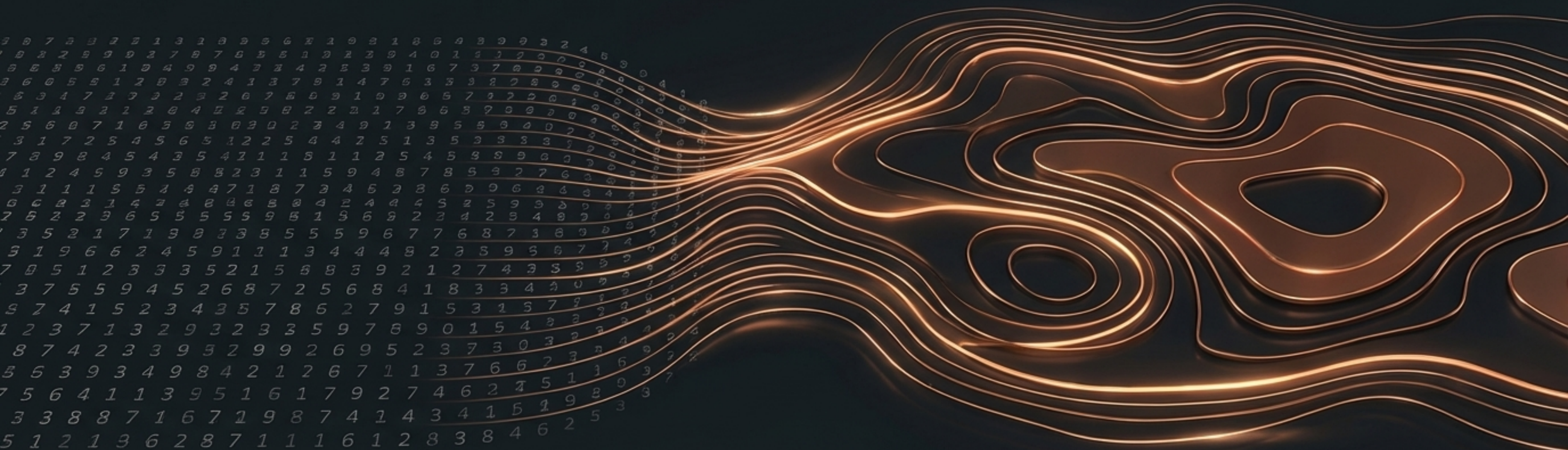


Görüntü Bölümlemenin Matematiği: Piksellerden Anlama Giden Yol



Bilgisayar Gerçekte Ne Görür?

4	3	2	3	4	4	4	4
5	10	23	29	32	27	13	4
6	22	31	32	32	33	28	4
6	31	29	3	1	31	35	4
4	31	30	4	1	32	32	4
4	24	31	29	30	34	25	3
3	13	27	35	35	28	14	3
1	3	4	4	4	3	1	0

$$\int \frac{\partial}{\partial x}, \frac{dy}{dx}$$

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

Bizim için nesne, arka plan ve derinlik; bilgisayar için yalnızca devasa bir sayı matrisi.

Pikselleri birer nesneye dönüştüren köprü, algoritmaların ardındaki denklemlerdir.

Görüntü bölümleme (Image Segmentation), bu sayı yığını global kararlar, yerel uçurumlar ve büyüyen bölgeler aracılığıyla anlamlandırır.

Işıđı ve Karanlığı Dengelemek: Yinelemeli Eşikleme


$$\lambda_{k+1} = 0.5(m_1 + m_2)$$

Görüntüyü **ön plan** ve **arka plan** olarak ayırmanın en temel yolu.

m_1 ve m_2 : O anki eşik değeriyle ikiye ayrılan bölgelerin (siyahlar ve beyazlar) ortalama gri seviyeleri. Kefelerdeki ağırlıklar.

λ_{k+1} : Sistemin bu iki ortalamanın tam merkezini alarak güncellediđi **yeni denge noktası**.

Terazi mükemmel dengeye ulaşana kadar **iterasyon** devam eder.

