

コンピュータグラフィックス

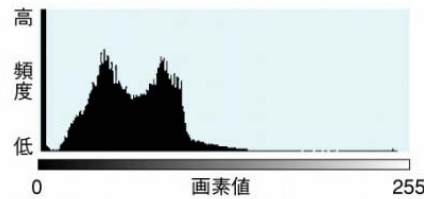
13. 画像処理2

ヒストグラム

- 横軸に画素値を，縦軸に頻度を示すヒストグラムにより画像全体の性質が見える
- 明暗の分布は，明るい画像では明るいところに，暗い画像は暗いところに集まる
- コントラストの低い画像は分布が狭い範囲に塊．高い画像は画素値が全体に広がる



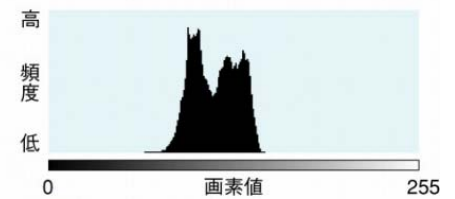
[a] 暗い画像



[b] 画像[a]のヒストグラム



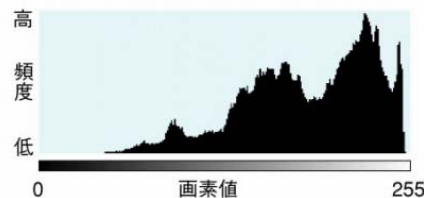
[a] コントラストの低い画像



[b] 画像[a]のヒストグラム



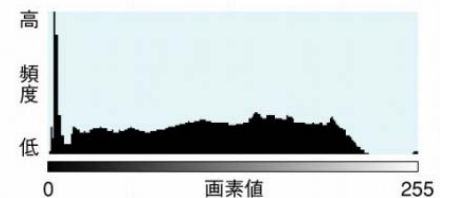
[c] 明るい画像



[d] 画像[c]のヒストグラム



[c] コントラストの高い画像



[d] 画像[c]のヒストグラム

画像の明暗とヒストグラム

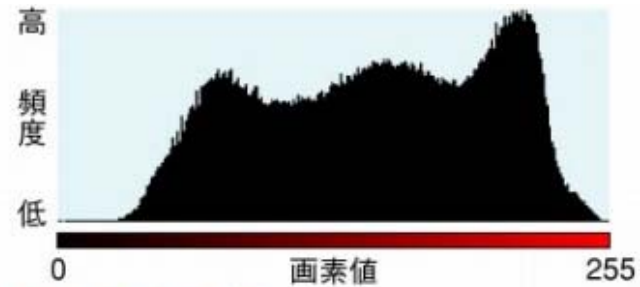
画像のコントラストとヒストグラム

ヒストグラム

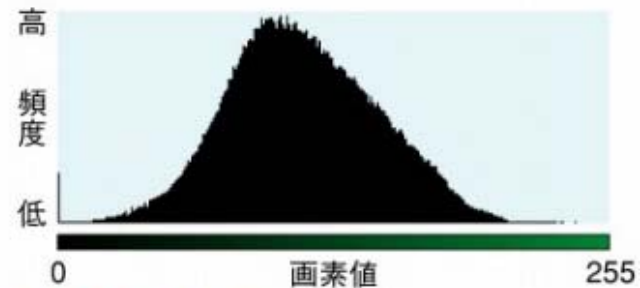
- カラー画像はR,G,Bそれぞれに対してヒストグラムがある



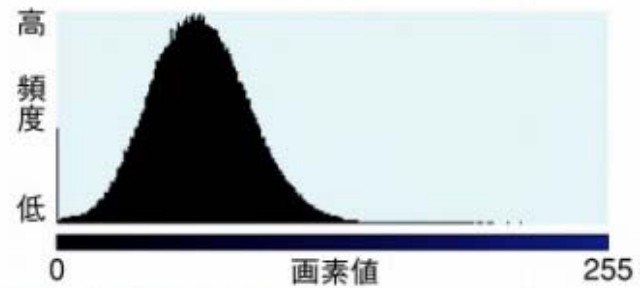
[a] カラー画像



[b] ヒストグラム (R)



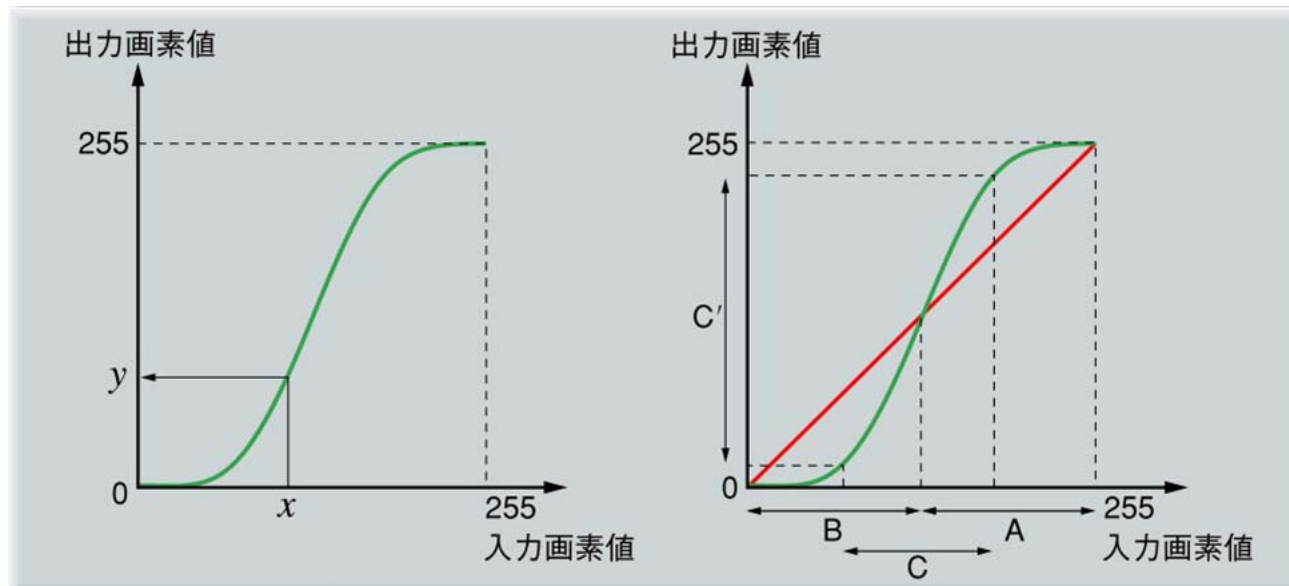
[c] ヒストグラム (G)



