1

画像の特徴抽出(II) 一不変特徴量 -

2

各種の不変特徴記述子

- 平行移動に不変な特徴記述子 重心モーメント, Zernike moment, …
- •回転に不変な特徴記述子

Zernike moment, Log - Polar, SIFT, SURF, Ferns, ···

- ・拡大・縮小に対して不変な特徴記述子 SIFT, SURF, Ferns,…
- •濃度変化に対して不変な特徴記述子

Local Binary Pattern(LBP), ···

3

各種モーメント量(1/2)

画素(x, y)の濃度を d_{xy} , d_{xy} の生起確率を $p(d_{xy})$ とする.

画素(x, y)における(m+n)次モーメント:

$$M_{mn} = \sum \sum x^m y^n p(d_{xy}), \quad$$
ただし, $\sum_d p(d) = 1$

画素
$$(x, y)$$
における重心回り $(m+n)$ 次モーメント:

$$\hat{M}_{mn} = \sum \sum (x - \bar{x})^m (y - \bar{y})^n p(d_{xy})$$

$$\hbar \bar{z} = \frac{M_{10}}{M_{00}}, \ \bar{y} = \frac{M_{01}}{M_{00}}$$



知的画像処理(4)

4

各種モーメント量(2/2)

回転・平行移動,相似変換に対して不変な特徴記述子:

$$\frac{(\hat{M}_{20} + \hat{M}_{02})}{(\hat{M}_{00})^2}$$

平行移動、回転移動、相似変換、さらに反射に対して 不変なモーメント: Zernike Moment

(参考)

Zernike多項式:単位円内で完全直交系をなす複素多項式

$$V_{n,m}(x,y) = r_{n,m}(\rho)e^{jm\theta}$$

Zernikeモーメント:

$$A_{n,m} = C_n \iint_E f(x, y) V_{n,m}^*(x, y) dx dy$$

高次局所自己相関関数(1/3)

(Higher-order Local Autocorrelation)

相関関数

$$r_{x}(a_{1},a_{2},...,a_{N}) = \int_{D} f(x)f(x+a_{1})...f(x+a_{N})dx$$

x:注目画素点, N:自己相関の次元, D:画像領域



相関関数マスク (N=2の場合)



6

高次局所自己相関関数(2/2)

- ●画像を複数のブロックに分割
- ブロック内の画素に対して全ての相関係数を計算。
- •各係数毎に積算し,25次元度数分布生成.
- •ブロック数分の度数分布を連結して特徴表現



大津展之, "パターン認識に関する数理的研究", 電子技術総合研究所研究報告, Vol.818, (1991)

7

LBP-Local Binary Pattern(1/2)

手法の特徴

- ●濃度変化に対して不変な特徴量
- ●計算コストが少ない

LBPの計算法

注目点の画素値と、その8近傍にある画素値を比較



LBP Pattern = 10101001

LBP value = 128 + 32 + 8 + 1 = 169

LBP-Local Binary Pattern(2/2)

- ●画像を複数のブロックに分割
- ●ブロック内の画素に対してLBPを計算.
- •ブロック毎にLBPのヒストグラム生成.
- ブロック数分のヒストグラムを連結して特徴表現



T.Ojala,M.Pietikäinen and D.Harwood,"A comparative study of texture measures with classification based on feature distribution",Pattern Recognition, Vol.29, No.1, pp.51-59 (1996)



•ガウス関数による平滑化

ガウス関数 $\sigma: スケールファクタ$ $G(x, y; \sigma) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$

画像f(x, y)の平滑化: $G(x, y; \sigma)$ による畳込み処理 $L(x, y; \sigma) = (G \otimes f)$

平滑化画像 $L(x, y; \sigma)$ の微分

$$L_{x} = \frac{\partial L}{\partial x} = \frac{\partial}{\partial x} (G \otimes f) = \frac{\partial G}{\partial x} \otimes f = G_{x} \otimes f$$
$$L_{y} = \frac{\partial L}{\partial y} = \frac{\partial}{\partial y} (G \otimes f) = \frac{\partial G}{\partial y} \otimes f = G_{y} \otimes f$$