Pikseller Arası Uçurumlar: Gradyan Operatörleri

$$|\nabla f| = \sqrt{f_x^2 + f_y^2}$$

$$\theta = \tan^{-1}(f_y/f_x)$$

Matematikte türev, değişimi ölçer. Bir görüntüde aydınlık aniden değişiyorsa, orada fiziksel bir kenar vardır.

$|\nabla f|$ (**Büyükük**): Işık değişiminin şiddeti. Uçurumun ne kadar keskin ve derin olduğunu söyler.

θ (**Yön**): Dijital pusulamız. Kenarın hangi açıya baktığını, yani uçurumun yönünü hesaplar.



Matematiksel Pusulalar: Roberts, Prewitt ve Sobel

Roberts (2x2): Çapraz komşulara bakan en hızlı, en basit pusula.

$$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$



Prewitt (3x3): 3x3 ızgarada ortalama farkı alarak daha geniş bir ufka bakar.

$$\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$



Sobel (3x3): Merkeze ağırlık (x2 çarpanı) verir. Gürültüyü ezer ve en net gradyanı ortaya çıkarır.

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$



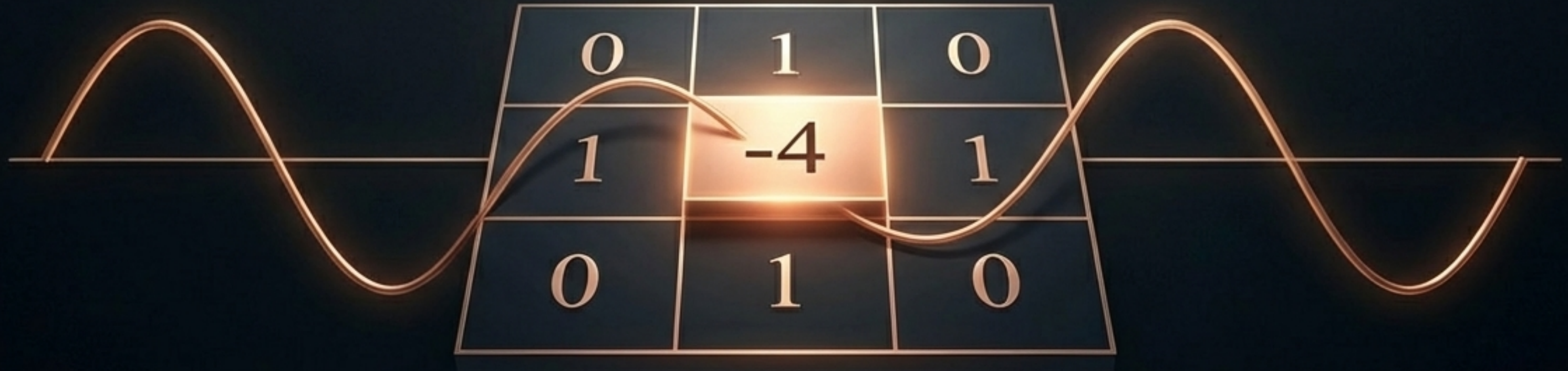
Dijital dünyada türev, komşu piksellerin birbirine olan matematiksel feryadıdır.

Zirveleri ve Çukurları Bulmak: İkinci Türev ve Laplacian

$$\nabla^2 f = f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1) - 4f(x, y)$$