[d] ヒストグラム (B)

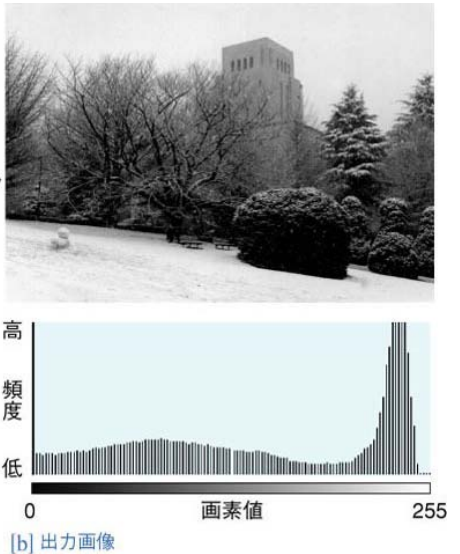
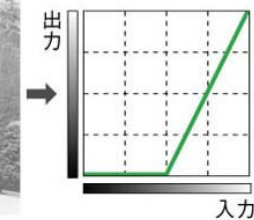
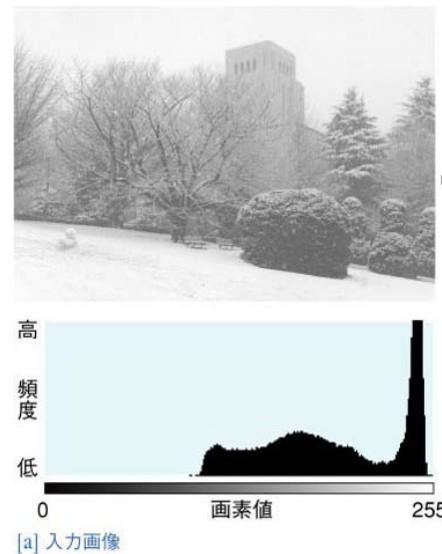
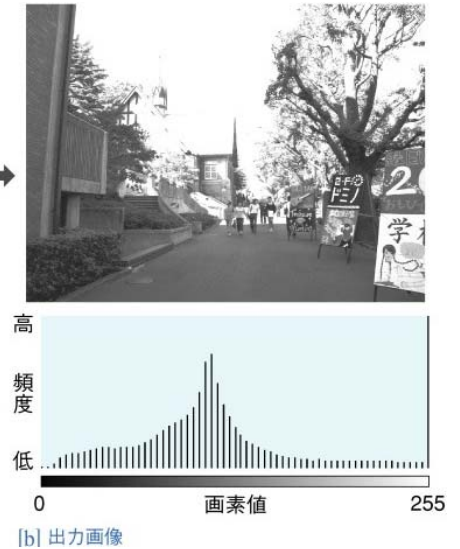
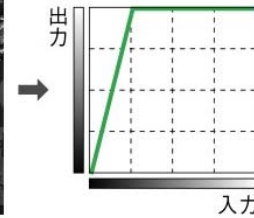
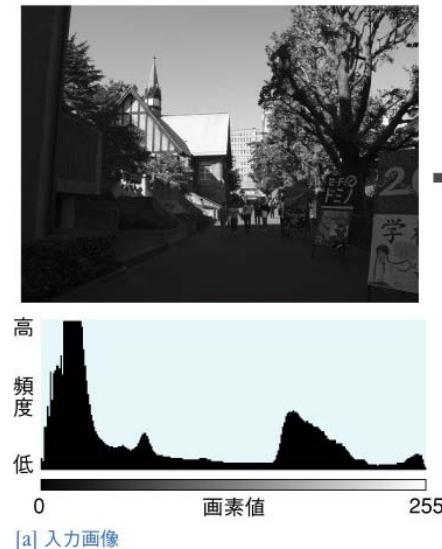
トーンカーブ

- 画像の濃淡を変化させるには，入力画像と出力画像の画素値の対応付け(階調変換関数)を与える
- 階調変換関数をグラフ表示したものをトーンカーブと呼ぶ
- カーブが直線よりも上にあれば明るく，下にあれば暗く変換される
- 傾きが直線よりも大きいとコントラストが高くなる



