画像情報処理1
－ 第10回 －

立命館大学　情報理工学部
画像・音メディアコース
岩本 祐太郎、徐　剛

シラバス

- 画像解析するための画像特徴を抽出する方法について学ぶ
- 画像認識方法(顔，物体etc.)について学ぶ

9 / 徐剛	画像特徴の抽出１	画像の微分、勾配、エッジ抽出、Sobelフィルタ ※【BCPレベル1~2】対面で実施、【BCPレベル3~4】webで実施
10 / 徐剛	画像特徴の抽出２	Cannyフィルタ、ヒステリシス閾値処理 ※【BCPレベル1~2】対面で実施、【BCPレベル3~4】webで実施
11 / 徐剛	画像特徴の抽出３	２次元特徴、コーナーの抽出、Harrisオペレータ ※【BCPレベル1~2】対面で実施、【BCPレベル3~4】webで実施
12 / 徐剛	画像特徴の抽出４	ハフ空間、ハフ変換、直接抽出 ※【BCPレベル1~2】対面で実施、【BCPレベル3~4】webで実施
13 / 徐剛	画像照合と認識１	テンプレートマッチング、輝度の線形変換、正規化相関 ※【BCPレベル1~2】対面で実施、【BCPレベル3~4】webで実施
14 / 徐剛	画像照合と認識２	ディスタンスマップ、２次元パターンの探索、影や隠れにもロバストなエッジマッチング ※【BCPレベル1~2】対面で実施、【BCPレベル3~4】webで実施
15 / 徐剛	確認テスト(60分)と解説(30分)	第９～１４回の授業内容についてのテスト ※【BCPレベル1~2】対面で実施、【BCPレベル3~4】webで実施

参考文献・データセット

- CG-ARTS, ”ディジタル画像処理[改訂第二版]”, 2020/2/26.
- 神奈川工科大学標準画像/サンプルデータ

  http://www.ess.ic.kanagawait.ac.jp/app_images_j.html

  ※スライドの画像は画像処理でよく用いられる上記の標準画像を利用
- John Canny, “A Computational Approach to Edge Detection”, IEEE Trans. On Pattern analysis and machine intelligence, Vol.8, No. 6, 1986.

今日の目的

- Canny エッジ検出アルゴリズムの原理理解
- Cannyエッジ検出アルゴリズムを実装できる

～レポートあり～

Canny エッジ検出器をMATLAB言語で実装

入力画像　スムージング　画像の微分(sobel)　非最大値抑制　ヒステリシス閾値

Cannyエッジ検出の目的

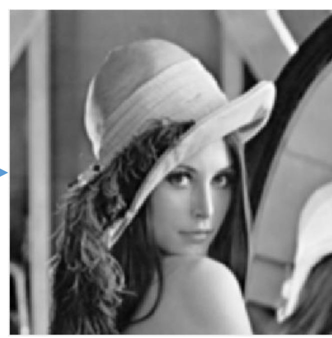
- 画像の位置合わせやエッジマッチング等に利用される.
- 画像内のオブジェクト境界となるため画像の領域分割(セグメンテーション)やオブジェクト認識に利用される.

John Canny, “A Computational Approach to Edge Detection”, IEEE Trans. On Pattern analysis and machine intelligence, Vol.8, No. 6, 1986.

Cannyエッジ検出アルゴリズム(概要)

入力画像

Gaussian smoothingにより画像内のノイズを取り除く



スムージング

画像の微分によりエッジを強調する(ただし，エッジに幅がある)

画像の微分(sobel)

勾配方向の大きさを確認し，極大値のみ抽出することで細線化する

勾配の極大位置の検出

2段階の閾値処理により，一定値以上の勾配を持つ画素のみエッジとする

ヒステリシス閾値

ディジタル画像処理[改訂第二版] 教科書P116-120

Cannyエッジ検出アルゴリズム

John Canny, “A Computational Approach to Edge Detection”, IEEE Trans. On PAMI, 8(6), 1986.

この二つを統合したフィルターがCanny Filter

画像の平滑化(ノイズの低減)

↓

画像の微分(Sobelフィルタ等)

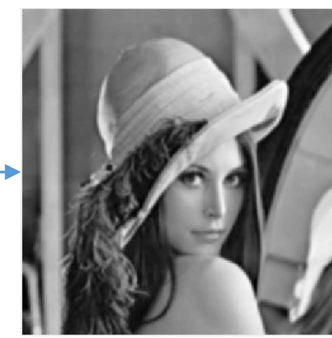
↓

勾配の極大位置の検出

↓

ヒステリシス閾値処理

入力画像



スムージング

Cannyエッジ検出アルゴリズム

John Canny, “A Computational Approach to Edge Detection”, IEEE Trans. On PAMI, 8(6), 1986.

