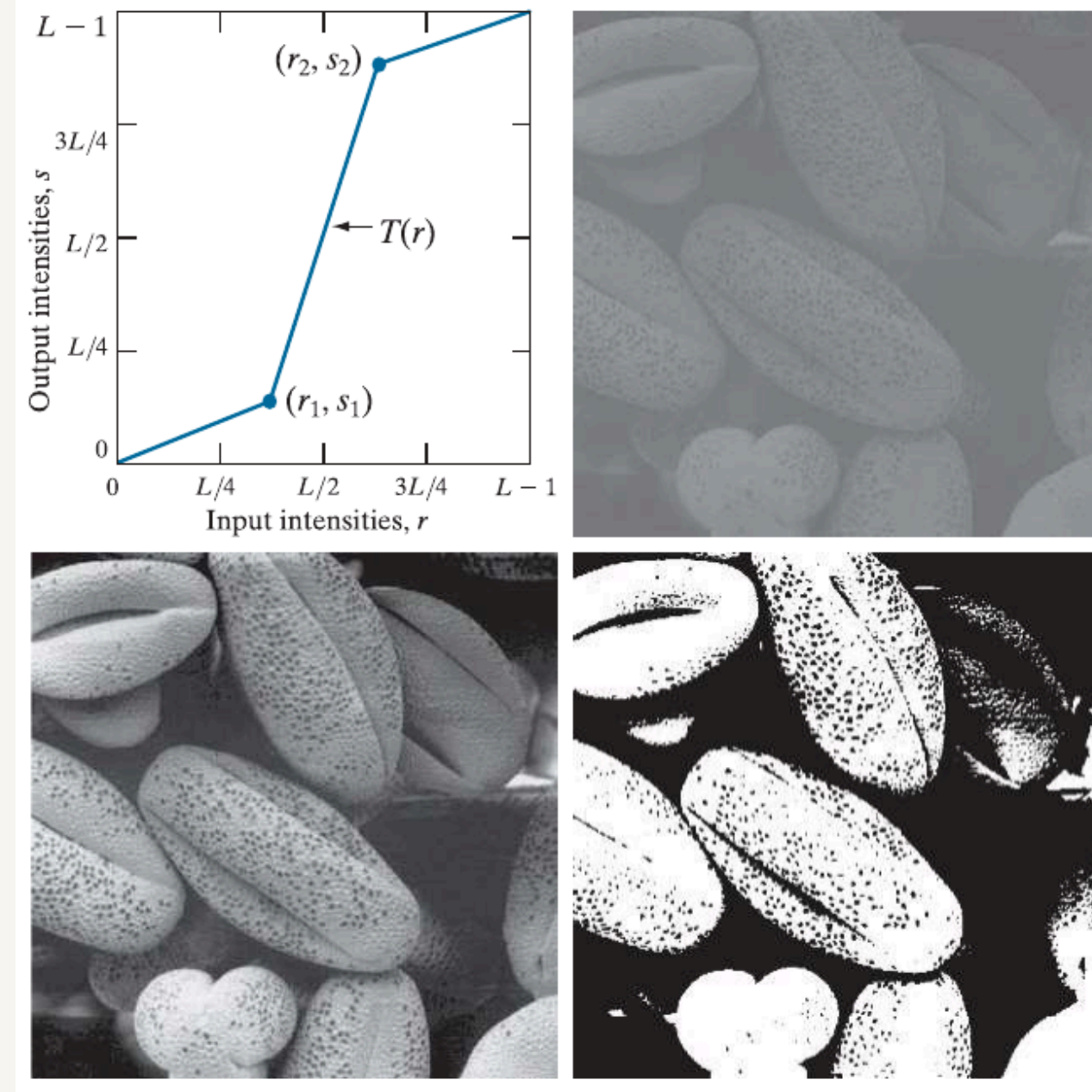
$$0 \leq r < r_1 : s = \frac{s_1}{r_1} r$$

$$r_1 \leq r < r_2 : s = s_1 + \frac{s_2 - s_1}{r_2 - r_1} (r - r_1)$$

$$r_2 \leq r \leq L - 1 : s = s_2 + \frac{(L - 1) - s_2}{(L - 1) - r_2} (r - r_2)$$

Genelde r_1 = en küçük, r_2 = en büyük yoğunluk seçilir; $s_1 = 0$, $s_2 = L - 1$ alınır. Fonksiyonun **tek artan** olması yoğunluk sıralamasını korur — aksi hâlde görüntüde olmayan yapay kenarlar oluşur.

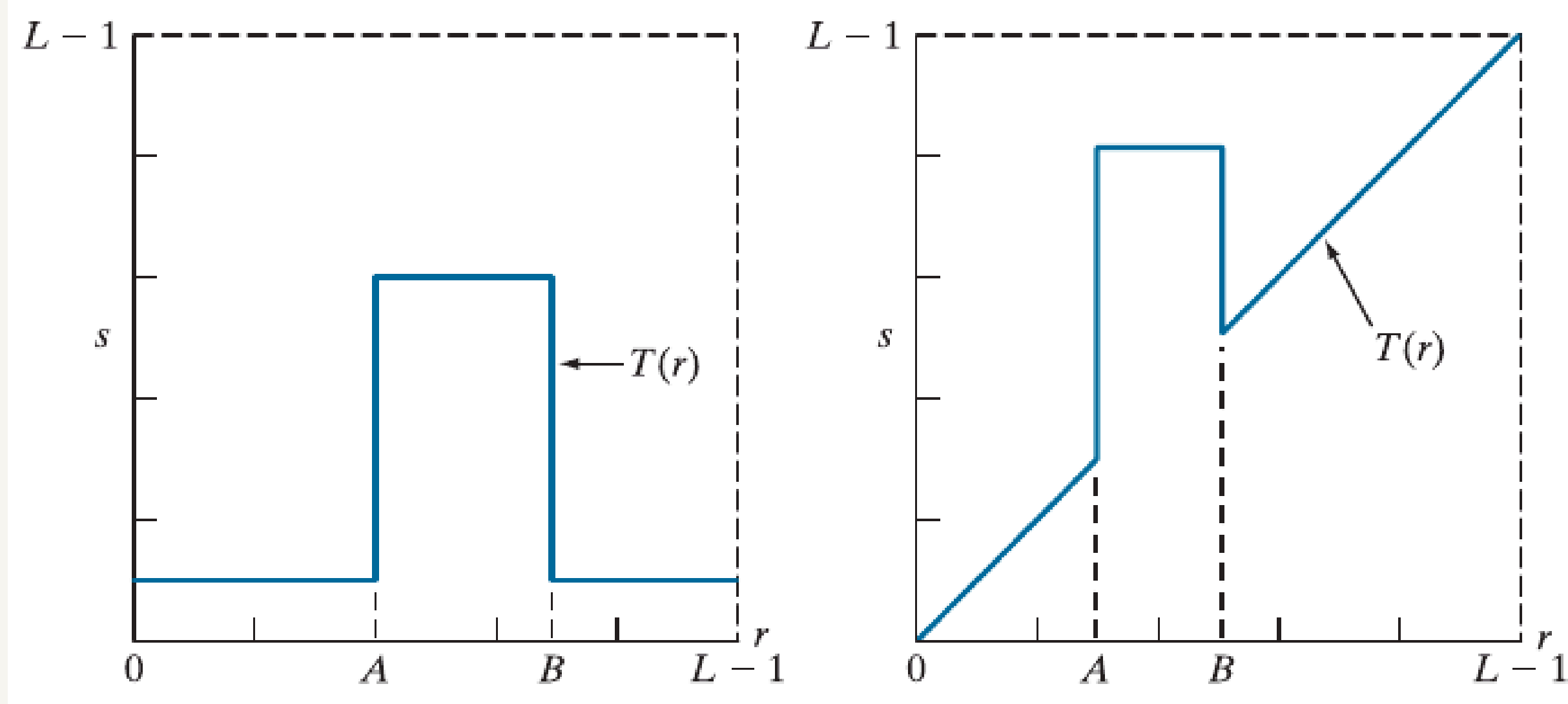
$r_1 = r_2$ seçilirse orta parça dikleşip sonsuz eğime gider: fonksiyon **eşiklemeye** dönüşür ve çıktı ikili görüntü olur.



Şekil 3.10 · Kontrast germe ve eşikleme sonucu

Yoğunluk düzeyi dilimleme

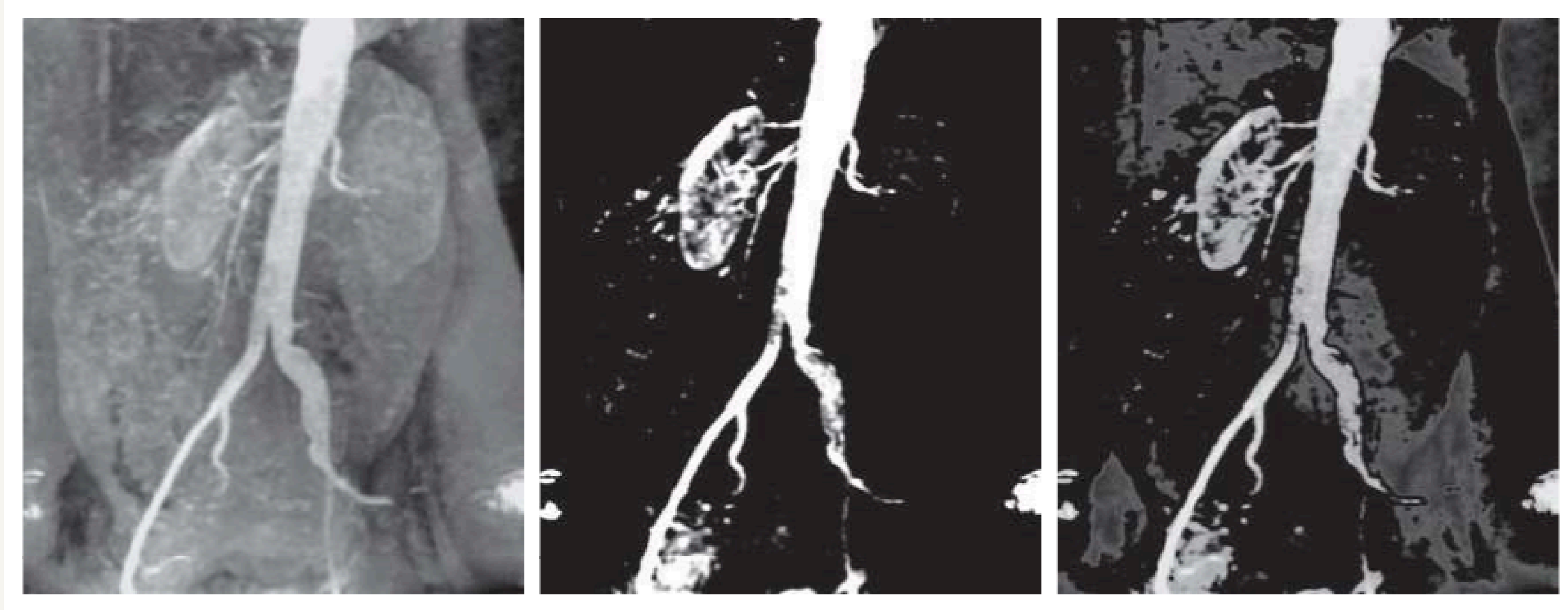
Belirli bir **[A, B]** aralığındaki yoğunlukları öne çıkarır. İki yaygın biçim vardır: aralık dışını bastırmak, ya da aralığı vurgularken diğer yoğunlukları korumak.



Şekil 3.11 · (a) Aralık dışını bastıran (b) Diğerlerini koruyan dönüşüm

Örnek: aort anjiyogramı

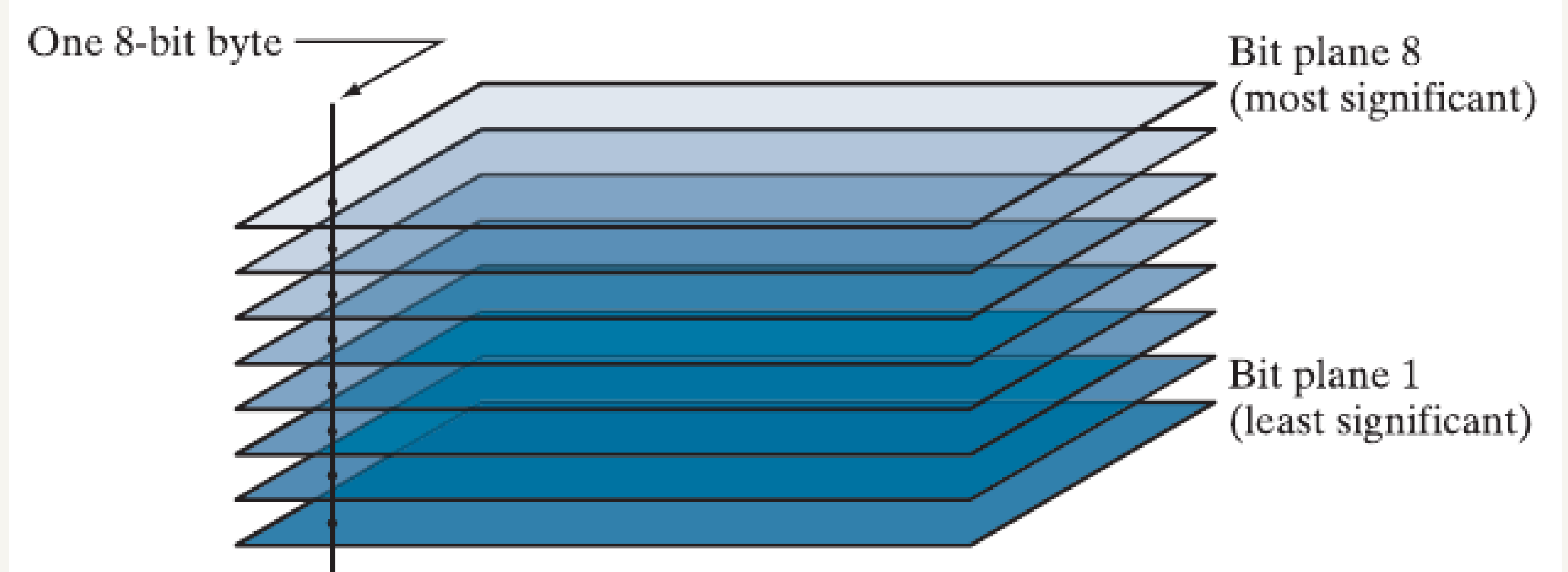
Enjekte edilen kontrast maddesi damarları arka plandan daha parlak yapar. Dilimleme ile bu yoğunluk bandı yalıtılır; ikinci biçimde ise damarlar vurgulanırken arka plan yapısı da korunur.