折れ線型トーンカーブによる変換

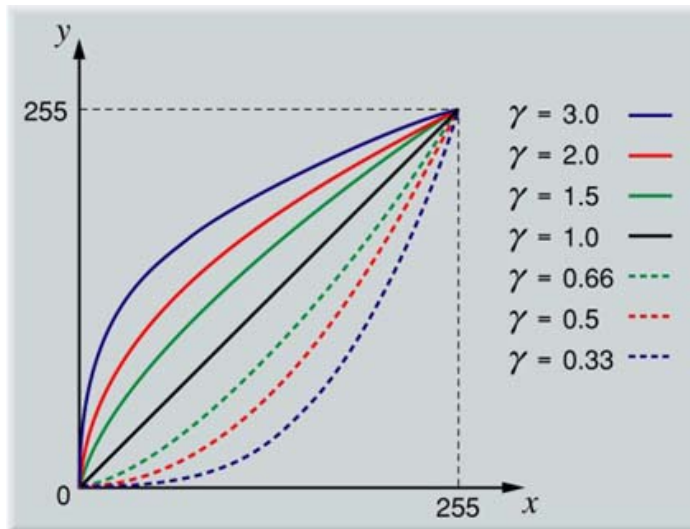
- 折れ線のトーンカーブは単純だが設定が容易なのでよく利用される
- 分布がくし状となるのは傾きが1よりも大きいトーンカーブで濃淡範囲が広げられたため



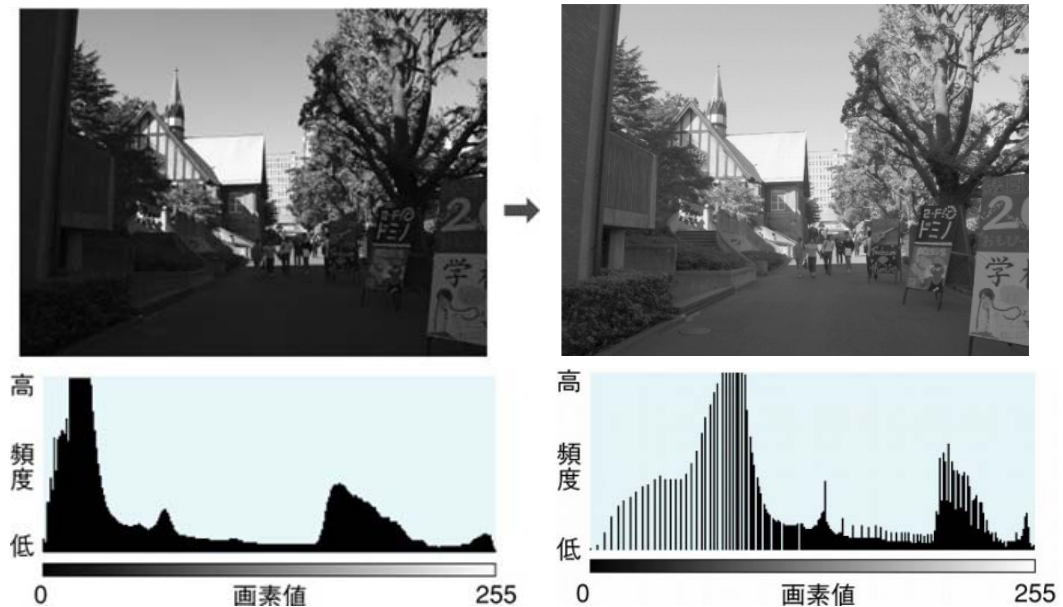
ガンマ変換

- 折れ線はトーンカーブが曲がっている点の前後で変化の性質が急激に変化し，水平な部分では出力画像の画素地が全て同じなためその部分の濃淡変化が失われる
- 次式の曲線のトーンカーブによる変換をガンマ変換と呼ぶ
 - $\gamma = 2$ の分布の右側もくし状となるのは，傾きが1以下では出力画像の画素値によって対応する入力画像の画素値の数が異なるため

$$y = 255 \left(\frac{x}{255} \right)^{\frac{1}{\gamma}}$$



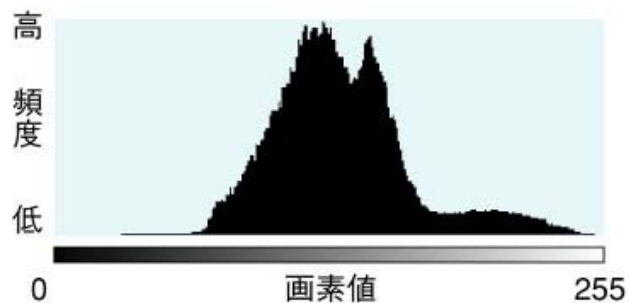
ガンマ変換のトーンカーブ



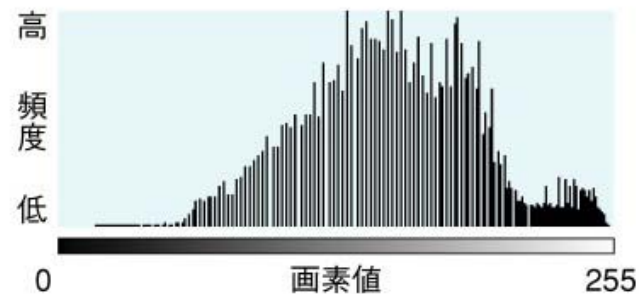
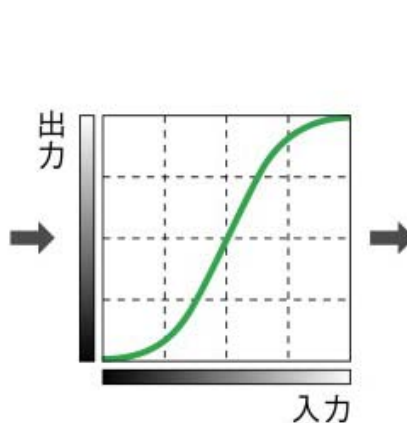
$\gamma=2$ による画像変換