知的 阃 像 则 理 (4) ガウス関数とその導関数(2/3)

平滑化画像L(x, y)の微分(続き)

$$L_{xx} = \frac{\partial L_x}{\partial x} = G_{xx} \otimes f, \quad L_{yy} = \frac{\partial L_y}{\partial y} = G_{yy} \otimes f, \quad L_{xy} = \frac{\partial L_x}{\partial y} = G_{xy} \otimes f$$

ガウス導関数



 $G_{y} = -\frac{y}{2\pi\sigma^{4}} \exp\left(-\frac{x^{2}+y^{2}}{2\sigma^{2}}\right)$

10

平滑化画像 L(x, y)の微分 ガウス導関数による畳込み

ガウス関数とその導関数(3/3)



 $G_{xy}(x, y; 1.0)$

 $G_{xx}(x, y; 1.0)$

 $G_{yy}(x, y; 1.0)$

11

ガウス導関数と画像の畳込み

ガウス導関数による画像の畳込み



L(x, y; 2.0)



 $L_{xv}(x, y; 2.0)$





 $L_{xx}(x, y; 2.0)$







 $L_{yy}(x, y; 2.0)$

$$g(t;\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{t^2}{2\sigma^2}\right)$$
とおくと、(ガウスカーネル)

 $G(x, y; \sigma) = g(x; \sigma) \cdot g(y; \sigma)$ と表される.

ガウス関数 $G(x, y; \sigma)$ は分離型フィルタの一種であり、そのインパルス応答 $h_{G(x,y;\sigma)}$ は、 $g(t; \sigma)$ に対応するインパルス応答 $h_{g(t;\sigma)}$ によって、以下のようなカスケード接続で表現可能.



ガウス導関数のカスケード構成

ガウス導関数についても $G_x(x,y;\sigma) = g_x(x;\sigma) \cdot g(y;\sigma), \quad G_y(x,y;\sigma) = g(x;\sigma) \cdot g_y(y;\sigma)$ $G_{xx}(x,y;\sigma) = g_{xx}(x;\sigma) \cdot g(y;\sigma), \quad G_{yy}(x,y;\sigma) = g(x;\sigma) \cdot g_{yy}(y;\sigma)$ $G_{xy}(x,y;\sigma) = g_x(x;\sigma) \cdot g_y(y;\sigma),$ \vdots $h_{G_x(x,y;\sigma)} \longrightarrow h_{g_x(x;\sigma)} h_{g_y(y;\sigma)}$





ガウス核導関数の離散化例



16

知的画像処理(4) 画像の多重スケール表現(1/3)

• 画像の多重スケール表現の背景

画像計測 : あるアパーチャを持つ計測装置に依存 ある場合は、高解像度での計測を、またあるとき は粗い解像度で広い範囲を計測したい。

→ 最適アパーチャサイズを決めておくことは困難

アパーチャサイズ(画像のスケール)をパラメータとして、複数の スケール画像を表現する方法が考案

Multi-scale Representation

知的画像処理(4) 画像の多重スケール表現(2/3)

• 初期の多重スケール表現

Quad Tree (四分木表現) (1971)

均一性に基づいて画像を再帰的に分割し、木構造で表現.(均一とされた領域は木の葉として表現)初期の濃淡画像分割に利用:"Split-and merge"法

Pyramids(ピラミッド表現))(1981) サブサンプリングとスムージングの組合せ

Coarse-to-fine表現法

ピラミッド表現:現在も利用.

