

情報科学 【AI・データサイエンス】

第7回 画像解析

画像解析の概要

基本的な画像解析 1 : フィルタ処理

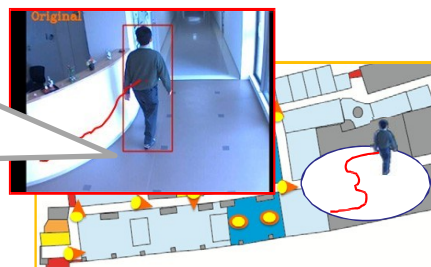
基本的な画像解析 2 : 2 値化

画像解析の概要

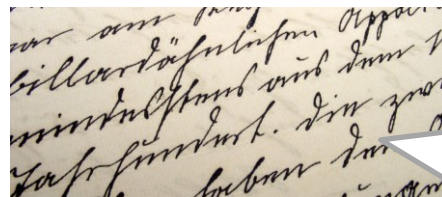
画像：理系・文系分野で問わず，有用なデータ

- 様々な現象・物体の状況・形状・広がり・個数などを調べたいときに便利

ショッピングモールでの
客の移動を解析して
店舗配置をしたい



文書を
デジタル化して
分析したい



野菜の生育状況に
異常がないか
手間なく気づきたい



ロボットに
自動で
片づけてほしい



画像（2次元）

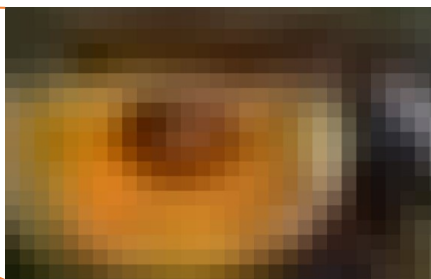
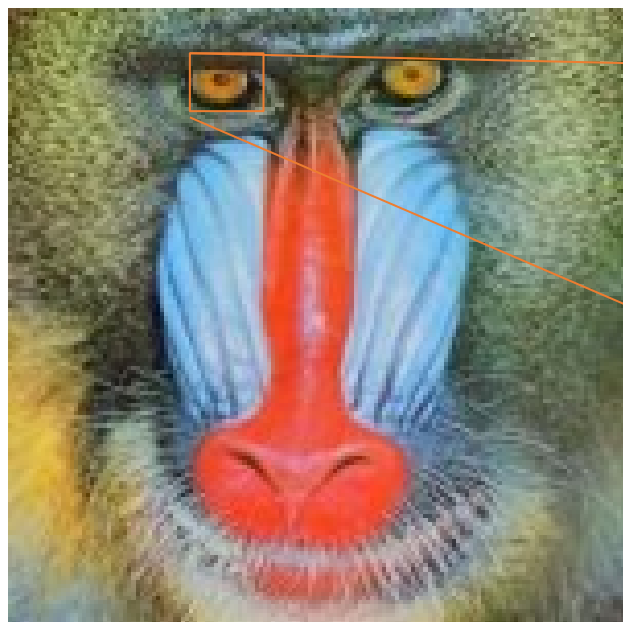
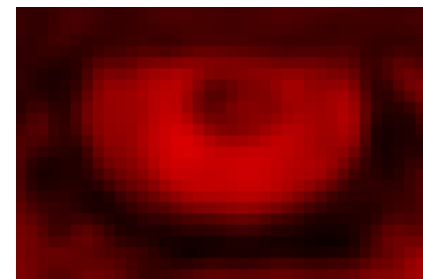
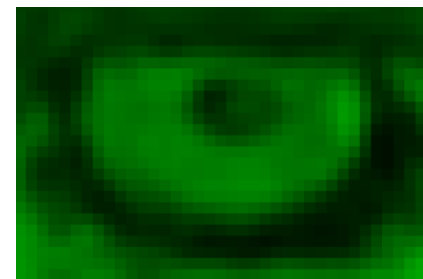
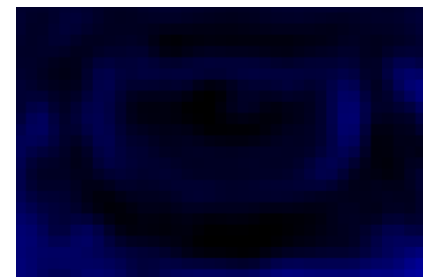
2次元上に数値を規則的に配置したデータ

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

「白 or 黒」の2値の画像
(8x8=64画素)

画像（2次元）

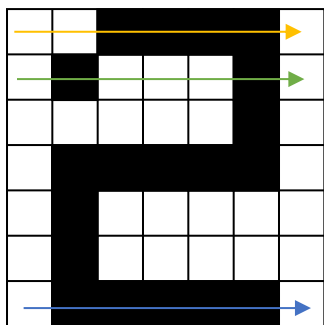
カラー画像は赤・緑・青に分解される

 $=$  $+$  $+$ 

画像ってデータ？

Yes！ 数値データのカタマリ

- 2値画像の例

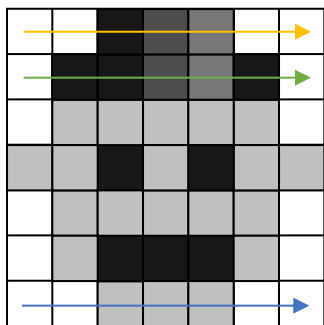


7×7画素

$$(1, 1, 0, 0, 0, 0, 1, 1, 0, \dots, 0, 1)$$

49個の0か1の組み合わせ
(49次元ベクトル)

- グレースケール画像(灰色もある画像)の例

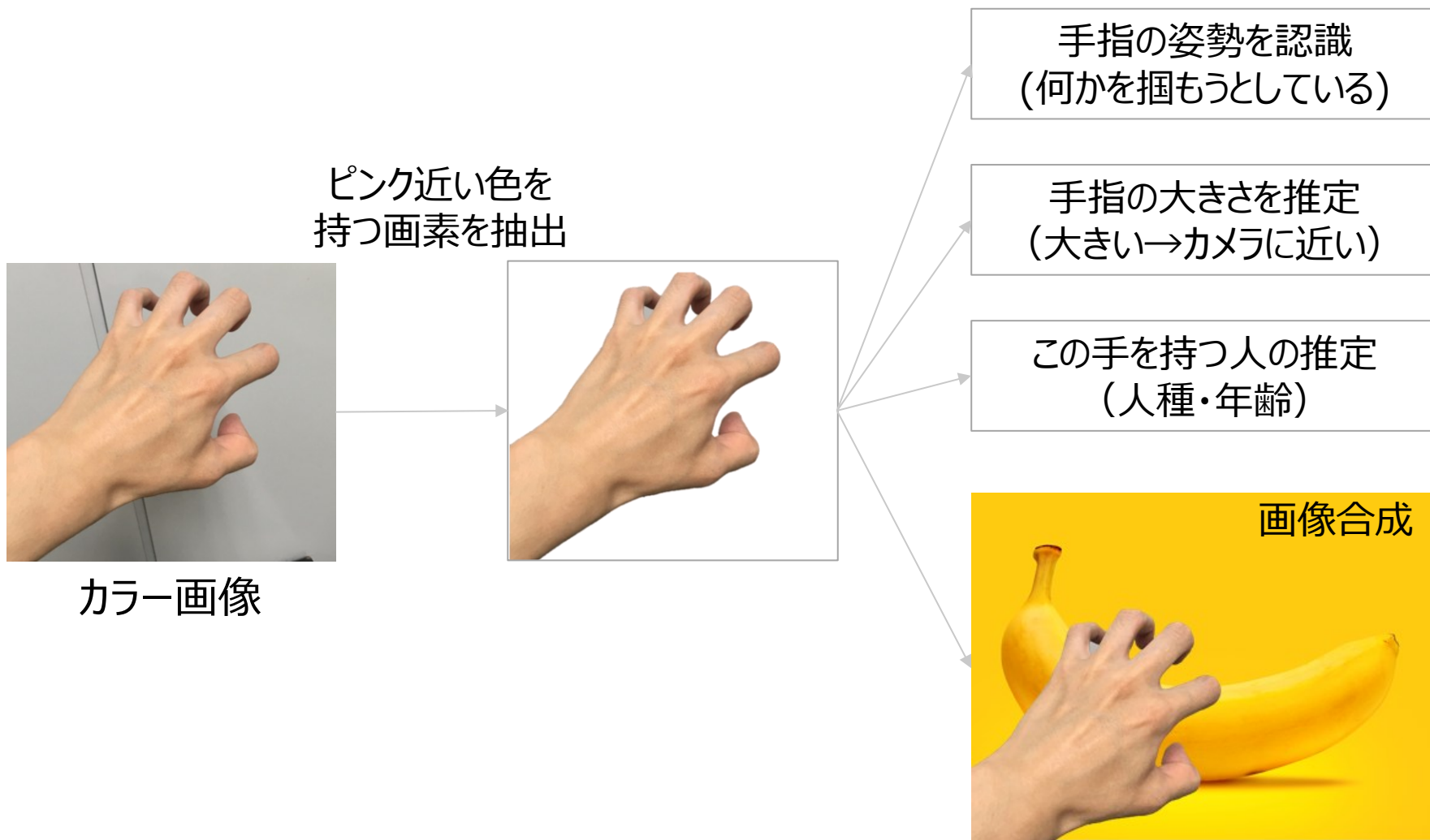


7×7画素

$$(255, 245, 10, 35, 92, 231, 254, \dots, 249)$$

49個の0～255の組み合わせ
(49次元ベクトル)

画像解析の目的は様々 例えば...



