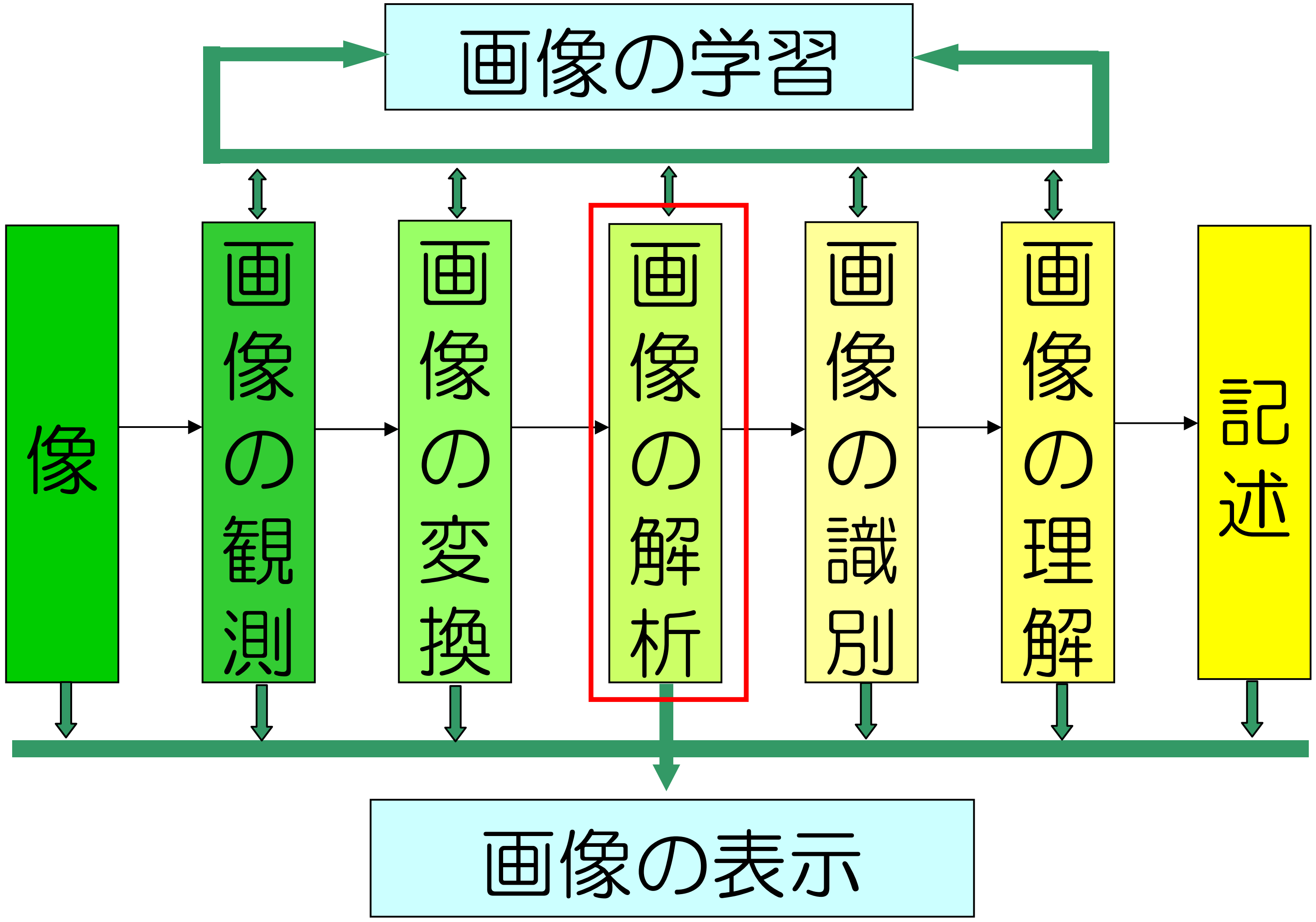
画像の特徴抽出(I)

点，線，領域，スケール

画像の処理工程



画像の分割

- 点検出

＊コーナ検出（Harris法，KLT法，主曲率法など）

- 線・輪郭線検出（濃淡値の不連続部の検出）

＊エッジ強調による線検出（Canny法，ゼロ交差法，弛緩法，非極値抑制法など）

＊パラメータ空間での線検出（Hough変換）

＊輪郭線検出（動的輪郭法，レベルセット法など）

- 領域分割

＊画質の均一性による画素の分類

＊テクスチャによる分類

コーナ検出(Harris/Plessey法)

各画素毎に，以下の行列$\boldsymbol{M}$を計算

ただし，画像Lは，$L = G \otimes f$（f：原画像，G：ガウス関数）

$$\boldsymbol{M} = \begin{bmatrix} L_x L_x & L_x L_y \\ L_x L_y & L_y L_y \end{bmatrix} = \begin{bmatrix} A & C \\ C & B \end{bmatrix},$$

行列列$\boldsymbol{M}$の固有値をλ_1, λ_2とするとし，各画素に対して以下のコーナらしさ$c(x, y)$を計算

$$c(x, y) = \det(\boldsymbol{M}) - k(tr(\boldsymbol{M}))^2, \qquad k = \text{constant}$$

$$\det(\boldsymbol{M}) = \lambda_1 \lambda_2 = AB - C^2$$

$$tr(\boldsymbol{M}) = \lambda_1 + \lambda_2 = A + B$$

もし，$c(x, y) < T$ ならば $c(x, y) = 0$

全画素処理後 ⇒ $c(x, y)$に対して非極値抑制処理を実行．ゼロでない点をコーナとして検出．

コーナ検出(KLT法)

各画素毎に，以下の行列$\boldsymbol{M}$を計算

ただし，画像Lは，$L = G \otimes f$（f：原画像，G：ガウス関数）

$$\boldsymbol{M} = \begin{bmatrix} L_x L_x & L_x L_y \\ L_x L_y & L_y L_y \end{bmatrix} = \begin{bmatrix} A & C \\ C & B \end{bmatrix},$$

行列Mの固有値を以下の式から計算

$$\det(\boldsymbol{M} - \lambda \boldsymbol{I}) = 0 \qquad \lambda^2 - (A+B)\lambda + AB - C^2 = 0$$

KLT法では，上式の固有値をλ_1, λ_2とするとき，

$$\min(\lambda_1, \lambda_2) > T$$

を満たす点をコーナとして検出.

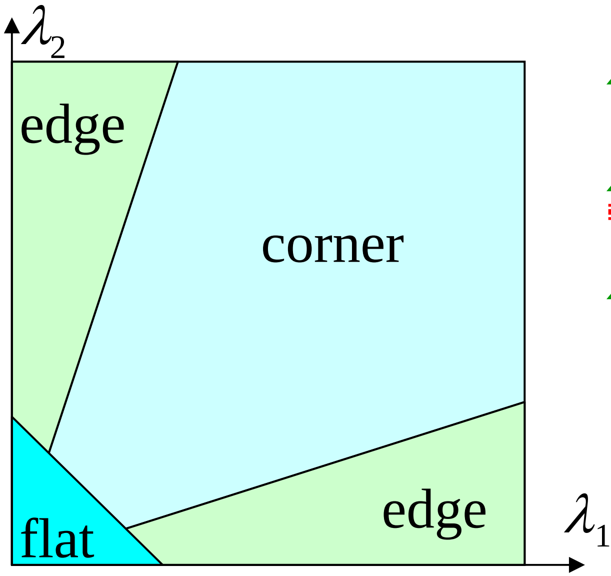
主曲率によるコーナ検出

ヘッセ行列の固有値から得られる主曲率を用いて特徴点を検出。（Hessian法）

各画素毎に，ヘッセ行列$\boldsymbol{M}$を計算． $\boldsymbol{M} = \begin{bmatrix} L_{xx} & L_{xy} \\ L_{xy} & L_{yy} \end{bmatrix},$

ただし，$L = G \otimes f$

行列$\boldsymbol{M}$の固有値をλ_1, λ_2とすると，以下のような分類が可能．

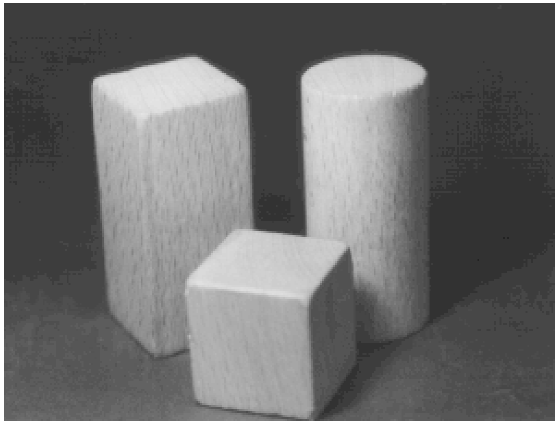


λ_1, λ_2ともに小さい　：フラット

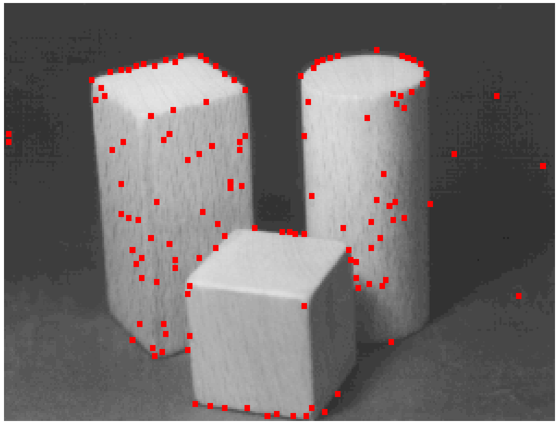
λ_1, λ_2ともに大きい　：コーナ

$\lambda_1 >> \lambda_2$或いは$\lambda_2 >> \lambda_1$：エッジ

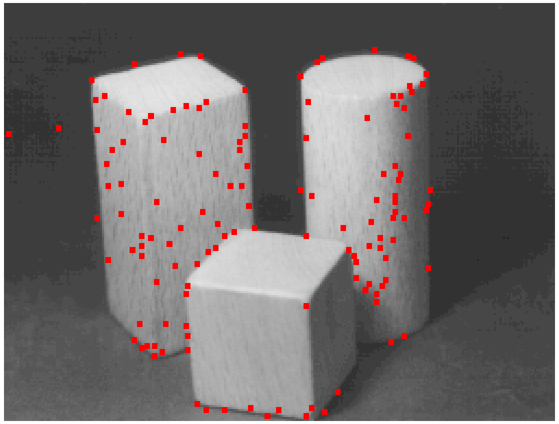
コーナ検出例(1/2)