Şekil 3.12 · Anjiyogram ve iki farklı dilimleme sonucu

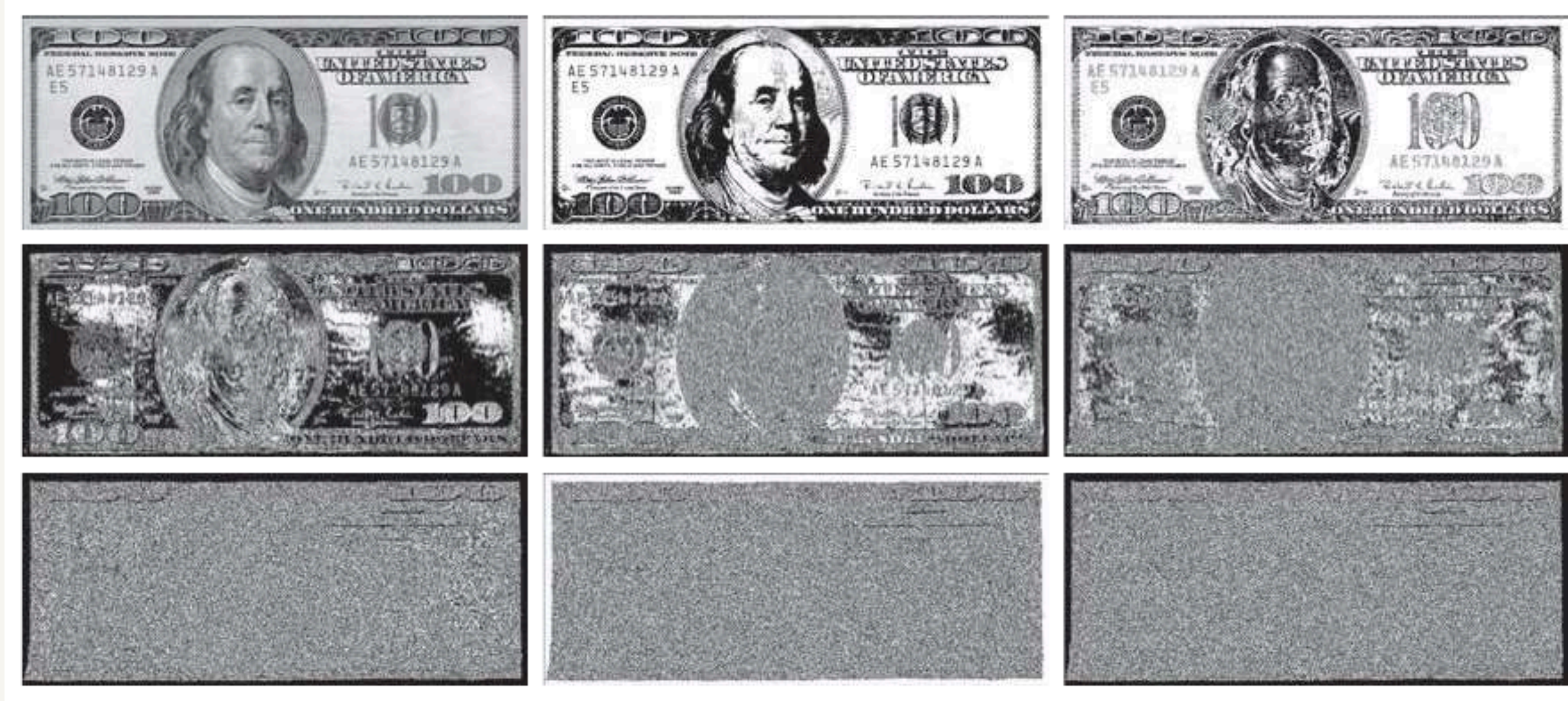
Bit düzlemi dilimleme

8 bitlik bir görüntü, her biri ikili olan **8 bit düzleminin** üst üste binmesi olarak düşünülebilir. Yüksek dereceli düzlemler görsel olarak anlamlı yapıyı, düşük dereceliler ince ayrıntı ve gürültüyü taşır.



Bit düzlemleri: örnek

100 dolarlık banknotun bit düzlemleri. **8. ve 7. düzlem** görüntünün tanınabilir yapısını taşıırken, alt düzlemler gürültüye benzeyen ince ayrıntıdan oluşur.



BÖLÜM 3.3

Histogram işleme

Görüntünün yoğunluk dağılımını doğrudan hedefleyen yöntemler: histogram eşitleme, histogram eşleme ve yerel uygulamaları.

Histogram nedir?

Histogram, her yoğunluk düzeyinde kaç piksel bulunduğunu gösterir. Histogramın **şekli ve yayılımı**, görüntünün parlaklığı ve kontrastı hakkında doğrudan bilgi verir.

$$h(r_k) = n_k \quad p(r_k) = \frac{n_k}{MN}$$

n_k : r_k yoğunluğundaki piksel sayısı · MN : toplam piksel

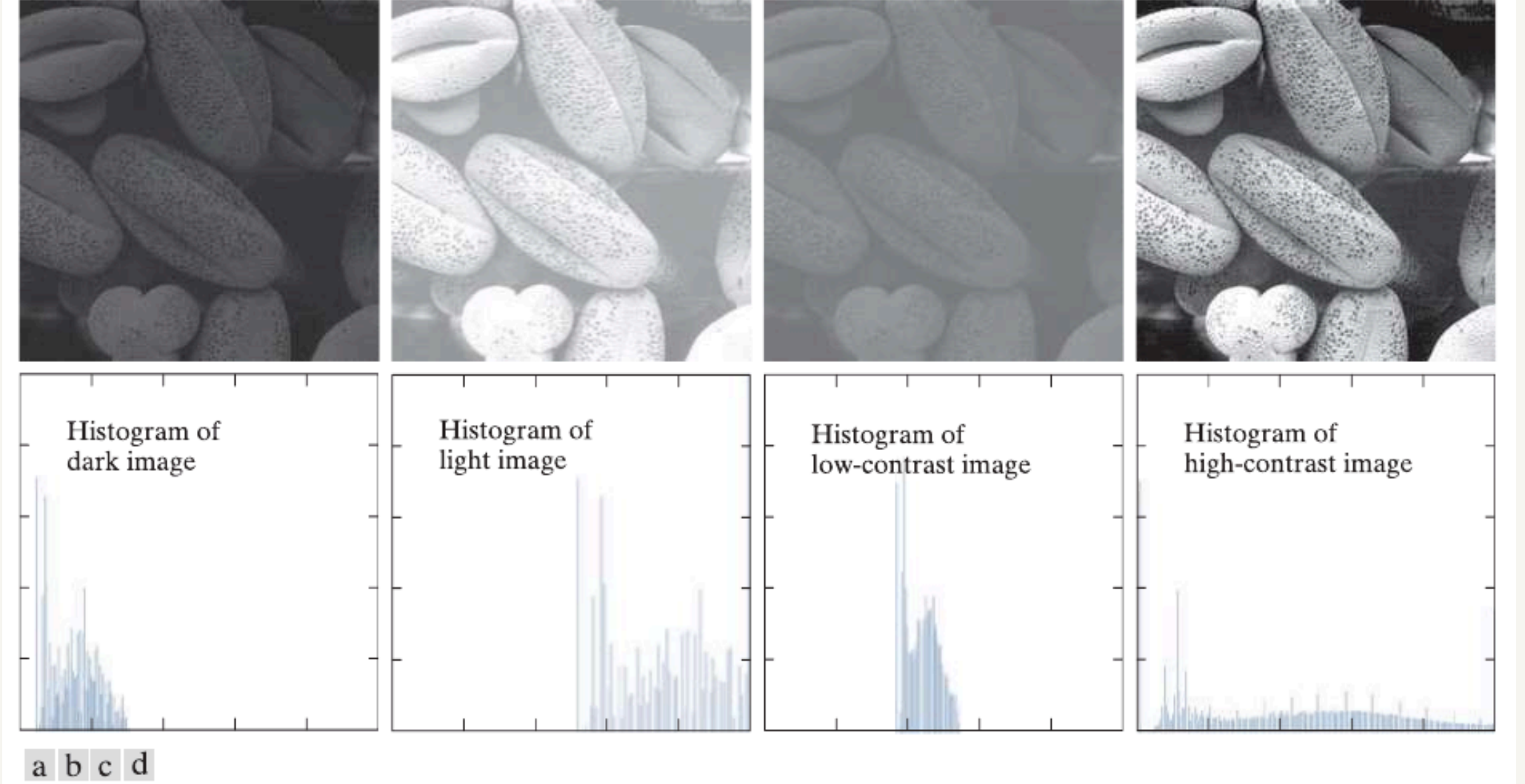


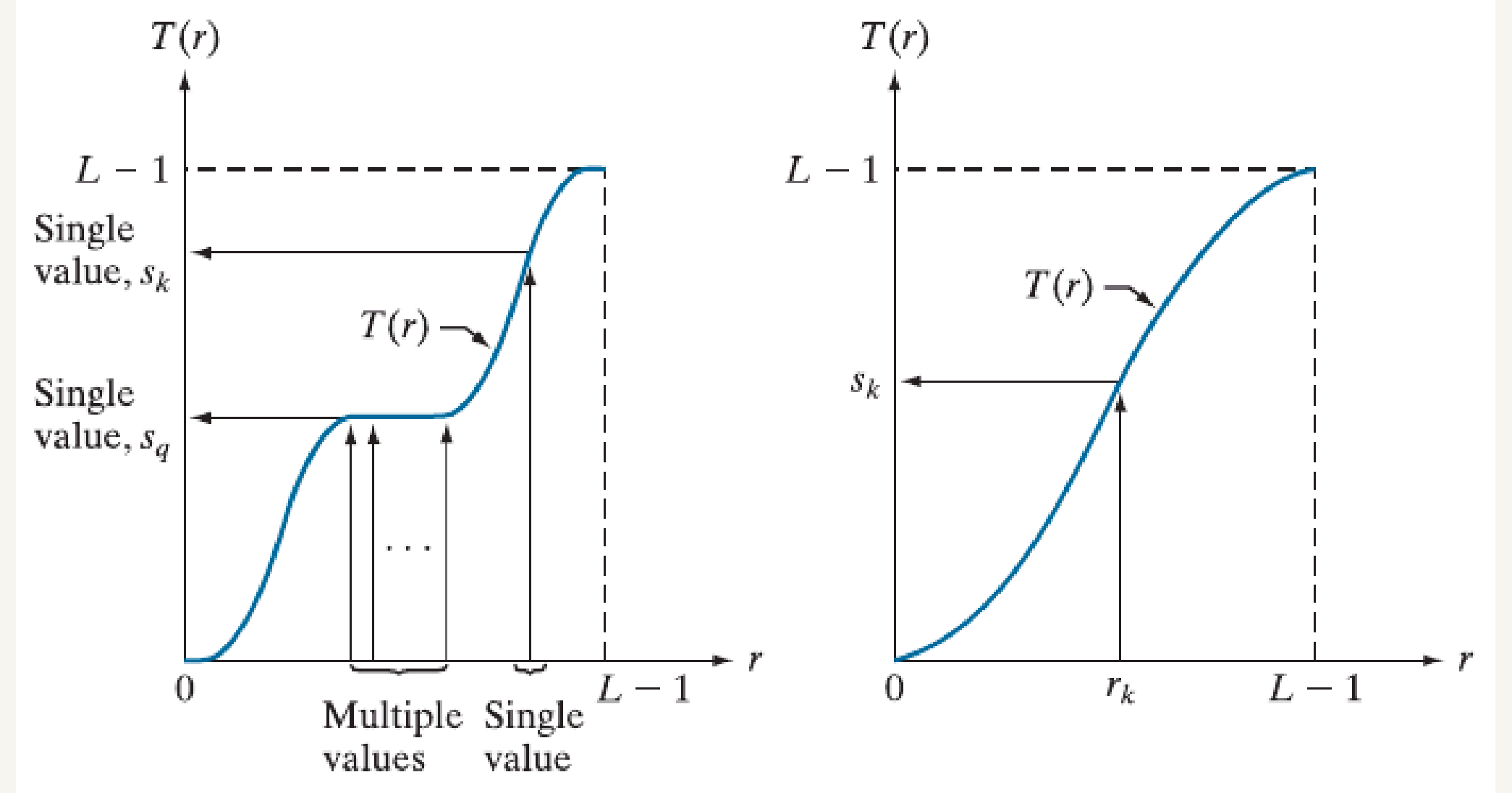
FIGURE 3.16 Four image types and their corresponding histograms. (a) dark; (b) light; (c) low contrast; (d) high contrast. The horizontal axis of the histograms are values of r_k and the vertical axis are values of $p(r_k)$.

Histogram eşitlemenin temeli

Dönüşümün **tek artan** olması yoğunluk sıralamasını korur; **kesin artan** olması ise dönüşümün tersinir olmasını sağlar. Bu iki koşul, eşitlemenin matematiksel dayanağıdır.

$$s = T(r) = (L - 1) \int_0^r p_r(w) dw$$

Kümülatif dağılım fonksiyonu (CDF). Tek artan olduğu için yoğunluk sıralaması korunur.



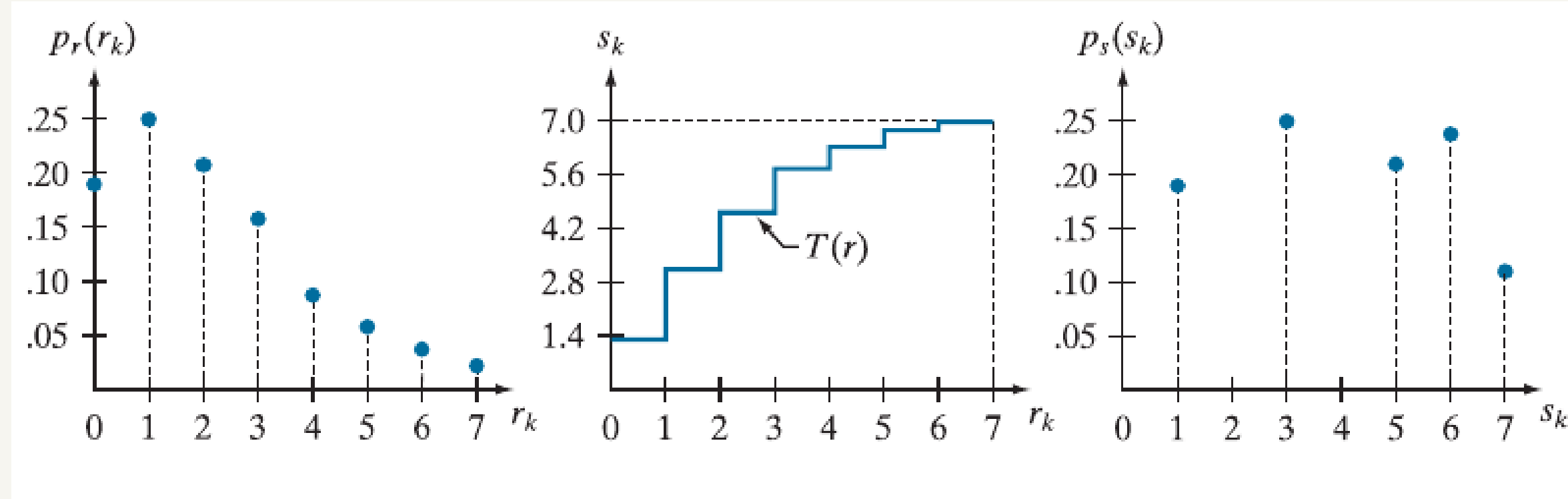
Şekil 3.17 · (a) Tek artan (b) Kesin artan dönüşüm