S字トーンカーブによる変換

- ヒストグラムの分布が中央に偏ったコントラストの低い画像は，中間調が引き伸ばされコントラストが上がる



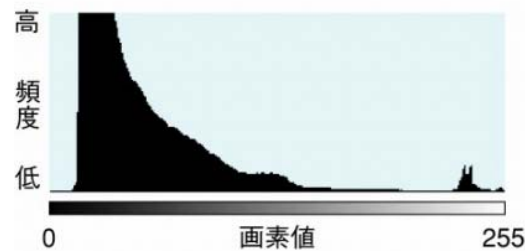
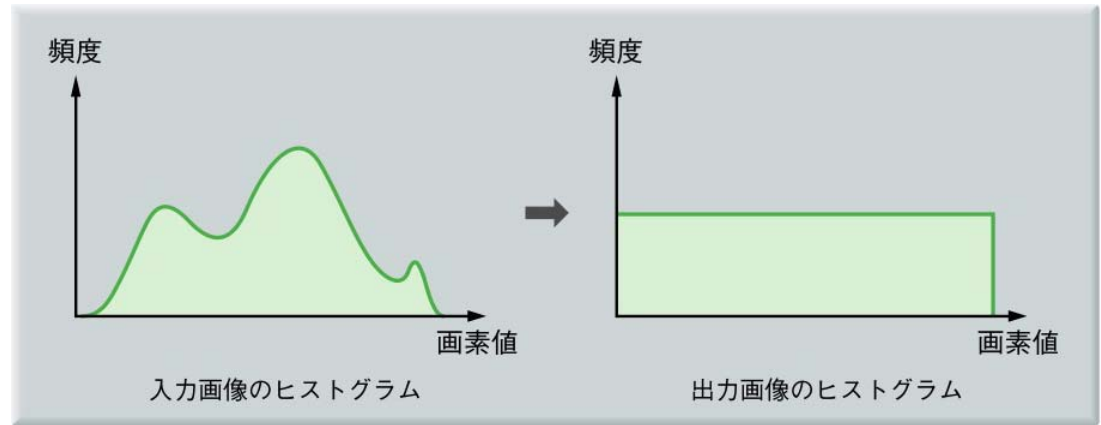
[a] 入力画像



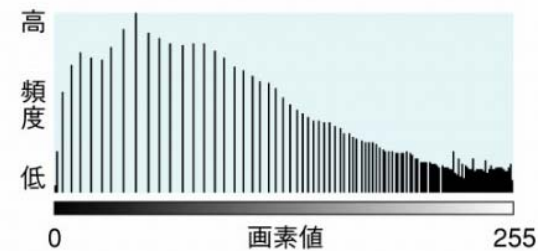
[b] 出力画像

ヒストグラム平坦化

- ヒストグラムの分布が集中している部分を引き伸ばす
ヒストグラム平坦化によってコントラストを上げることができる



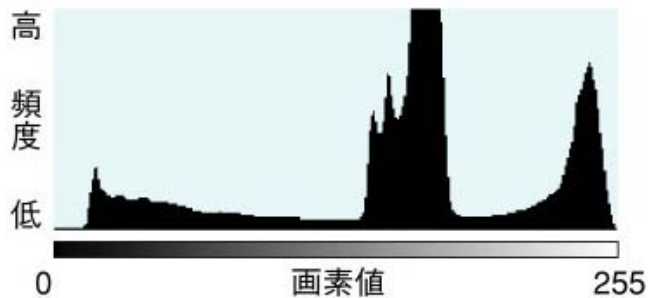
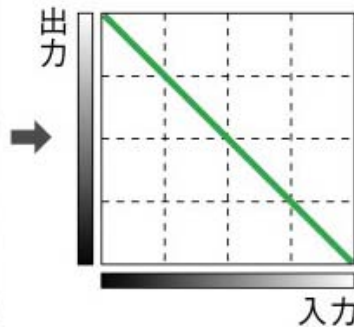
[a] 入力画像



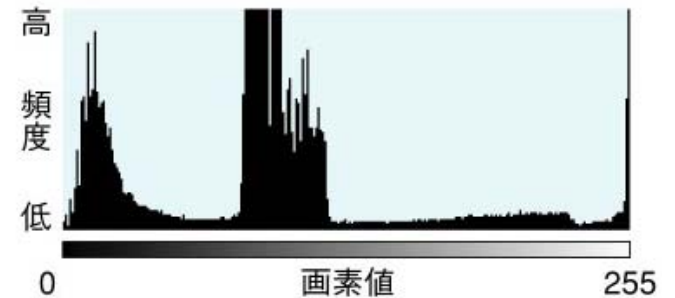
[b] 出力画像

濃淡の反転

- ヒストグラム分布を折り返す処理はネガ・ポジ反転と呼ばれる



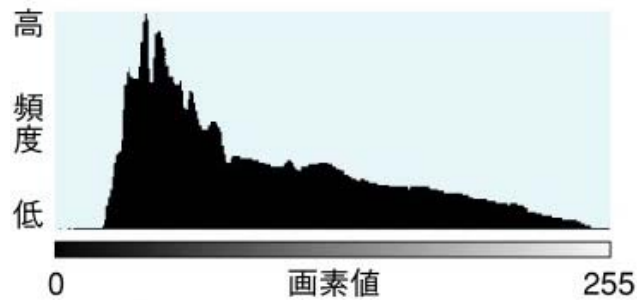
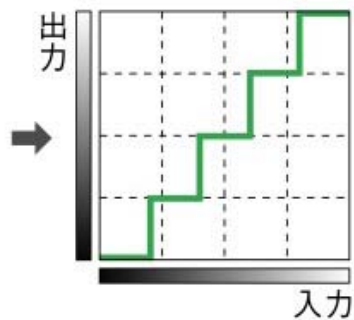
[a] 入力画像