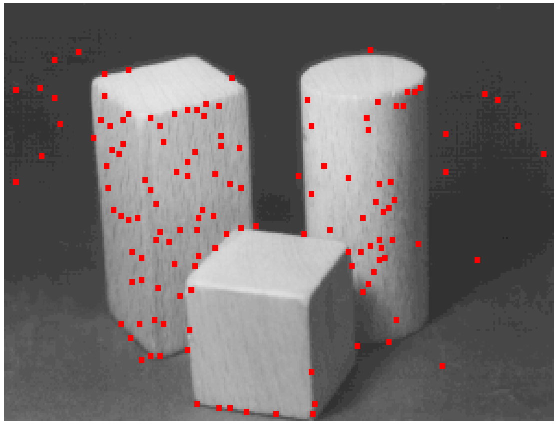
ガウス核は3×3のmask patternを使用.



KLT法
$T = 8$, point no. $= 130$

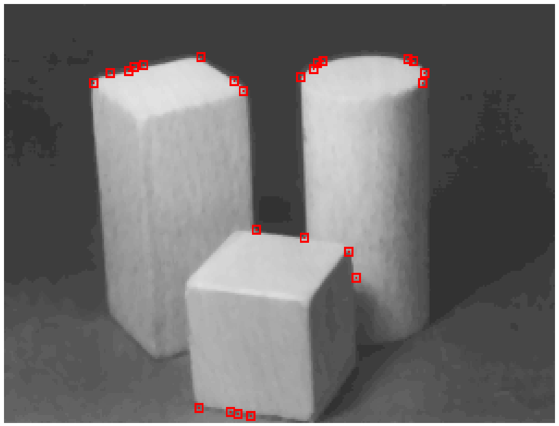


Harris/Plessey法
$T = 1000$, $k = 0.04$

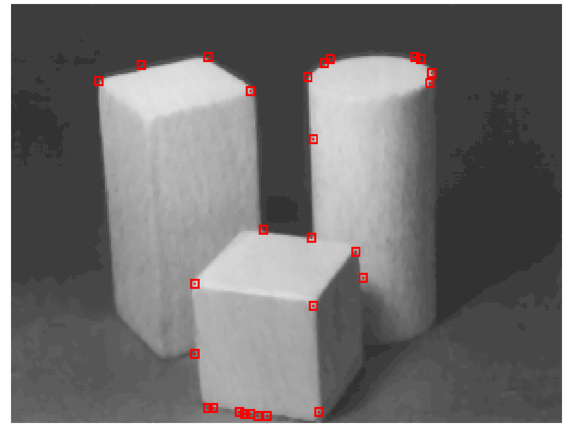


Hessian 法
$T = 1$, point no. $= 134$

コーナ検出例(2/2)



KLT 法
$T = 6$



Harris/Plessey法
$T = 1000, k = 0.04$

$$\left(\begin{array}{ll} \text{前処理：} & \text{Median Filter,} \\ \text{ガウス核：} & \text{５×５マスク} \\ \text{特徴点数：} & \text{２４点} \end{array} \right)$$

$$\left(\begin{array}{ll} \text{前処理：} & \text{Median Filter,} \\ \text{ガウス核：} & \text{５×５マスク} \\ \text{特徴点数：} & \text{２７点} \end{array} \right)$$