Ayrık histogram eşitleme

Pratikte kullanılan biçim, normalize histogramın **kümülatif toplamıdır**. Hesap tamamen histogramdan yapılır; görüntü hakkında başka hiçbir parametre gerekmez — yöntem **otomatiktir**.

$$s_k = T(r_k) = (L - 1) \sum_{j=0}^k p_r(r_j)$$

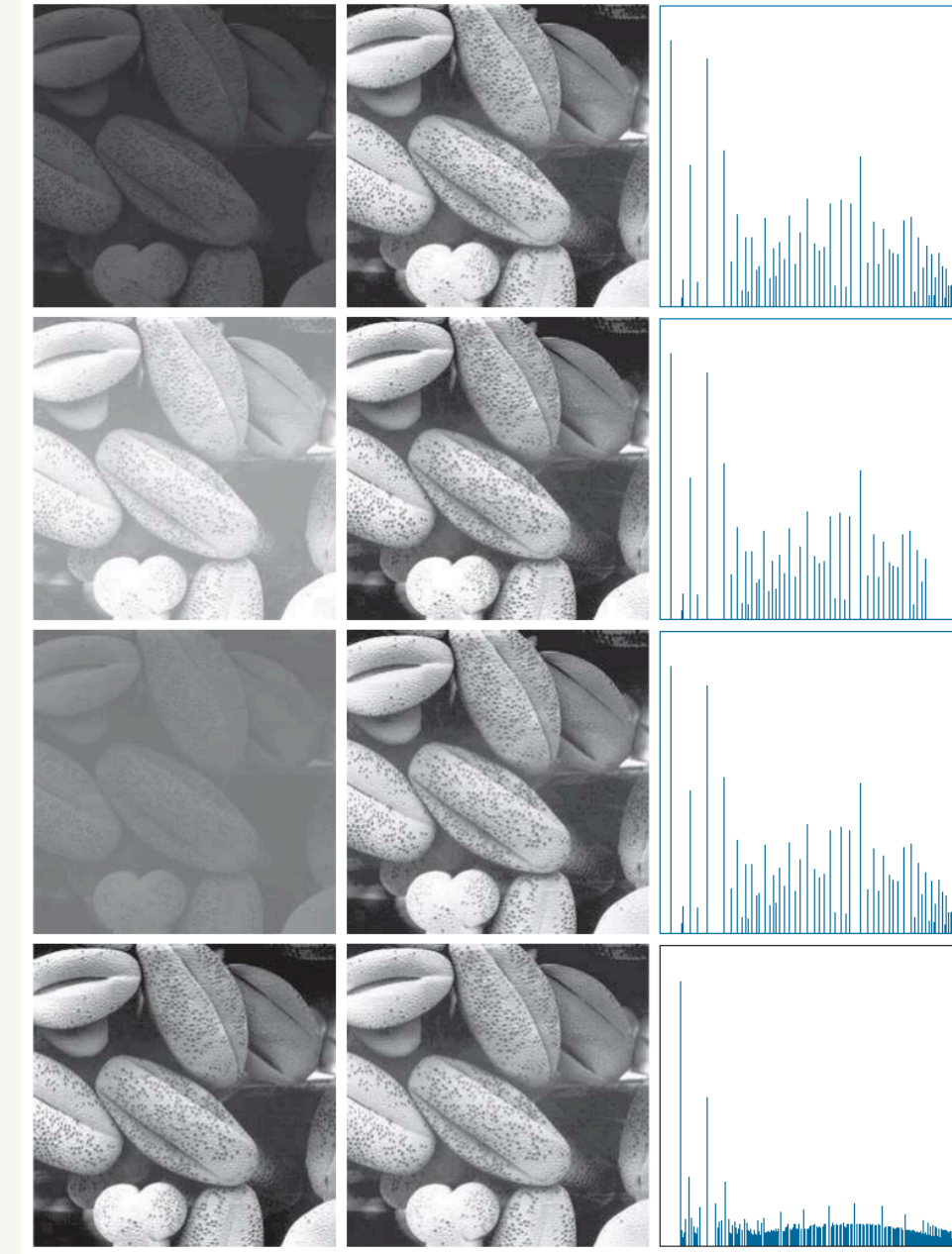
$k = 0, 1, \dots, L - 1$ · sonuç en yakın tam sayıya yuvarlanır



Şekil 3.19 · (a) Özgün histogram (b) Dönüşüm fonksiyonu (c) Eşitlenmiş histogram

Histogram eşitleme: sonuçlar

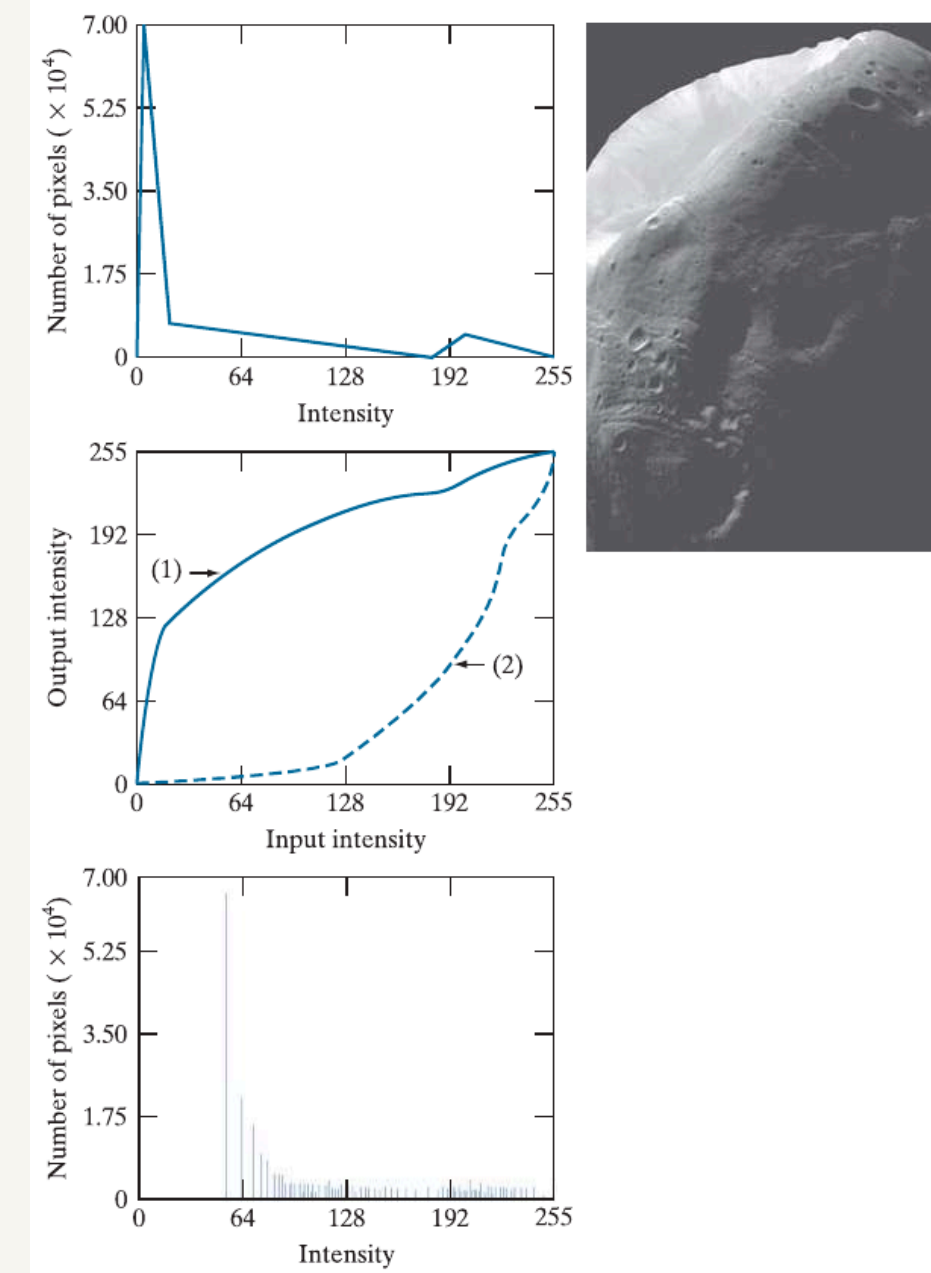
Şekil 3.16'daki dört görüntünün eşitlenmiş hâlleri. Dördü de **birbirine çok benzer** hâle gelir — çünkü eşitleme, kaynak dağılım ne olursa olsun aynı hedefe (düzgün dağılıma) yönelir.



Histogram eşleme (belirtme)

Eşitleme her zaman düzgün dağılımı hedefler; bu bazen istenmeyen sonuç verir. **Histogram eşleme**, çıkış histogramını bizim belirlediğimiz bir şekle yaklaştırır.

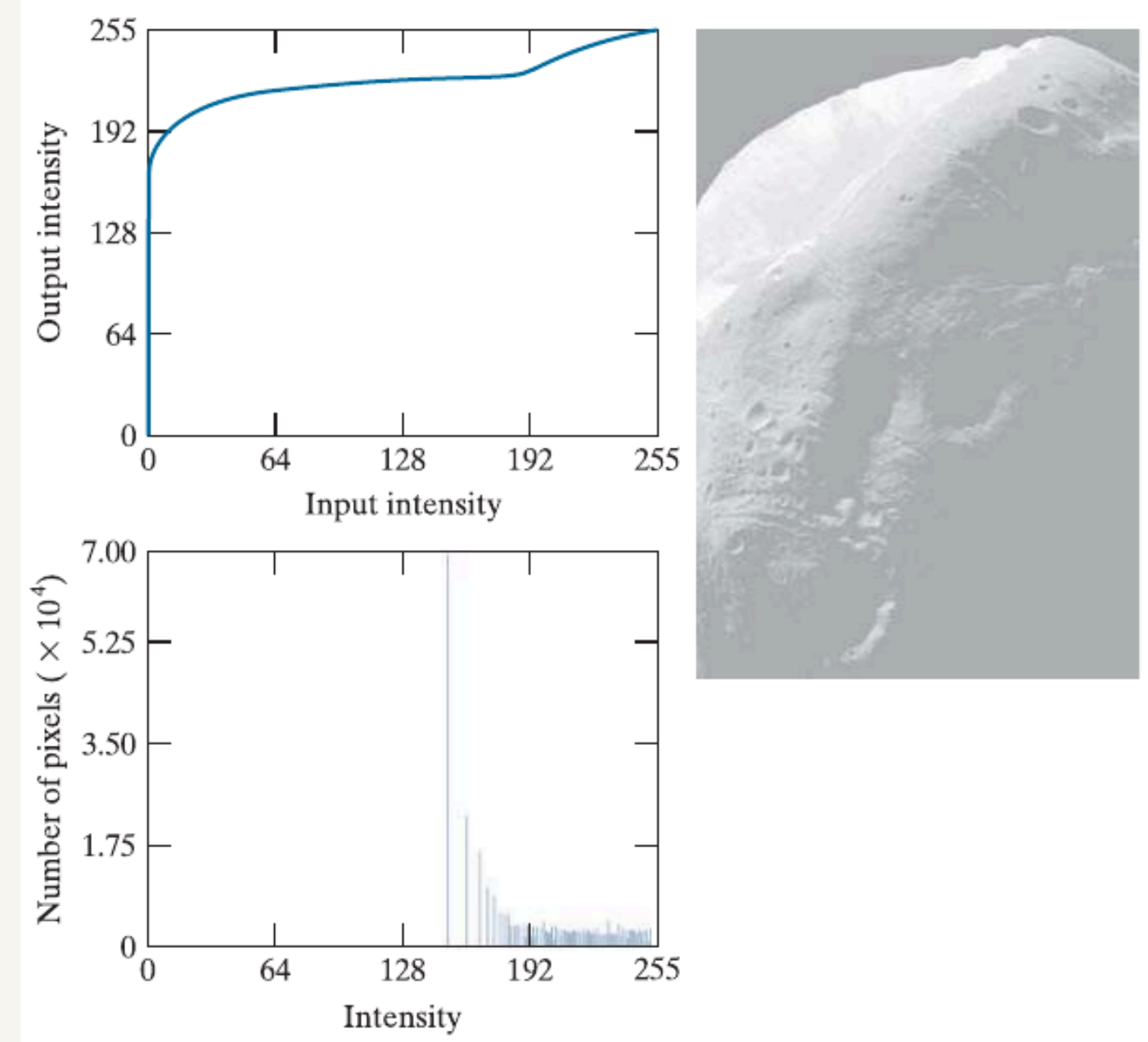
- Adım 1: giriş histogramını eşitle $\rightarrow s$
- Adım 2: hedef histogramı eşitle $\rightarrow G(z)$
- Adım 3: $z = G^{-1}(s)$ ile geri eşle



Şekil 3.25 · Belirtilen histogram, dönüşüm fonksiyonları ve sonuç

Neden eşleme gerekir? — Phobos örneği

Mars'ın uydusu Phobos'un görüntüsü **geniş karanlık alanlarla** doludur. Histogram eşitleme bu alanları aşırı aydınlatır ve görüntüyü yıkanmış hâle getirir; eşleme ise kontrolü bizde bırakır.

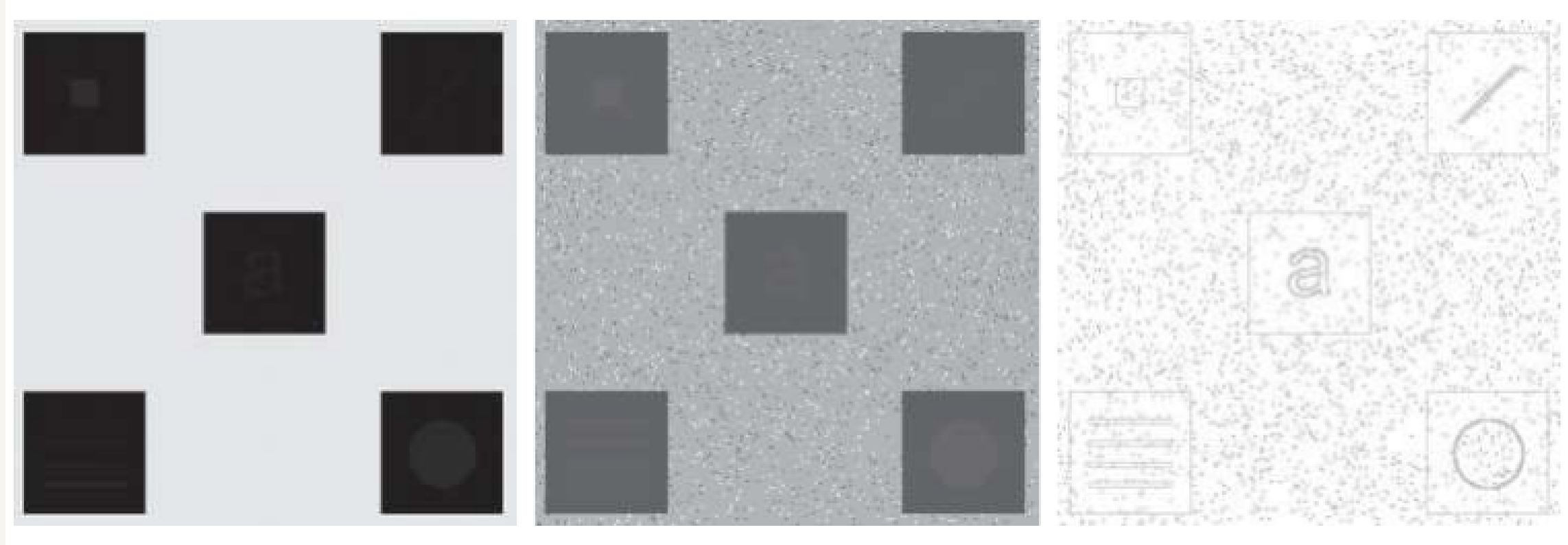


Şekil 3.24 · Eşitleme dönüşümü ve aşırı aydınlanmış sonuç

Yerel histogram eşitleme

Küresel eşitleme tüm görüntü için tek bir dönüşüm kullanır ve **küçük bölgelerdeki** ayrıntıyı kaçıır. Yerel yöntemde her piksel, kendi komşuluğunun histogramıyla hesaplanan dönüşümden geçer.