知的画像処理(4) 画像の多重スケール表現(3/3)

ピラミッド表現

入力画像*I*:2^{*K*}×2^{*K*}, *I*^(K) = *I* スムージングフィルタ:等方性,分離可能性を仮定 一次元処理として以下のように表現

$$f^{(k-1)}(x) = \sum_{n=-N}^{N} c(n) \cdot f^{(k)}(2x-n)$$

例えば、N = 1の場合、 $c(n) = \left(\frac{1}{4}, \frac{1}{2}, \frac{1}{4}\right)$

N = 2の場合, $c(n) = \left(\frac{1}{20}, \frac{1}{4}, \frac{2}{5}, \frac{1}{4}, \frac{1}{20}\right)$

19

 (256×256)



(128×128)





スケール(尺度) 空間とは

• 四分木表現・ピラミッド表現:

サイズ(スケール)を縮小(疎化)することで多重スケー ル表現する方法。 • スケール空間:

連続なスケール変数を持ち、全てのスケールで同じ画像 サイズ(同じ空間サンプリング数)を保持する多重ス ケール表現法。

1次元信号に対するガウシャン・スケール空間:1983

因果性: スケール変数の増加に伴って、新たなレベル面 (局所的最適値)が出現することはない。(1984)

 $L_{G_{derivative}}(x, y; \sigma) = G_{derivative}(x, y; \sigma) \otimes f(x, y)$:画像 f(x, y) の $G_{derivative}$ 尺度空間表現

スケール空間の例(1/2)







 $\sigma = 3.0$



 $\sigma = 4.0$



 $\sigma = 9.0$



 $\sigma = 8.0$

 L_G 尺度空間











スケール空間の例(2/2)

 $\sigma = 1$







 $\sigma = 4.0$





 $\sigma = 10.0$

 $\sigma = 9.0$



 $\sigma = 8.0$



 $\sigma = 6.0$





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)



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





24

スケール特徴検出(3/10)



L_{2²C}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)



^{知的画像処理(4)} スケール特徴検出(5/10)

LoG vs. DoG



 $LoG(\cdot; 3.0)$



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) $k^0 \sigma_0$



(d) $k^3 \sigma_0$



(b) $k^1 \sigma_0$



(e) $k^4 \sigma_0$



(c) $k^2 \sigma_0$



(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、 $L_{G}^{(0)}(\cdot;\sigma_{1}) = \frac{1}{2}$ 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

 $\sigma_{1} D 52 \sigma_{1} O 間 [C \sigma_{1}, k \sigma_{1}, \dots, k^{n} \sigma_{1} (= 2\sigma_{1}) O (n+1) (\square O)]$ $Z \mathcal{T} - \mathcal{V} \mathcal{E} \mathcal{E} \mathcal{E} \mathcal{A}, \quad n \mathcal{H} O D O G \square \mathcal{B} \mathcal{E} \mathcal{E} \mathcal{K}.$ $L_{DoG}^{(i)} (\cdot; \sigma_{1}) = L_{G}^{(i)} (\cdot; k \sigma_{1}) - L_{G}^{(i)} (\cdot; \sigma_{1})$ $L_{DoG}^{(i)} (\cdot; k \sigma_{1}) = L_{G}^{(i)} (\cdot; k^{2} \sigma_{1}) - L_{G}^{(i)} (\cdot; k \sigma_{1})$ $\vdots \qquad \vdots$ $L_{DoG}^{(i)} (\cdot; k^{n-1} \sigma_{1}) = L_{G}^{(i)} (\cdot; k^{n} \sigma_{1}) - L_{G}^{(i)} (\cdot; k^{n-1} \sigma_{1})$

スケール特徴検出(8/10)



スケール特徴検出(9/10)





スケール特徴検出(10/10)





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

SIFT-Scale Invariant Feature Transform

by Lowe (2004)

手法の特徴

- スケール空間での各点ごとのスケール特徴の定義と 処理の高速化
- ●スケールに基づく特徴領域の決定とオリエンテーションの導入

●特徴領域のHOG(Histogram of Gradient)に基づく 特徴量記述

•D.G.Lowe, "Distinctive image features from scale-invariant key points", Proc. of Int. Journal of Computer Vision (IJCV), 60(2) pp.91-110 (2004)