エッジ強調による線検出

処理の概要

- エッジ強調画像の生成

  弛緩法，ゼロ交差法，Canny法

  非極値抑制エッジ強調法などを利用

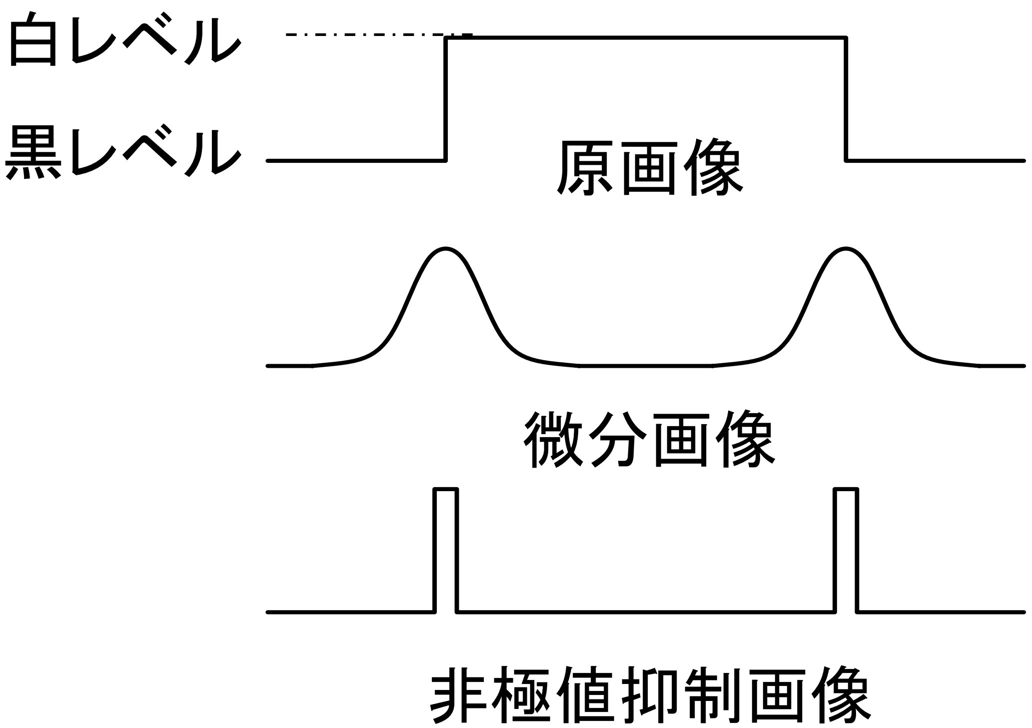
- 閾値以上のエッジ抽出
- 断片的なエッジの連結

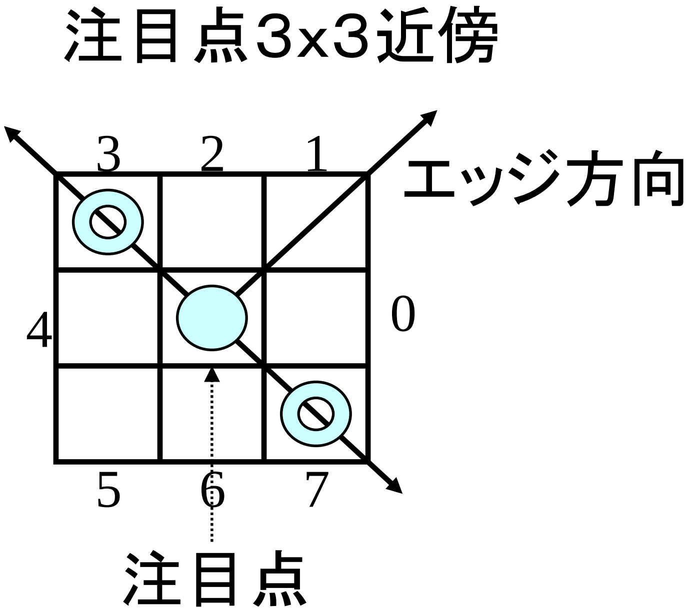
  エッジ方向に基づく延長処理，

  線の膨張処理など

非極値抑制エッジ強調法

- 非極値抑制の概要

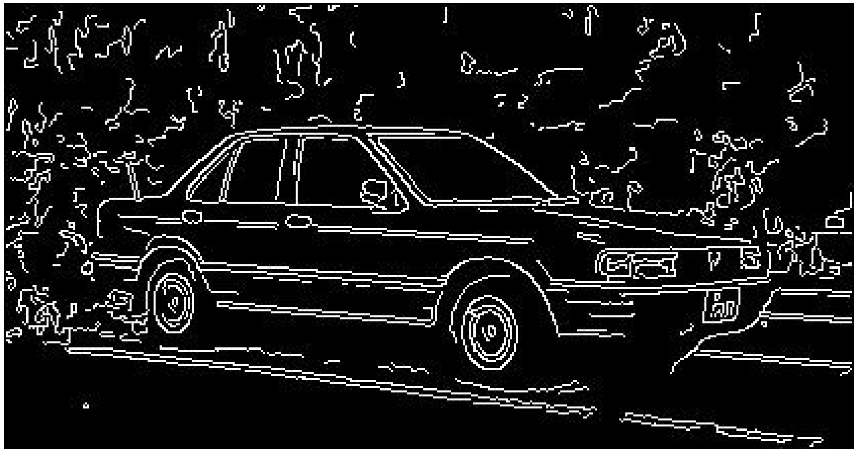




非極値抑制エッジ強調処理の例

原画像

エッジ伸長処理なし



エッジ伸長処理あり

線検出（Hough変換）

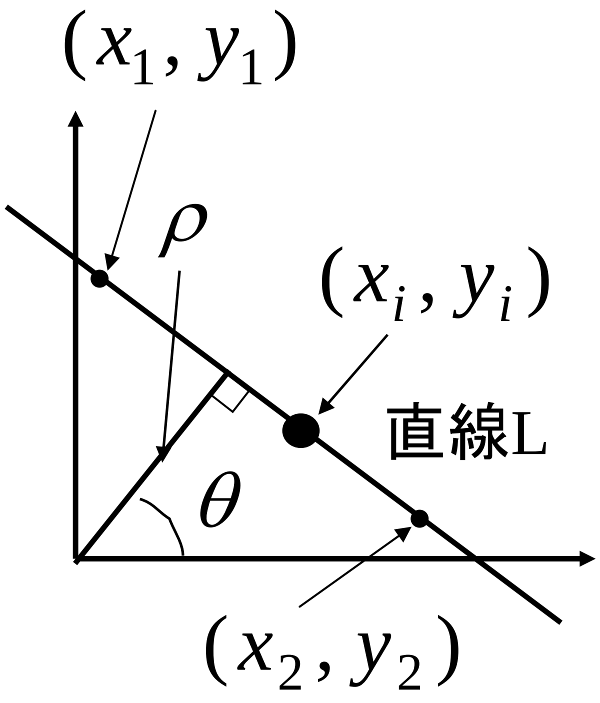
- 直線のパラメータ表示

$$x\cos\theta + y\sin\theta = \rho$$

画素点 (x_i, y_i)を通る直線

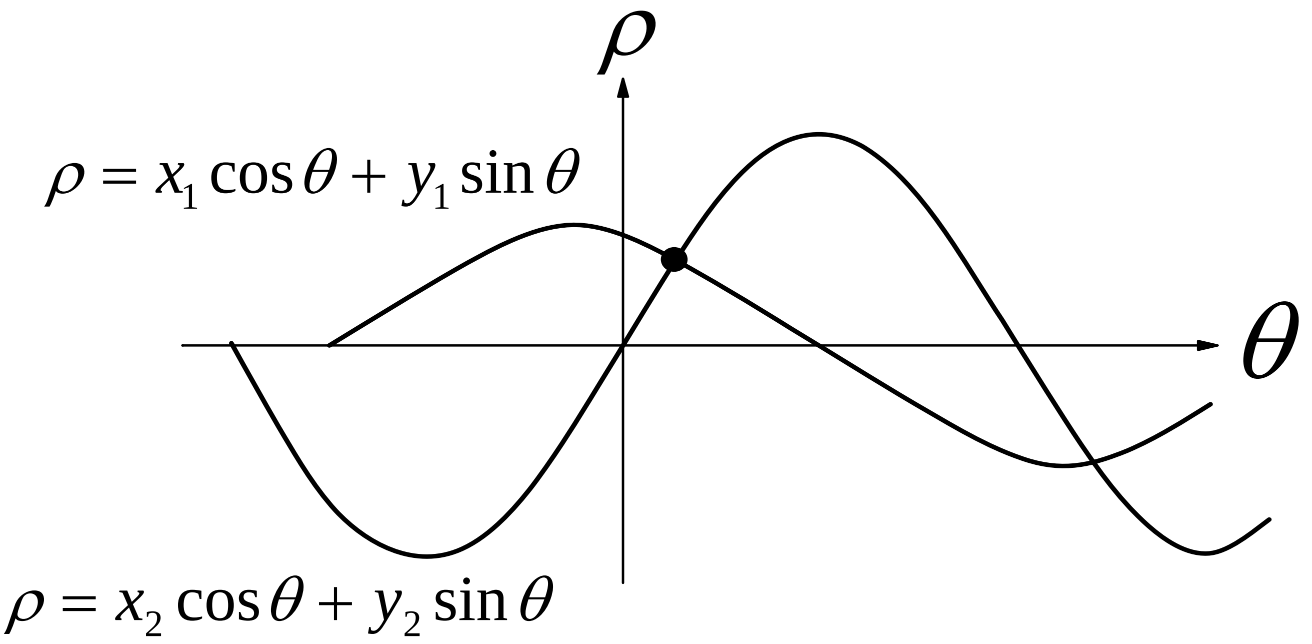
(θ, ρ)空間に写像

$$x_i\cos\theta + y_i\sin\theta = \rho$$



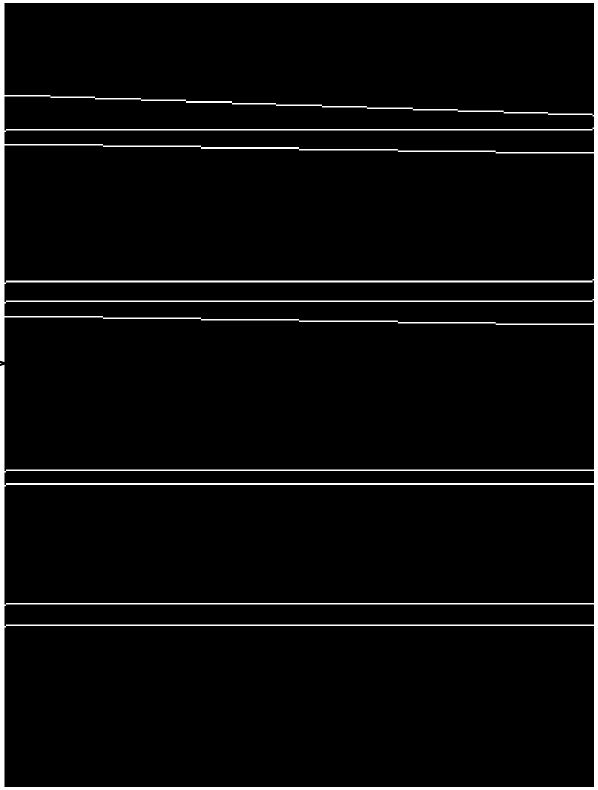
Hough変換

$(x_1, y_1), (x_2, y_2)$を通る直線の(θ, ρ)関係



Hough変換による直線検出の例

横方向直線の検出



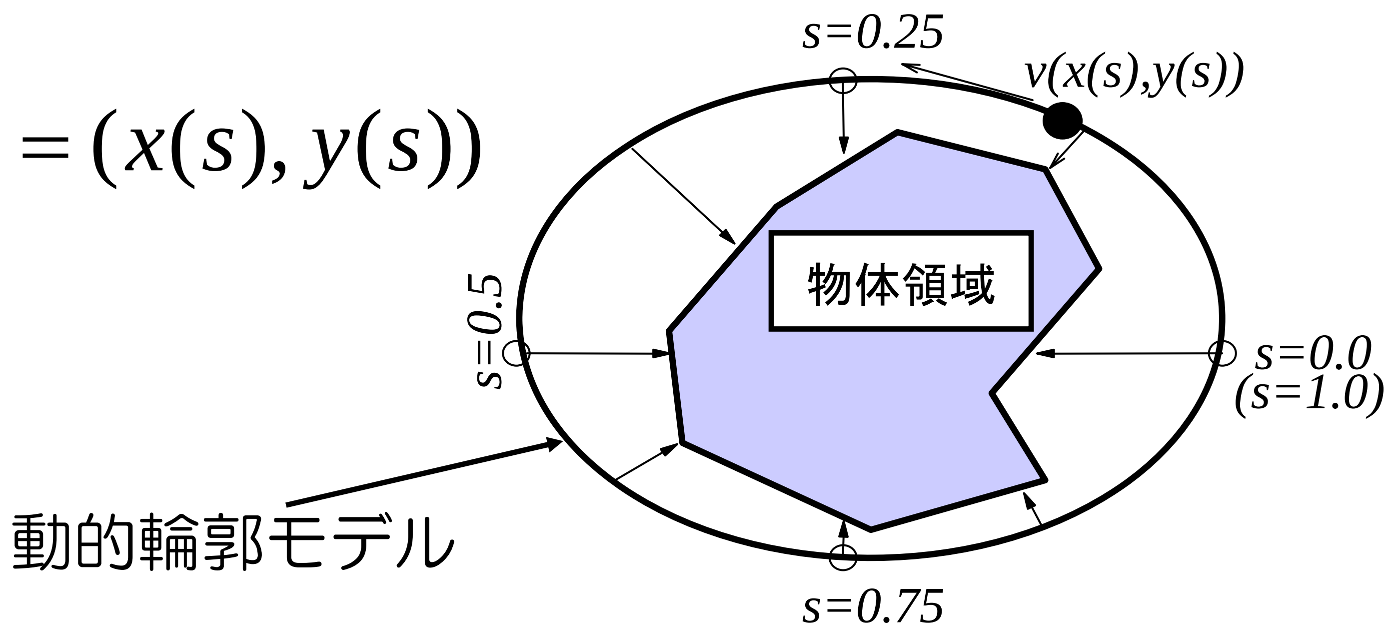
原画像の例

直線検出の例

輪郭線抽出（SNAKES)

- 動的輪郭モデル（Active Contour Model)
  - エネルギー（評価関数）の導入とその最小化
  - 輪郭線のパラメータ表現

$$v(s) = (x(s), y(s))$$



エネルギー関数 E_{snakes}

$$E_{snakes} = \int \left\{ E_{in}(v(s)) + E_{img}(v(s)) + E_{con}(v(s)) \right\} ds$$