画像の平滑化(ノイズの低減)

画像の微分(Sobelフィルタ等)

勾配の極大位置の検出

ヒステリシス閾値処理

入力画像

スムージング

様々な平均

- 算術平均 $\mu = \frac{1}{n}\sum_{i=1}^{n} x_i = \frac{1}{n}(x_1 + \cdots + x_n)$
- 加重平均 $\mu = \frac{1}{\sum_{i=1}^{n} w_i}\sum_{i=1}^{n} w_i x_i = \frac{1}{w_1 + \cdots + w_n}(w_1 x_1 + \cdots + w_n x_n)$
- 幾何平均 $\mu = \sqrt[n]{\prod_{i=1}^{n} x_i} = \sqrt[n]{x_1 \cdots x_n}$
- 調和平均 $\mu = \frac{n}{\sum_{i=1}^{n} \frac{1}{x_i}} = \frac{n}{\frac{1}{x_1} + \cdots + \frac{1}{x_n}}$
- 一般化平均 $\mu = \left(\frac{1}{n}\sum_{i=1}^{n} x_i^p\right)^{1/p}$

画像の平滑化

加重平均化フィルタにより，画像のノイズを低減する．重みは**ガウス分布**が良く用いられる．

■ 2次元ガウス分布

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

赤枠内の式で重みの計算を行い，最後にフィルタの合計が1になるように正規化する．

入力画像

ガウシアンフィルタ：

0.023247	0.033824	0.038328	0.033824	0.023247
0.033824	0.049214	0.055766	0.049214	0.033824
0.038328	0.055766	0.063191	0.055766	0.038328
0.033824	0.049214	0.055766	0.049214	0.033824
0.023247	0.033824	0.038328	0.033824	0.023247

標準偏差=2のガウス分布

スムージング

【補足】ガウシアンフィルタの作成

例: フィルタサイズ 5x5 の場合

1. x, yの範囲を[-2,1,0,1,2]とする.

(-2,-2)	(-1,-2)	(0,-2)	(1,-2)	(2,-2)
(-2,-1)	(-1,-1)	(0,-1)	(1,-1)	(2,-2)
(-2,0)	(-1,0)	(0,0)	(1,0)	(2,0)
(-2,1)	(-1,1)	(0,1)	(1,1)	(2,1)
(-2,2)	(-1,2)	(0,2)	(1,2)	(2,2)

2. 以下の式に代入し，各要素の値を埋める.

$$Gauss(x,y) = exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)$$

3. フィルタの合計が1になるように正規化する.

$$Gauss(x,y) = \frac{Gauss(x,y)}{\sum \mathrm{Gauss(x,y)}}$$

0.023247	0.033824	0.038328	0.033824	0.023247
0.033824	0.049214	0.055766	0.049214	0.033824
0.038328	0.055766	0.063191	0.055766	0.038328
0.033824	0.049214	0.055766	0.049214	0.033824
0.023247	0.033824	0.038328	0.033824	0.023247

【補足】加重平均化フィルタ

整数演算(int)でフィルタリングした後に各画素を16で割ればよいため、高速に平滑化できる.
真ん中から周辺に行けば行くほど、値が減っていく.
ピークの整数値（何ビットを使うか）に対して縁の整数値が0になれば、それ以上のサイズは不要.