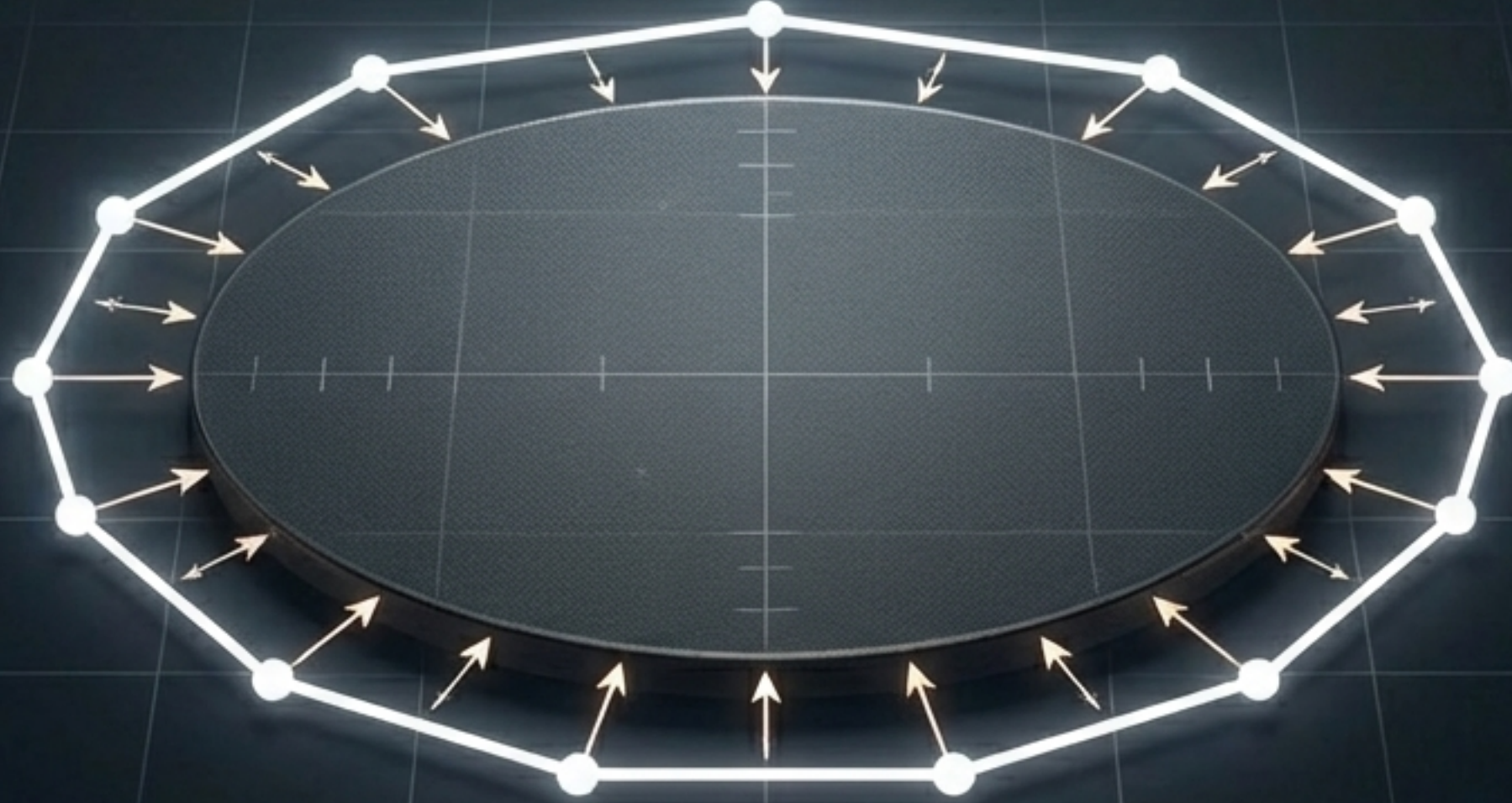
Eğimin hızını ölçer. Türevin zirve yaptığı noktada sıfır geçişi (zero-crossing) yaparak kenarı milimetrik hassasiyetle bulur.

Maskenin Merkezindeki -4: Bir pikselin, dört ana yönündeki (sağ, sol, alt, üst) komşularıyla olan maksimum farkının matematiksel kanıtıdır. İzotropiktir, yönden bağımsız tepki verir.



Fiziksel Bir Metafor Olarak Yılan (Snake) Modeli

$$E_{snake}(v) = \alpha E_{cont}(v) + \beta E_{curv}(v) + \gamma E_{image}(v)$$



Sınırları sadece piksellerle değil, fiziksel bir enerji denklemiyle tanımlarız.

Amaç: Toplam enerjiyi (E_{snake}) minimize edecek en rahat, en uyumlu pozisyonu bulmak.