E_{in} ：輪郭線の持つ内部エネルギー

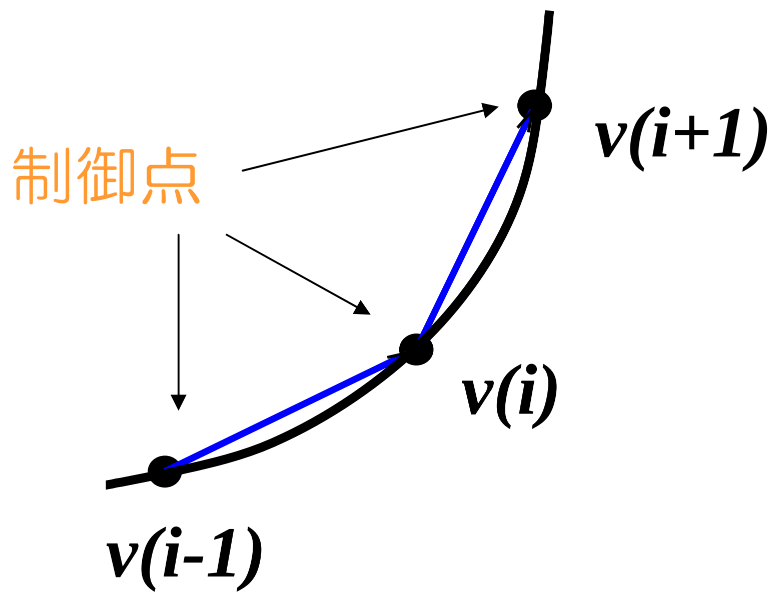
E_{img} ：画像のエッジ部分で極小となるエネルギー

E_{con} ：外部からの拘束力によるエネルギー

$s(0.0\sim1.0)$ を離散化して輪郭上に N 個の制御点 $v(1),\cdots,v(N)$ を配置.

内部エネルギーの例

$$E_{in} = \alpha\,|\,\dot{\mathbf{v}}(s)\,| + \beta\,|\,\ddot{\mathbf{v}}(s)\,|$$

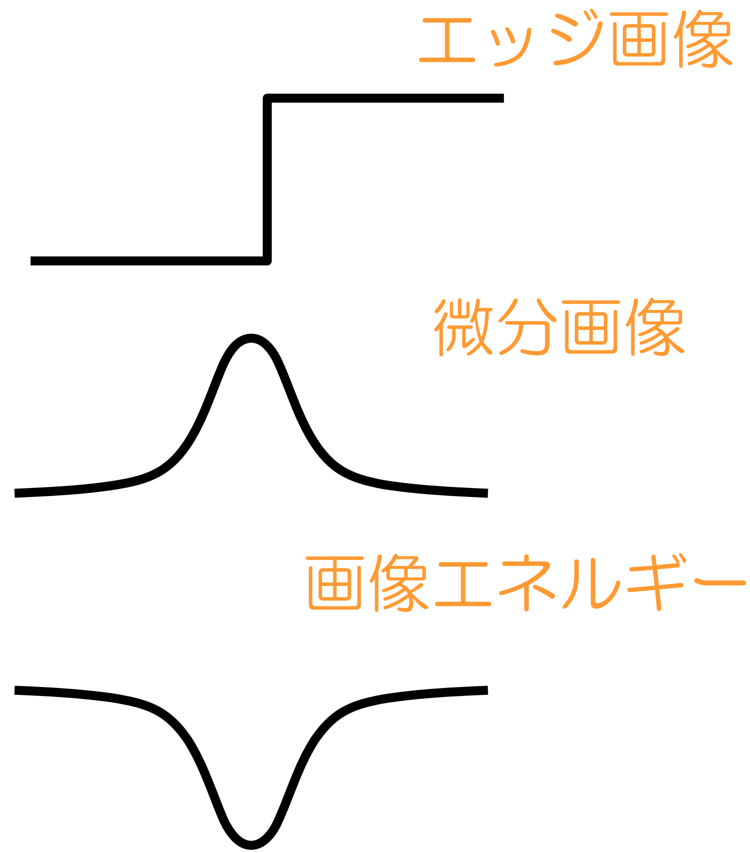


$$\dot{\mathbf{v}}(i) = \mathbf{v}(i) - \mathbf{v}(i-1)$$

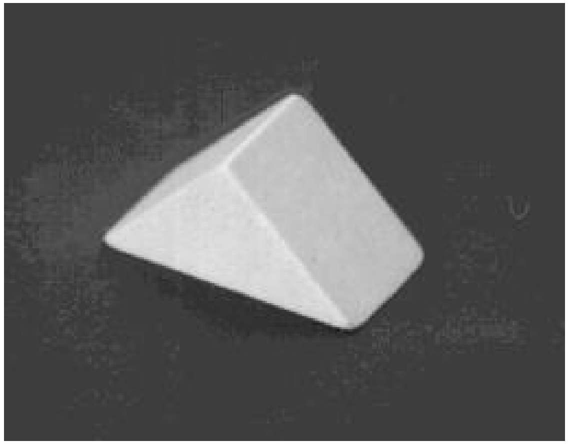
$$\ddot{\mathbf{v}}(i) = \dot{\mathbf{v}}(i+1) - \dot{\mathbf{v}}(i)$$

画像エネルギーの例

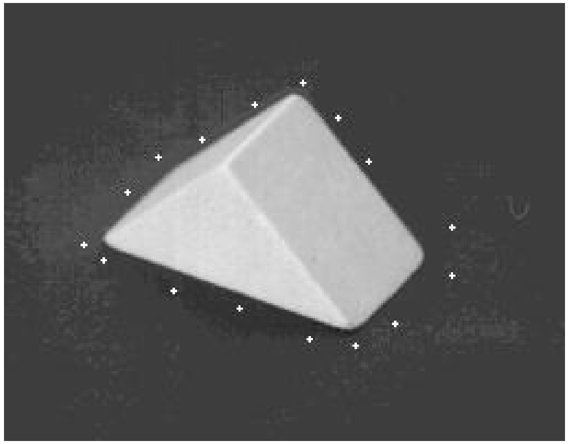
$$E_{img} = -\gamma \cdot \nabla I(x, y)$$