1	2	1
2	4	2
1	2	1

/16

3x3画素

1	4	6	4	1
4	16	24	16	4
6	24	36	24	6
4	16	24	16	4
1	4	6	4	1

/256

5x5画素

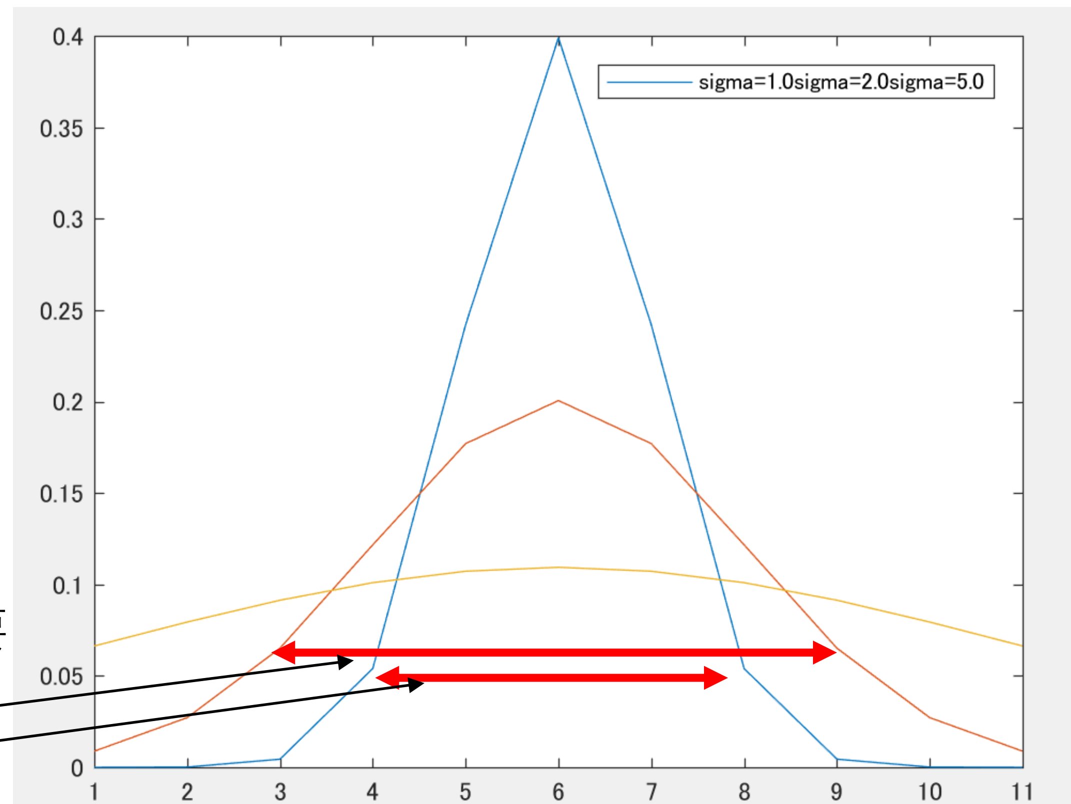
【補足】ガウシアンフィルタのパラメータ

標準偏差が大きくなるにしたがって，平滑化の効果が強まり，大きなフィルタサイズが必要となる．

パラメータ：

・標準偏差σ

・フィルタサイズ[N,N]



σ=5のとき，大きなフィルタが必要

σ=2のとき，[7x7] 程度

σ=1のとき，[5x5] 程度

【補足】ガウシアンフィルタの標準偏差σ

標準偏差が大きくなるほど，関数の裾野が広がるため，画像がぼける．

フィルタリング＝たたみ込み積分

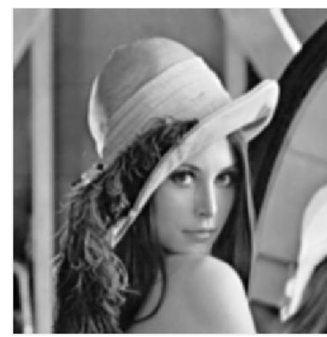
- 連続系の畳み込み積分
- １次元　f*g(t)= $\int f(u)g(t-u)$du
- ２次元　f*g(s,t)= $\int f(u,v)g(s-u,t-v)$dudv
- 離散系の畳み込み積分
- １次元　F*G(t)= $\sum_{i=-N}^{N} F(i)G(t-i)$
- ２次元　F*G(s,t)= $\sum_{j=-M}^{M} \sum_{i=-N}^{N} F(i,j)G(s-i,t-j)$

ディジタル画像処理[改訂第二版] 教科書P116-120

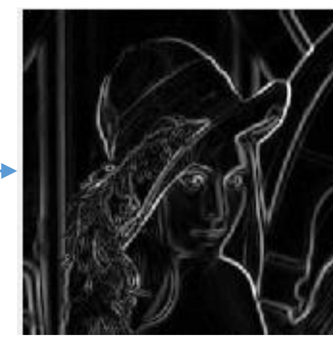
Cannyエッジ検出アルゴリズム

John Canny, “A Computational Approach to Edge Detection”, IEEE Trans. On PAMI, 8(6), 1986.