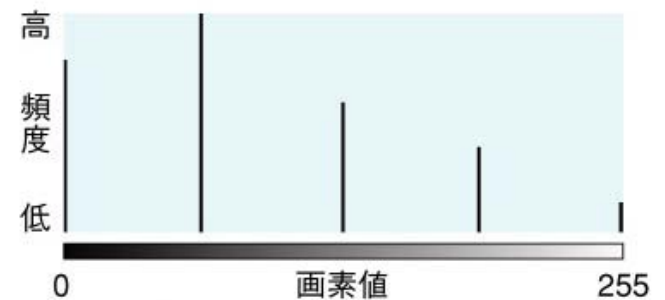
[b] 出力画像

ポスタリゼーション

- 階段状のトーンカーブにより出力画像の値が飛び飛びとなり色の等高線が現れる変換



[a] 入力画像



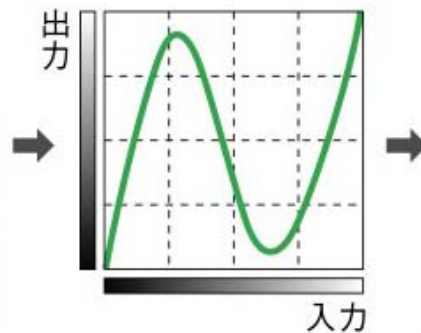
[b] 出力画像

ソラリゼーション

- 画像の濃淡の一部を反転させることで、ネガ画像とポジ画像が混ざったような効果が得られる



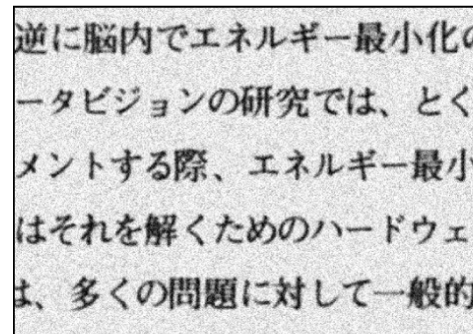
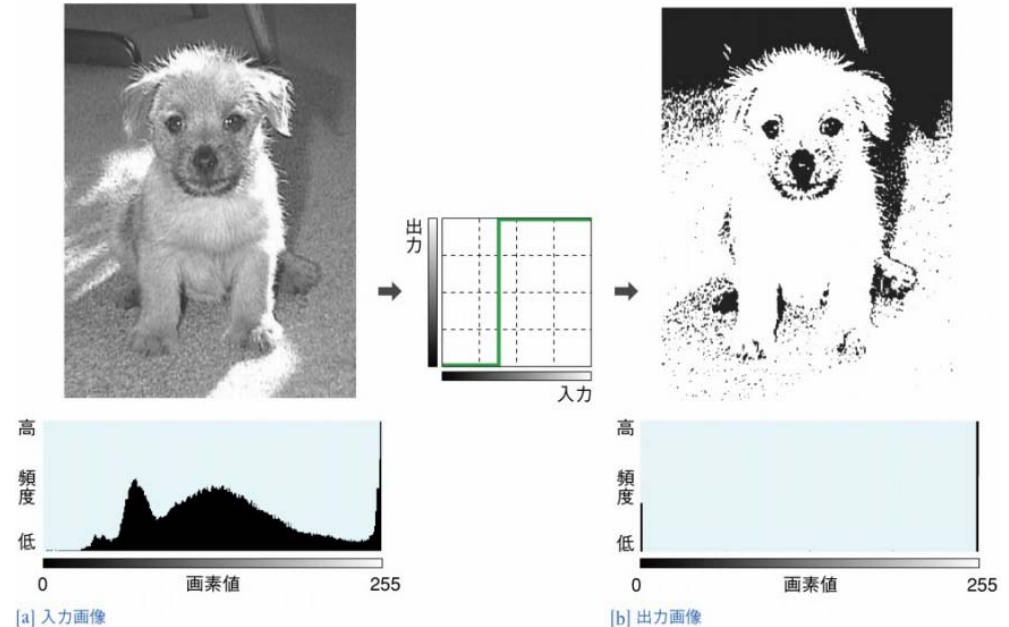
[a] 入力画像



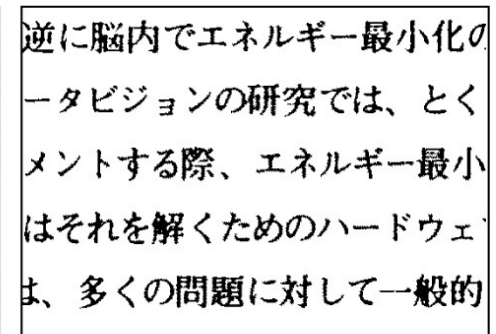
[b] 出力画像

2値化

- グレー画像中の画素地がある閾値より大きい小さいかによって黒か白の2値に変換する処理
- 閾値を自動的に決める方法は
 - ① 画像全体に占める対象物の面積比が一定と仮定してそれを満たすように決めるP-タイル法
 - ② 文書等のヒストグラムでは画素値の小さな部分と大きな部分に分かれるのでその間に閾値を置くモード法
 - ③ 統計処理による判別分岐法
 - ④ 画像の部分ごとの性質に合わせて変える動的閾値法