SIFT-処理の概要

1) キーポイント候補 点の検出 *L_{DoG}*尺度空間上での局所探索によるキーポイント候補点と そのスケール抽出

2) キーポイント候補点の絞込み

DoG値の低い候補点およびエッジ上の候補点の除去 キーポイントのサブピクセル推定(位置,スケール)

3)キーポイント周りの特徴領域の決定

オリエンテーションの計算、勾配方向ヒストグラム作成

4) キーポイント周りの特徴量表現

オリエンテーション:キーポイントにおける方向.
 この方向により向きの正規化:回転不変性

キーポイントp:
$$(u,v,\sigma_0)$$

平滑化画像: $L_G(\cdot;\sigma_0)$ を参照.
キーポイントpの近傍: $N(u,v,\sigma_0)$
 $(x,y) \in N(u,v,\sigma_0)$ に対し,

$$M(x,y) = \sqrt{\Delta_x(x,y)^2 + \Delta_y(x,y)^2} \qquad (勾配の強さ)$$
 $\theta(x,y) = \tan^{-1} \frac{\Delta_y(x,y)}{\Delta_x(x,y)} \qquad (勾配の向き)$
ただし,
 $\Delta_x(x,y) = L_G(u+1,v;\sigma_0) - L_G(u-1,v;\sigma_0)$
 $\Delta_y(x,y) = L_G(u,v+1;\sigma_0) - L_G(u,v-1;\sigma_0)$

重み付き方向ヒストグラムの作成

 $\theta(x, y)$ を36方向に量子化 $\Rightarrow \hat{\theta}(x, y)$ さらに、 $\hat{\theta}(x, y)$ 方向のビンに以下の重みを付けて投票。 $w(x, y) = G(x - u, y - v; \sigma_0) \cdot M(x, y)$ $G(\cdot; \sigma_0) :$ ガウス関数



^{知的画像処理(4)} SIFT-スケールと方向の抽出例











SIFT-特徵量記述



39

SURF:Speeded Up Robust Features

By Herbert Bay, Tinne Tuytelaars, and Luc Van Gool (2006)

手法の特徴

- ●SIFTと同様に、キーポイントを不変特徴量で記述
- ●計算の高速化のため、ガウス関数による畳込みを 近似処理で代用
- ●直接マスク処理による各点でのスケール計算
 (ピラミッド画像,反復的フィルタ処理を使用せず)

H.Bay,T.Tuytelaars, and L.V.Gool,"SURF:Speed Up Robust Features",Proc. of Int. Conf. of ECCV, (2006)

H.Bay,A.Ess, T.Tuytelaars,L.V. Gool,"SURF:Speeded Up Robust Features", Computer Vision and Image Understanding(CVIU), vol.110,No.3,pp.346—359 (2008)

40

SURF – 処理の概要

- 1) ヘッセ行列に基づくキーポイントの検出 ガウス導関数の近似による処理の高速化
- 2) LoG簡易フィルタによるスケール特徴計算 画像のダウンサンプリングを用いない、スケール抽出用 フィルタ群によるスケールの直接計算処理
- 3)オリエンテーションの決定

扇型検出器による勾配分布の最大方向選択

4) スケールに基づく特徴領域の決定と特徴量記述

知的画像処理(4) SURF - キーポイント検出

41

ヘッセ行列を利用

$$H(\mathbf{x},\sigma) = \begin{bmatrix} L_{xx}(\mathbf{x},\sigma) & L_{xy}(\mathbf{x},\sigma) \\ L_{xy}(\mathbf{x},\sigma) & L_{yy}(\mathbf{x},\sigma) \end{bmatrix}$$

$$L_{xx}(x,\sigma), L_{xy}(x,\sigma), L_{yy}(x,\sigma)$$
:
ガウス2次導関数による画像の畳込み
 $x = (x, y)$ は、画像中の点

ガウス導関数との畳込み処理に要する時間が大



知的画像処理(4) SURF-ガウス導関数とその近似







 G_{xx}

42

SURF - 近似処理

• ガウス関数の畳み込みを矩形フィルタで近似



• ヘッセ行列の固有値よりキーポイント検出

SURF – フィルタ構造



44

SURF - オリエンテーション(1)

- ♦得られたキーポイントのスケールσより、3σの 半径を持つ円領域を設定.
- ◆円領域内の各点について、以下のHaar型フィルタを 用いて*dx*,*dy*を計算.