- Her piksel için $m \times n$ komşuluk
- Komşuluk kaydırıldıkça histogram güncellenir
- Küresel yöntemin kaçırdığı ayrıntıyı ortaya çıkarır

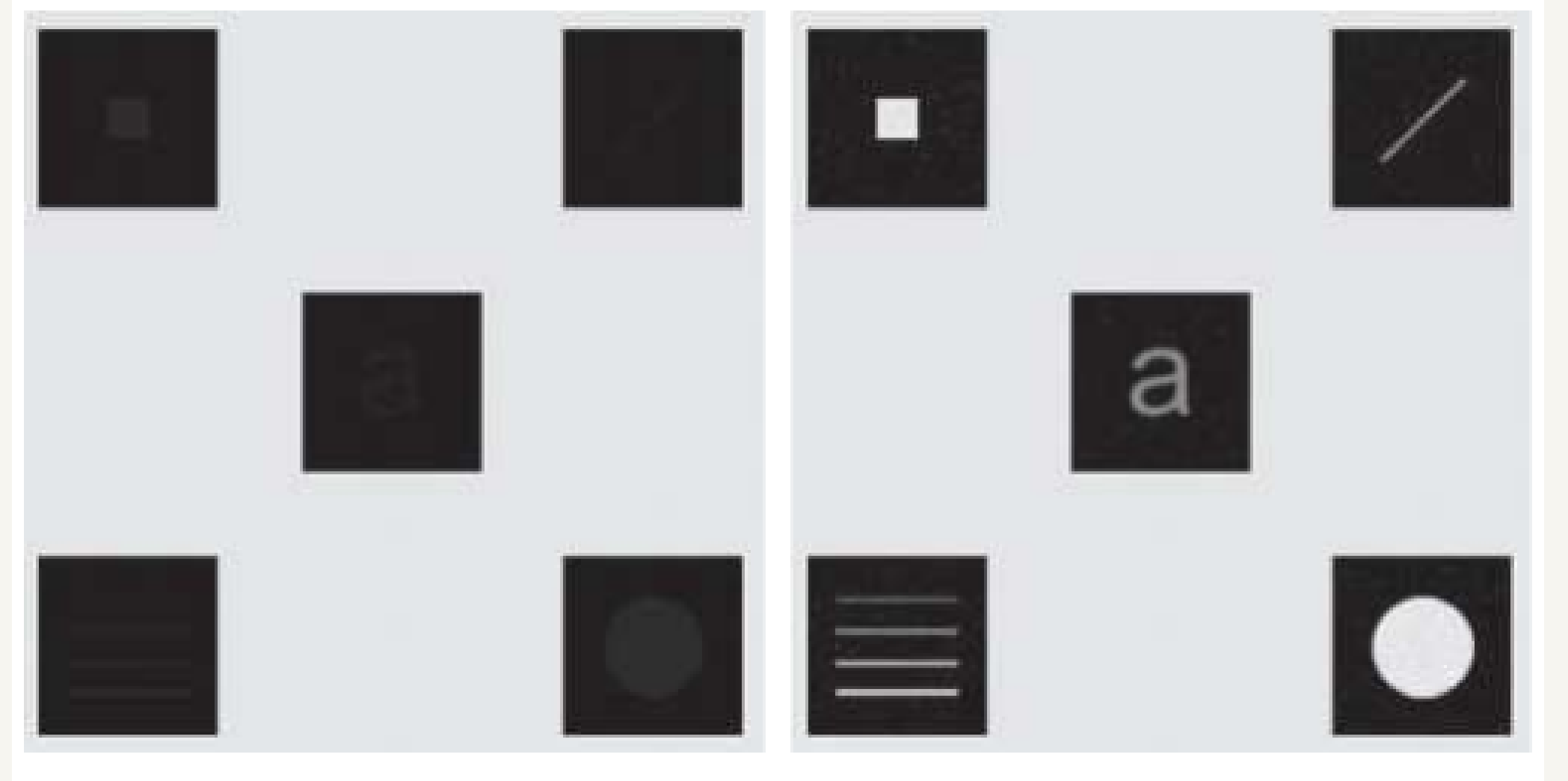


Şekil 3.26 · (a) Özgün (b) Küresel eşitleme (c) Yerel eşitleme

Histogram istatistikleriyle yerel iyileştirme

Histogramın tamamı yerine yalnızca **ortalama ve standart sapma** kullanılarak da yerel iyileştirme yapılabilir: koyu ve düşük kontrastlı komşuluklar seçilerek yükseltilir.

- Yerel ortalama → parlaklık ölçüsü
- Yerel standart sapma → kontrast ölçüsü
- Koşulu sağlayan pikseller bir E katsayısıyla yükseltilir



BÖLÜM 3.4

Uzamsal filtrelemenin temelleri

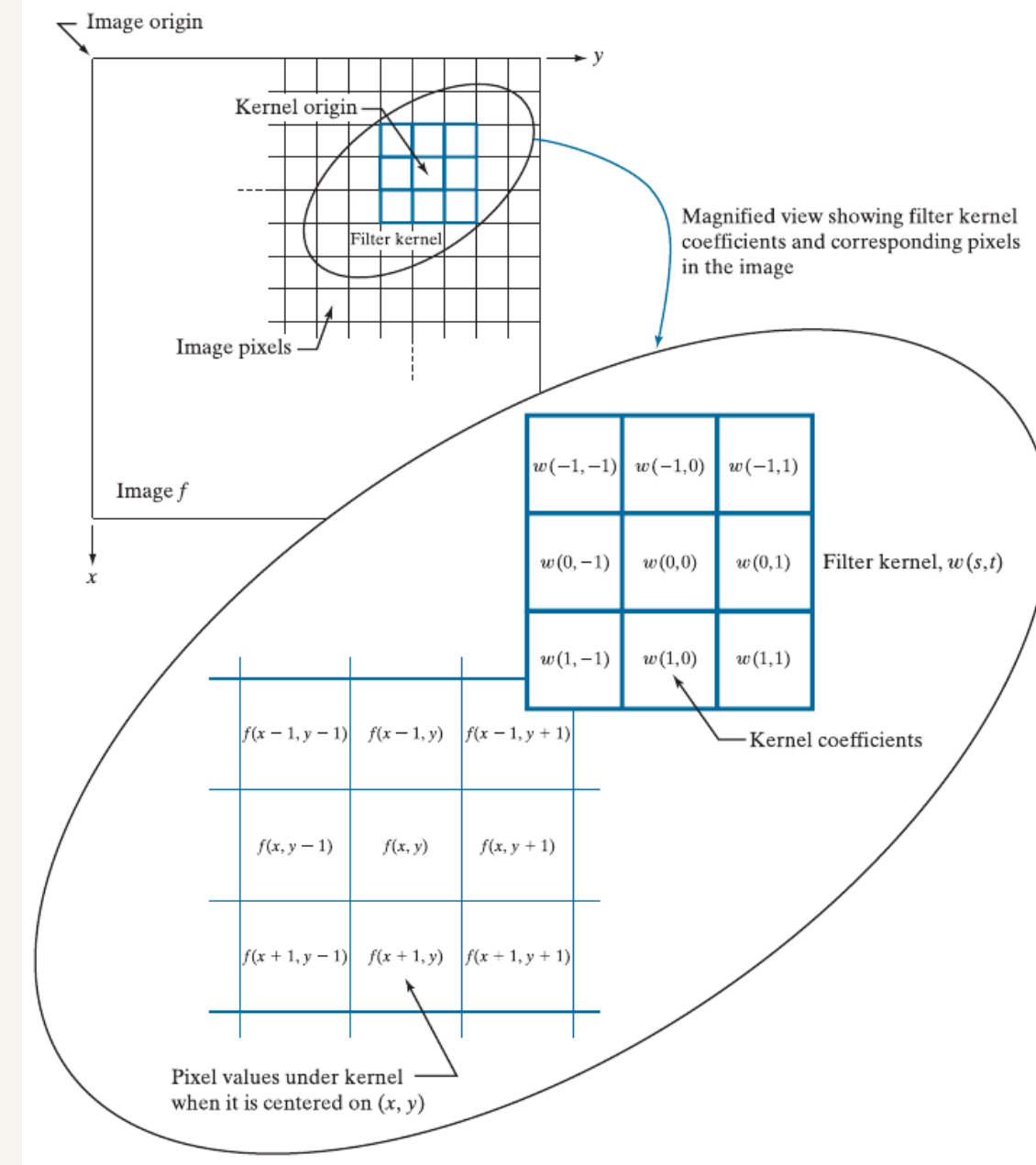
Komşuluk işlemleri: bir çekirdeğin görüntü üzerinde kaydırılması, korelasyon ile konvolüsyon arasındaki fark ve ayrılabilirlik.

Uzamsal filtrelemenin mekaniği

Bir **çekirdek** (kernel/maske) görüntü üzerinde piksel piksel kaydırılır. Her konumda çekirdek katsayıları ile altlarındaki piksel değerleri çarpılıp toplanır; sonuç, çekirdek merkezinin bulunduğu çıkış pikselidir.

$$g(x, y) = \sum_{s=-a}^a \sum_{t=-b}^b w(s, t) f(x + s, y + t)$$

$m \times n$ çekirdek, tek sayılı $\cdot a = (m-1)/2, b = (n-1)/2$



Korelasyon ve konvolüsyon — tek boyutta

İkisi de aynı kaydır–çarp–topla işlemidir; **tek fark**, konvolüsyonda çekirdeğin önce 180° döndürülmesidir. Birim dürtüyle korelasyon çekirdeğin *döndürülmüş* hâlini, konvolüsyon ise *aynısını* verir.