E_{image} (**Dış Kuvvet**): Görüntüdeki kenarların, yani uçurumların bizim lastik bandımızı kendine doğru çeken manyetik kuvvetidir.

İç Enerji: Süreklilik ve Eğrilik Sınırları

Yılanın (lastik bandın) kendi formunu korumasını sağlayan iskelet yapısı.

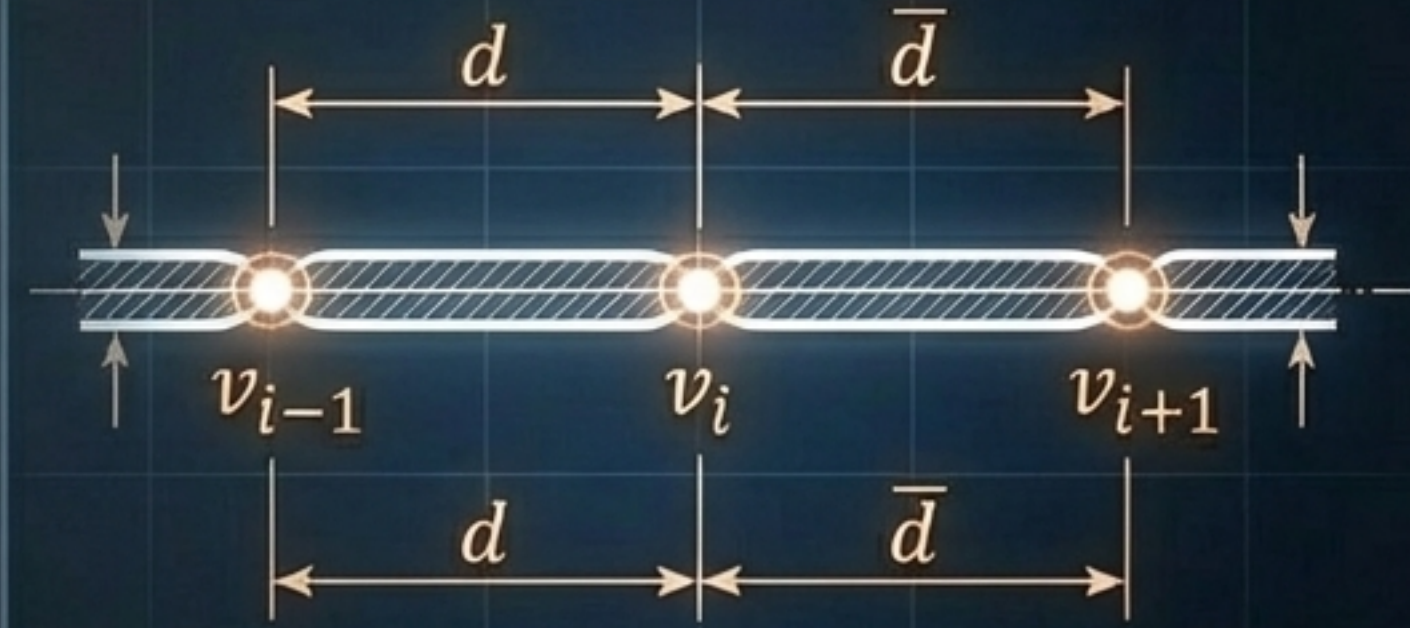
E_{cont} (Süreklilik Enerjisi): Noktalar arası mesafeyi eşit tutmaya çalışır. Bandın kopmasını veya orantısız esnemesini engeller.

E_{curv} (Eğrilik Enerjisi): İkinci türevi kullanarak konturun çok keskin köşeler (kırılmalar) yapmasını engeller, yumuşak ve doğal akmasını garantiler.

$$E_{cont} = \bar{d} - |v_i - v_{i-1}|$$

$$E_{curv} = |v_{i-1} - 2v_i + v_{i+1}|$$

Süreklilik



Eğrilik



İçeriden Dışarıya: Tohum Tabanlı Bölge Büyütme (SRG)

Sınırları dışarıdan sarmak her zaman işe yaramaz. Bazen nesneyi içeriden fethetmek gerekir.

Konsept: Homojen ve güvenilir bir merkez noktasına (Tohum) bir kristal atılır.