画像認識

そこに写っているものは何かを当てる



正解→

GT: coucal

1: coucal

2: indigo bunting

3: lorikeet

4: walking stick

5: custard apple

認識結果
(上位5位)



GT: komondor

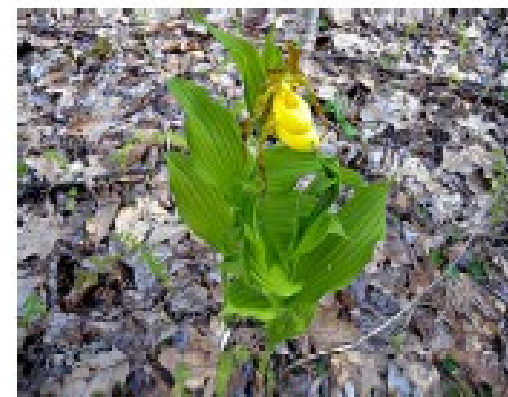
1: komondor

2: patio

3: llama

4: mobile home

5: Old English sheepdog



GT: yellow lady's slipper

1: yellow lady's slipper

2: slug

3: hen-of-the-woods

4: stinkhorn

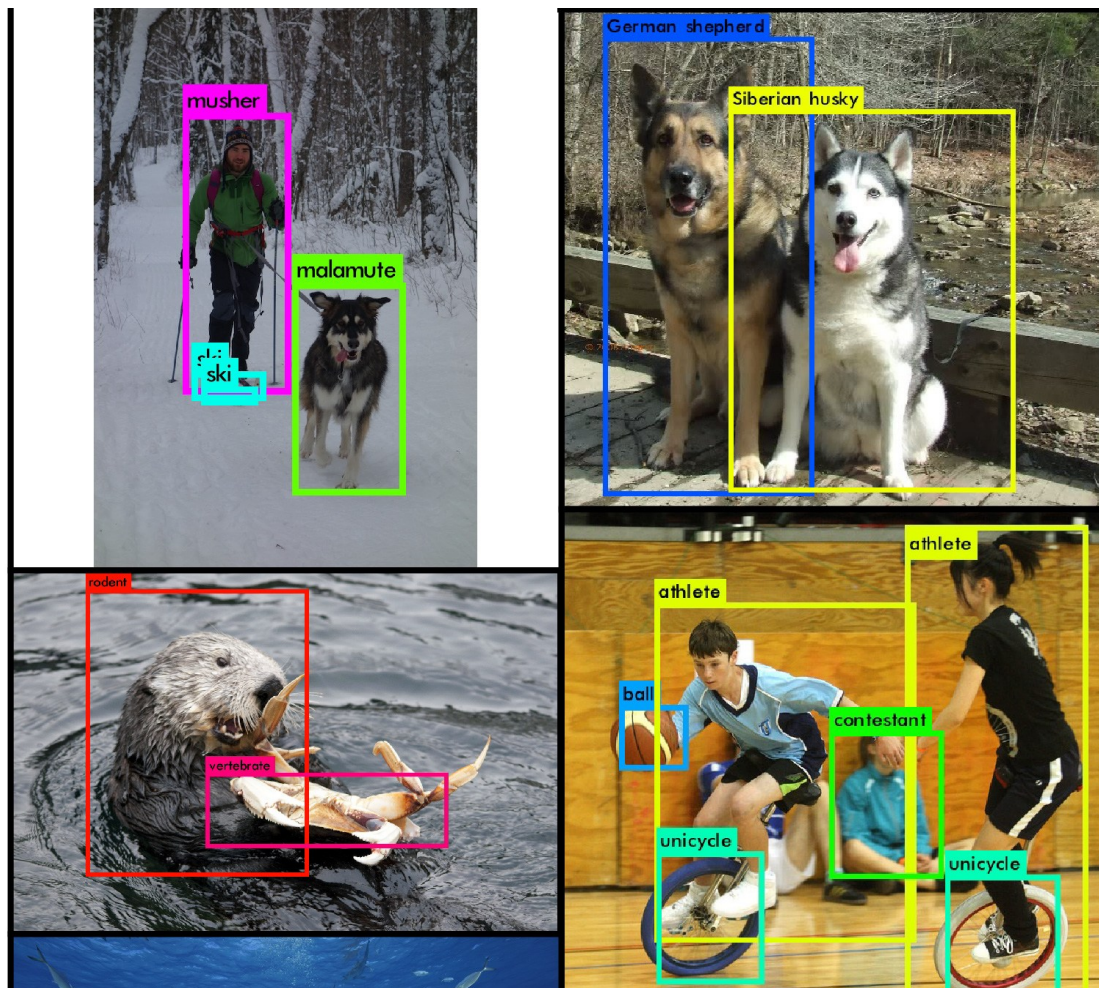
5: coral fungus

人間よりも
当てられてる？



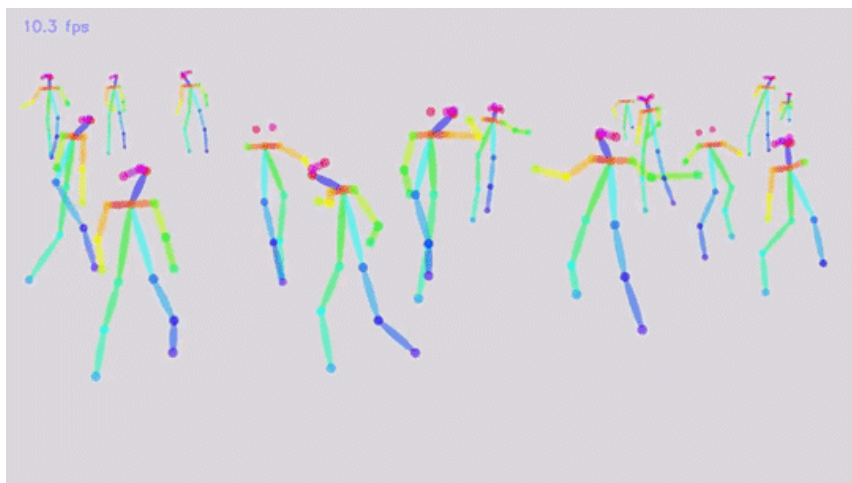
物体検出

9000種類の物体検出



[Redmon&Farhadi, "YOLO9000: Better, Faster, Stronger", CVPR2017]

人体骨格抽出 OpenPose



バイオ系の方は、DeepLabCutも要チェック

<https://github.com/CMU-Perceptual-Computing-Lab/openpose>

九州大学 数理・データサイエンス教育研究センター

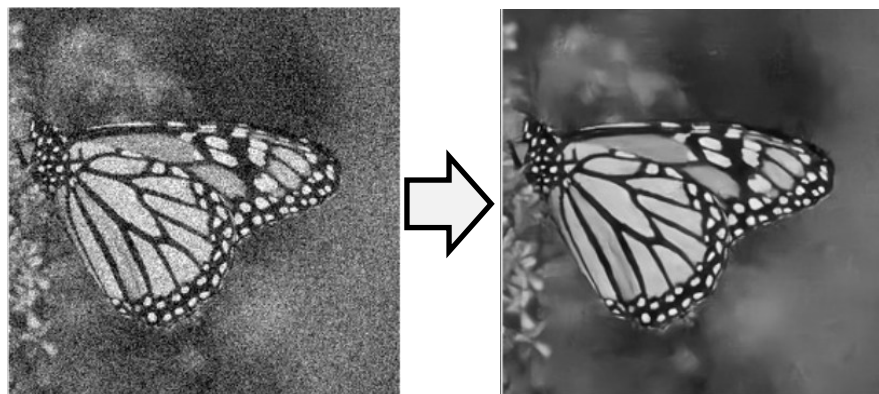
画像の領域分割 Segment Anything Model



<https://ai.facebook.com/blog/segment-anything-foundation-model-image-segmentation/>

様々な画像変換

- 高度なノイズ除去



[Mao+, NIPS2016]

- 超解像

- 低解像度画像の高解像度化



<https://towardsdatascience.com/image-super-resolution-an-overview-of-the-current-state-of-research-94294a77ed5a>

- 自動着色



(a) Colorado National Park, 1941

(b) Textile Mill, June 1937

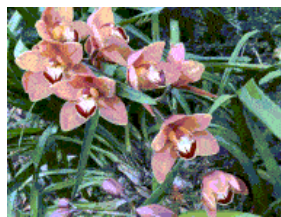
(c) Berry Field, June 1909

(d) Hamilton, 1936

[Iizuka+, TOG 2016]

複数静止画からの3次元再構成

- NeRF (2020)



[Mildenhall+, ECCV2020]

- Gaussian Splatting (2023)



<https://aras-p.info/blog/2023/09/27/Making-Gaussian-Splats-more-smaller/>



Jurdein@Wikimedia Commons

画像生成（拡散モデル）

「こういう画像を作って」という指示に応じて画像を作る

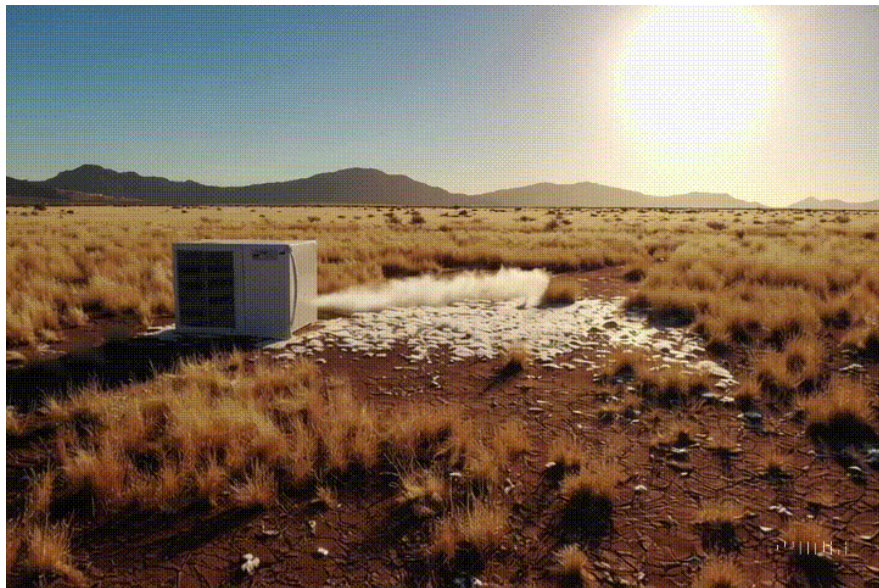


Kreis+, "Latent Diffusion Models," NeurIPS2023 tutorial

動画生成

「こういう動画を作って」という指示に応じて動画を作る

- プロンプト：「アフリカの大地を巨大なエアコンが徐々に冷やしている」



- プロンプト：「カピバラの群れがライオンを追いかけている」



[Open AI社 Soraで生成(2025年9月)]