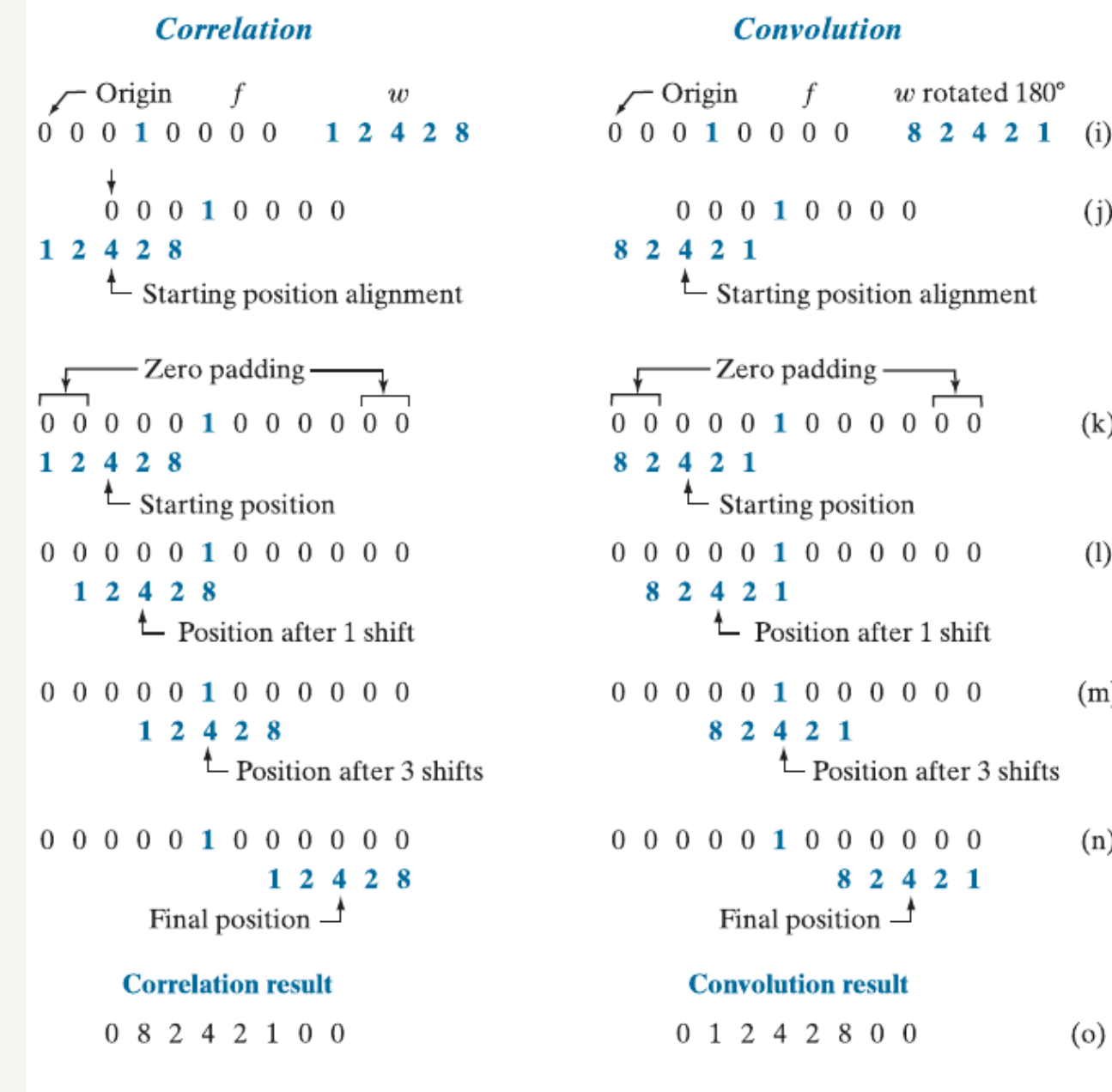
KORELASYON

$$(w \star f)(x) = \sum_s w(s) f(x + s)$$

KONVOLÜSYON

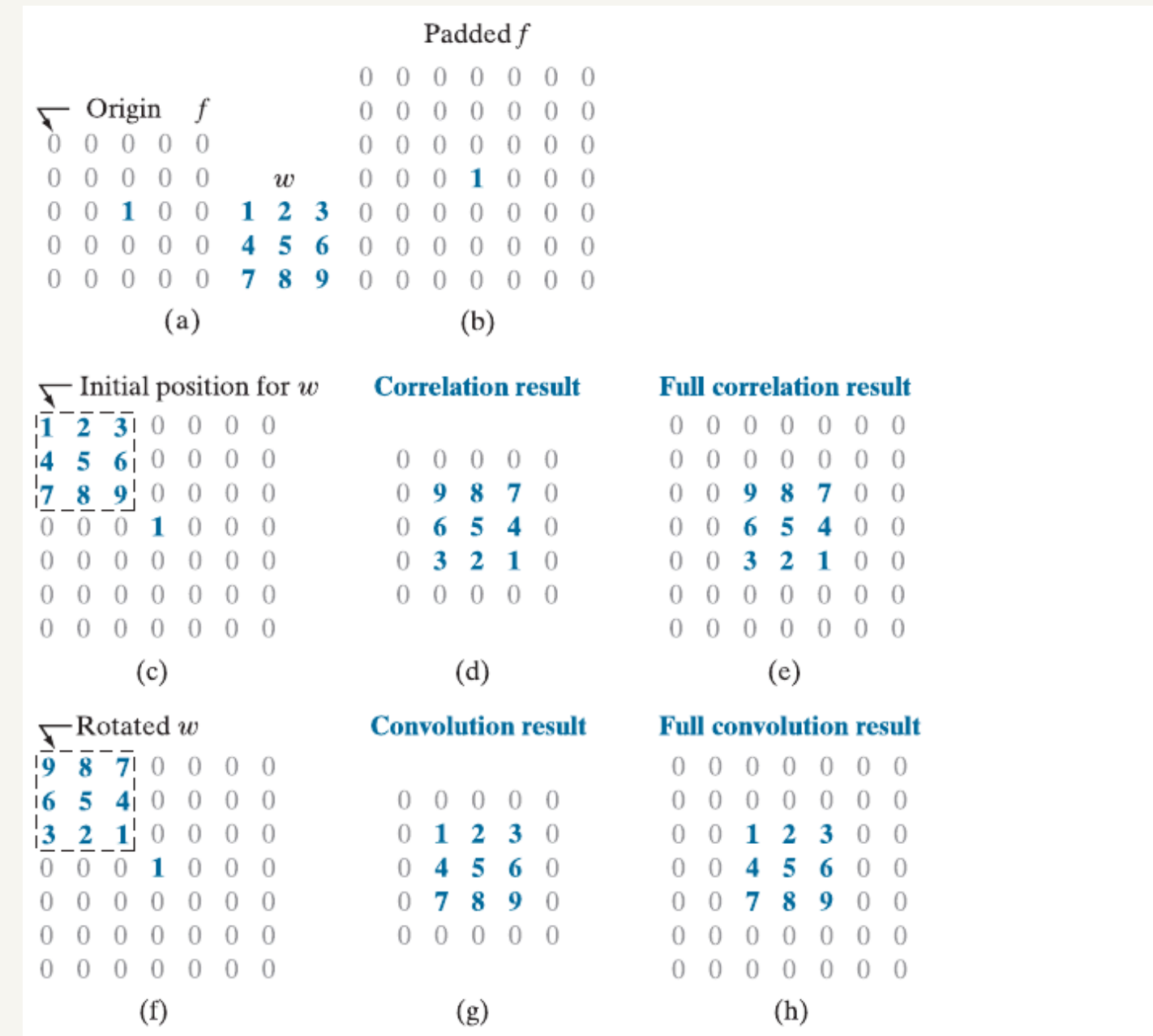
$$(w * f)(x) = \sum_s w(s) f(x - s)$$

Tek fark işaret: konvolüsyonda çekirdek 180° döndürülür.



İki boyutta korelasyon ve konvolüsyon

Aynı ayırım 2 boyutta da geçerlidir. Görüntü **sıfırlarla dolgulanır**, çekirdek her konuma yerleştirilir ve çarpımlar toplanır. Konvolüsyonda çekirdek her iki ekseninde de çevrilir.



Ayrılabilir çekirdekler ve hız

Bir çekirdek iki vektörün çarpımı olarak yazılabiliyorsa **ayrılabilir**dir. O zaman 2-B filtreleme, ardışık iki 1-B filtrelemeye indirgenir ve hesap yükü büyük ölçüde düşer.

$$w = w_1 w_2^T \quad \text{kazanc} = \frac{mn}{m+n}$$

Örnek: 41×41 çekirdekte ≈ 20 kat hızlanma

$$\frac{1}{9} \times \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array} \quad \frac{1}{4.8976} \times \begin{array}{|c|c|c|} \hline 0.3679 & 0.6065 & 0.3679 \\ \hline 0.6065 & 1.0000 & 0.6065 \\ \hline 0.3679 & 0.6065 & 0.3679 \\ \hline \end{array}$$

Şekil 3.31 · (a) Kutu çekirdeği (b) Gauss çekirdeği

BÖLÜM 3.5

Düzgünleştirme (alçak geçiren) filtreler

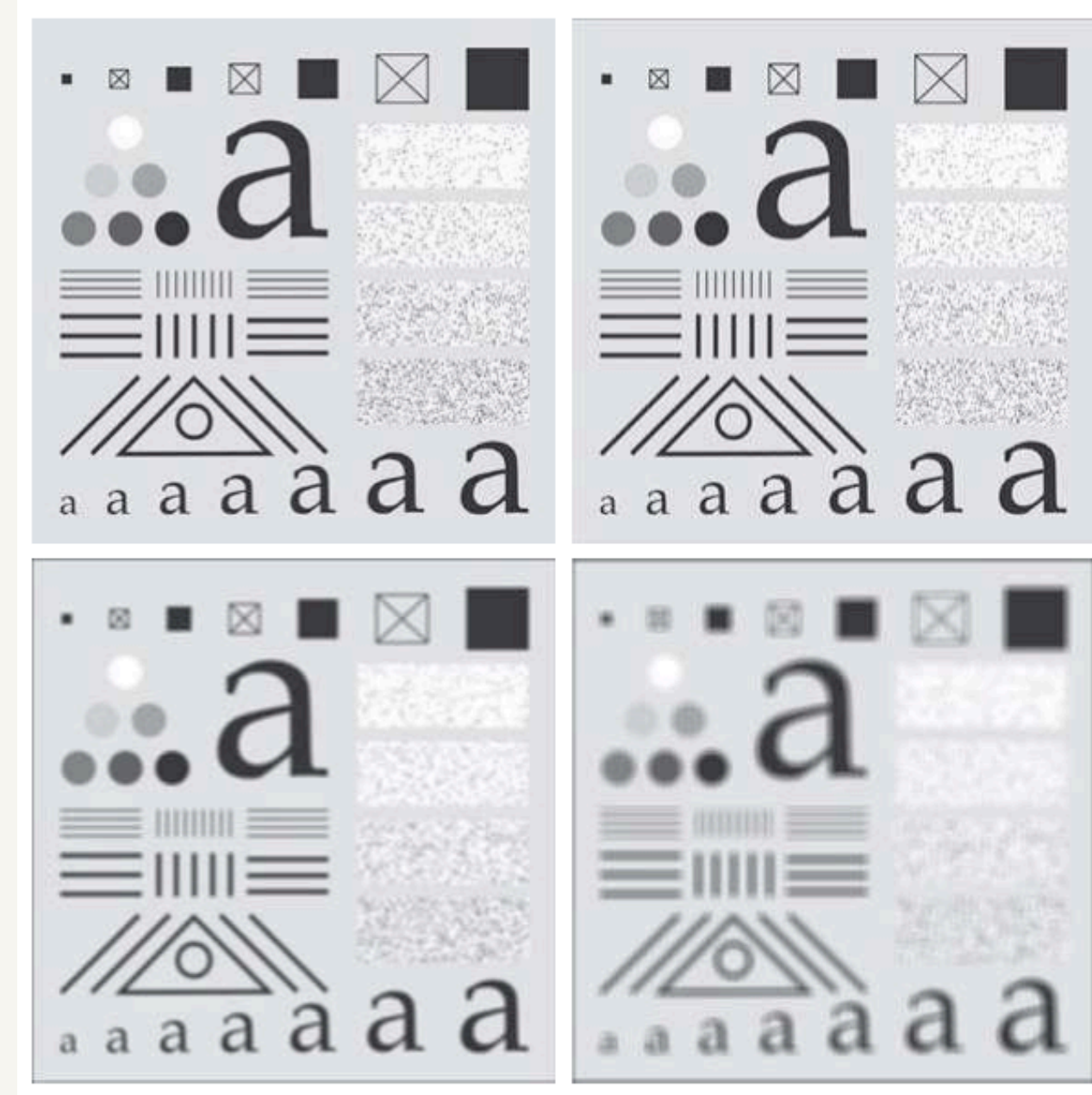
Gürültü azaltma ve bulanıklaştırma: kutu filtresi, Gauss filtresi, dolgulama seçenekleri ve doğrusal olmayan medyan filtresi.

Kutu filtresi (ortalama filtre)

En basit alçak geçiren filtre: komşuluktaki piksellerin **aritmetik ortalaması**. Uygulaması kolaydır, ancak yönlü yapay etkiler üretir ve kenarları fazla bozar.

$$w(s, t) = \frac{1}{mn}$$

Tüm katsayılar eşit ve toplamaları 1

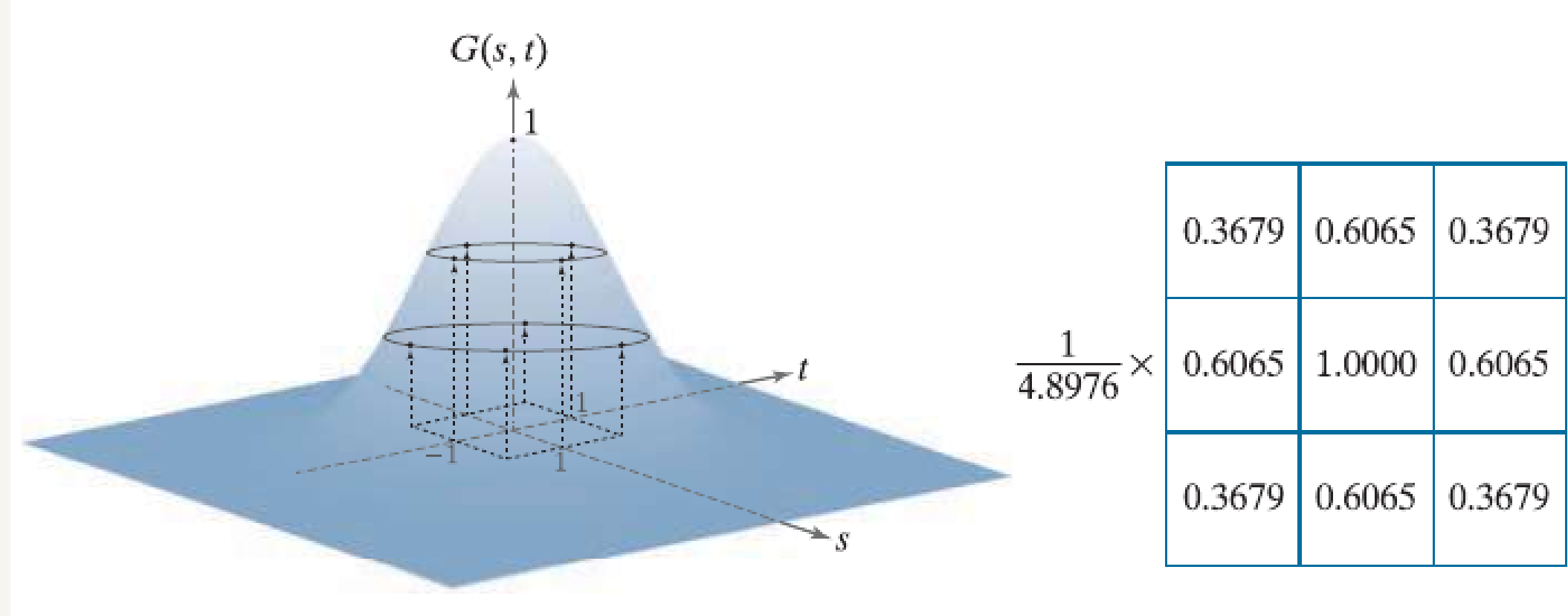


Gauss filtresi

Gauss çekirdeği **dairesel simetriktir** (izotropik) ve tek parametreyle (σ) kontrol edilir. Kutu filtresinden daha yumuşak geçişler üretir; ayrıca iki Gauss'un konvolüsyonu yine bir Gauss'tur.

$$G(s, t) = K e^{-\frac{s^2 + t^2}{2\sigma^2}}$$

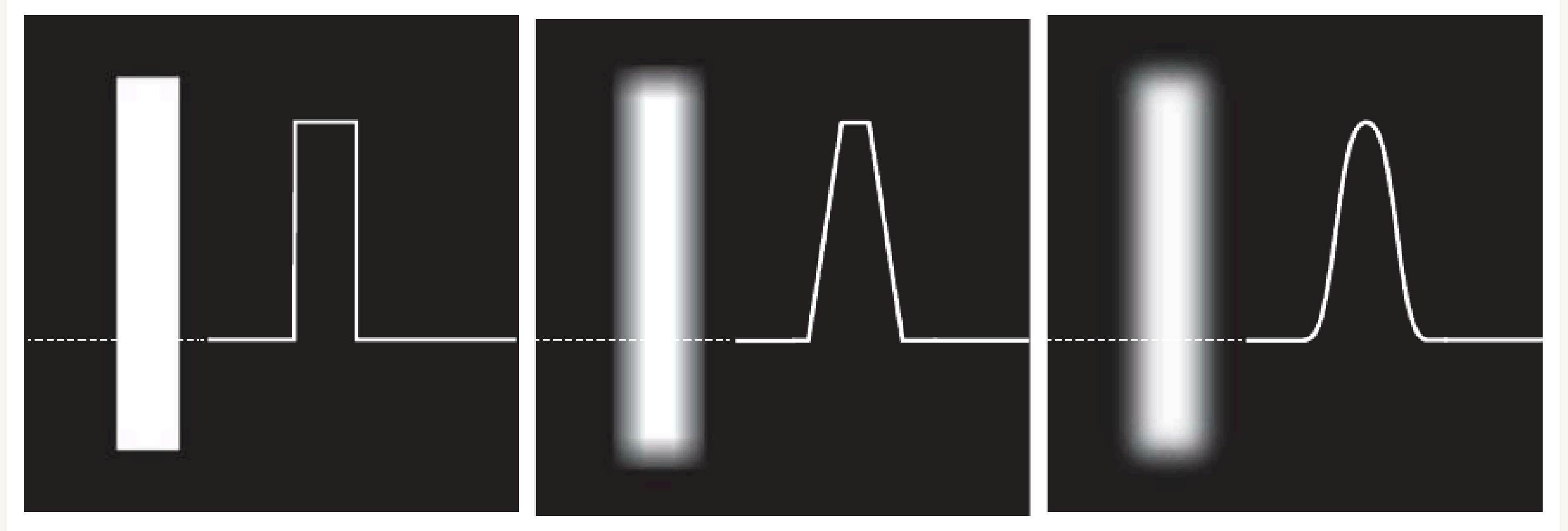
Çekirdek boyutu $\approx [6\sigma] \times [6\sigma]$ (tek sayıya yuvarlanmış)



