

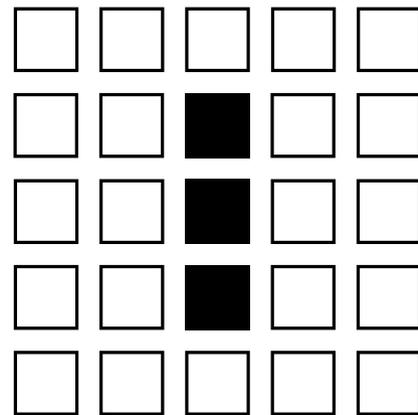
# 画像解析について

# ここでいう画像解析とは

画像データの中に現れる、  
ある特定のパターンで出現する値を  
抽出すること。

# 例えば

「1」だったら、以下の様なパターンで出現する  
※ピクセルを□で表現する。



# 例えば



# 今回の目標

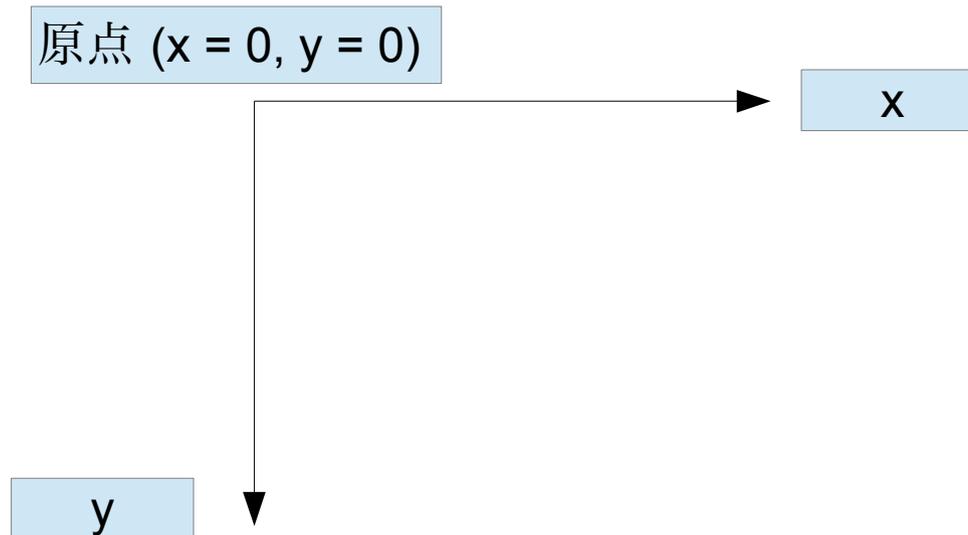
- 1.画像データを解析しやすいように、加工するフィルタリング処理を実装する。
- 2.比較元(テンプレート)、比較先(画像解析対象)を比較し、一致度を求めるマッチング処理を実装する。

# 前提知識

- 1.画像データの構成
- 2.画像データのフォーマット

# 1.画像データの構成

画像データは左上始まりの、 $x,y$ 座標で表す。



## 2.画像データのフォーマット

pgm形式を採用(簡単なので)

# pgmの構成

P5 (256階調のグレースケール)

# コメント

横ピクセル数 縦ピクセル数

1ピクセルのデータの大きさ

画像データ (バイナリ)

ヘッダ部

ボディ部

pgmのviewer

gimpを採用



フィルタリングとは

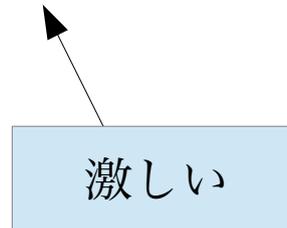
エッジを抽出する処理。

# エッジとは

画像の中で、色の変化が激しい境界

ex.画像データを10進数で表した場合

00087000



フィルタリングするとどうなる？

画像の輪郭がハッキリする。

→ 解析に不要な、余計なデータが除去される。

## フィルタリングの考え方

エッジには方向がある。

→  $X, Y$ の方向、大きさを持つベクトルデータ。

$X, Y$ に対して係数を掛け、微分することで  
変化率が大きい箇所を抽出する。

# フィルタリングの種類

1.Gradient

2.Prewitt

3.Sobel

etc...