♦(*dx*, *dy*)平面上に各点をプロット.

知的画像処理(4) SURF - オリエンテーション(2)

46

♦ 6 0 度の扇型(灰色部分)内 にプロットされている点の dy和を計算。 ◆扇型をずらし、和の最も大きい 扇型を検出. dx♦検出された扇型の向きをキーポイ ントのオリエンテーションとする.

SURF - 特徴記述(1)

◆オリエンテーションの方向に回転.

- ◆スケール値によって決まる円領域を4×4=16のブロックに分割.
- ◆各ブロック内で、25個のサンプル点について Haar型フィルタにより、*dx,dy*を計算.



SURF - 特徴記述(2)

 ◆ブロック内の各サンプル点における *dx*, |*dx*|, *dy*, |*dy*|の 積和 ∑*dx*, ∑|*dx*|, ∑*dy*, ∑|*dy*| を計算. (4次元特徴)
 ◆4×4×4=64次元の特徴ベクトルでキーポイント 特徴量と定義.



SIFT, SURFの特徴

•一般的特徴

スケールの変化に強い

回転に強い

照明変化に対して比較的頑強

微小なアフィン変換に対して比較的頑強

• 応用分野

対応点探索と画像マッチング 物体認識、画像分類、特徴点追跡など、 多くの応用が可能.



(応用)特徴ベクトルによる対応点探索

対応点探索を行う画像 $I_R, I_T; I_R, I_T$ での特徴点集合を $V_R = \left\{ \mathbf{v}_i^{(R)} \mid i = 1, \cdots, N_R \right\}, V_T = \left\{ \mathbf{v}_i^{(T)} \mid i = 1, \cdots, N_T \right\}$

とする. $v_i^{(R)} \in V_R \mathcal{E} v_j^{(T)} \in V_T$ に対して、特徴点間距離を定義.

$$d(\mathbf{v}_{i}^{(R)},\mathbf{v}_{j}^{(T)}) = \sqrt{\sum_{k=1}^{N} (v_{i,k}^{(R)} - v_{j,k}^{(T)})^{2}}, \quad N: 特徴の次元数$$

ただし、 $v_{i,k}^{(R)}, v_{j,k}^{(T)}$ は特徴ベクトル $v_i^{(R)}, v_j^{(T)}$ のk番目の要素 ある閾値Tに対して

対応点探索実験 一対応点決定における閾値の影響実験一

• 使用画像





画像 (A)

画像(B) 画像サイズ: 640×480画素

- SIFTによる対応点探索
- ●探索の閾値 Tを変化

^{知的画像処理(4)} SIFT対応点探索処理例(1)



T = 0.49

T = 0.70

52

^{知的画像処理(4)} 対応点探索実験 – SIFTとSURF

• SIFT

モデル画像学習時間: 691ms 使用プログラム

- Rob Hess
- http://web.engr.oregonstate.edu/~hess/
- SURF

モデル画像学習時間: 31ms 使用プログラム

- OpenCV
- http://sourceforge.net/projects/opencvlibrary/

モデル画像



画像サイズ: 328×244画素



(1342, 124)



SIFT

SURF

知的画像処理(4) 54



(1284, 128)



(135, 65)

(所要時間:ms,対応点数))

(102, 57)



(1800, 26)



(174, 3)

SIFT

SURF



(1533, 41)



(103, 6)

まとめ

SIFT:

照合時間の点で実時間処理にはやや不向き 精度はやや不安定 学習時間はやや短く,追加学習は可能

SURF:

照合時間は短く、実時間処理も可能

精度はほぼ安定

学習時間は非常に短く、追加学習は容易