1. 画像の平滑化(ノイズの低減)
2. **画像の微分**
3. 勾配の極大位置の検出
4. ヒステリシス閾値処理



スムージング



画像の微分

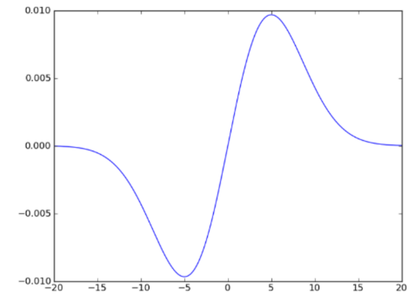
畳み込みの微分＝微分の畳み込み

$$\frac{d(\mathrm{f}*\mathrm{g})}{dt} = \frac{df}{dt}*\mathrm{g} = f * \frac{dg}{dt}$$

$$\frac{\partial(\mathrm{f}*\mathrm{g}(\mathrm{s},\mathrm{t}))}{\partial s} = \frac{\partial f}{\partial s}*\mathrm{g} = f * \frac{\partial g}{\partial s}$$

$$\frac{\partial(\mathrm{f}*\mathrm{g}(\mathrm{s},\mathrm{t}))}{\partial t} = \frac{\partial f}{\partial t}*\mathrm{g} = f * \frac{\partial g}{\partial t}$$

ガウス関数の微分



- 1次元ガウス関数の微分 $G'(x)=\frac{-x}{\sqrt{2\pi}\sigma^3}exp\left(-\frac{x^2}{2\sigma^2}\right)$
- 2次元x方向の一次微分 $G_x(x,y)=\frac{-x}{2\pi\sigma^4}exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)$=G'(x)G(y)
- 2次元y方向の一次微分 $G_y(x,y)=\frac{-y}{2\pi\sigma^4}exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)$=G(x)G'(y)
- $G_x(x,y)$と$G_y(x,y)$がCanny Filter：平滑化＋微分

Sobel FilterはCanny Filter

スムージング

-1	0	1
-2	0	2
-1	0	1

fx

Gx

-1	-2	-1
0	0	0
1	2	1

fy

Gy

$$\sqrt{(G_x(i,j))^2 + (G_y(i,j))^2}$$

強度画像

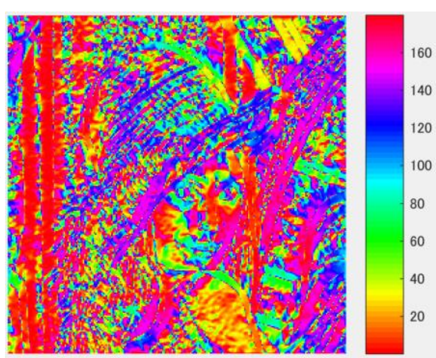
勾配方向の計算

次のステップ(非最大値抑制)のために勾配方向の計算と角度の量子化を行う.

■ STEP1 勾配方向の計算

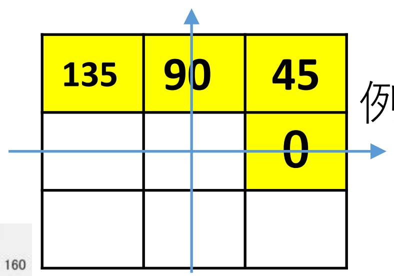
$$\tan^{-1}\frac{G_y(i,j)}{G_x(i,j)}$$

強度画像

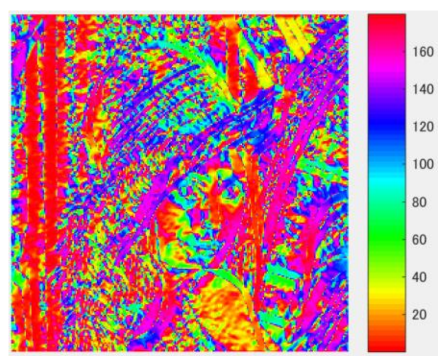


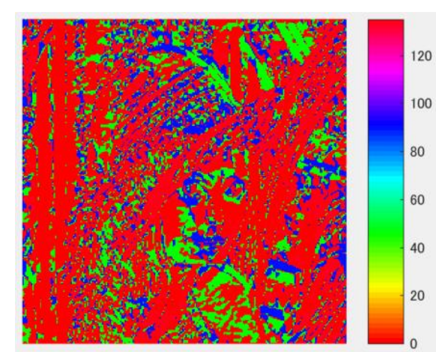
■ STEP2 角度の量子化

角度がこの格子のどれかに割り当たるよう量子化する.



