２値画像処理

２値化

- しきい値
  - ２値化の結果を見ながら試行錯誤する
  - ヒストグラムの谷にしきい値を設定する（モード法）
  - 統計的処理による（判別分析法）
  - 画像中の部分領域ごとにしきい値を決定する（動的しきい値法）

幾何学的性質

- ４近傍と８近傍
- 連結画素

形状特徴

面積

- 画素の数

周囲長

- 領域の周囲の画素の数

円形度

- 図形がどれだけ円に近いか

オイラー数

- 連結成分の個数－穴の数
- 穴は４連結と８連結とで異なる

２値画像の符号化

チェーン符号化

- 画素の連結方向にコードを割り当てて符号化

ランレングス符号化

- 走査線方向に連続する画素の数で符号化

変換

濃淡情報の変換

- コントラスト変換
- 疑似カラー表示

空間的情報の変換

- 先鋭化
- 平滑化

幾何学的情報の変換

- 座標変換
- 画像の再配列
- 画像データの内挿

濃淡情報の変換

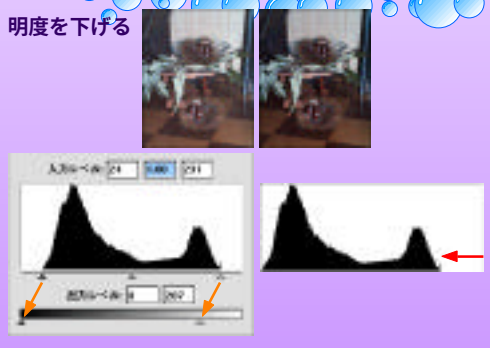
- 明度変換
- コントラスト変換
- 反転
- ソラリゼーション

明度変換

明度を上げる

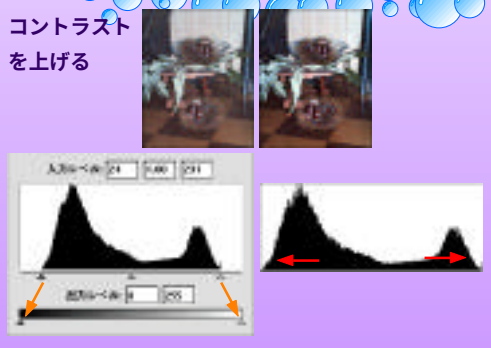
明度変換

明度を下げる



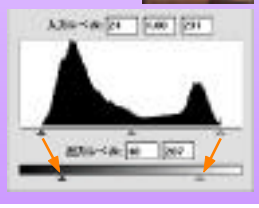
コントラスト変換

コントラスト
を上げる



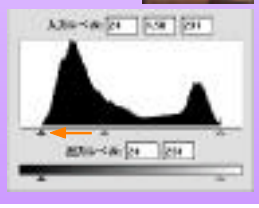
コントラスト変換

コントラスト
を下げる



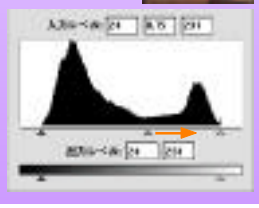
コントラスト変換

中間調を
上げる



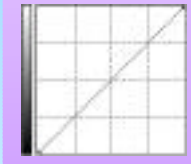
コントラスト変換

中間調を
下げる

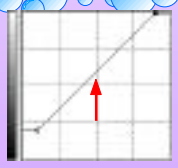


コントラスト変換

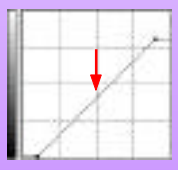
トーンカーブ



明度を上げる

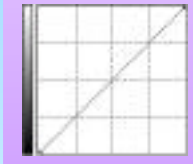


明度を下げる

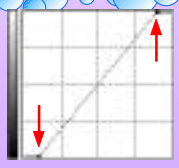


コントラスト変換

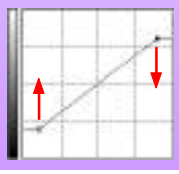
トーンカーブ



コントラスト
を上げる

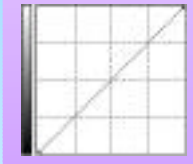


コントラスト
を下げる

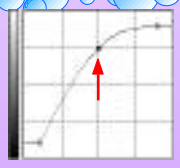


コントラスト変換

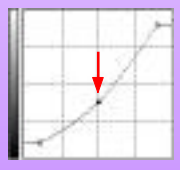
トーンカーブ



中間調を
上げる

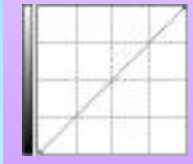


中間調を
下げる

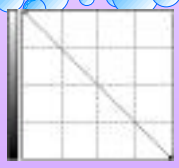


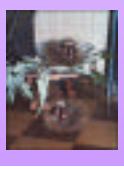
コントラスト変換

トーンカーブ



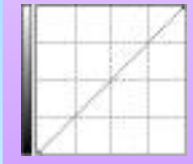
反転



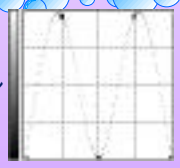


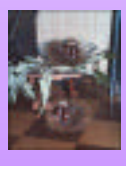
コントラスト変換

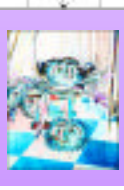