非常に基本的な画像解析 1 : フィルタ処理

フィルタ処理とは

次の2画像を見比べてみましょう

before



after



Microsoft スtock画像

- ボケたけど、おかげで細かい「シワ」は目立たなくなったかも
- これを画像の「平滑化」と呼びます

フィルタ処理とは？ 画像の加工法

● 平滑化

- 画像に含まれる不要な濃淡変動を軽減できる（前スライド）

● エッジ抽出

- エッジ = 画像中の明るさが急激に変化する部分
- 物体境界の候補や輪郭線を抽出することができる



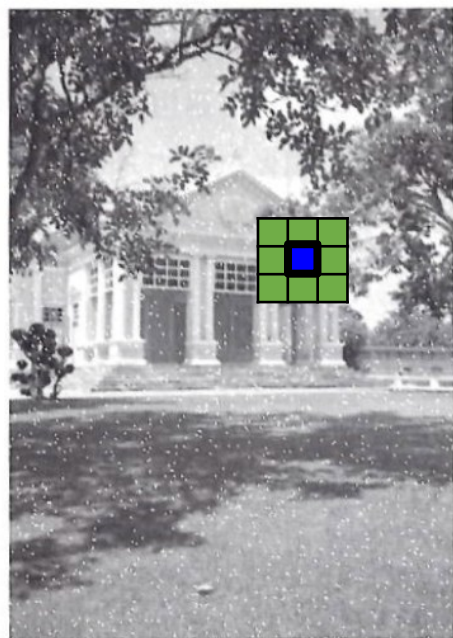
● 画像の鮮鋭化

- メリハリをつける



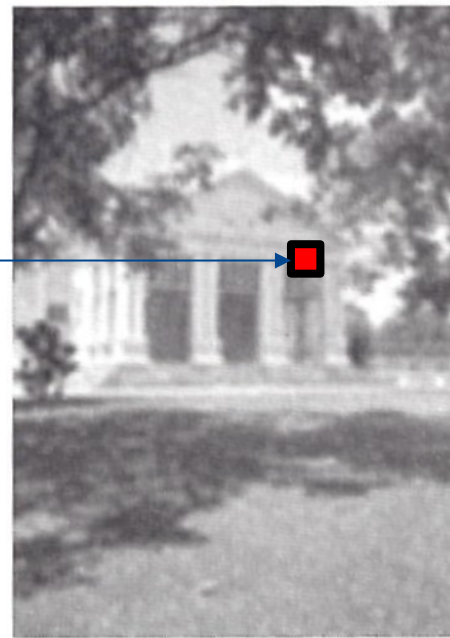
出典：CG-ARTS協会 デジタル画像処理

フィルタ処理の基本的考え方：ある画素の値（明るさ）をその付近の画素値を用いて変更



入力画像

何かしらの
計算

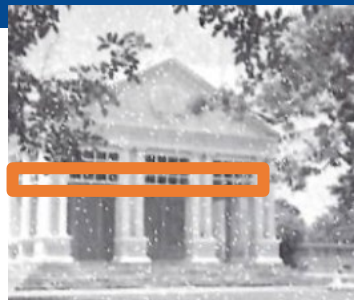


出力画像

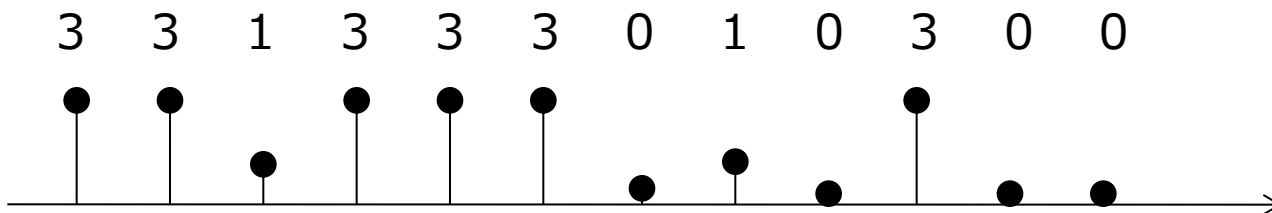
- これで平滑化やエッジ抽出ができる！
- 具体的には？ → 次スライド以降

フィルタ処理（単純化した例）（1/2）

本当は画像は
2次元だけど、
簡単のため、
1次元で考える

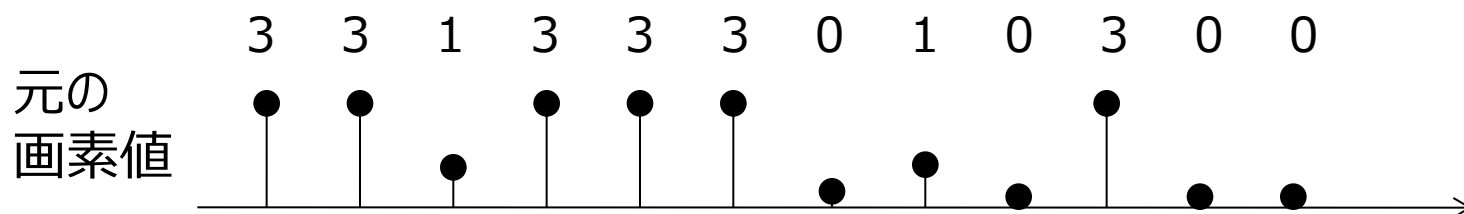
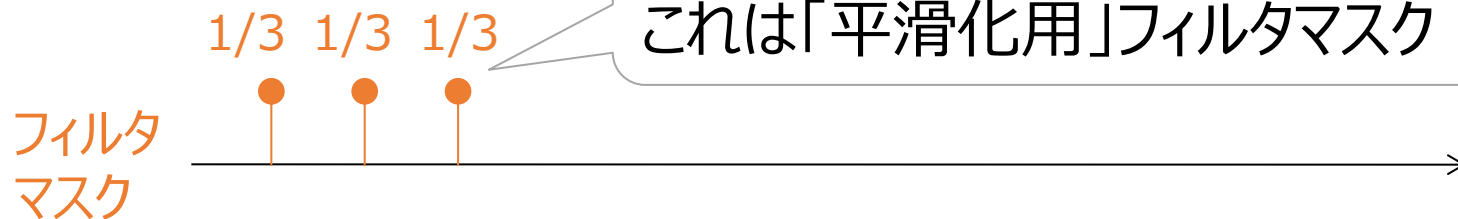