Tohum, çevresindeki pikselleri tek tek inceler. Kendisine benzeyenleri yutarak büyür, ta ki nesnenin doğal sınırlarına çarpana dek.



Büyümenin Kalbi: Standart Sapma ve Renk Mesafesi

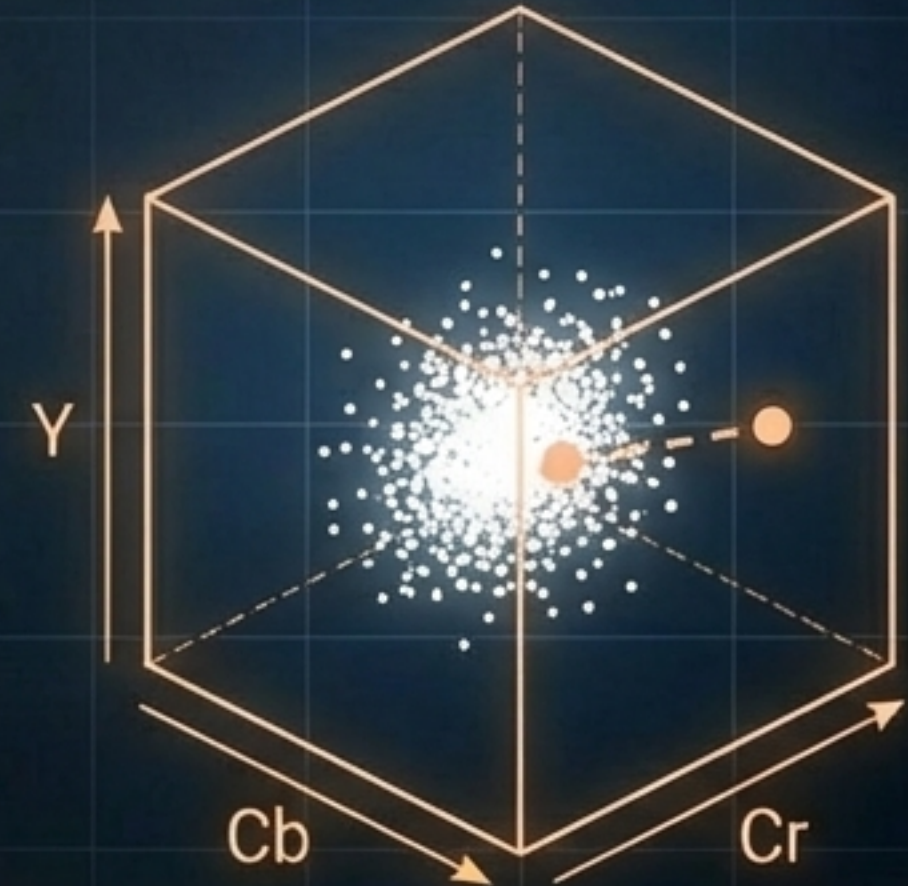
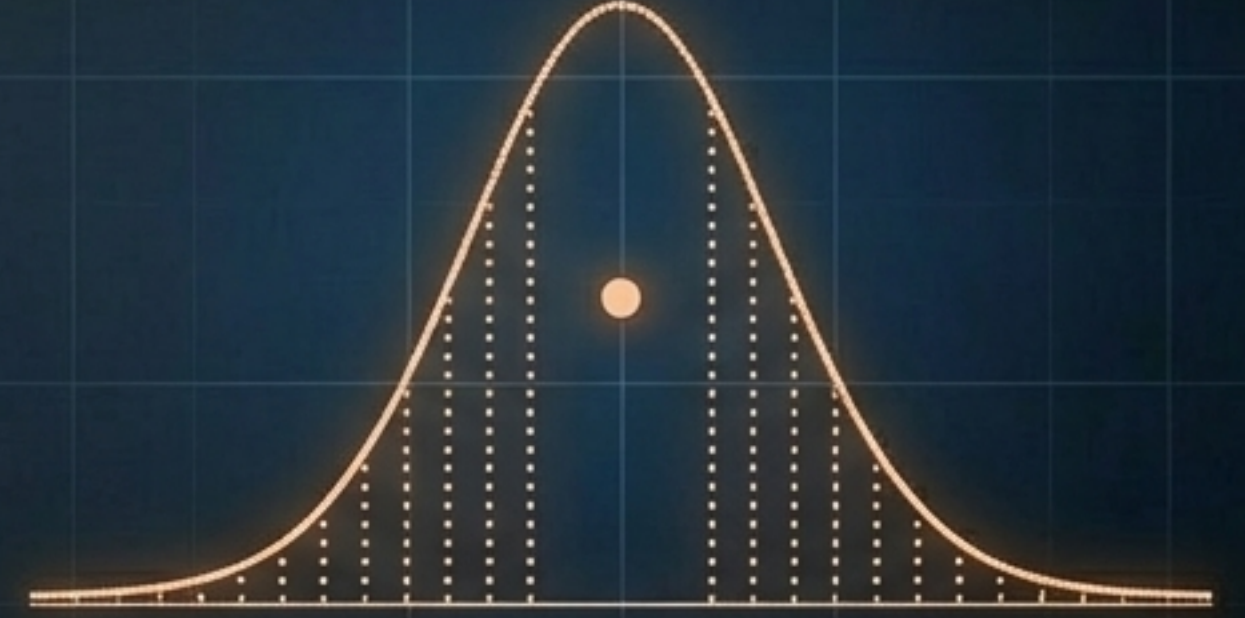
Bir pikselin tohuma katılması için iki matematiksel sınavdan geçmesi gerekir:

$$\sigma_N = \sigma / \sigma_{\max}$$

1) σ_N (Benzerlik Sınavı): Bölgedeki yapısal değişimi ölçer. Piksel, komşularıyla barışık mı?

$$d_i = \sqrt{(Y - Y_i)^2 + (Cb - Cb_i)^2 + (Cr - Cr_i)^2}$$

2) d_i (Renk Sınavı): YCbCr renk uzayında, pikselin mevcut bölgenin ortalama rengine olan göreceli Öklid mesafesi. Uzayda mesafe ne kadar küçükse, renk o kadar uyumlu, katılım o kadar hızlıdır.



Bütünü Parçalamak: Yukarıdan Aşağıya Bölge Bölme

Tohum büyütmenin tam tersi bir strateji:
Tıpkı bir **heykeltıraş** gibi çalışmak.

- Tüm görüntüyü devasa, tek parça bir mermer bloğu (**bölge**) olarak kabul et.
- Bu bloğun içindeki kusurlu, heterojen (farklılık içeren) fay hatlarını bul ve bütünü alt bölgelere böl. Çatlatma işlemi, her bir parça kendi içinde tamamen pürüzsüz olana kadar devam eder.



Başarıyı Ölçmek: Sınıf İçi Standart Sapma (WCSD)

$$WCSD = \sqrt{\sum_{i=1}^m \frac{N_i s_i^2}{N_{Total}}}$$

Parçalanan her bir bölgenin gerçekten kusursuz olup olmadığını denetleyen matematiksel hakem.

m alt bölgenin her birindeki piksel sayısı (N_i) ve o bölgenin kendi içindeki varyansının (s_i^2) ağırlıklı ortalaması.

Anlamı: WCSD ne kadar düşükse, oluşan her bir bölge kendi içinde o kadar homojen demektir. Bu, algoritmanın piksellerden anlamlı nesnelere başarıyla kestiğinin ispatıdır.

Senfoni: Matematiğin Kusursuz Uyumu

Görüntüleri anlamlandırmak,
kuru bir piksel ayrımı değildir;
zarif bir **algoritmalar senfonisidir**.

Eşiklerin o **kusursuz sadeliği**,
Türevlerin **milimetrik keskinliği**,
Snake modelinin doğayı taklit eden
fiziksel uyumu,
Ve Bölge Büyütmenin **kristalleşen
renk derinliği...**

Hepsi, bilgisayara kaotik piksellerden
anamlı dünyamızı göstermek için
yazılmış denklemlerdir.