Şekil 3.35 · Gauss fonksiyonu ve ondan örneklenen 3×3 çekirdek

Kutu ile Gauss'un karşılaştırması

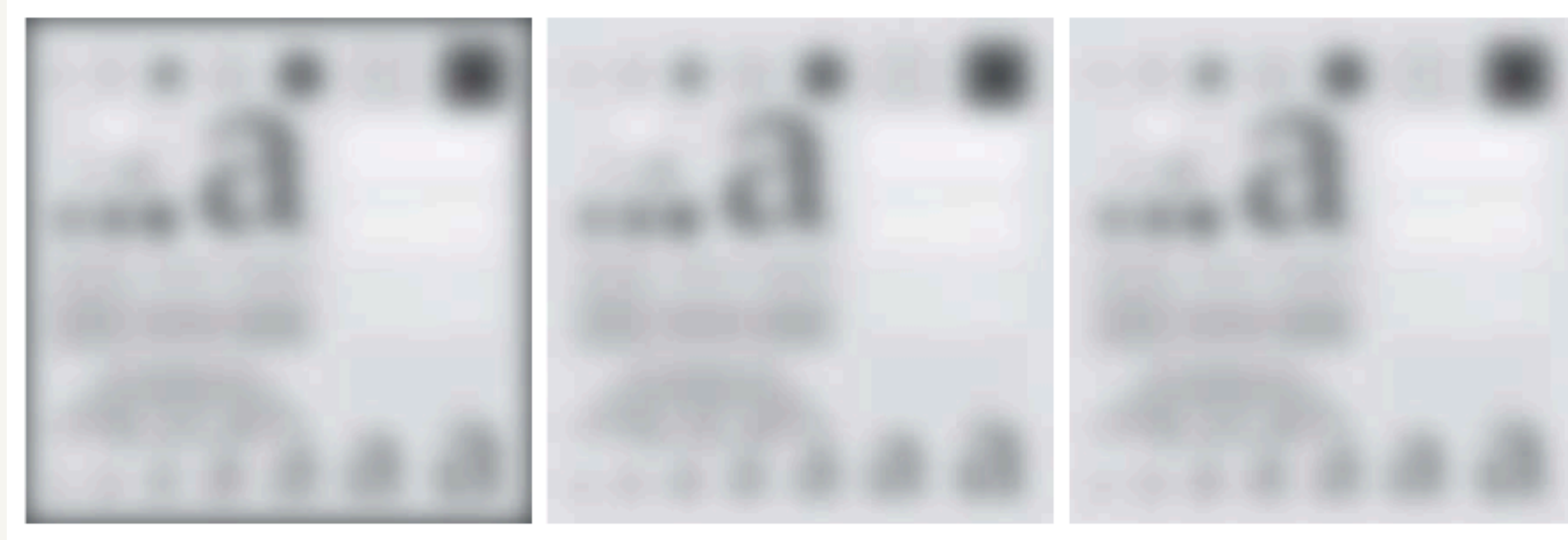
Beyaz bir dikdörtgenin kenarındaki yoğunluk profili. Kutu filtresi kenarı **doğrusal bir rampaya** çevirirken, Gauss filtresi **yumuşak, S biçimli** bir geçiş üretir — gözün doğal bulanıklık algısına daha yakındır.



Dolgulama (padding) seçimi

Çekirdek kenarlarda görüntünün dışına taşar. Seçilen dolgulama yöntemi **kenar bölgesinde belirgin farklar** yaratır: sıfır dolgulama koyu bir çerçeve bırakır, ayna ve tekrarlama ise doğal görünür.

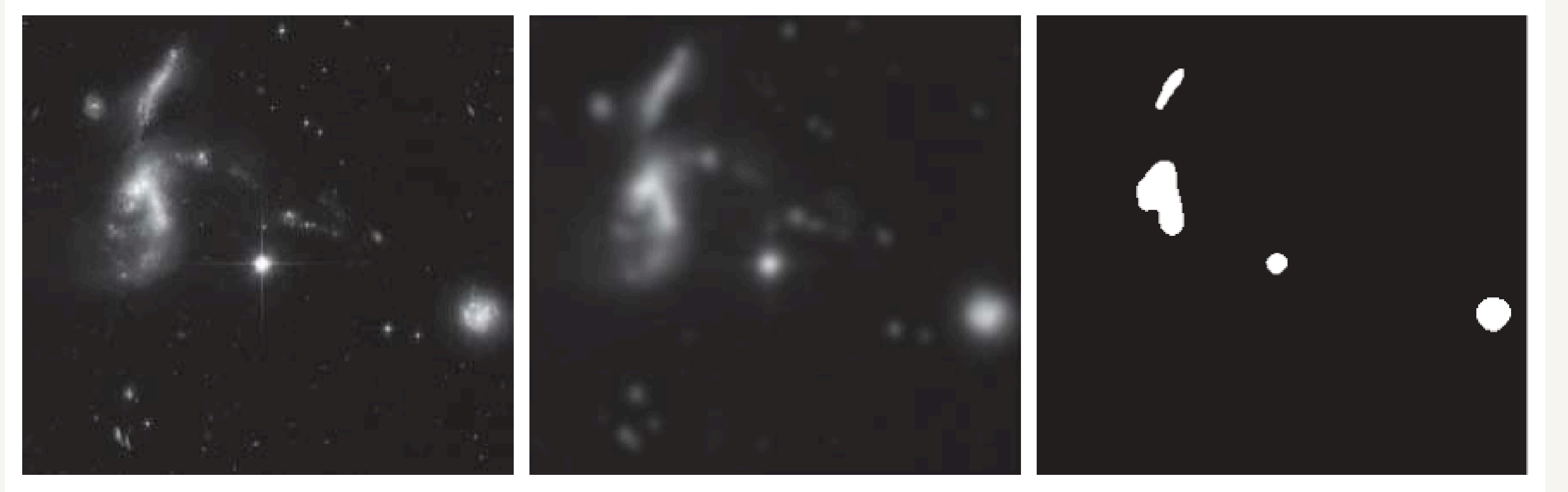
- **Sıfır:** kenarda koyu bant
- **Ayna (mirror):** kenar değerleri yansıtılır
- **Tekrarlama (replicate):** kenar pikseli kopyalanır



Şekil 3.39 · (a) Sıfır (b) Ayna (c) Tekrarlama dolgulaması

Uygulama: bulanıklaştırma + eşikleme ile bölge çıkarma

Hubble görüntüsünde amaç, **en büyük ve en parlak** nesneleri ayırmaktır. Geniş bir Gauss çekirdeğiyle küçük nesneler silinir, ardından eşikleme ile kalan bölgeler yalıtılır.



Şekil 3.41 · Özgün görüntü, 151×151 Gauss ile filtrelenmiş hâli ve eşiklenmiş sonuç

Uygulama: gölge (aydınlatma) düzeltmesi

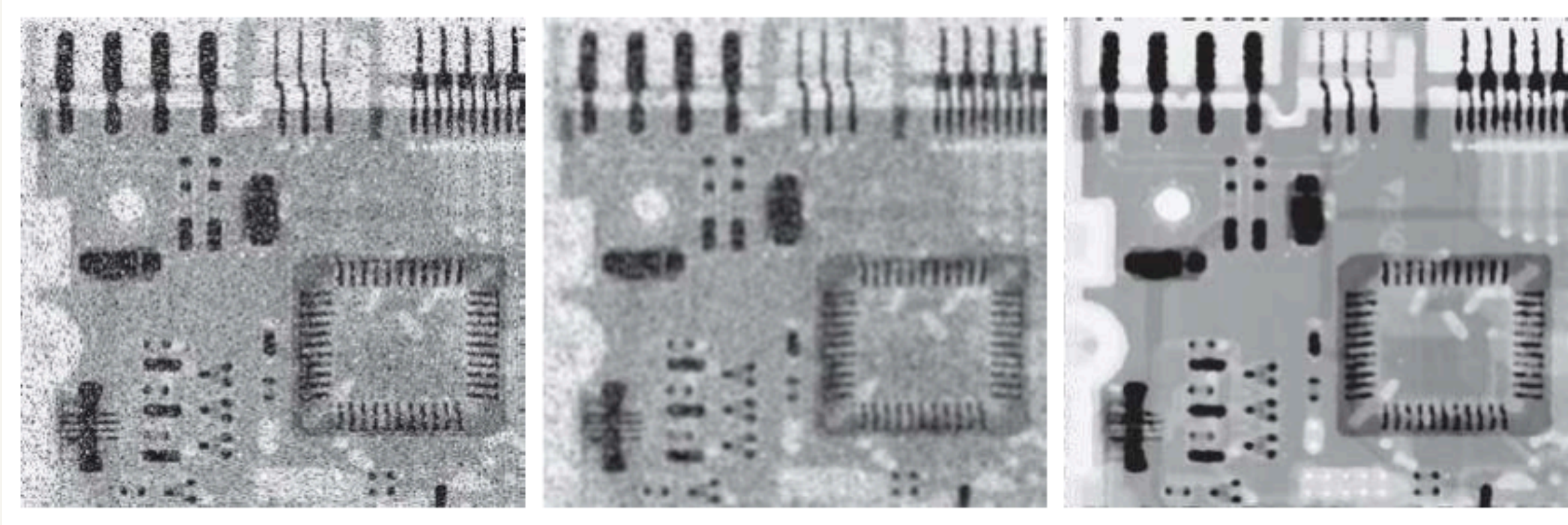
Düzgün olmayan aydınlatma, görüntü üzerine yavaş değişen bir **gölge deseni** bindirir. Çok geniş bir alçak geçiren filtre bu deseni tahmin eder; özgün görüntüyü bu tahmine bölmek gölgeyi kaldırır.



Doğrusal olmayan: medyan filtresi

Sıra istatistiği filtreleridir; komşuluktaki değerleri sıralayıp **ortancayı** seçerler. **Tuz-biber gürültüsüne** karşı olağanüstü etkilidir ve kenarları ortalama filtreden çok daha iyi korur.

- **Medyan:** uç değerli gürültüyü siler
- **Maks:** en parlak noktaları bulur
- **Min:** en karanlık noktaları bulur



Şekil 3.43 · Gürültülü röntgen, ortalama filtre ve medyan filtre sonucu

BÖLÜM 3.6

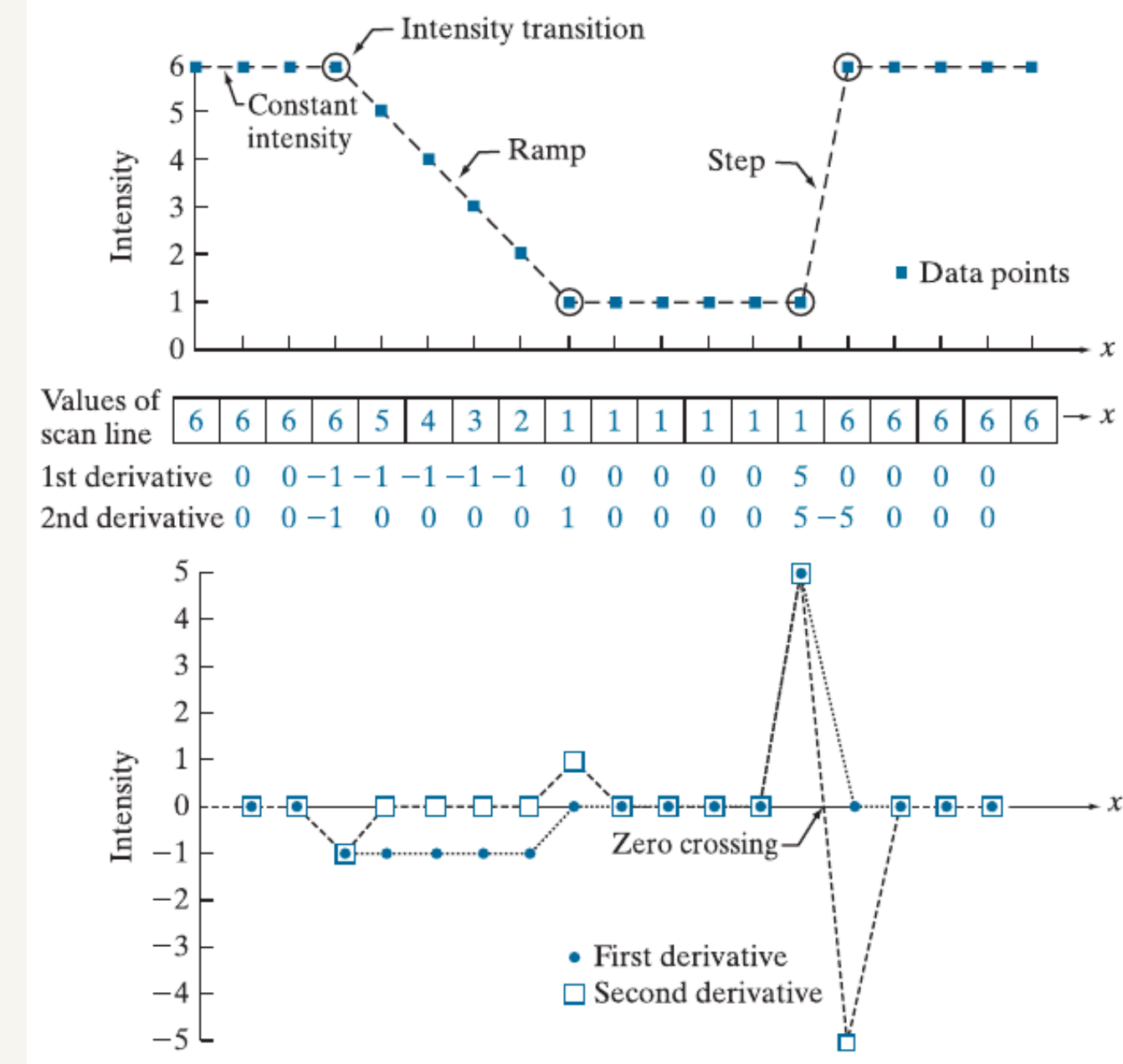
Keskinleştirme (yüksek geçiren) filtreler

Türev temelli yöntemler: Laplas operatörü, keskin olmayan maskeleme / yüksek destekli filtreleme ve gradyan (Sobel).

Türevler ne yapar?

Düzgünleştirme bir **integralleme**,
keskinleştirme ise **türev alma** işlemidir. Birinci
türev rampalar boyunca sıfırdan farklıdır; ikinci
türev ise basamaklarda çift tepe ve bir **sıfır**
geçişi üretir.

- **1. türev:** kalın kenarlar, rampalara güçlü tepki
- **2. türev:** ince ayrıntıya güçlü tepki, sıfır geçişi
- Sabit bölgelerde her iki türev de sıfırdır



Şekil 3.44 · Bir tarama satırı, değerleri ve birinci/ikinci türevleri

Laplas operatörü

En basit **izotropik** ikinci türev operatörüdür — sonucu görüntünün dönme açısından bağımsızdır. Çekirdek katsayılarının toplamı sıfırdır; bu yüzden tek başına uygulandığında çoğunlukla koyu bir görüntü verir.