元の
画素値

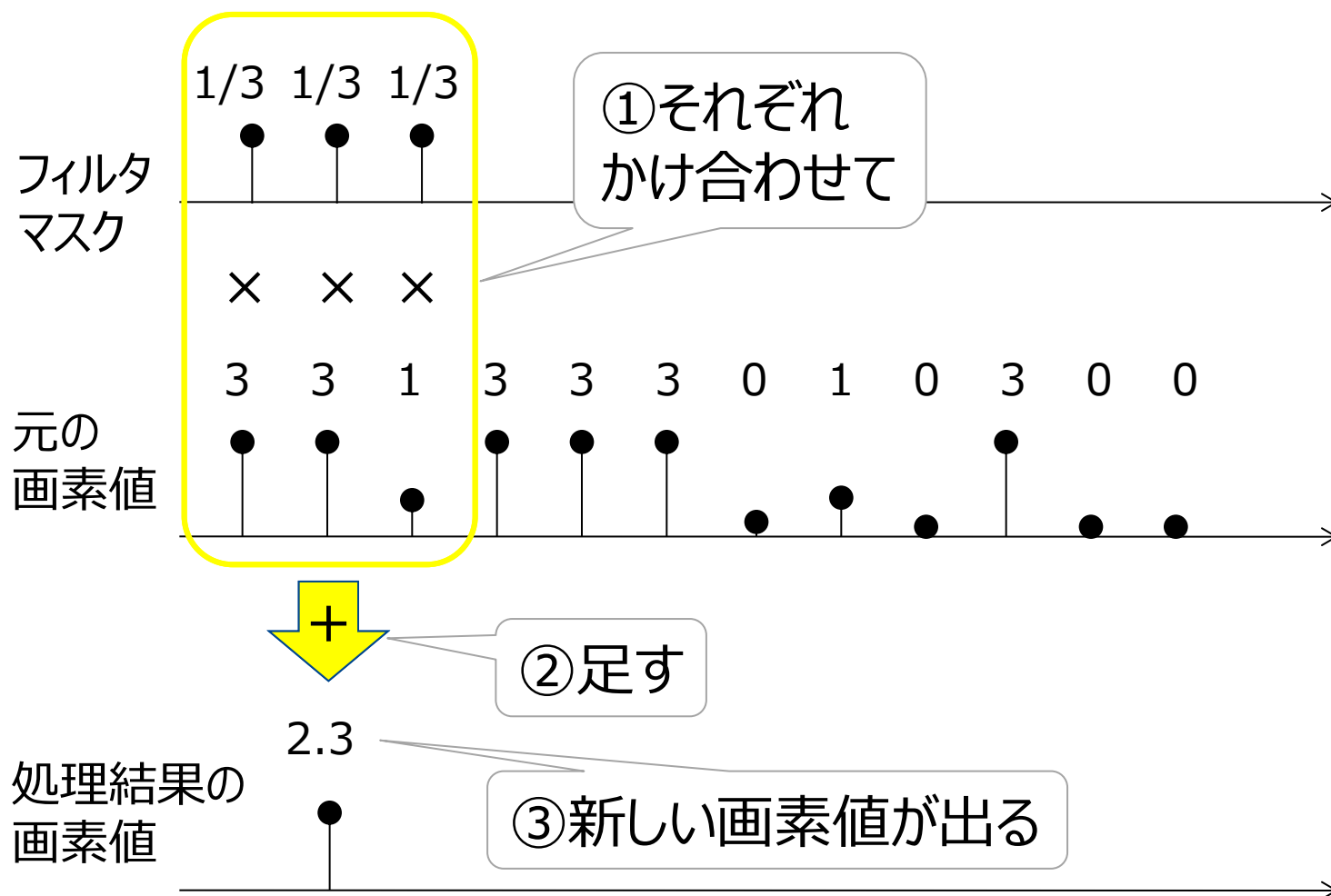


フィルタ処理（単純化した例）（1/2）

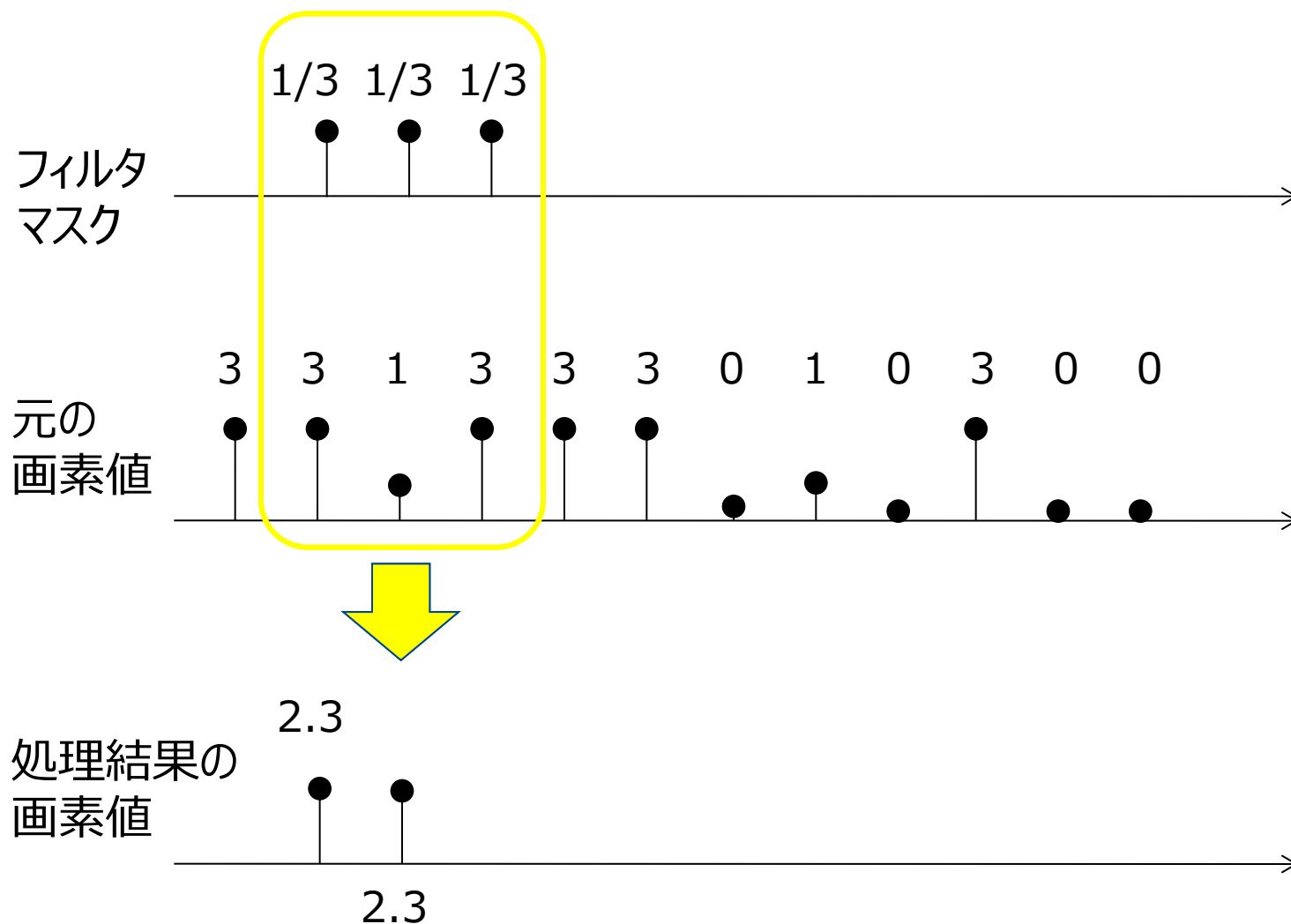
これがフィルタ処理結果を決める！
これは「平滑化用」フィルタマスク



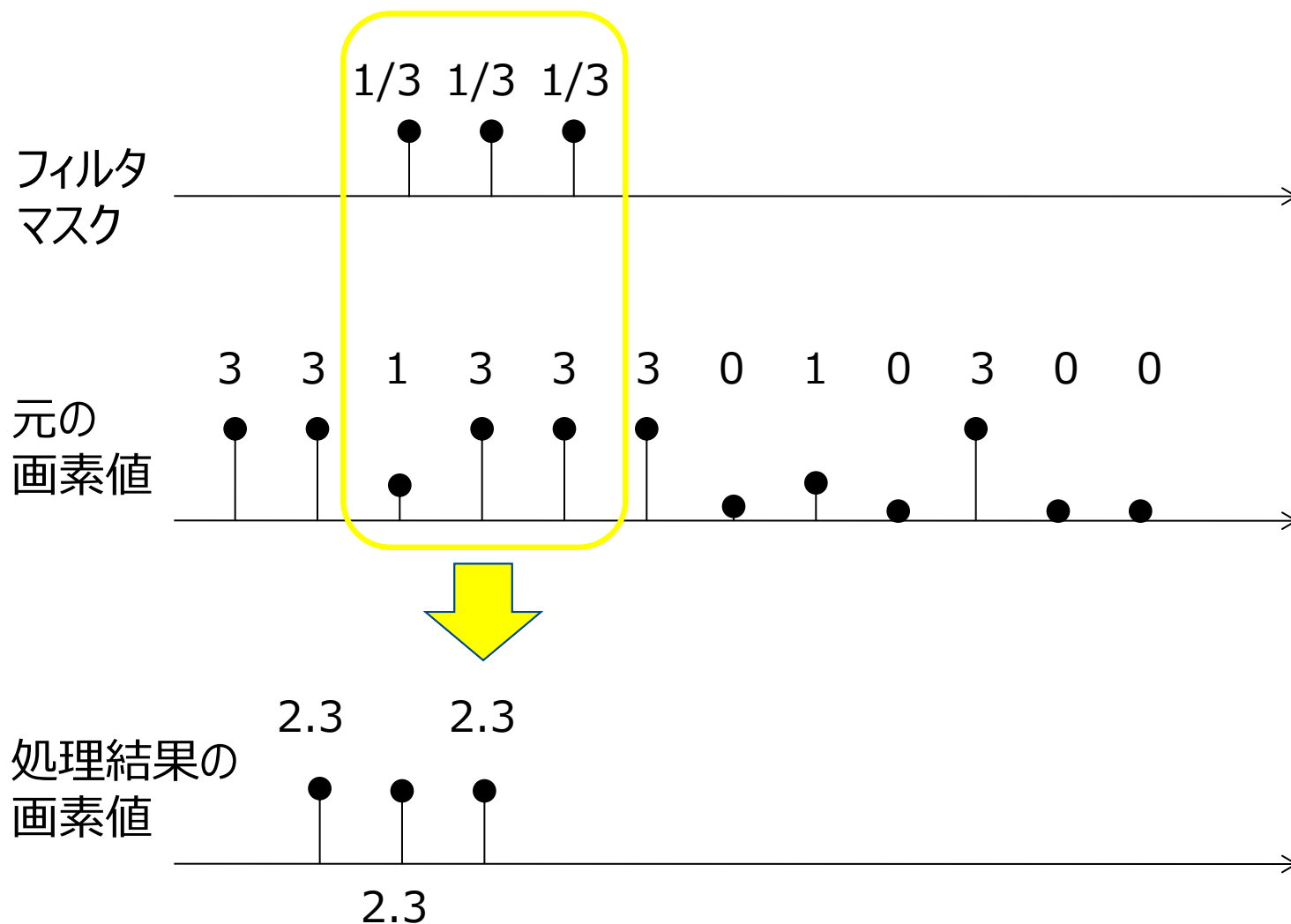
フィルタ処理（単純化した例）（1/2）



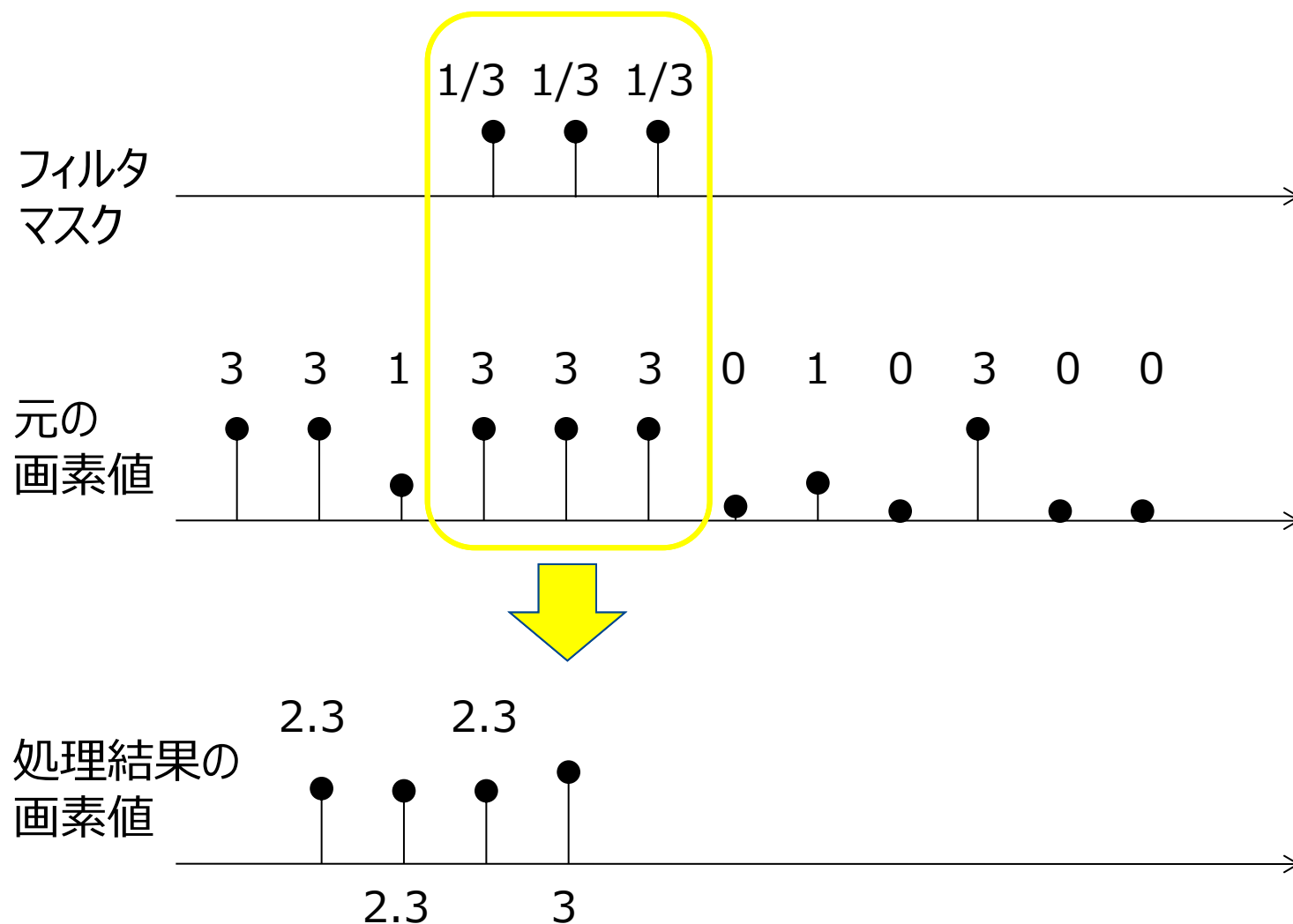
フィルタ処理（単純化した例）（1/2）



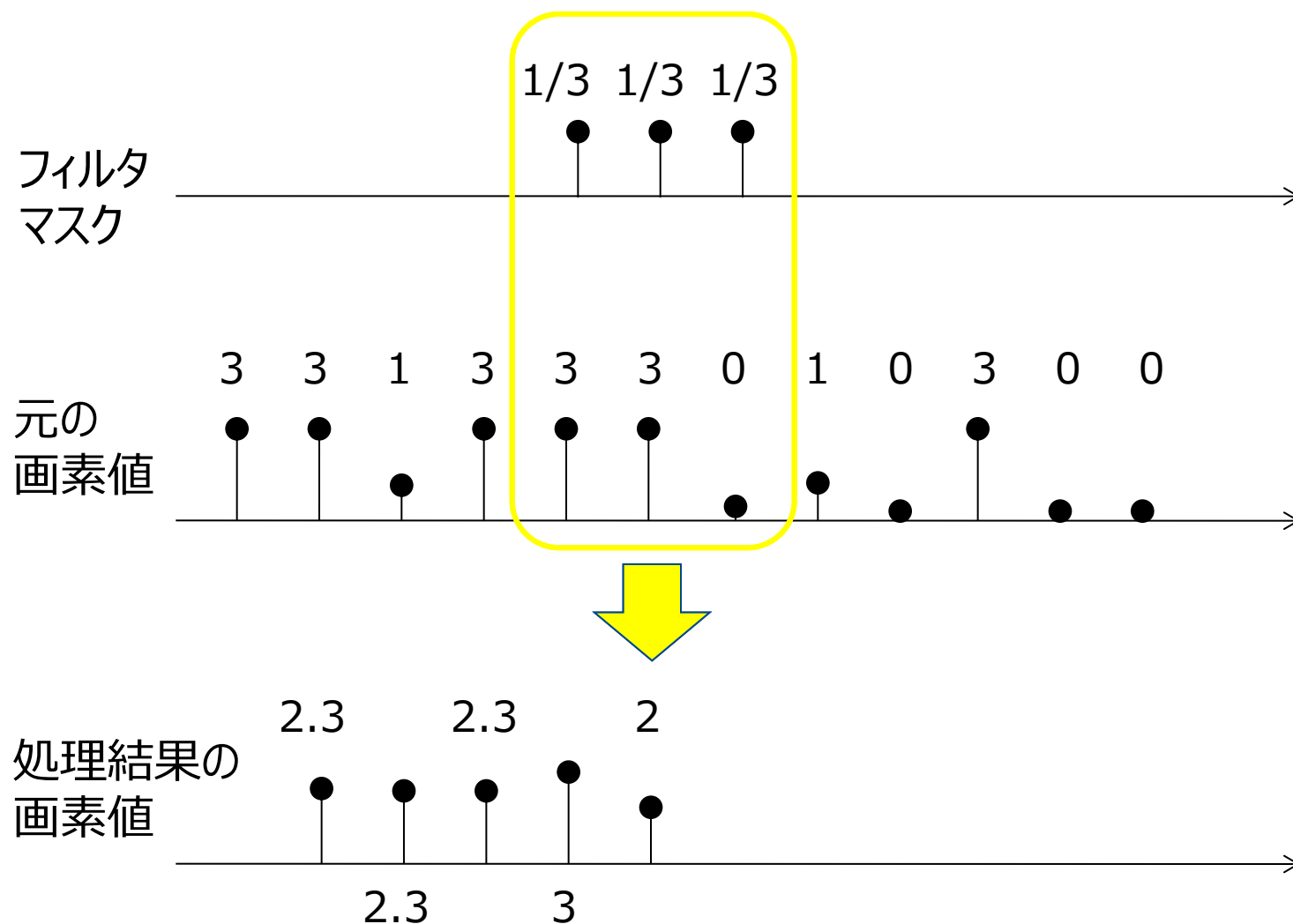
フィルタ処理（単純化した例）（1/2）



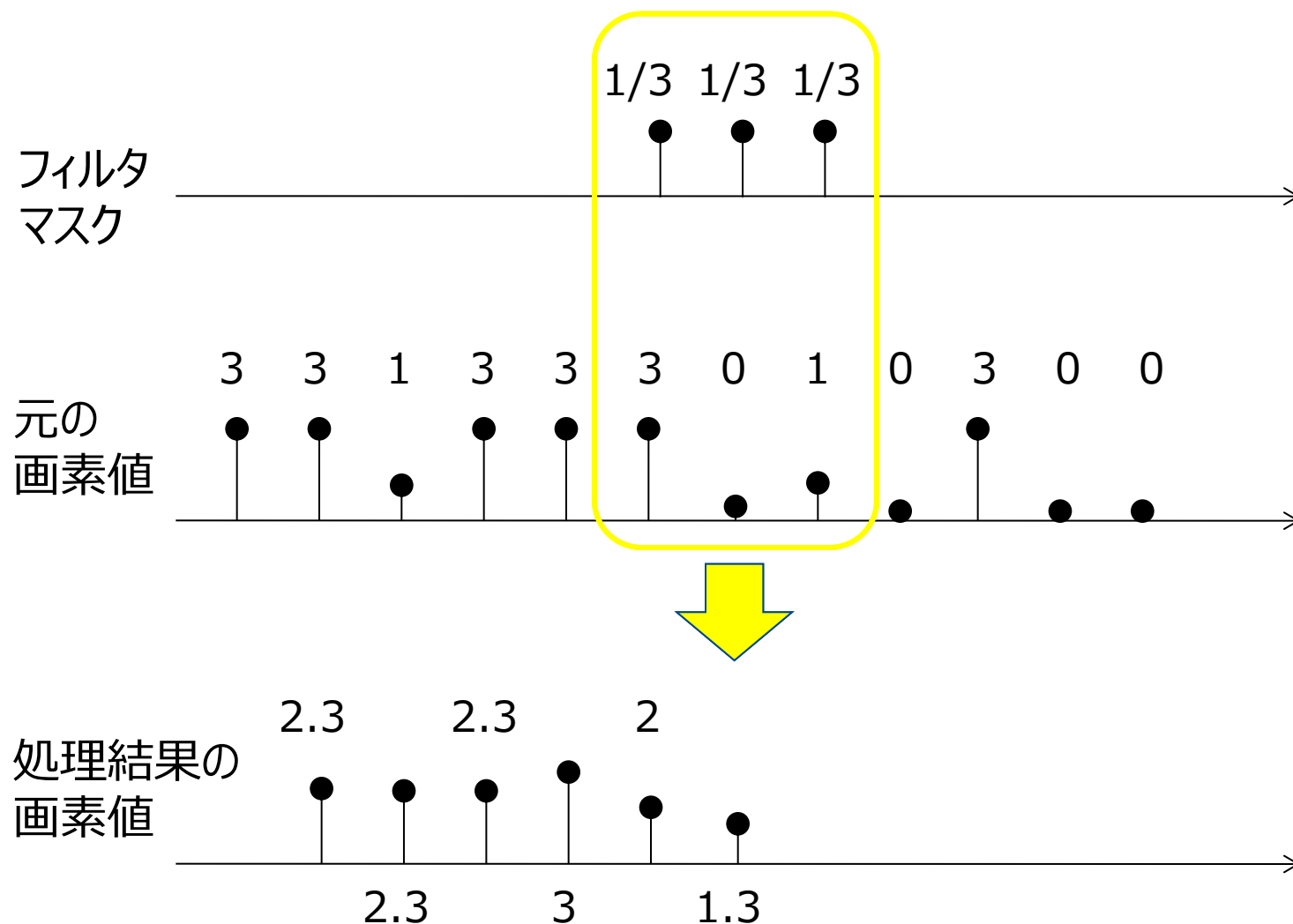
フィルタ処理（単純化した例）（1/2）



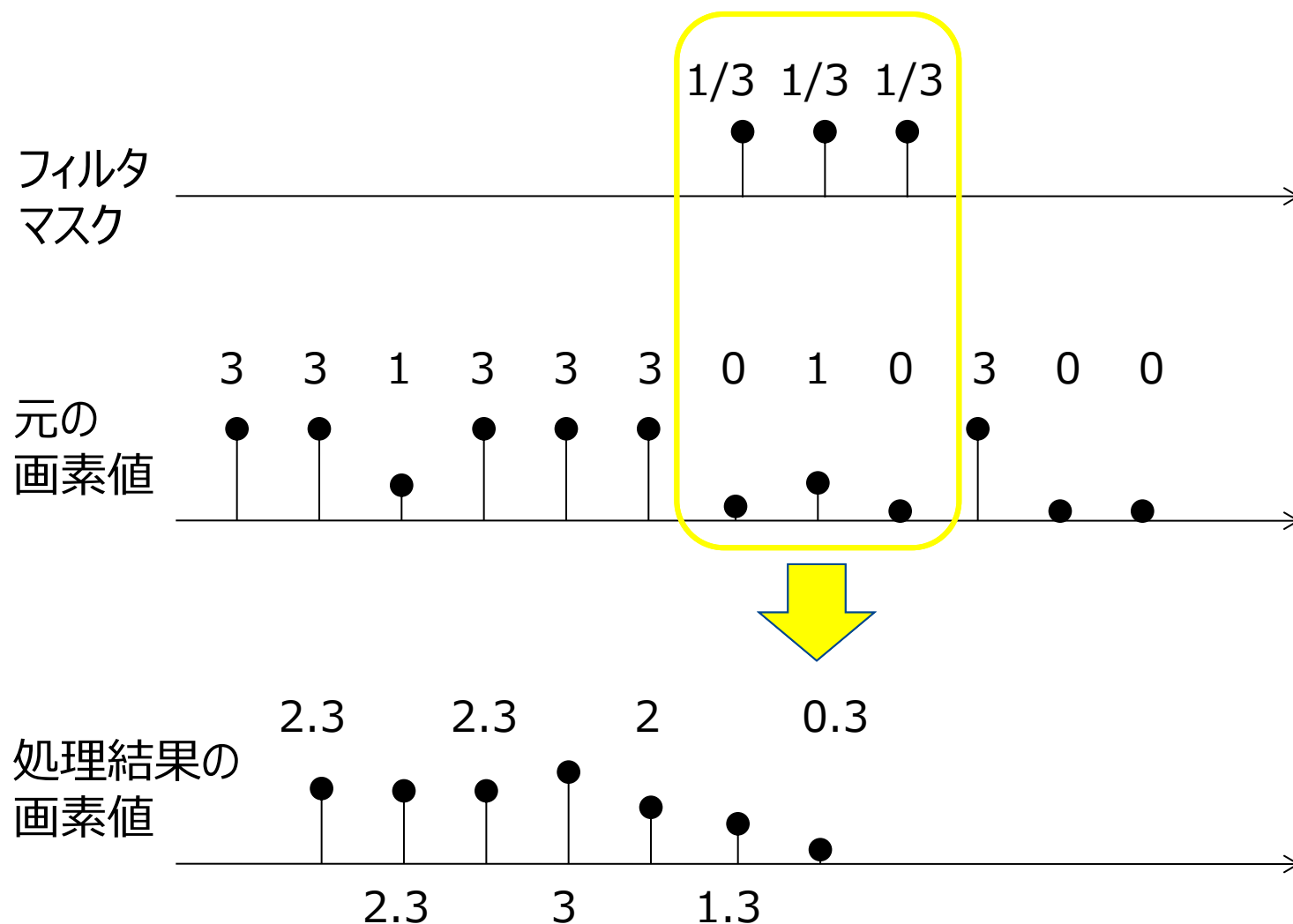
フィルタ処理（単純化した例）（1/2）



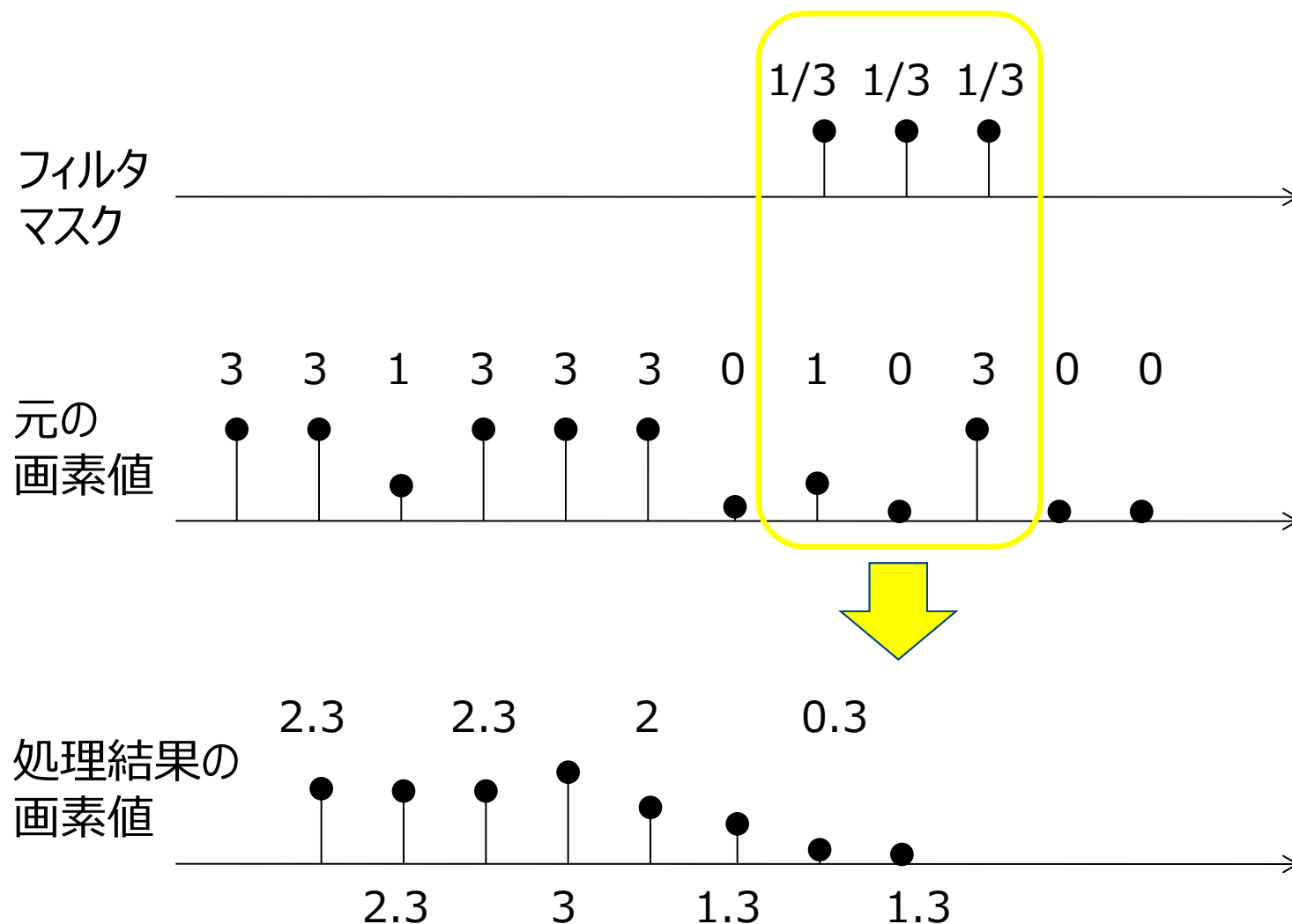