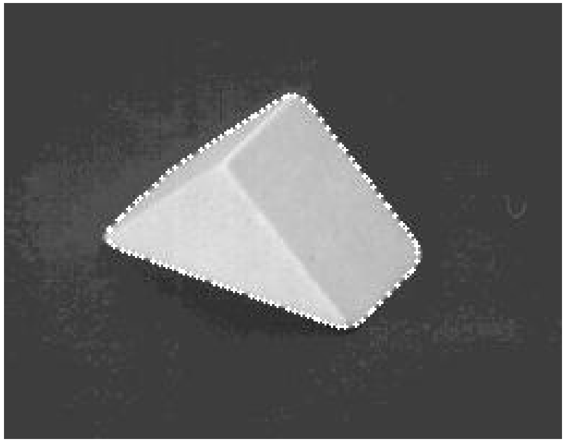
動的輪郭線処理の例



積み木画像



初期制御点配置



収束後制御点配置

領域分割

- 基本的概念

  画質の均一性とそれに基づく画素の分類

- 画質評価のための画像特性

  濃度，色，テクスチャ，スペクトルなど

- 画像分割のための空間

  画像空間上での分割

  特徴空間上での分割

- 分割領域数

  領域数２の分割 → 2値化処理

２値化処理

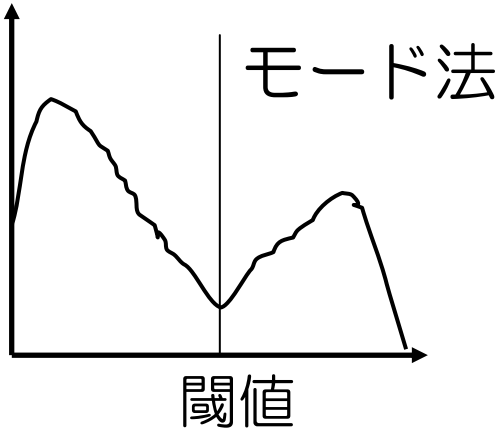
- カテゴリ数２の領域分割
- 主な手法（特徴空間上での分割）

　＊モード法

　＊事前情報に基づく2値化法

　＊区分的2値化法

　＊判別分析法



双峰性のヒストグラム

- 画像に占める対象物体の大きさの割合
- 対象物体の色情報

- 画像を分割し，各ブロック毎に２値化

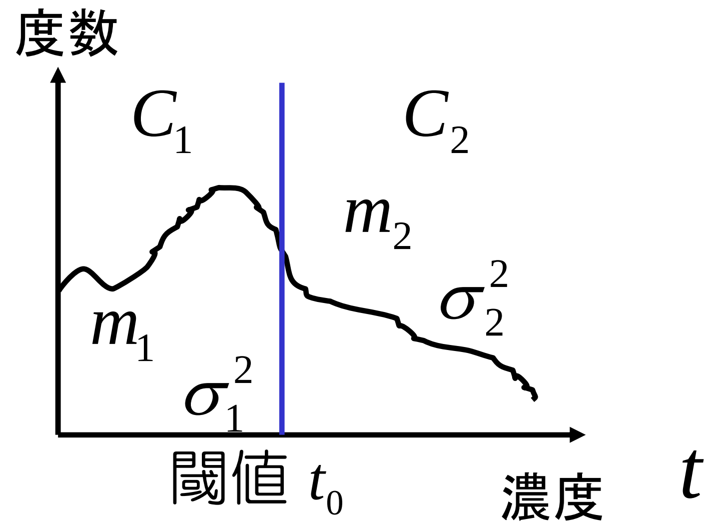
判別分析法

- 領域の分離性を評価

$$\eta(t) = \frac{\sigma_I^2(t)}{\sigma_T^2(t)} \Rightarrow \text{最大化}$$

（tを選択）

濃度ヒストグラム



$$\sigma_T^2(t) = \sigma_1^2 + \sigma_2^2$$ （クラス内分散）

$$\sigma_I^2(t) = \frac{n_1(m_1 - m_0)^2 + n_2(m_2 - m_0)^2}{n_1 + n_2}$$

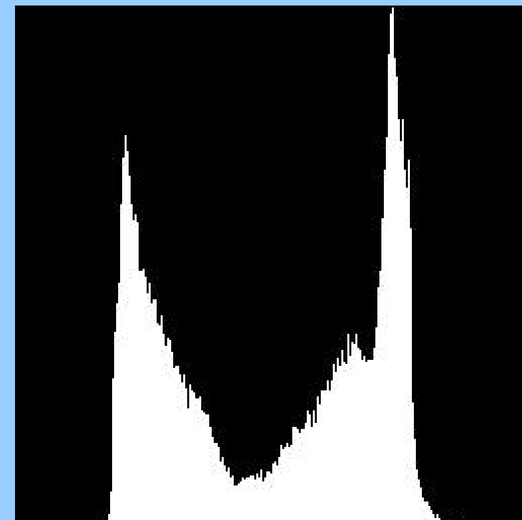
（クラス間分散）

n_1, n_2 :各クラスの個数

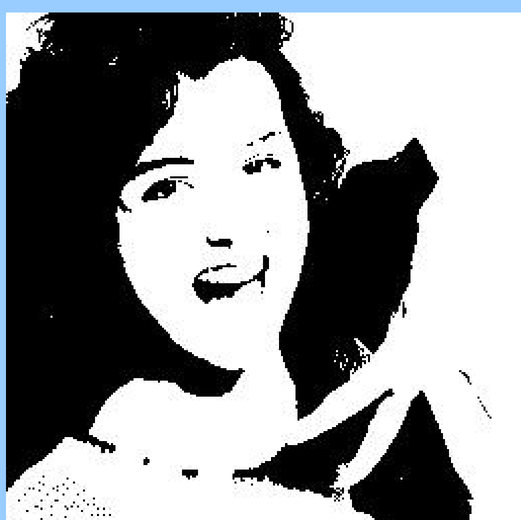
m_1, m_2 :各クラスの平均、m_0 :全体平均

２値化処理の例

原画像

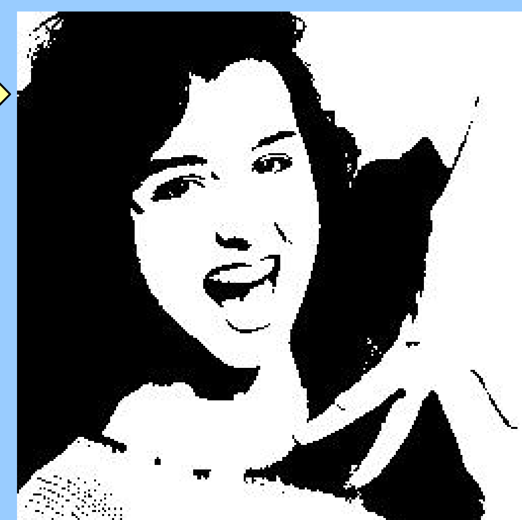


ヒストグラム



モード法(108)

判別分析法（121）



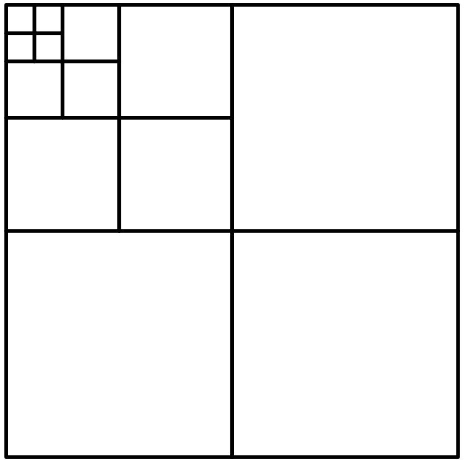
領域分割法

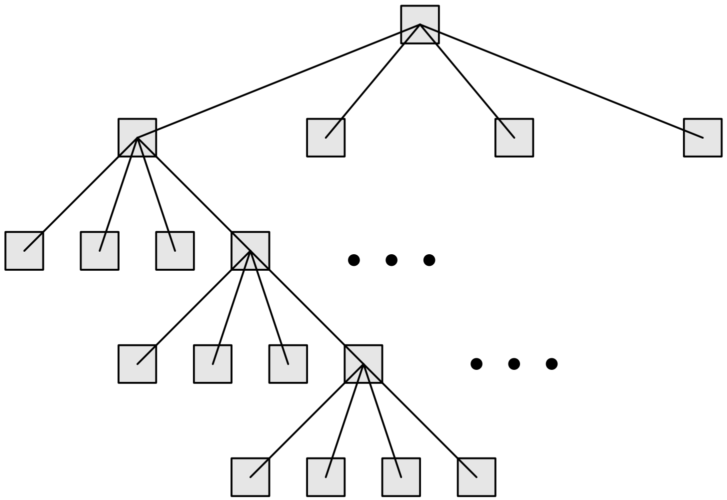
（カテゴリ数２以上）

- 分離統合による分割（Split and Merge法）
- 非線形平滑化による分割（ランクフィルタ等）
- 特徴空間でのクラスタリングによる分割
- 特徴空間と画像空間の再帰的分割
- 対象の知識を利用した分割
- ベイズ統計処理による分割
- Graph Cut法による分割
- 統計的手法（ＭＲＦなど）による分割

Split and Merge法

- 画像の4進木表現


画像

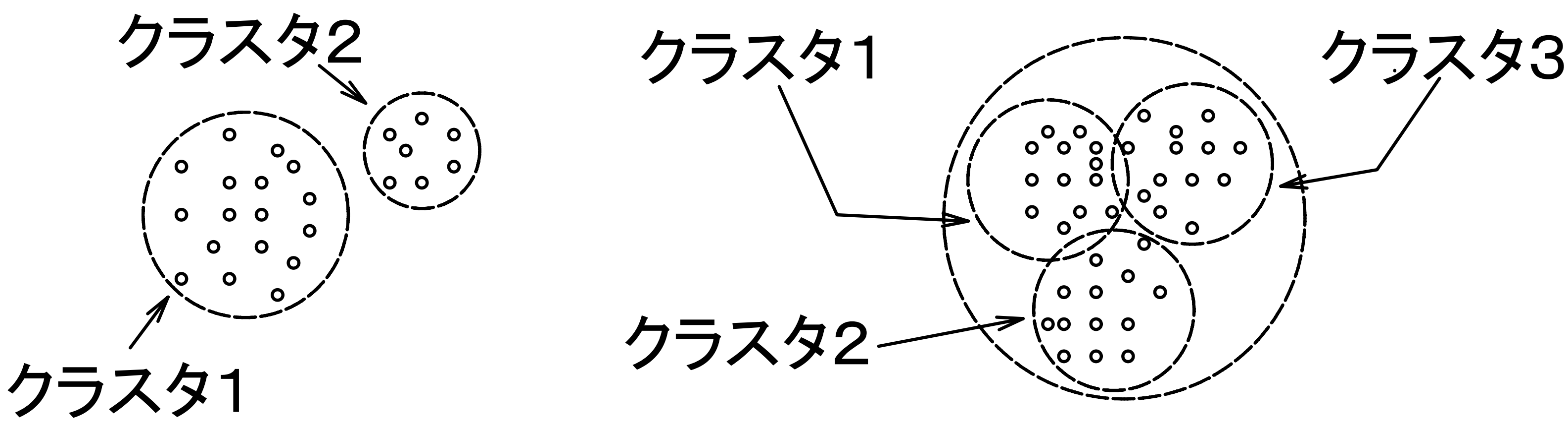


中間レベルから各スクエアの均一性を評価

不均一なら分離，隣接４近傍が均一なら統合

クラスタリングによる領域分割

ある特徴空間上の点を，その分布状態に応じて
いくつかのグループ（クラスタ）に分類すること．



クラスタリングの例

代表的クラスタリング法

- Nearest Neighbor Algorithm（NN法）
- K Nearest Neighbor Algorithm（K-NN法）
- K Mean Algorithm（K平均法）

準備

特徴空間上の N 個の点　：$\{P_1, P_2, \cdots, P_N\}$

クラスタ c_k の代表パターン　：C_k

パターン P_i と P_j の距離　：$d(P_i, P_j)$

Nearest Neighbor法

各パターン $P_l(l=1,\cdots,N)$ に対して，次式を計算．

$$d_{\min} = \min_{1\le i\le N_c}\{d(P_l, C_i)\}$$

(N_Cはクラス数，C_iはクラスc_iの代表ベクトル）

- $d_{\min} \le T$ならば，P_l をクラスc_i に帰属させる．
- $d_{\min} > T$ ならば，P_l を新たなクラスとして生成．

（Ncをインクリメントし，$C_{Nc} = P_l$とする．）

$d(P_l, C_i)$ ：**パターン P_l と C_i の距離**

K-Nearest Neighbor法

各パターン$P_l\,(l=1,\cdots,N)$に対して次式を計算.

$$d_l(i,j)=D(P_l,C_{ij})\quad for\;\; 1\le i\le N_C,\,1\le j\le n$$

C_{ij}は，クラスc_iのj番目の代表ベクトル.

クラス数，各代表ベクトルは予め決定.