例)
- 0° : Angle < 22.5 || Angle >=157.5
- 45° : 22.5 < Angle <= 67.5
- 90° : 67 .5 < Angle <= 112.5
- 135° : 112.5 < Angle < 157.5





※【プログラム】画像座標はy軸が下に対して、角度の量子化はy軸が上になっているので、上下反転が必要。もしくはy軸方向のsobelフィルタを正負反転する必要がある。 Matlabではatan2d を利用すると-180°～180° の範囲で逆正接を計算できる。

ディジタル画像処理[改訂第二版] 教科書P116-120

Cannyエッジ検出アルゴリズム

John Canny, “A Computational Approach to Edge Detection”, IEEE Trans. On PAMI, 8(6), 1986.

画像の平滑化(ノイズの低減)

↓

画像の微分(Sobelフィルタ等)

↓

勾配の極大位置の検出

↓

ヒステリシス閾値処理

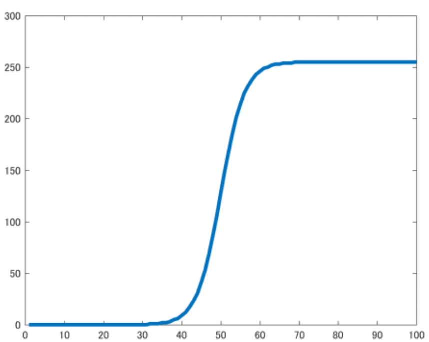
画像の微分(sobel)

勾配の極大位置の検出

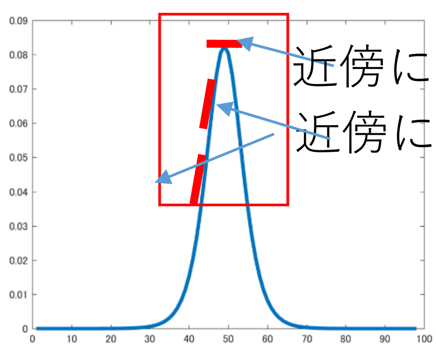
勾配方向の極大位置の検出による細線化

勾配方向の近傍画素の勾配の大きさを確認し，近傍画素より勾配が大きい画素のみエッジとみなす．

一次元イメージ



勾配の高い領域



近傍により高い勾配なし ⇒ 極大値⇒エッジとする

近傍により高い勾配あり ⇒ 非極大値⇒除外する

勾配

勾配方向の極大位置の検出による細線化

勾配方向の近傍画素の勾配の大きさを確認し，近傍画素より勾配が大きい画素のみエッジとみなす．

二次元イメージ

近傍の方が高い

30	30	70
30	50	30
20	30	30

勾配強度

→

30	30	70
30	0	30
20	30	30

削除

近傍の方が低い

30	40	30
30	50	40
20	30	30

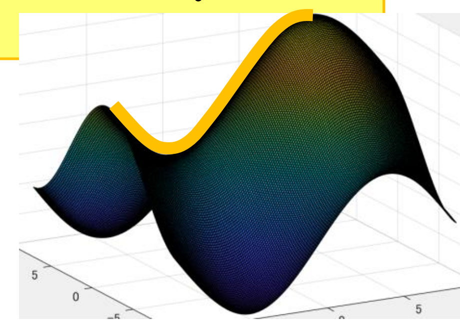
勾配強度

→

30	40	30
30	50	40
20	30	30

残す

山の尾根の部分のみ抽出するイメージ！



エッジ検出(sobel)

細線化

勾配の極大位置の検出

Cannyエッジ検出アルゴリズム

John Canny, “A Computational Approach to Edge Detection”, IEEE Trans. On PAMI, 8(6), 1986.

画像の平滑化(ノイズの低減)

↓

画像の微分(Sobelフィルタ等)

↓

勾配の極大位置の検出

↓

ヒステリシス閾値処理

勾配の極大位置の検出

ヒステリシス閾値

ヒステリシス閾値処理

二段階閾値処理(ヒステリシス閾値処理)により，途切れの少ないエッジを抽出する.