フィルタ処理（単純化した例）（1/2）



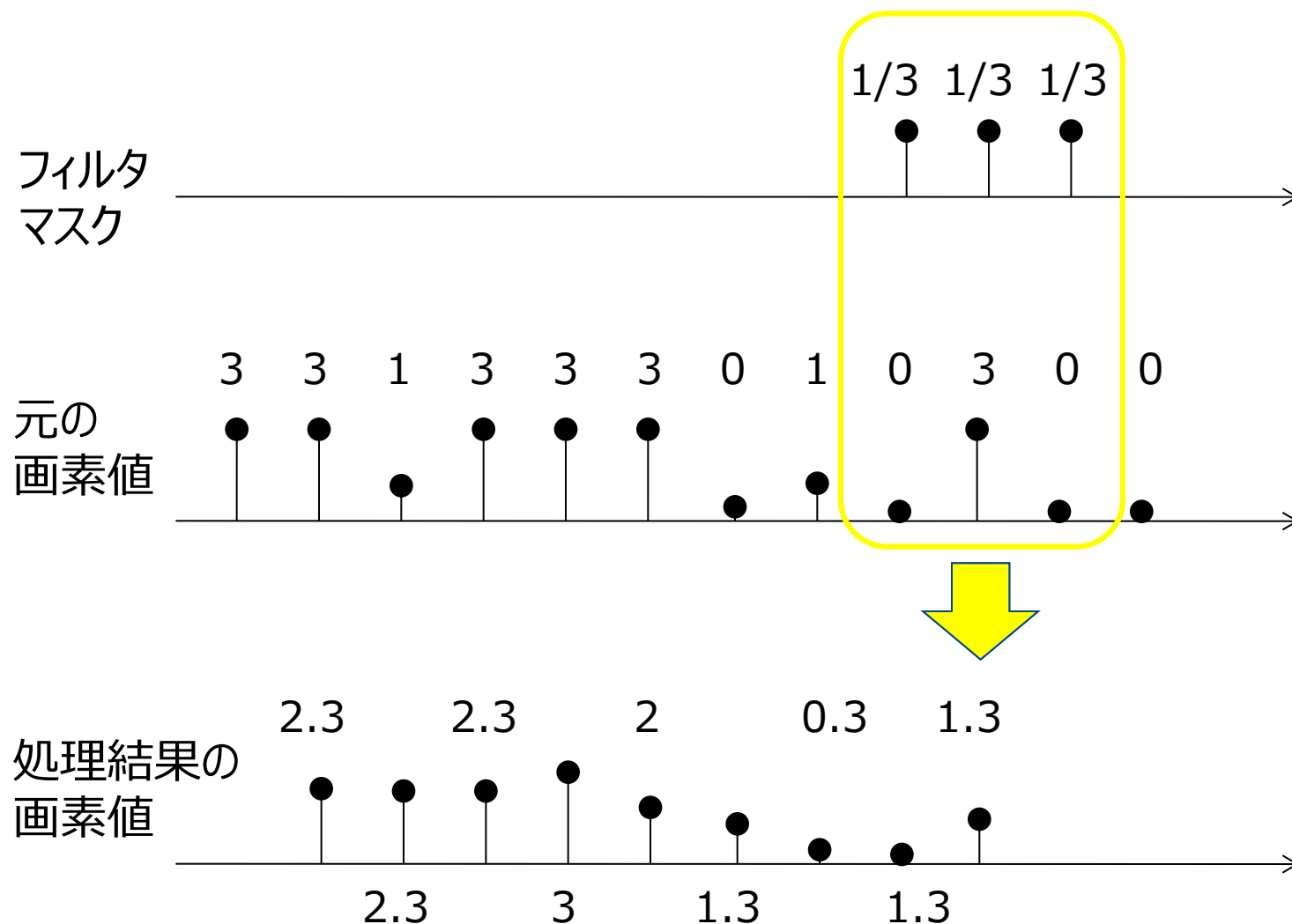
フィルタ処理（単純化した例）（1/2）



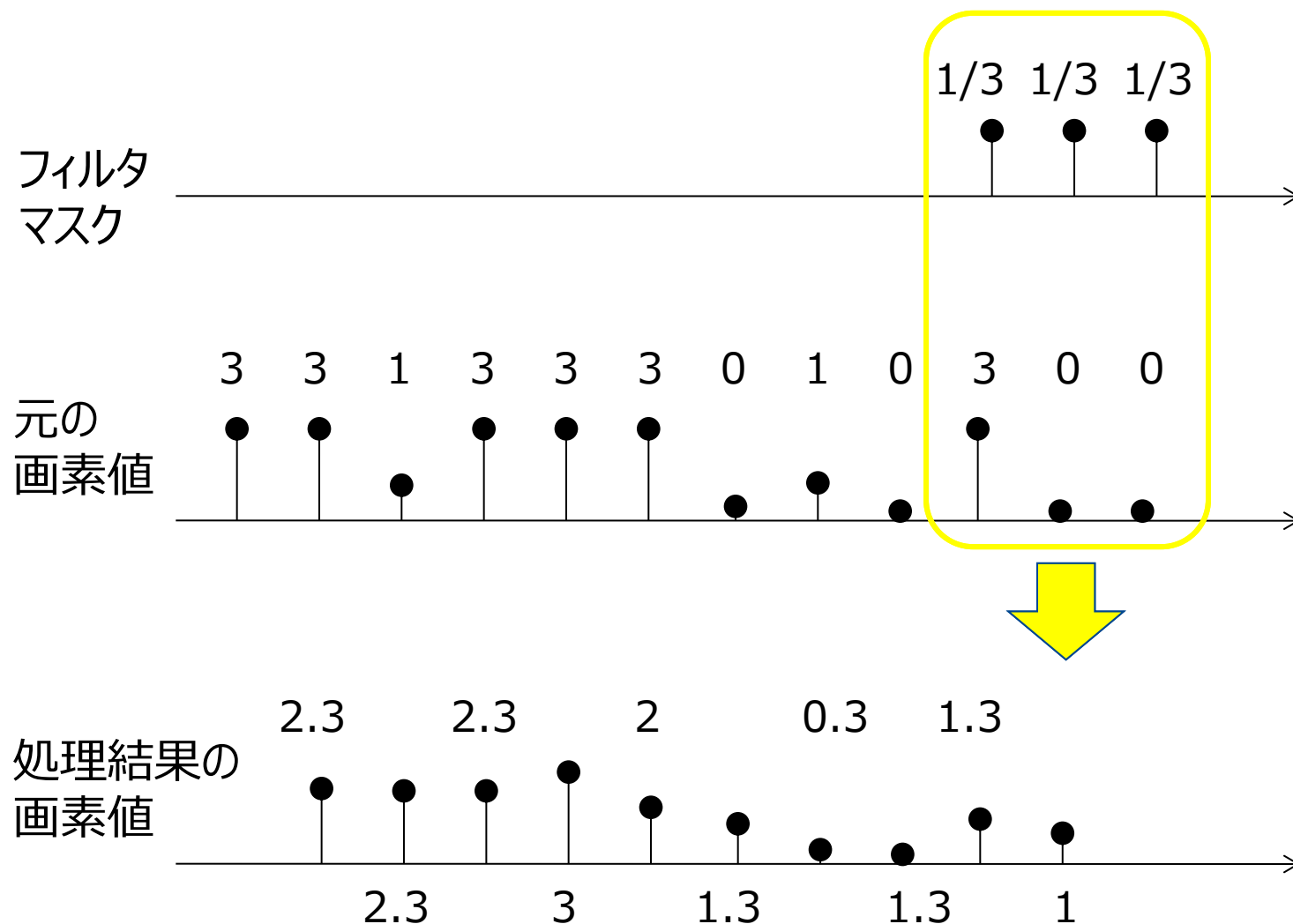
フィルタ処理（単純化した例）（1/2）



フィルタ処理（単純化した例）（1/2）

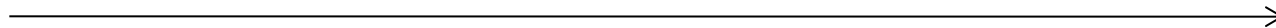


フィルタ処理（単純化した例）（1/2）

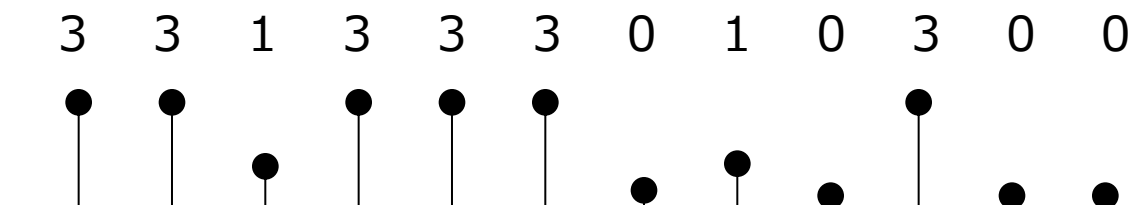


フィルタ処理（単純化した例）（1/2）

フィルタ
マスク

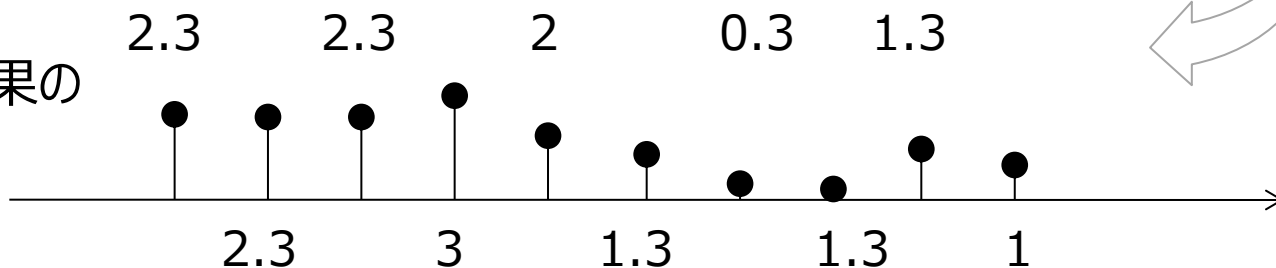


元の
画素値



元の画素値に比べ、変化が滑らかに（平滑化）！

処理結果の
画素値



フィルタ処理（単純化した例）（1/2）

このマスク, 実は, 結局3つの画素値の平均をとっていることになる

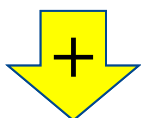