- $d_l(i,j), 1\le i\le N_c, 1\le j\le n$を，距離の順に並び替える.
- 距離の小さい順にK個の代表ベクトルを選択.
- 選択された各代表ベクトルのクラスを調べ、最も多いクラスにP_lを帰属させる.

K-平均法(K-Mean Algorithm)

- K個のパターンを選択し，クラスタ$c_1, c_2, \cdots, c_k$の代表パターンとする.
- パターンP_lが,

$$d(P_l, C_i) < d(P_l, C_j),\ for\ \ j = 1, \cdots, K (j \neq i)$$

  のとき，パターンP_lをc_iに帰属させる.
- すべてのパターンの分類後，代表パターンを更新.

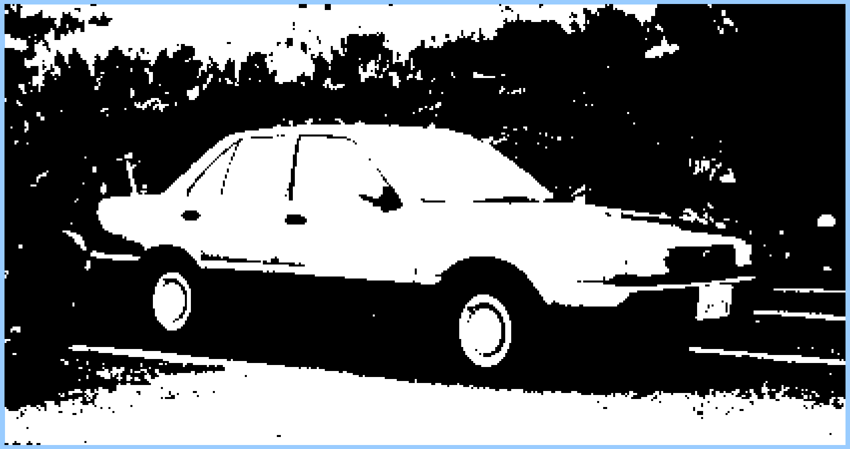
$$C_i = \frac{1}{N_i} \sum_{P_j \in c_i} P_j,$$

  N_iはクラスタc_iに属するパターン数
- 全ての代表パターンが変化しなくなるまで上記を反復.

領域分割の例

原画像

２値画像
(判別法)



K平均法による領域分割の例

$k=2$

$k=3$

$k=4$

$k=5$

$k=6$

$k=7$

画像特徴の記述(線）

- 線の記述
  - ：線の連結性：4連結と８連結
  - ：細線化処理
  - ：連結性に基づく点追跡
  - ：端点，分岐点，交差点，連結点
  - ：点列，チェーンコード，グラフ

5	6	7
4	r	0
3	2	1

８近傍

$$N_c^{(4)}(r) = \sum_{k \in S_1} \left(f(x_k) - f(x_k) f(x_{k+1}) f(x_{k+2}) \right)$$

$$N_c^{(8)}(r) = \sum_{k \in S_1} \left(\overline{f}(x_k) - \overline{f}(x_k) \overline{f}(x_{k+1}) \overline{f}(x_{k+2}) \right)$$

$$S_1 = \{0,2,4,6\} \qquad k : \mathrm{mod}\ 8$$

距離速度　D

- 距離の条件

$$D(p,r) \le D(p,q) + D(q,r)$$

（1）ユークリッド距離(Euclidean distance)

$$D_e((i,j),(k,l)) = ((i-k)^2 + (j-l)^2)^{1/2}$$

（2）街区画距離(city block distance)

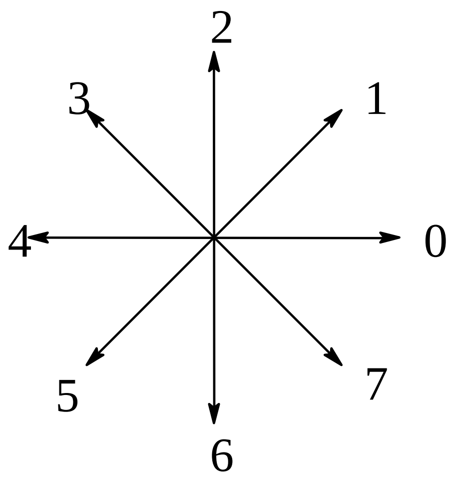
$$D_4((i,j),(k,l)) = |i-k| + |j-l|$$

（3）チェス距離(chase board distance)

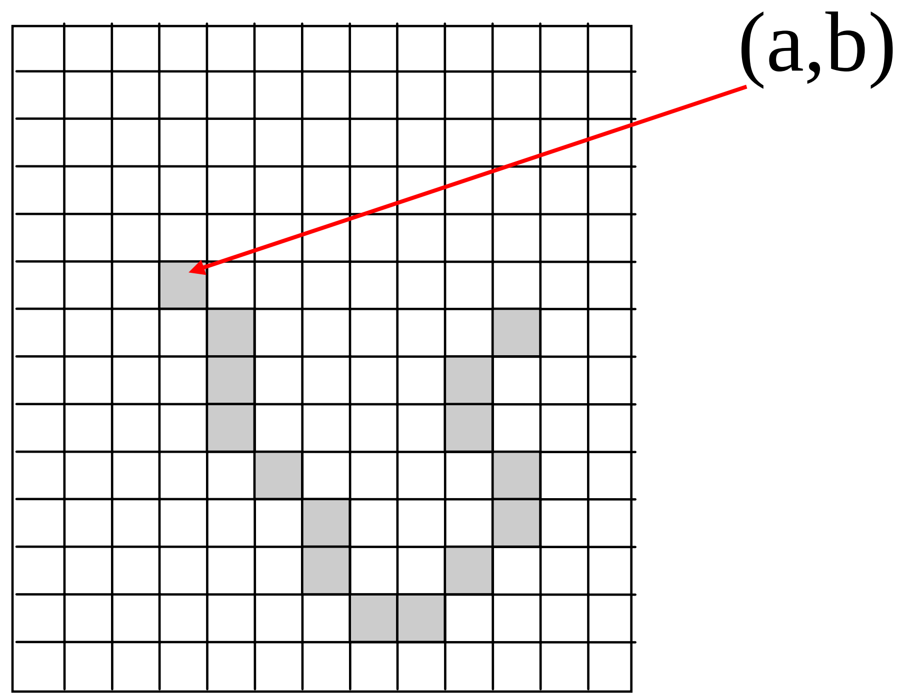
$$D_8((i,j),(k,l)) = \max(|i-k|,|j-l|)$$

Chain Code符号化法

Chian Code



primitive



start: (a,b)
chain codes: 76677670112321

画像特徴の記述（領域）

- 位相幾何特徴表現

  領域の連結数，孔数，オイラー数

- モーメント表現

$$M_{pq} = \sum_{y=1}^{N} \sum_{x=1}^{M} x^p y^q f(x, y),$$

$$\hat{M}_{pq} = \sum_{y=1}^{N} \sum_{x=1}^{M} (x - \bar{x})^p (y - \bar{y})^q f(x, y)$$

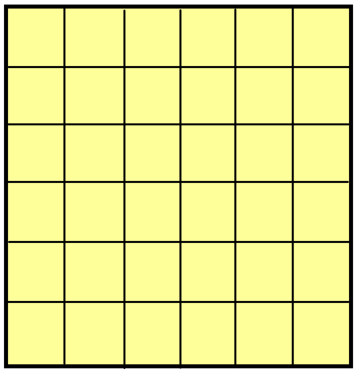
- 領域計量表現

  連結画素数，境界画素数，円形度，凹率など

- 特徴抽出表現

  各種直交変換，色ヒストグラム，HOG等の利用

HOG(Histogram of Oriented Gradients)



画像$I(x, y)$を$n\times n$画素の重ならないセルに分割
（$M\times N$セル）

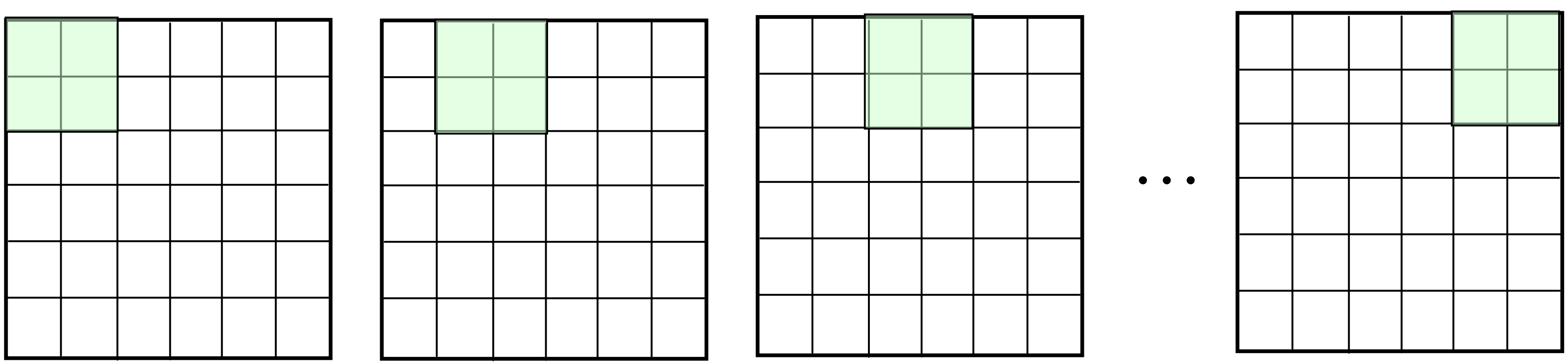
セル内の各画素の勾配を計算

$$m(x, y) = \sqrt{I_x^2 + I_y^2}, \quad \theta(x, y) = \tan^{-1}\frac{I_y}{I_x}$$

勾配方向の９方向量子化，セル内度数分布作成

$p\times q$のセルを単位とし，１セル毎にシフトしたブロックを構成

（2×2）



Navneet Dalal and Bill Triggs, “Histograms of Oriented Gradients for Human Detection”, Proc. of Int. Conf. on Computer Vision and Pattern Recognition, 2005

合計： $(M-p+1)\times(N-q+1)$個のブロック

k番目のブロックがu行v列目のセルを先頭とするとき,

$$V_k = [h_{u;v}, h_{u;v+1}, \cdots, h_{u;v+q-1}, h_{u+1;v}, \cdots, h_{u+p-1;v}, \cdots, h_{u+p-1;v+q-1}]$$

となる特徴ベクトルを作成.