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

$$g(x, y) = f(x, y) + c \nabla^2 f(x, y)$$

$c = -1$ (merkez katsayı negatifse) · $c = +1$ (pozitifse)

0	1	0	1	1	1	0	-1	0	-1	-1	-1
1	-4	1	1	-8	1	-1	4	-1	-1	8	-1
0	1	0	1	1	1	0	-1	0	-1	-1	-1

Laplas ile keskinleştirme: Ay örneđi

Hafifçe bulanık Ay görüntüsü. Laplas sonucu tek başına çođunlukla siyaha yakındır; **özgün görüntüye eklendiğinde** arka plan tonlaması korunurken ayrıntı belirginleşir.



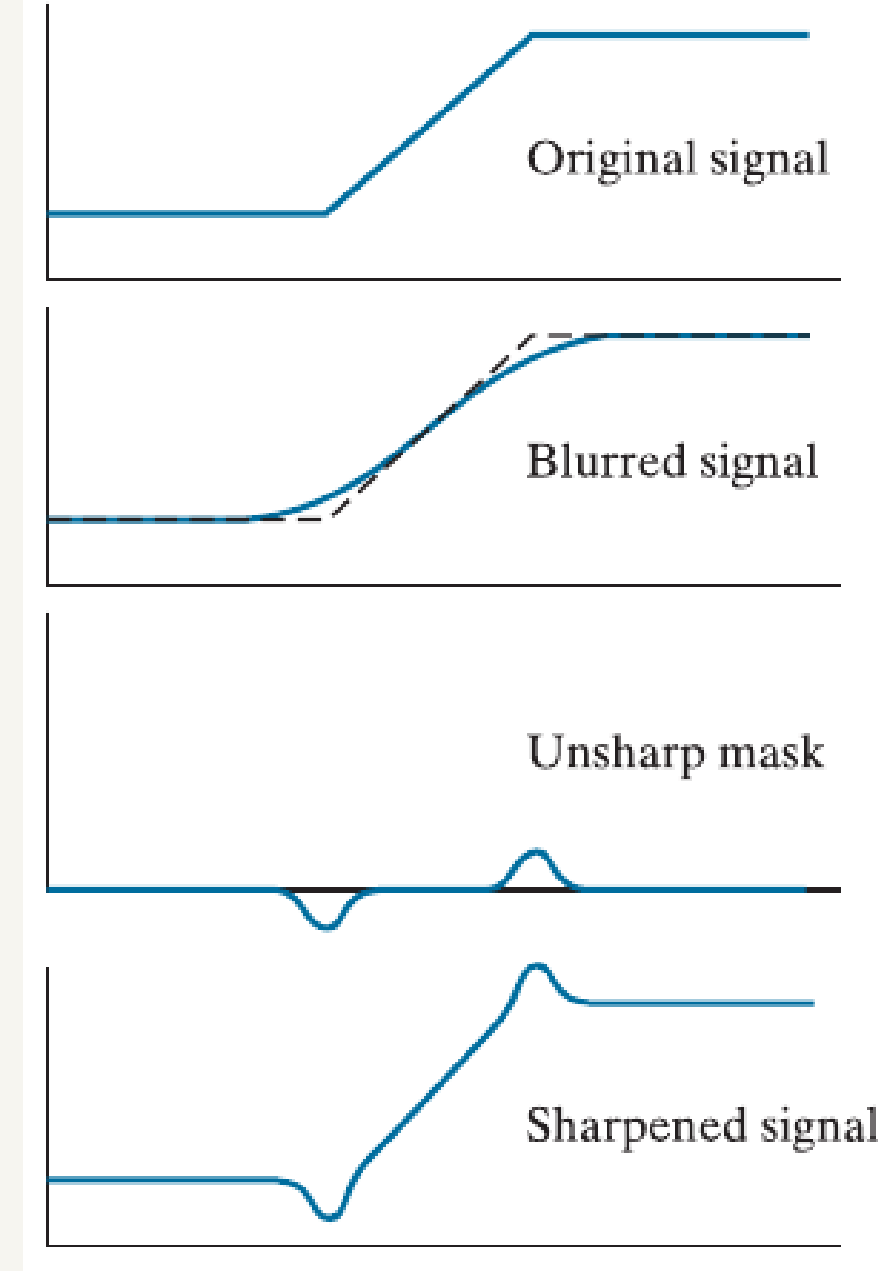
Keskin olmayan maskeleme ve yüksek destekli filtreleme

Onlarca yıldır matbaacılıkta kullanılan klasik yöntem: görüntünün **bulanık bir kopyasını çıkar**, kalan maskeyi bir k katsayısıyla geri ekle. $k > 1$ olduğunda **yüksek destekli (highboost)** filtreleme adını alır.

$$g_{mask}(x, y) = f(x, y) - \bar{f}(x, y)$$

$$g(x, y) = f(x, y) + k g_{mask}(x, y)$$

f : bulanık kopya · $k = 1$ klasik · $k > 1$ yüksek destekli (highboost)



Keskin olmayan maskeleme: k 'nın etkisi

Aynı görüntü artan k değerleriyle işlendiğinde metin giderek keskinleşir; ancak **k çok büyükse** kenarların çevresinde belirgin parlak ve koyu halkalar (halo) oluşur.

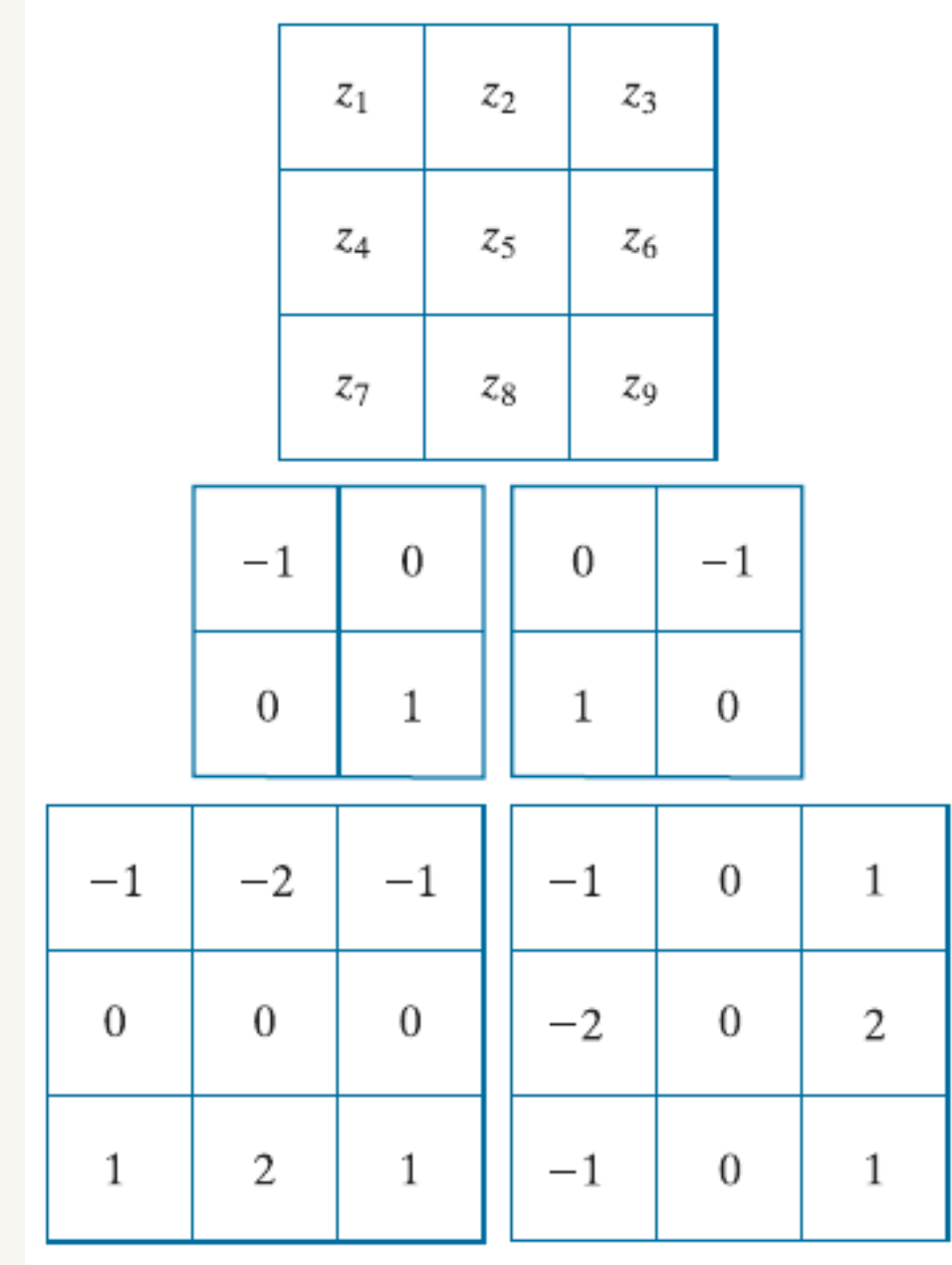


Gradyan ve Sobel operatörleri

Birinci türev, gradyan vektörünün **büyüklüğü** ile hesaplanır. Sobel çekirdekleri, merkez satır/sütuna 2 ağırlığı vererek düzgünleştirmeyi de içerir; bu yüzden gürültüye Roberts'tan daha dayanıklıdır.

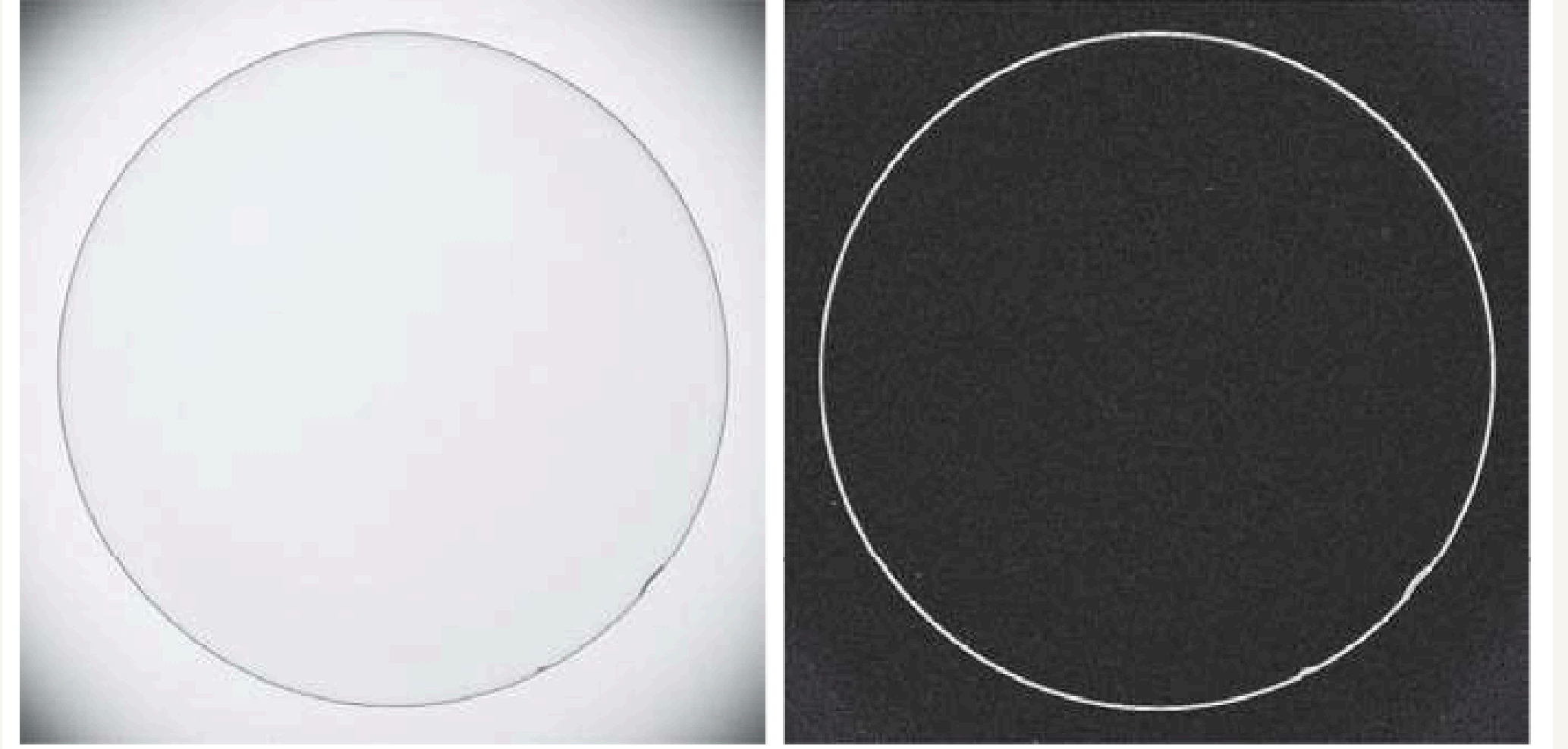
$$M(x, y) = \sqrt{g_x^2 + g_y^2} \approx |g_x| + |g_y|$$

g_x , g_y : yatay ve dikey Sobel yanıtları



Uygulama: kontakt lens kusur denetimi

Kontakt lensin kenarındaki kusurların **saat 4 ve 5 yönünde** olduğu görülüyor. Sobel gradyanı sabit bölgeleri bastırıp yalnızca kenarları bıraktığı için otomatik denetim için idealdir.



BÖLÜM 3.7

Filtre türleri arasındaki ilişki

Yüksek geçiren, bant durduran ve bant geçiren filtrelerin hepsi alçak geçiren filtrelerden türetilir.