フィルタ
マスク

$1/3$ $1/3$ $1/3$

× × ×

3 3 1 3 3 3 0 1 0 3 0 0

元の
画素値



2.3

$$=(3+3+1)/3$$

要するに, 局所的に
平均をとることで,
画素値が均(なら)されて
滑らかになる

フィルタ処理（単純化した例）（2/2）

このようにマスクを変えると、
結果も変わる(これはエッジ抽出用)

フィルタ
マスク

$-1/2$ 1 $-1/2$

元の
画素値

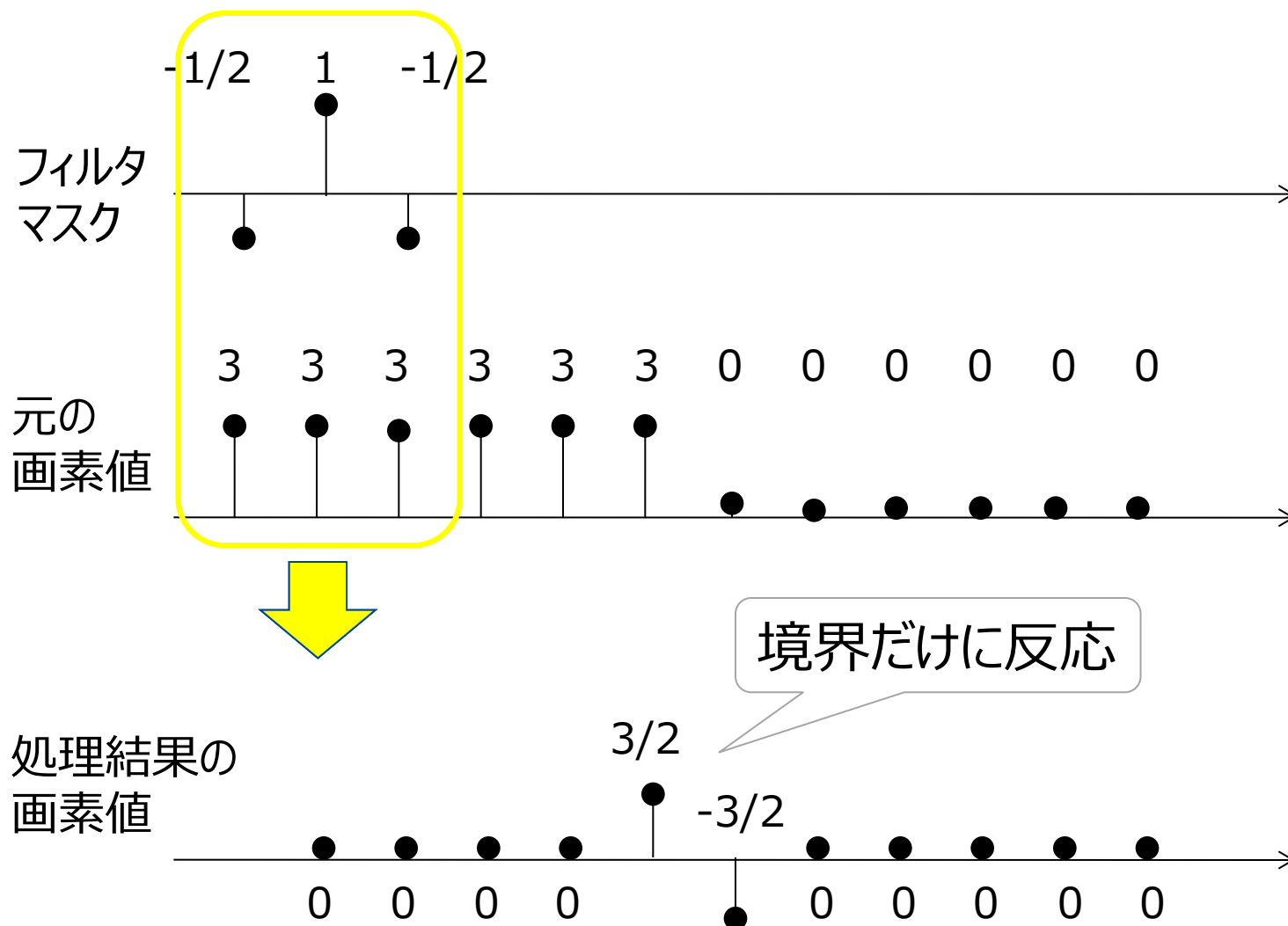
3 3 1 3 3 3 0 1 0 3 0 0

どうなるかやってみよう(時間があれば)

処理結果の
画素値

フィルタ処理（単純化した例）（2/2）

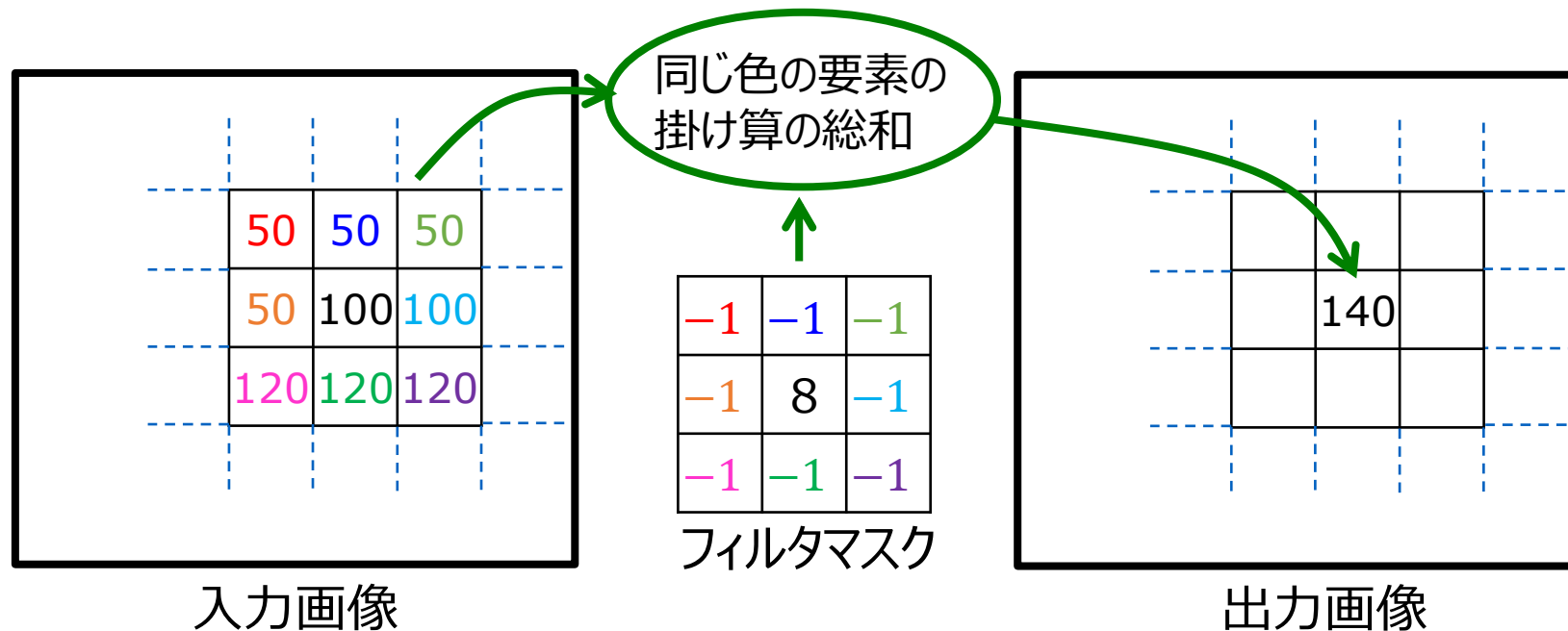
こちらのケースだともっとわかりやすい



実際の画像用のフィルタは、2次元の

- ただし「やること」はこれまでの「単純化した例」と同じ

$$\begin{array}{rclcl}