($h_{u;v}$は(u,v)番目のセルの度数分布）

さらに特徴ベクトルを正規化

$$\tilde{h}_{i;j} = \frac{h_{i;j}}{\sqrt{\|V_k\|^2 + 1.0}}, h_{ij} \in V_k \quad \Rightarrow \tilde{V}_k$$

（例）

画像サイズ：　72×48画素,

セルサイズ：　8×8 画素,

ブロックサイズ：　2×2セル

⇒

ブロック数：40

特徴ベクトル次元数：360

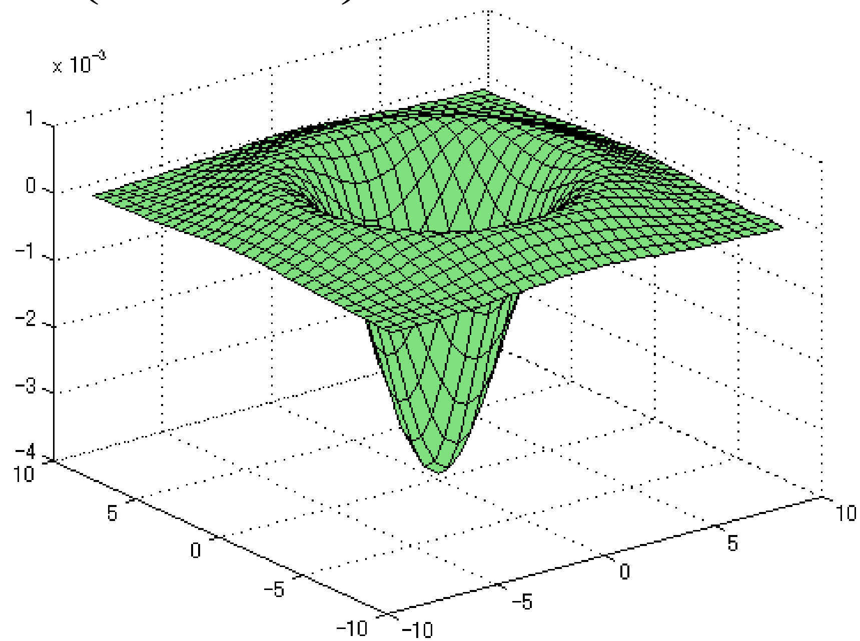
スケール特徴検出(1/10)

SIFT(Scale Invariant Feature Transform)で用いられている特徴量計算.

ガウス関数のラプラシアン(Laplacian of Gaussian; LoG)

$$\nabla^2 G = G_{xx} + G_{yy} = \frac{x^2 + y^2 - 2\sigma^2}{2\pi\sigma^6} \exp\left(-\frac{x^2 + y^2}{2\sigma^2} \right)$$

$\sigma = 3.0$の場合のグラフ



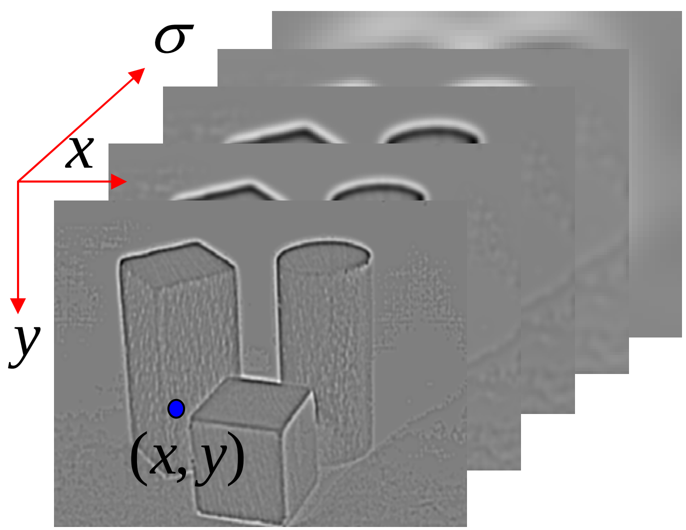
スケール特徴検出(2/10)

- 画像$f(x, y)$に対して，以下のLoG尺度空間を作成．

$$L_{\nabla^2 G}(x, y; \sigma) = \left(\nabla^2 G(x, y; \sigma)\right) \otimes f(x, y)$$

(参考）

$$L_G(x, y; \sigma) = \left(G(x, y; \sigma)\right) \otimes f(x, y)$$



$L_{\nabla^2 G}$尺度空間

スケール特徴検出(3/10)

スケール σ の増大

⇩

ガウス核表現に必要なマスクサイズの増大

⇩

計算時間の増大

マスク処理不整合端領域の増大

$L_{\nabla^2 G}$尺度空間作成の負荷軽 減法

(1) DoG処理によるLoG計算の近似

(2) ガウス核平滑化とダウンサンプリングの関係を利用

スケール特徴検出(4/10)

(1) DoG処理によるLoG計算の近似

$$\frac{\partial G}{\partial \sigma} = \frac{\partial}{\partial \sigma}\left(\frac{1}{2\pi\sigma^2}\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)\right)$$

$$= \frac{x^2+y^2-2\sigma^2}{2\pi\sigma^5}\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) = \sigma\left(\nabla^2 G\right) = \sigma \cdot LoG$$

一方，

$$\frac{\partial G}{\partial \sigma} = \lim_{\Delta\sigma \to 0}\frac{G(x,y;\sigma+\Delta\sigma)-G(x,y;\sigma)}{\Delta\sigma} = \lim_{k \to 1}\frac{G(x,y;k\sigma)-G(x,y;\sigma)}{(k-1)\sigma}$$

$$\approx \frac{G(x,y;k_0\sigma)-G(x,y;\sigma)}{(k_0-1)\sigma}$$

$, k_0 (>1)$は１に近い定数

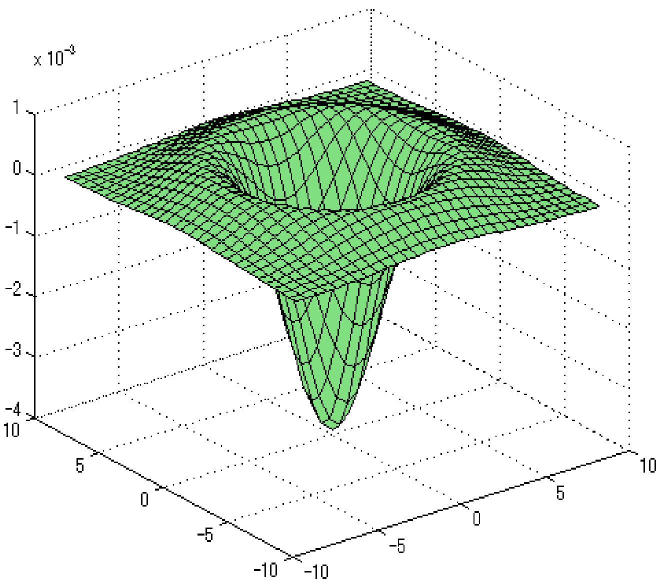
ここで，$DoG = G(x,y;k_0\sigma)-G(x,y;\sigma)$，

(Difference of Gaussian; DoG)とおくと，

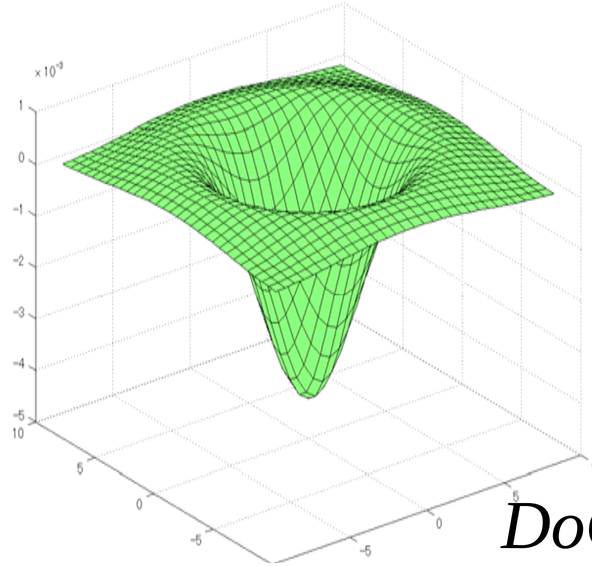
$\frac{\partial G}{\partial \sigma} = \sigma \cdot LoG$ の関係より $LoG \approx \frac{DoG}{\sigma^2(k_0-1)}$ ⇒ LoGの代りにDoGを利用

スケール特徴検出(5/10)