Dört filtre türü, tek bir yapı taşı

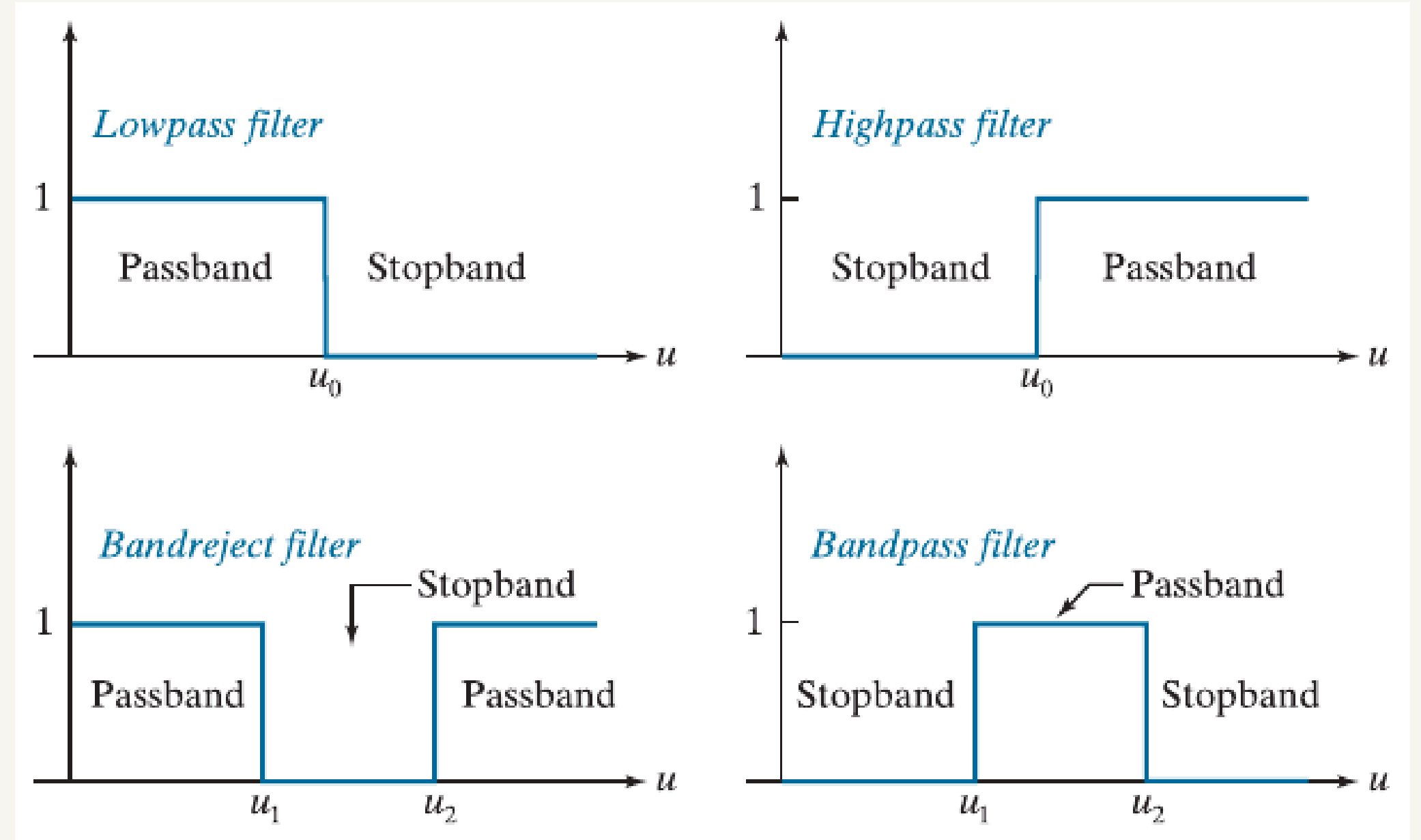
Alçak geçiren bir transfer fonksiyonundan başlayarak **diğer üçü doğrudan elde edilir**. Uzamsal alanda bu, birim dürtüden alçak geçiren çekirdeği çıkarmaya karşılık gelir.

$$H_{YG} = 1 - H_{AG}$$

$$H_{BD} = H_{AG} + H_{YG}$$

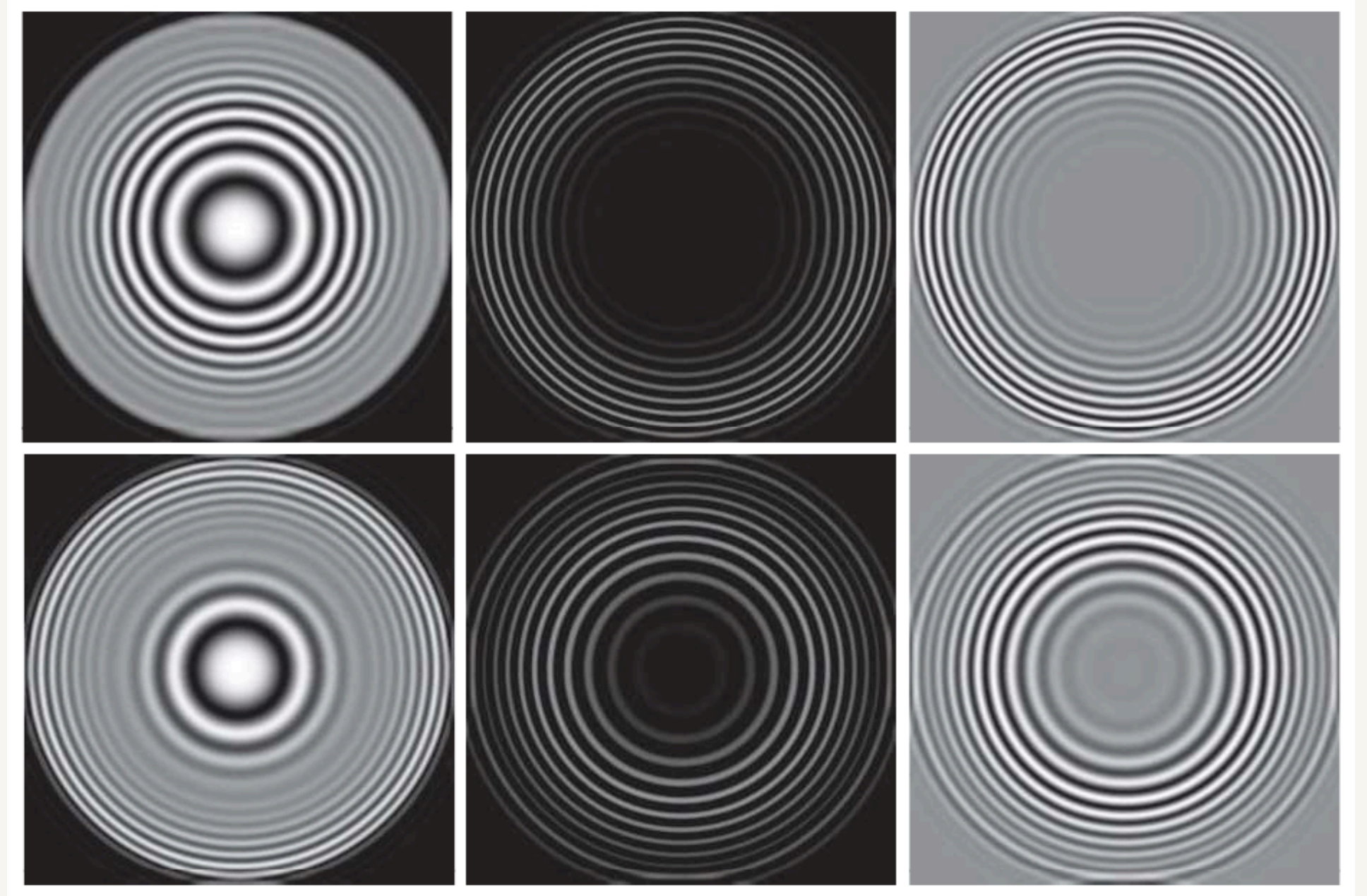
$$H_{BG} = 1 - H_{BD}$$

AG: alçak geçiren · YG: yüksek geçiren · BD: bant durduran · BG: bant geçiren



Dört filtrenin bir test deseni üzerindeki etkisi

Zone plate deseni, merkezden dışa doğru artan frekans içerdiği için filtre karakteristiklerini görsel olarak okumaya izin verir: hangi filtrenin hangi frekansları geçirdiği doğrudan görülür.



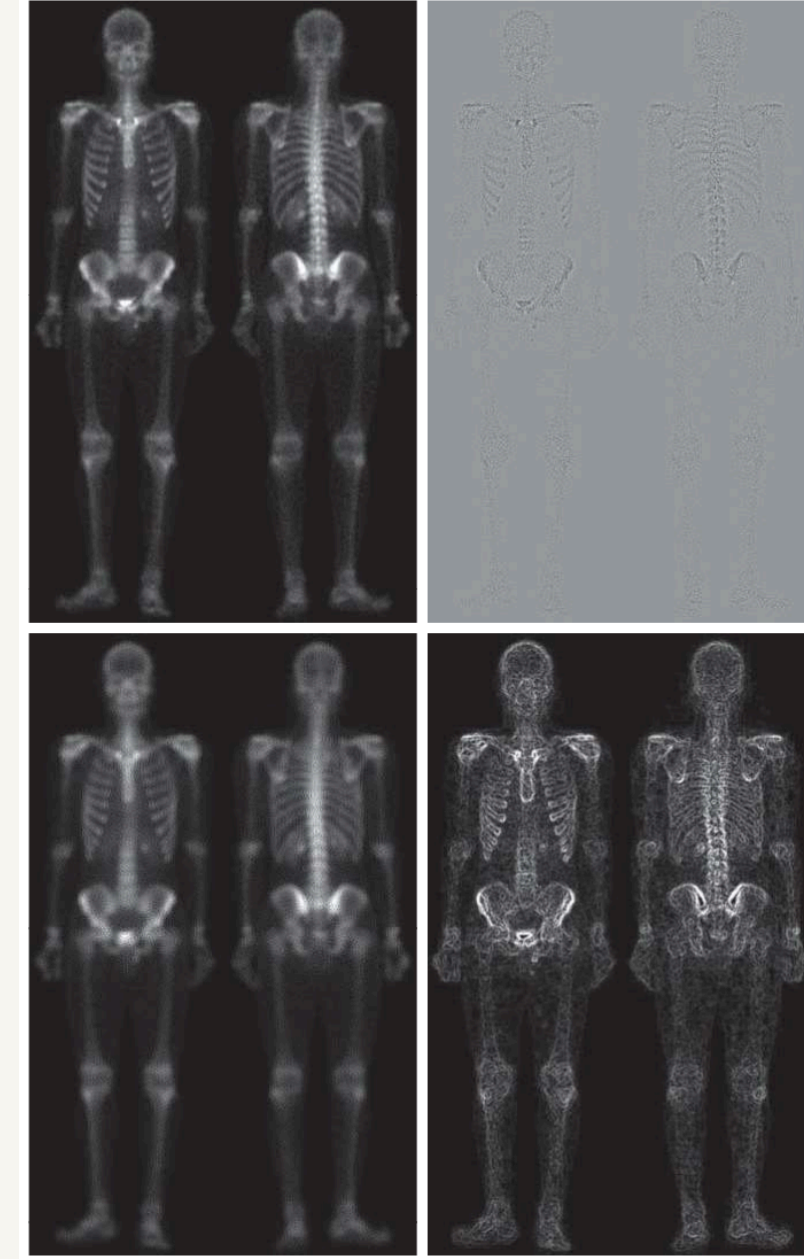
BÖLÜM 3.8

Yöntemlerin birleştirilmesi

Gerçek bir problem genellikle tek bir yöntemle çözülmez: Laplas, Sobel, düzgünleştirme ve gama dönüşümü birlikte kullanılır.

Tüm kemik taraması: zorlu bir örnek

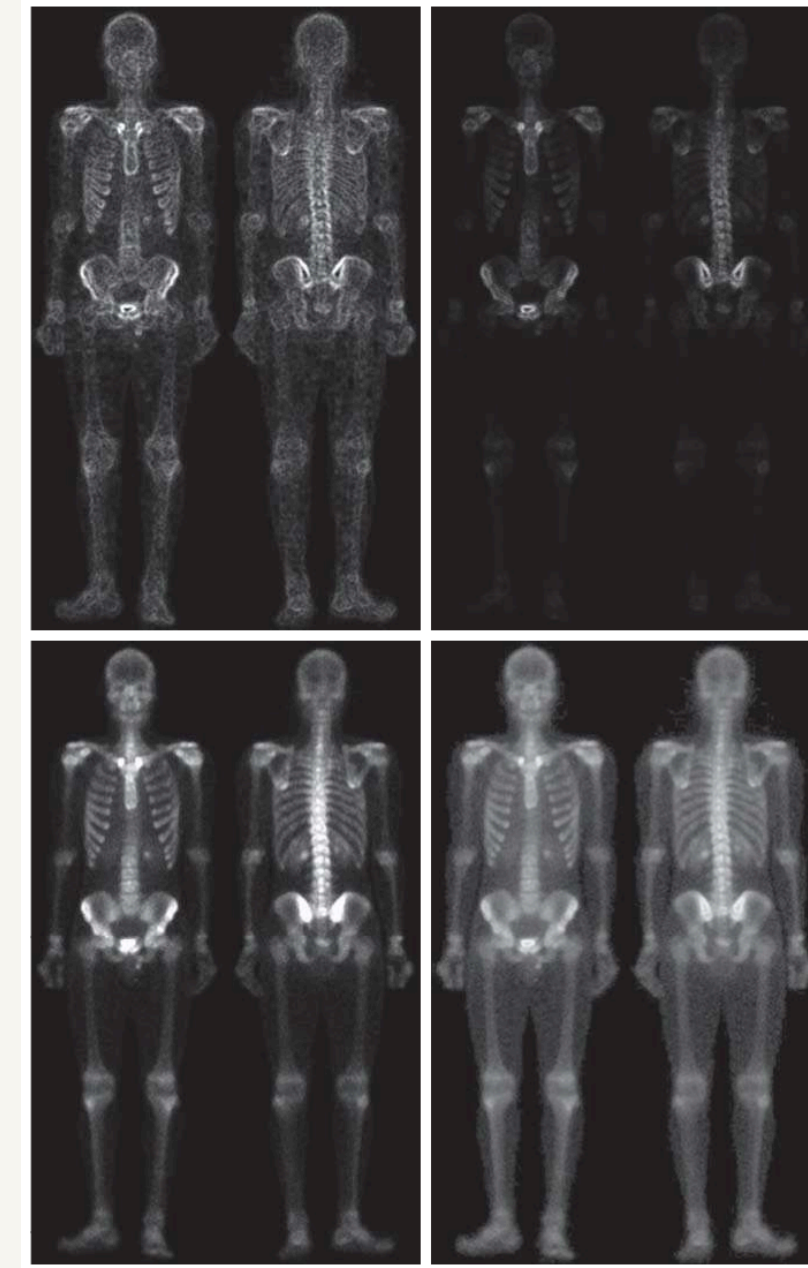
Nükleer tıp görüntüsünün **dinamik aralığı dar** ve **gürültü düzeyi yüksektir**. Tek bir yöntem yeterli değildir: Laplas ince ayrıntıyı, Sobel belirgin kenarları vurgular ve ikisi birlikte kullanılır.



Şekil 3.57 · (a) Özgün görüntü (b) Laplas (c) Keskinleştirilmiş (d) Sobel gradyanı

İyileştirme zincirinin sonucu

Düzgünleştirilmiş gradyan, Laplas görüntüsünü **maskeler**; toplam özgün görüntüye eklenir ve son adımda **gama dönüşümü** ile dinamik aralık genişletilir. Sonuç: iskelet yapısı belirgin, gürültü kontrol altında.



Özet

NOKTA İŞLEMLERİ

Negatif · Log · Gama ·
Parçalı doğrusal · Bit
düzlemi

Tek piksel, komşuluk yok

HİSTOGRAM YÖNTEMLERİ

Eşitleme · Eşleme · Yerel
eşitleme · İstatistiksel

Dağılımı doğrudan hedefler

ALÇAK GEÇİREN

Kutu · Gauss · Medyan
Gürültü azaltma, ön işleme

YÜKSEK GEÇİREN

Laplas · Keskin olmayan
maskeleme · Sobel

Ayrıntı ve kenar vurgulama

Ana fikir: Uzamsal alanda iyileştirme, ya tek bir pikseli bir eğriden geçirmek ya da bir komşuluğu bir çekirdekle özetlemektir. Hangi yöntemin işe yarayacağını görüntünün kendisi — çoğu zaman histogramı — söyler.

Bu yöntemlerin frekans alanındaki karşılıkları Bölüm 4’te ele alınacaktır.