[a] 入力画像



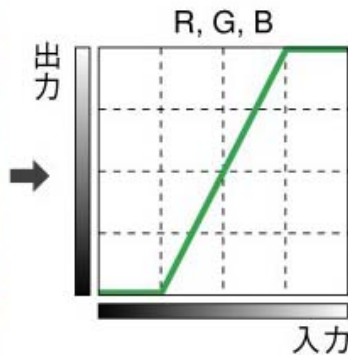
[b] 出力画像

色成分毎の濃淡変化

- カラー画像ではR,G,Bのトーンカーブを同一にすることが多いが、意図的に変えて色合いを変化させることができる



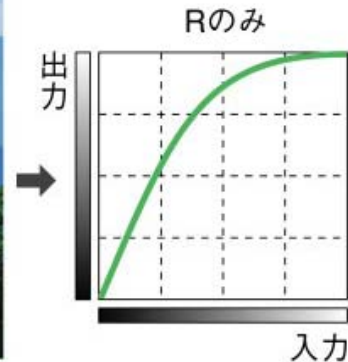
[a] 入力画像



[b] 出力画像



[a] 入力画像



[b] 出力画像

グレースケール画像への変換

- YIQモデルのY成分(輝度成分)だけを次式で取り出す

$$y = 0.299r + 0.587g + 0.114b$$

色相，彩度，明度の変化

- 人間にとって直感的に理解しやすい色相，彩度，明度に変換してそれぞれの属性を調整するほうほうもよく用いられる



[a] 入力画像



[b] 入力画像の色相を変える



[c] 入力画像の彩度を下げる



[d] 入力画像の彩度を上げる



[e] 入力画像の明度を下げる



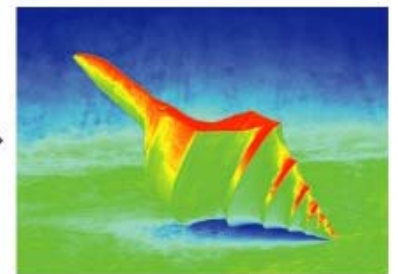
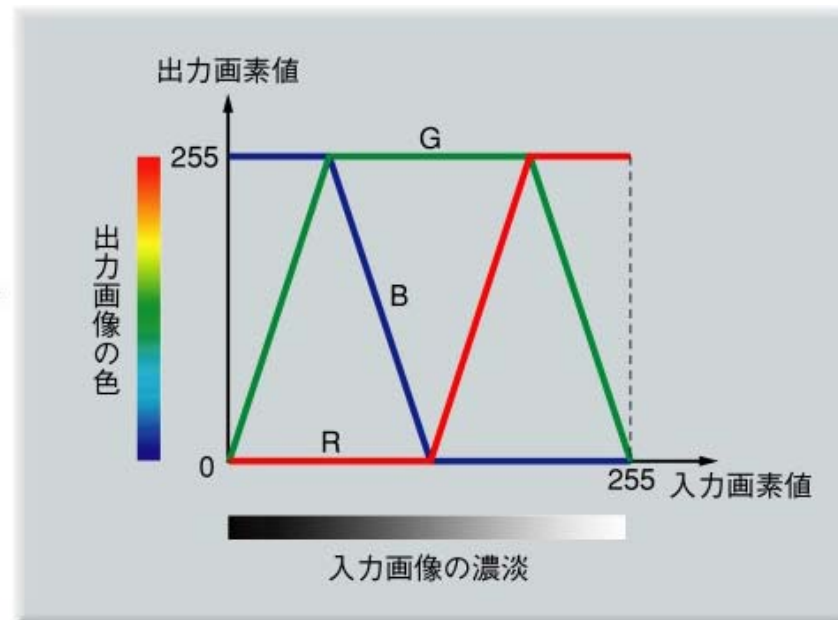
[f] 入力画像の明度を上げる

疑似カラー

- グレースケール画像はカラー画像のR,G,Bの画素値が全て等しい場合と見なすことができる.
- そのR,G,B各色に対して適当な異なるトーンカーブを割り当てカラー化することを疑似カラーと呼ぶ
- グレースケールでは区別しにくい微妙な濃淡の違いを色の違いとして表現し視覚的に識別しやすくできる



[a] 入力画像



[b] 出力画像(疑似カラー画像)

画像間演算

- 2枚以上の画像の同じ位置にある画素毎にある算術演算や論理演算を行う処理
- 2枚の画像の単純平均や重み付平均(アルファブレンディング)等

$$g = \frac{f_1 + f_2}{2}$$



[a] 入力画像1



[b] 入力画像2



[c] 出力画像

単純平均

$$g = \alpha f_1 + (1 - \alpha) f_2$$



[a] 入力画像1



[b] 入力画像2



[c] 出力画像

アルファブレンディング(左端 $\alpha=0$, 右端 $\alpha=1$)