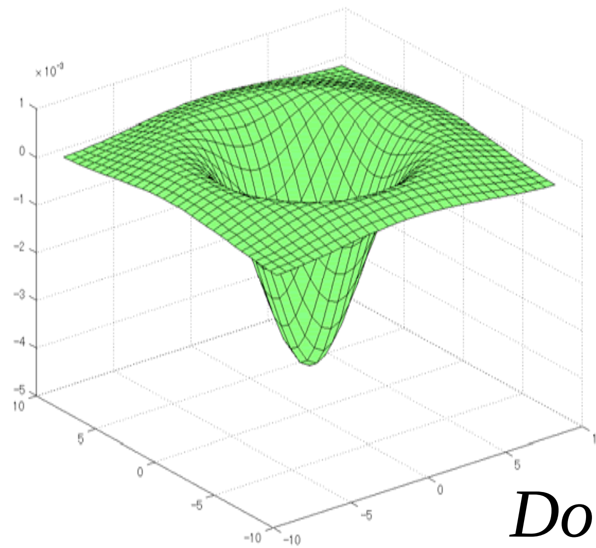
LoG vs. DoG



$LoG(\cdot\,; 3.0)$



$DoG(\cdot\,; 3.0, \sqrt[5]{2})$



$DoG(\cdot\,; 3.0, \sqrt[10]{2})$

スケール特徴検出(6/10)

DoG画像のスケールによる変化　$(\sigma_0 = 1.0, k^5 = 2.0)$

(a) $k^0\sigma_0$

(b) $k^1\sigma_0$

(c) $k^2\sigma_0$

(d) $k^3\sigma_0$

(e) $k^4\sigma_0$

(f) $k^5\sigma_0$

スケール特徴検出(7/10)

(2) ガウス核平滑化とダウンサンプリングの関係利用

$L_G^{(0)}(\cdot\,;\sigma_1)=G(\cdot\,;\sigma_1)\otimes f$ とし，$L_G^{(0)}(\cdot\,;\sigma_1)$を$1/2$にダウンサンプリングした画像を$L_G^{(1)}(\cdot\,;\sigma_1)$ とするとき，

$$L_G^{(1)}(\cdot\,;\sigma_1)\approx L_G^{(0)}(\cdot\,;2\sigma_1)$$

一般的に　$L_G^{(i+1)}(\cdot\,;\sigma_1)\approx L_G^{(i)}(\cdot\,;2\sigma_1)$

以上のことから

σ_1から$2\sigma_1$の間に$\sigma_1, k\sigma_1, \cdots, k^n\sigma_1(=2\sigma_1)$の$(n+1)$個のスケールを考え，$n$枚のDoG画像を生成．

$$\begin{aligned}
L_{DoG}^{(i)}(\cdot\,;\sigma_1) &= L_G^{(i)}(\cdot\,;k\sigma_1)-L_G^{(i)}(\cdot\,;\sigma_1)\\
L_{DoG}^{(i)}(\cdot\,;k\sigma_1) &= L_G^{(i)}(\cdot\,;k^2\sigma_1)-L_G^{(i)}(\cdot\,;k\sigma_1)\\
&\vdots\\
L_{DoG}^{(i)}(\cdot\,;k^{n-1}\sigma_1) &= L_G^{(i)}(\cdot\,;k^n\sigma_1)-L_G^{(i)}(\cdot\,;k^{n-1}\sigma_1)
\end{aligned}$$

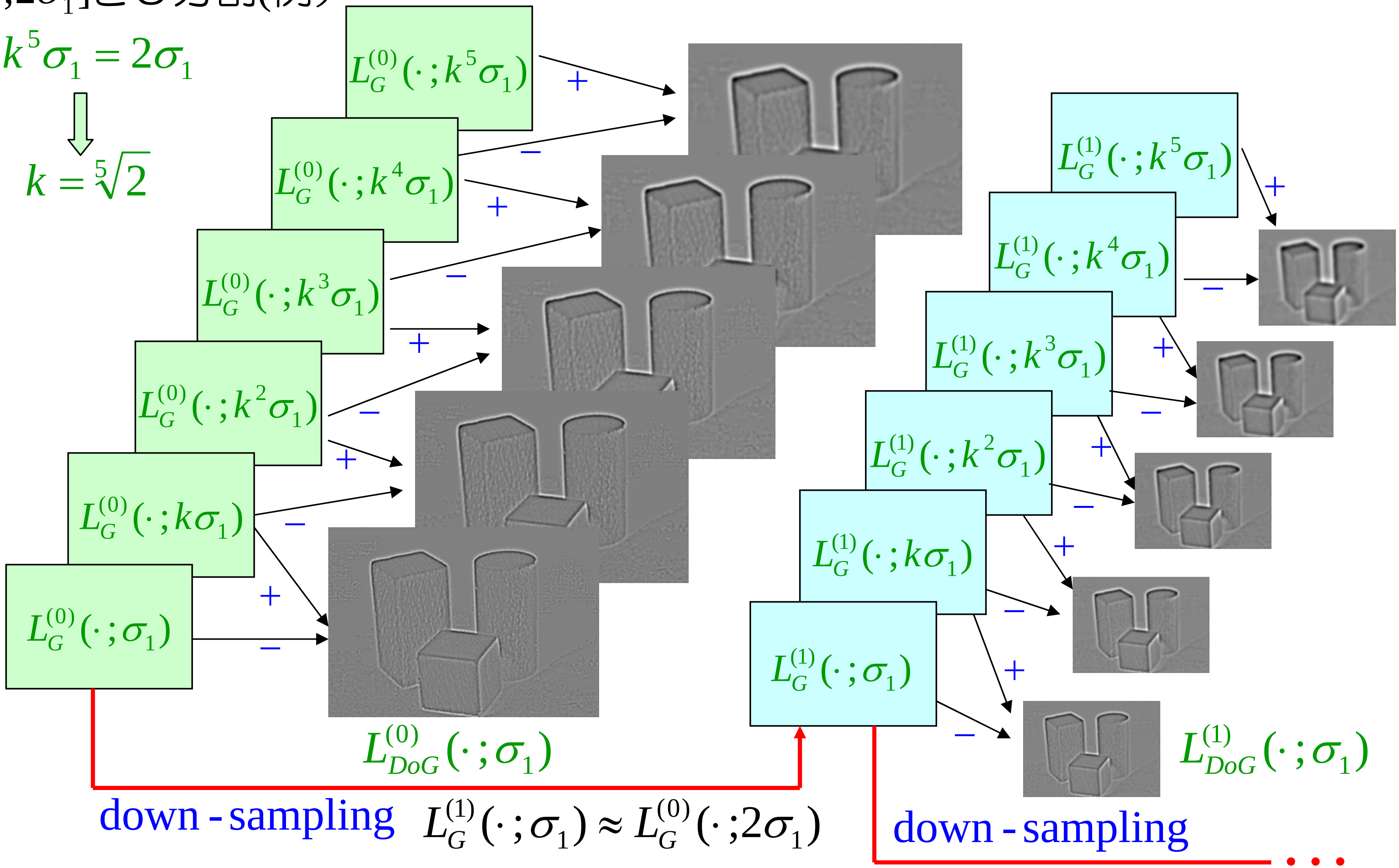
スケール特徴検出(8/10)

$[\sigma_1, 2\sigma_1]$を５分割(例)

$$k^5\sigma_1 = 2\sigma_1$$

$$\Downarrow$$

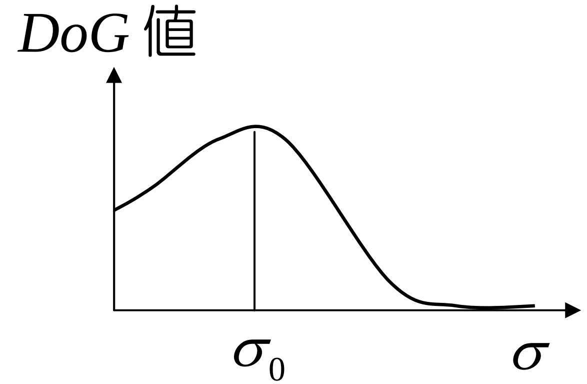
$$k = \sqrt[5]{2}$$



スケール特徴検出(9/10)

画素(x, y)におけるσをパラメータとするDoG値の変化に着目.

（極値の存在）

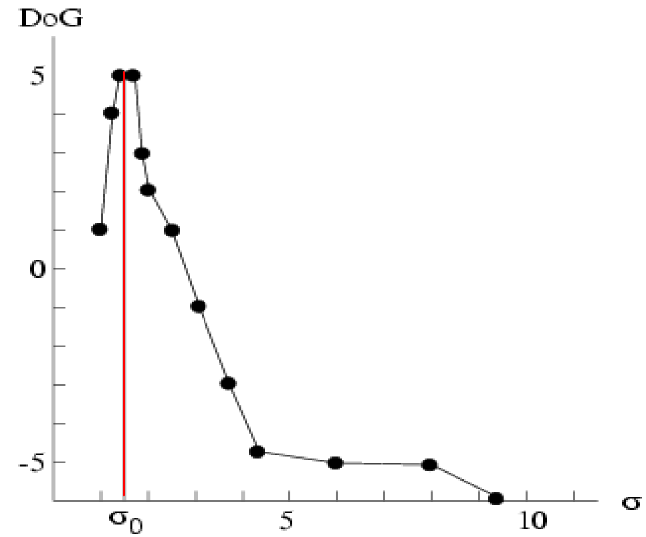


σ_0 : 画素(x, y)のスケール

スケール空間の因果性 に基づき，各画素毎の DoG 最大値を与えるスケールを，その点のスケール特徴 と定義.

スケール特徴検出(10/10)

中心点(128,128)での
DoG変化のグラフ



点(128,128)のスケールは， $\sigma = 1.4$

L_{DoG}尺度空間により得られる画素毎のスケール特徴可視化例