一次元イメージ

STEP1:

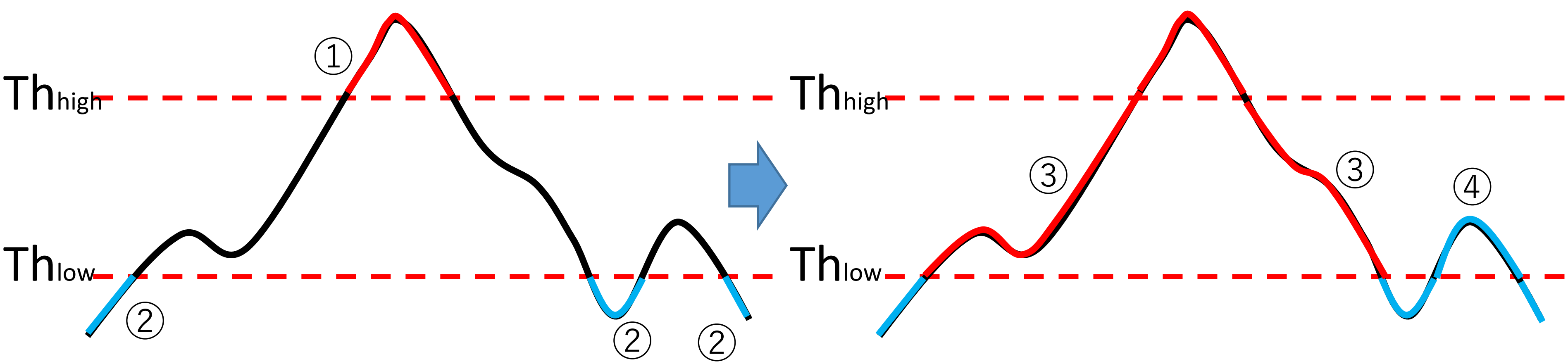
① Th_{high} 以上のエッジ候補はエッジである

② Th_{low} 以下のエッジ候補はエッジでない

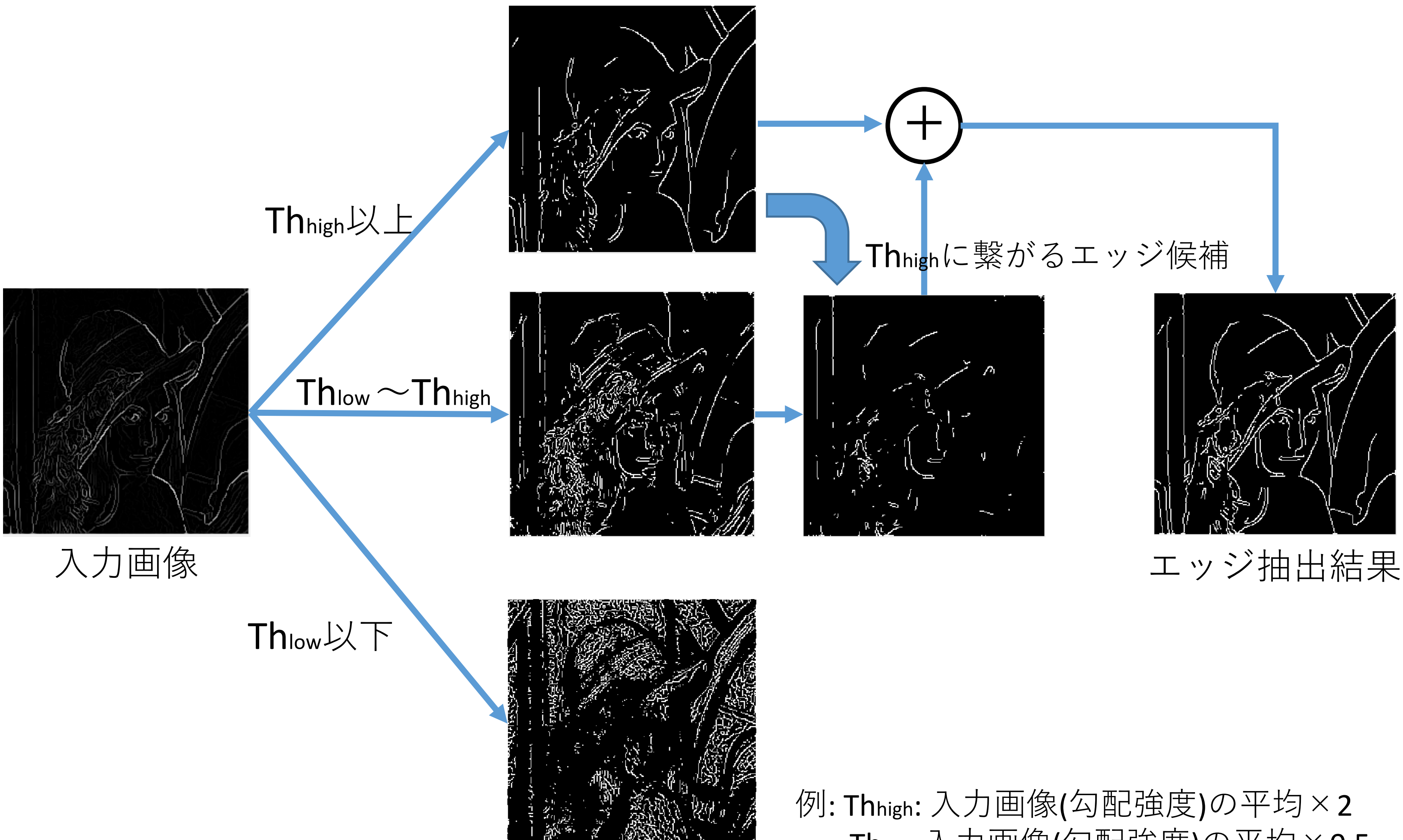
STEP2:

③ Th_{high} に繋がるエッジ候補はエッジである

④ Th_{high} に繋がらないエッジ候補はエッジでない



ヒステリシス閾値処理

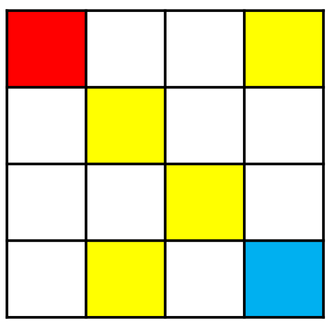


例: Th_{high}: 入力画像(勾配強度)の平均×2
Th_{low}:入力画像(勾配強度)の平均×0.5

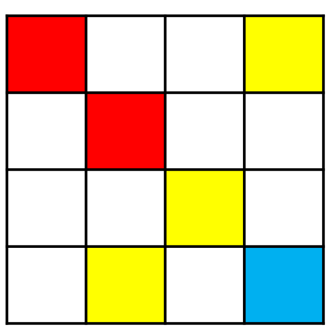
【補足】ヒステリシス閾値処理アルゴリズムのヒント

【アルゴリズム１】

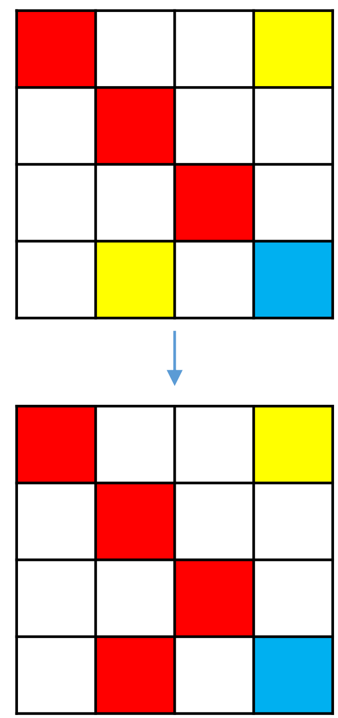
Step1 Th_{high}以上(赤)とエッジ候補(黄色)(Th_{low}〜 Th_{high})をすべて特定しておく.



Step2 Th_{high}以上(赤)の近傍でエッジ候補(黄色)がある場合は，エッジ(赤)とする.



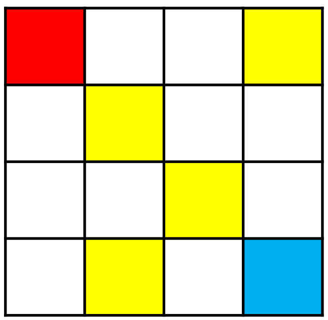
Step3 赤となった画素を含めStep2を変化がなくなるまで繰り返す.