 50 \times (-1) & + & 50 \times (-1) & + & 50 \times (-1) & + \\
 50 \times (-1) & + & 100 \times (8) & + & 100 \times (-1) & + \\
 120 \times (-1) & + & 120 \times (-1) & + & 120 \times (-1) & = 140
 \end{array}$$



- 同様の処理を位置をずらしながら画像全体にわたって行う

様々なフィルタ処理

平均化フィルタ

- フィルタによって覆われる領域内の画素値の平均を求めることと等価

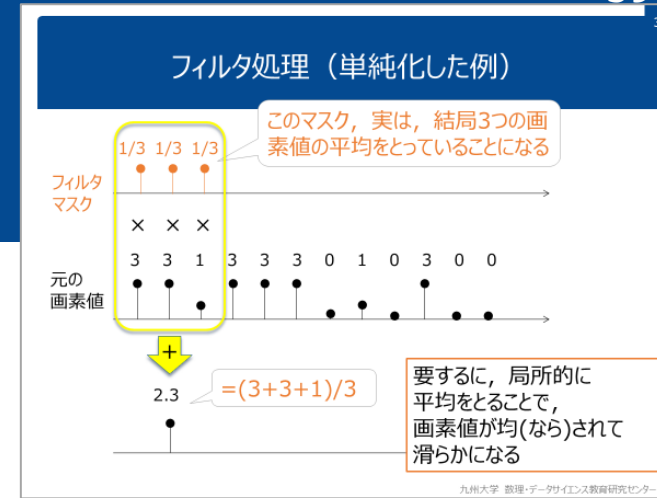
平均化フィルタの例：

$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$
$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$
$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$