# Gradient

隣合うセルに対するエッジを求める。  
→ 1方向のみ強調される。  
→ ノイズが残りやすい。

X

0	0	0
0	-1	1
0	0	0

Y

0	0	0
0	-1	0
0	1	0

# Prewitt

X,Y方向それぞれに対し、  
両隣+斜め方向のエッジも取る。  
→ より多くの方向との差分を取るため、  
単方向と比べノイズが残りにくい。

X

-1	0	1
-1	0	1
-1	0	1

Y

-1	-1	-1
0	0	0
1	1	1

# Sobel

Prewittに比べ、隣合う方向に対する  
重み付けが多い。  
→ 強調したい方向が、より強調される。

X

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

Y

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

## 試した結果

単純にフィルタリング処理をかけただけでは  
思った程輪郭が強調されない。

→ 演算結果に対し、閾値を設け  
この値以上ならば白、それ以外は黒と  
よりハッキリさせる処理をいれると  
更に輪郭が強調される。

→個人的にSobelフィルタの閾値10が  
良い感じに輪郭が取れた気がする。

→ 実行結果の画像があるのでご覧下さい。

実装はソースコードをどうぞ。

# マッチング

以下の式が基本的な考え方。

$$\sum |T_i - l_i|$$

どういうこと？

比較元、比較先の画像データを、  
左上から順番に差分を取っていき、  
どんどん足し込んでいった結果を  
求めるということ。  
(差分がマイナスになった場合、  
プラスにする)

→ 結果の数字が少ないほど、  
一致度が高いとされる。  
0だと完全一致。

## 得られた値に対して

一致度がどの程度ならば  
比較先の画像中に対象のデータがあると  
判断できるような閾値が必要だと思われる。

→ 現在のところ私はまだ解っていません。

実装はソースコードをどうぞ。

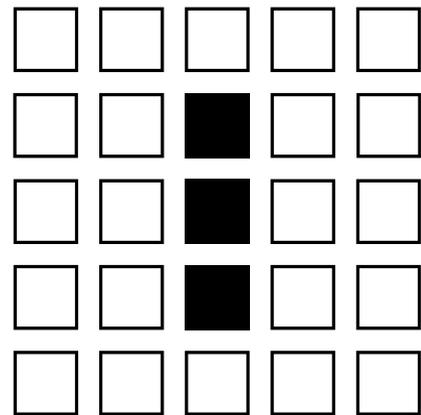
実は

これだけだと問題がある。

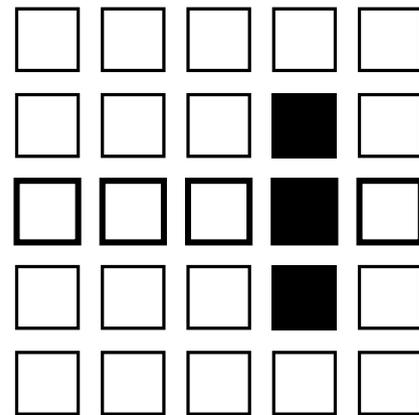
# 問題1！

以下は別もの扱いされてしまう。

比較元



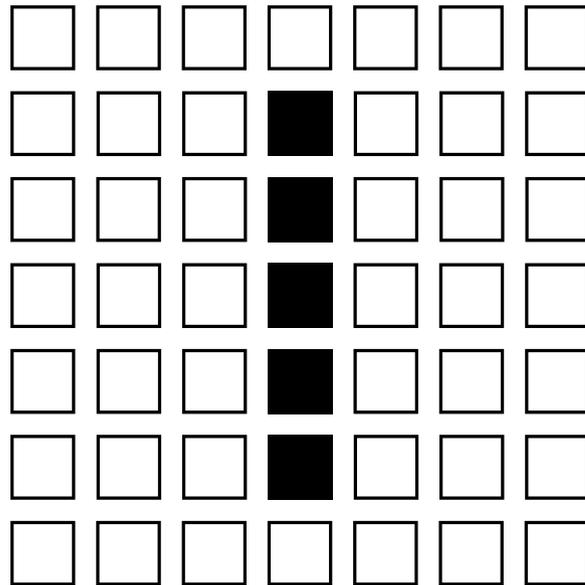
比較先



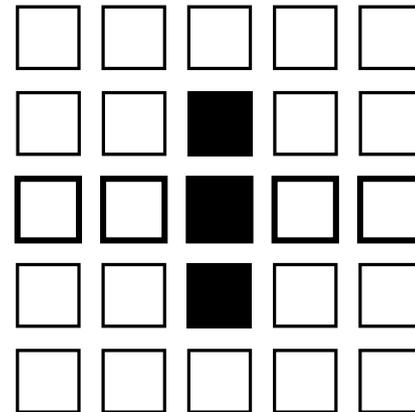
## 問題2！

大きさが違ってもダメ。

比較元



比較先



どうすれば良い??

問題 1 に対して

テンプレートの画像データの特徴を  
量化して、比較先に対しどれくらい  
一致する部分があるか計算する。

→ 糞難しい。。。

どうすれば良い??

問題2 に対して

比較元、比較先に対し拡大、縮小して  
比較しやすくする。

それぞれ次回の課題にしたいと思います。

## 参考資料

- ・ 画像研究入門

<http://image.onishi-lab.jp/index.html>

- ・ 詳解 画像処理プログラミング

C言語で実装する画像処理アルゴリズムのすべて  
<http://www.sbcr.jp/products/4797344370.html>