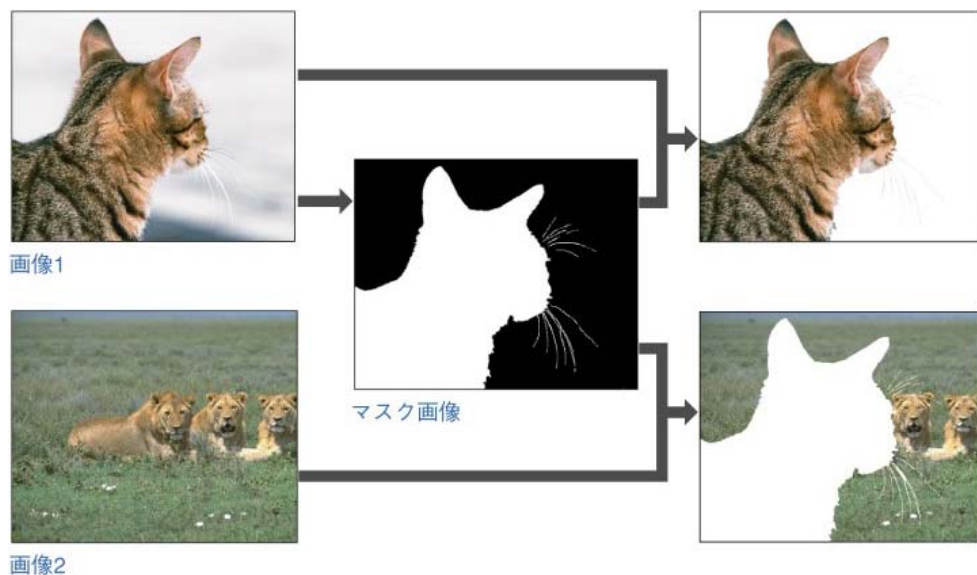
ディゾルブ

- 動画において α の値を時間的に変化させてあるシーンから別のシーンへ徐々に変化させる手法



マスク処理

- 2枚の画像を合成する際に、どの部分をどちらの画像から取ってくるかをマスク画像で指定する処理をマスク処理と呼ぶ
- クロマキーは単一色の背景などの領域を切り捨てる処理