トーンカーブ



ソラリゼーション



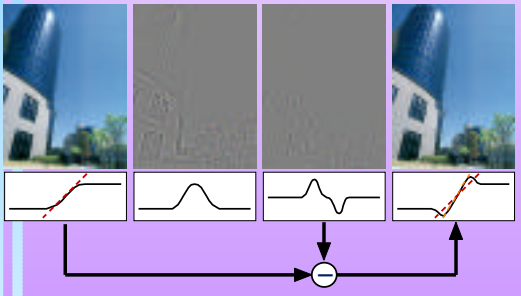




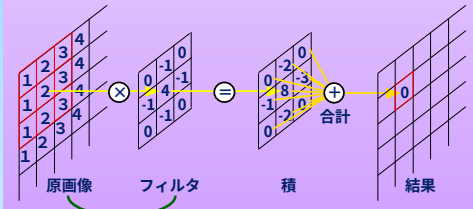
疑似カラー表示

（これは手抜きの方法）

先鋭化

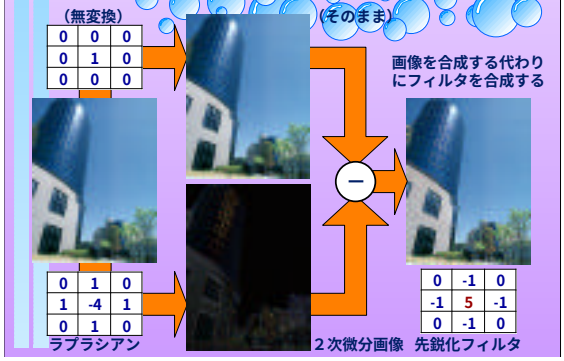


空間フィルタ



対応する画素を掛け合わせる

先鋭化フィルタ



平滑化

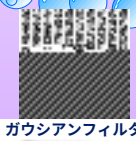
移動平均フィルタ

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

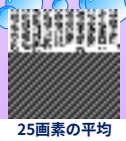
1/25	1/25	1/25	1/25	1/25
1/25	1/25	1/25	1/25	1/25
1/25	1/25	1/25	1/25	1/25
1/25	1/25	1/25	1/25	1/25
1/25	1/25	1/25	1/25	1/25

原画像　9画素の平均　25画素の平均

ガウシアンフィルタ



ガウシアンフィルタ



25画素の平均

ガウシアンフィルタ（加重平均フィルタ）

雑音除去

原画像　平滑化(ぼかす)　輪郭以外をぼかす　明るさの中間値（メディアンフィルタ）

座標変換

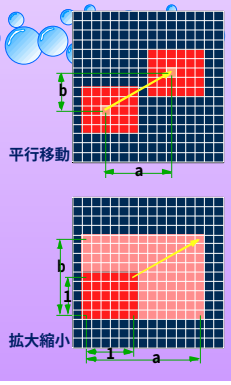
平行移動

- u = x + a
  v = y + b

拡大縮小

- u = ax
  v = by
- x = u / a
  y = v / b

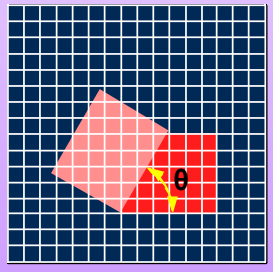
拡大縮小した結果の領域の各画素に，元の画像のどの画素の色を埋めるかという考え方をした場合



座標変換

回転

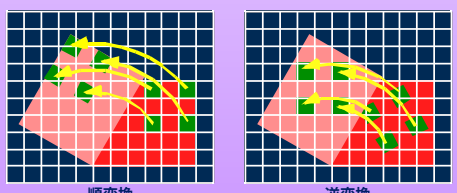
- $u = x \cos\theta + y \sin\theta$
  $v = -x \sin\theta + y \cos\theta$



画素の再配列

座標変換後の画像を再度標本化格子にはめ込む

- 順変換に基づく方法
  - 現画像の画素の位置から変換後の画素の位置を求めて画素を移動する
- 逆変換に基づく方法
  - 変換後の画素の位置から現画像の画素の位置を求めて画素を移動する



画素の再配列

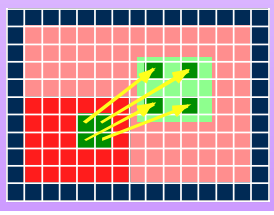
ラスタ画像の幾何変換では，逆変換が一般的

変換後の画素の座標値が整数値にはならない

- 内挿処理が必要

変換の際に対応する画素が無い場合がある

- 穴埋め処理が必要



画像データの内挿

最近隣内挿法

- ニアレストネイバー法

共１次内挿法

- バイリニア法

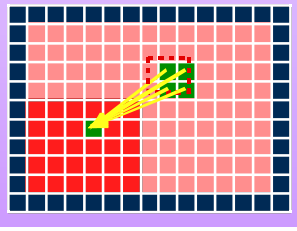
３次たたみ込み内挿法

- キュービックコンボリューション法

最近隣内挿法

内挿したい画素に最も近い画素の値を用いる

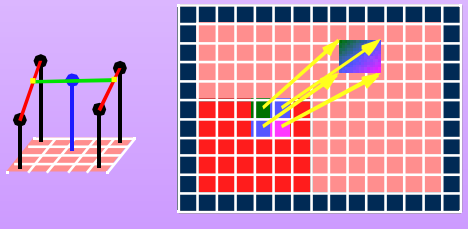
- 位置誤差は最大１／２画素
- オリジナルデータを壊さない
- 処理アルゴリズムが簡単



共１次内挿法

画素の周囲の４点の画素値から線形補間する

- オリジナルデータは壊される
- 平滑化の効果が得られる



３次たたみ込み内挿法

16点の格子点から３次関数を使って補間する

- オリジナルデータは壊される
- 平滑化と同時にエッジの部分では先鋭化の効果がある