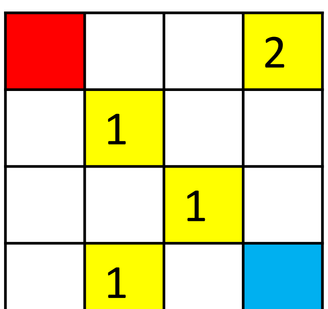
【補足】ヒステリシス閾値処理アルゴリズムのヒント

【アルゴリズム2】

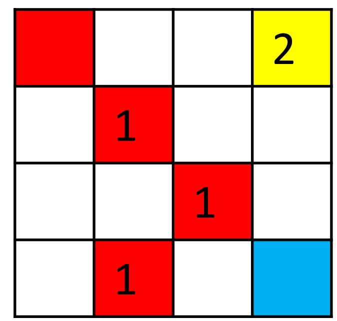
Step1 Th_{high}以上(赤)とエッジ候補(黄色)(Th_{low}〜 Th_{high})をすべて特定しておく.



Step2 エッジ候補(黄色)をラベリング処理する.



Step3 各ラベルで近傍にTh_{high}以上(赤)の画素があればそのラベルはすべてエッジとする.

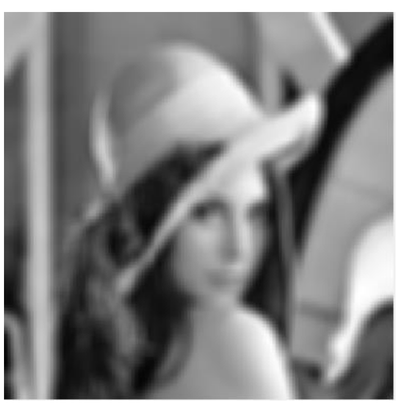


各パラメータでの結果(gaussian)

Gaussian σ＝0.1, [5,5]

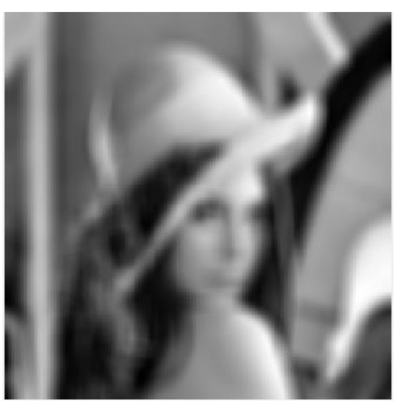
Gaussian σ＝1, [5,5]

Gaussian σ＝2, [5,5]



Gaussian σ＝5, [11,11]

Gaussian σ＝10, [13,13]



各パラメータでの結果（ヒステリシス閾値）

Th_{high}: 勾配強度の平均×2
Th_{low}: 勾配強度の平均×0.5

Th_{high}: 勾配強度の平均×1
Th_{low}: 勾配強度の平均×0.5

Th_{high}: 勾配強度の平均×2
Th_{low}: 勾配強度の平均×1

※ただし，すべてGaussian σ=1, [5,5]

レポート課題

- Cannyエッジ検出アルゴリズムをMATLAB言語で実装せよ.

レポート提出締切：第14回講義前まで(7月14日（金）12:55まで)

レポート提出先：manaba+R

【課題①】 画像の平滑化(ガウシアンフィルタ)をせよ

【課題②】 画像のエッジ検出(Sobelフィルタ)をせよ

【課題③】画像のエッジ検出(Cannyフィルタ)をせよ

【課題④】画像の勾配の極大位置を検出せよ

【課題⑤】 画像のヒステリシス閾値処理をせよ

処理内容(実装工夫した点を含む)及び**実行結果**を記載すること

- MATLAB言語の**プログラム実装部分**をレポート(.pdf)に記載すること
- **自分で撮影した画像**を用いること
- MATLABの**エッジ検出やフィルタリング関数を利用しない(imfiler, edge等)**こと
- MATLABの**画像読み込み、保存関数**は利用してよい

課題が全部できなくても、できた課題までレポートを作成して提出してください。
できた課題まで評価します。

練習問題10－１

数式G(x,y)=A*exp(-(x*x+y*y)/2)を用いて2次元ガウス関数のフィルタを設計する。フィルタの値は全て整数で８ビットで表現する。フィルタ中心点におけるフィルターの値を幾らとすべきか。フィルタのサイズ（2W+1）×（2W+1）におけるWを決めよ。
ただし、 exp(-1/2)≒0.6, exp(-2)≒0.135 , exp(-4.5)≒0.0111, exp(-8)≒0.0003。

練習問題10－２

Cannyフィルターは正方形が良いか、長方形が良いか。具体的数字例を用いてその理由を述べよ。

Cannyフィルターは、ガウス平滑化と微分を一つのフィルタで実現しているが、これは、ガウス平滑化のフィルターを適用したあとで再度微分フィルターを適用するよりも良い理由を述べよ。