3×3画素

$\frac{1}{25}$	$\frac{1}{25}$	$\frac{1}{25}$	$\frac{1}{25}$	$\frac{1}{25}$
$\frac{1}{25}$	$\frac{1}{25}$	$\frac{1}{25}$	$\frac{1}{25}$	$\frac{1}{25}$
$\frac{1}{25}$	$\frac{1}{25}$	$\frac{1}{25}$	$\frac{1}{25}$	$\frac{1}{25}$
$\frac{1}{25}$	$\frac{1}{25}$	$\frac{1}{25}$	$\frac{1}{25}$	$\frac{1}{25}$
$\frac{1}{25}$	$\frac{1}{25}$	$\frac{1}{25}$	$\frac{1}{25}$	$\frac{1}{25}$

5×5画素



平均化フィルタの結果

- 濃淡変動が滑らかになるが、画像全体がぼける



入力画像



平均化フィルタの結果



出典：CG-ARTS協会 デジタル画像処理

演習課題： 「手計算でやってみる」画像の平滑化

- 以下の入力画像に対して
3×3の平均化フィルタを適用した結果画像を求めなさい

0	0	3	3	3
0	0	3	3	3
0	0	3	3	3
0	0	3	3	3
0	0	3	3	3

入力画像

0	0	3	3	3
0		?		3
0				3
0				3
0				3
0	0	3	3	3

出力画像

エッジ抽出の結果

トリッキーですが、2つフィルタマスクを使う



入力画像



エッジ抽出結果

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

垂直エッジ
検出用マスク

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

水平エッジ
検出用マスク



エッジ抽出結果
(拡大)

鮮鋭化フィルタ

- エッジ部分の両側で，明るい部分をより明るく，暗い部分をより暗くし，エッジの傾斜を急にする

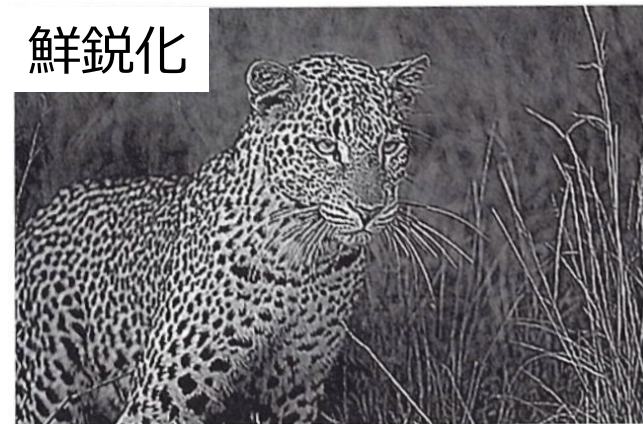
入力画像



0	1	0
1	-5	1
0	1	0

鮮鋭化フィルタ

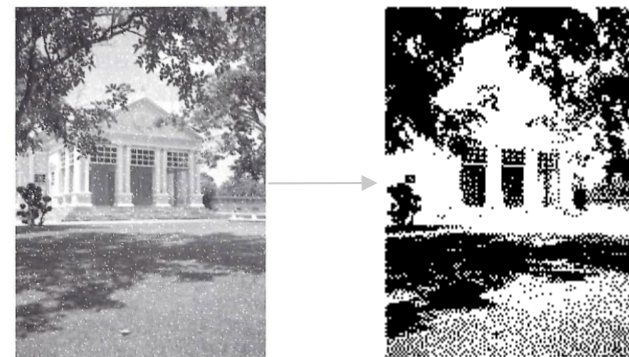
鮮鋭化



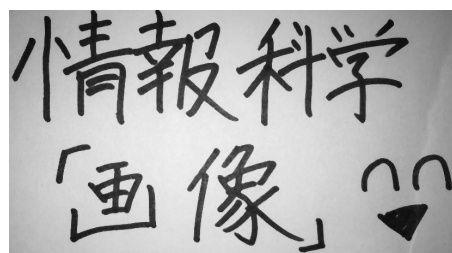
非常に基本的な画像解析 2 : 2値化

二値化とは

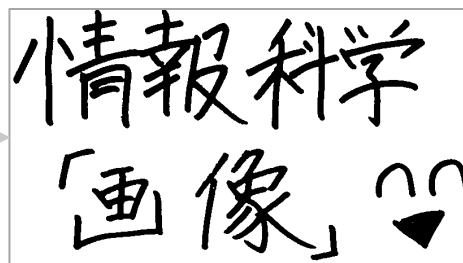
- 画像を二値画像に変換する処理
 - 画像中の様々なものの領域や形が明確になる



- 応用例：文書からの文字データ取得
 - 文書画像を二値化→黒画素領域はおそらく「文字」



文書画像

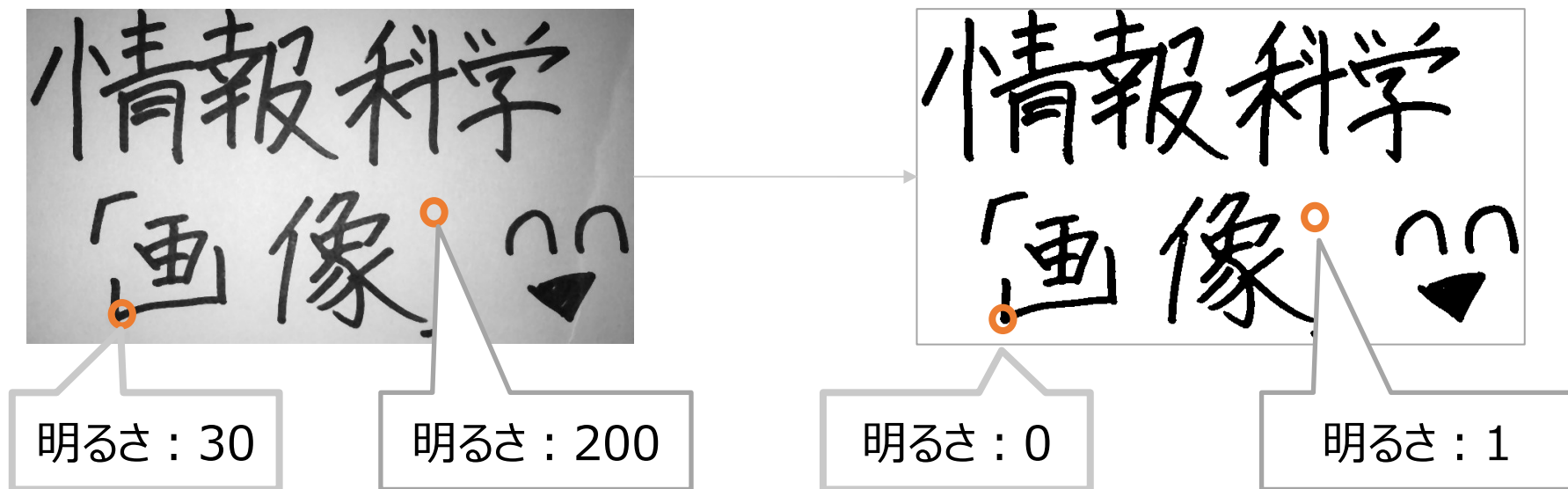


二値化結果画像

黒画素部分を
取り出して
文字認識

2値化の基本的考え方：明るさの「しきい値」処理

- 各画素がある値（しきい値）より明るい→白，暗い→黒
- 文書画像の二値化例（しきい値=110）



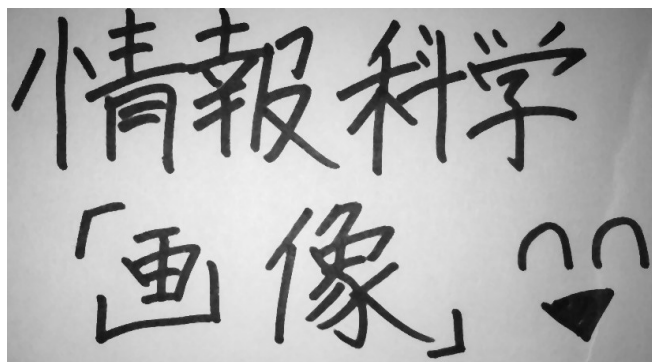
しきい値の設定は2値化の結果に大きく影響する



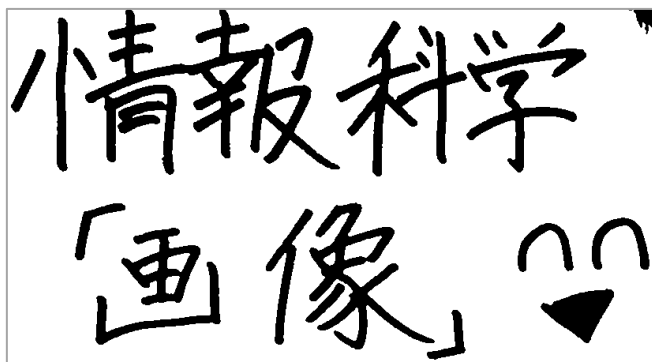
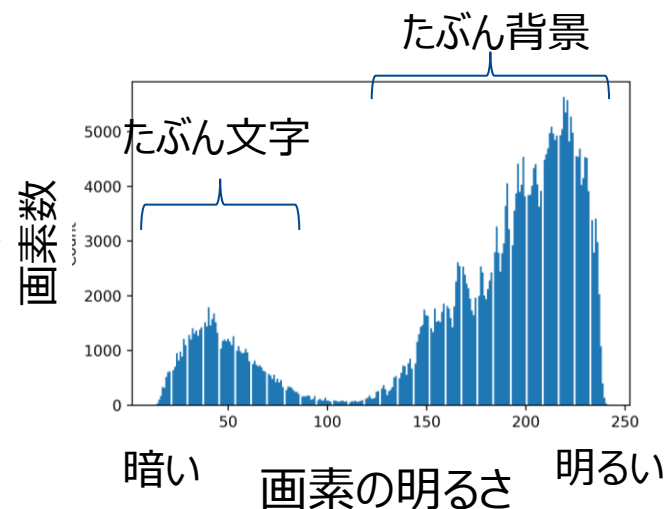
- 自動で適切な「しきい値」を決める方法はないか？
 - 次スライド

明るさのヒストグラムに基づいた 「しきい値」の自動設定

- 二つの山に分けることができる明るさがしきい値としてよさそう！



明るさの
ヒストグラム
化



しきい値
124で
2値化