[a] 入力画像

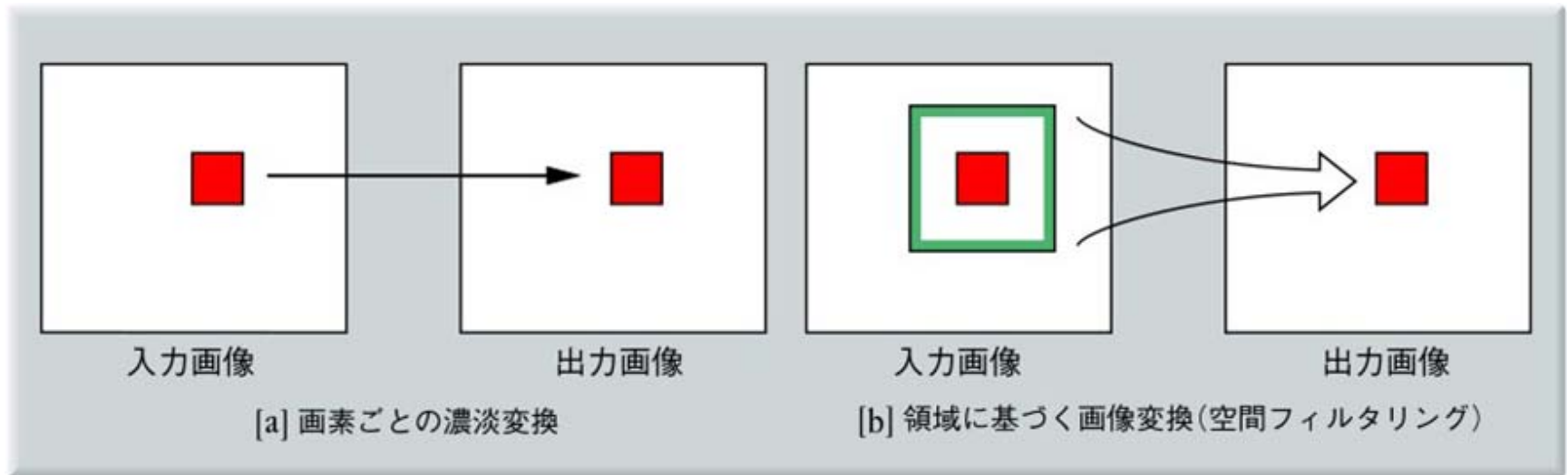
[b] マスク画像

クロマキーによるマスク生成

マスク処理による画像合成

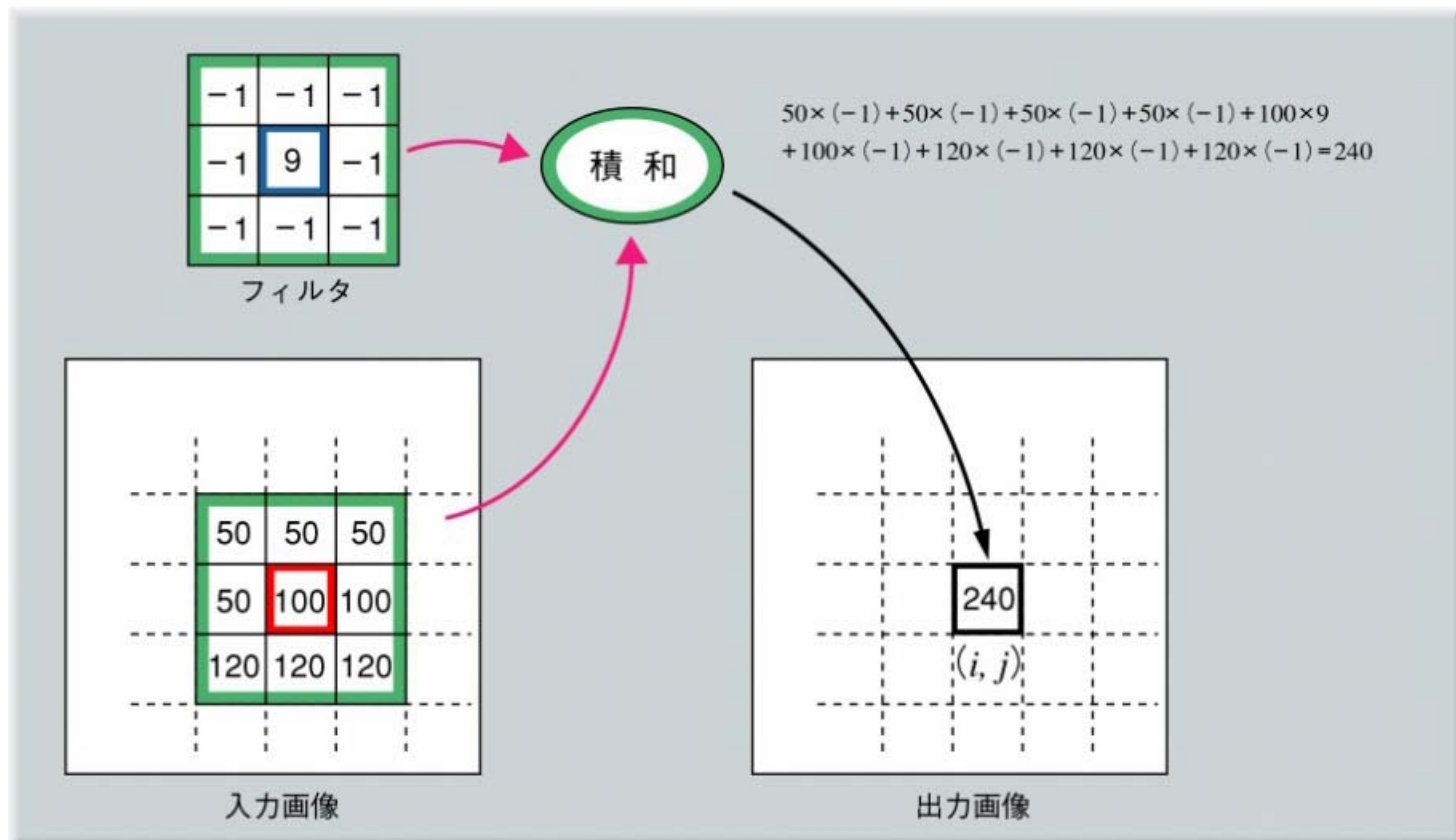
空間フィルタリング

- 出力画像の各画素値を入力画像の対応画素だけから求めるのではなく、対応画素の近傍領域も含めた処理を空間フィルタリングと呼ぶ



線形フィルタ

- フィルタの係数が図の様に3×3の画素に対して与えられているとき、中央を出力画素位置に合わせて3×3の入力画素値とフィルタ係数の積和演算結果を出力画素値とする
- 積和演算以外のフィルタを非線形フィルタと呼ぶ



平滑化

- ピンボケのようになめらかな濃淡変化を与える処理を平滑化と呼ぶ
- フィルタに覆われる領域内の画素値の平均値を求めるため平均化フィルタとも呼ばれる
 - 画素数が大きいフィルタほど効果が強い
- フィルタの中央ほど大きな重みを付ける加重平均は単純平均よりも自然に見える

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

[a] 3×3画素

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

[b] 5×5画素

平均化フィルタ

<table><tr><td>1</td><td>2</td><td>1</td></tr><tr><td>16</td><td>16</td><td>16</td></tr><tr><td>2</td><td>4</td><td>2</td></tr><tr><td>16</td><td>16</td><td>16</td></tr></table>			1	2	1	16	16	16	2	4	2	16	16	16	<table><tr><td>1</td><td>4</td><td>6</td><td>4</td><td>1</td></tr><tr><td>256</td><td>256</td><td>256</td><td>256</td><td>256</td></tr><tr><td>4</td><td>16</td><td>24</td><td>16</td><td>4</td></tr><tr><td>256</td><td>256</td><td>256</td><td>256</td><td>256</td></tr><tr><td>6</td><td>24</td><td>36</td><td>24</td><td>6</td></tr><tr><td>256</td><td>256</td><td>256</td><td>256</td><td>256</td></tr><tr><td>4</td><td>16</td><td>24</td><td>16</td><td>4</td></tr><tr><td>256</td><td>256</td><td>256</td><td>256</td><td>256</td></tr><tr><td>1</td><td>4</td><td>6</td><td>4</td><td>1</td></tr><tr><td>256</td><td>256</td><td>256</td><td>256</td><td>256</td></tr></table>					1	4	6	4	1	256	256	256	256	256	4	16	24	16	4	256	256	256	256	256	6	24	36	24	6	256	256	256	256	256	4	16	24	16	4	256	256	256	256	256	1	4	6	4	1	256	256	256	256	256
1	2	1																																																																			
16	16	16																																																																			
2	4	2																																																																			
16	16	16																																																																			
1	4	6	4	1																																																																	
256	256	256	256	256																																																																	
4	16	24	16	4																																																																	
256	256	256	256	256																																																																	
6	24	36	24	6																																																																	
256	256	256	256	256																																																																	
4	16	24	16	4																																																																	
256	256	256	256	256																																																																	
1	4	6	4	1																																																																	
256	256	256	256	256																																																																	
[a] 3×3画素			[b] 5×5画素																																																																		

加重平均による平均化フィルタ



[a] 入力画像



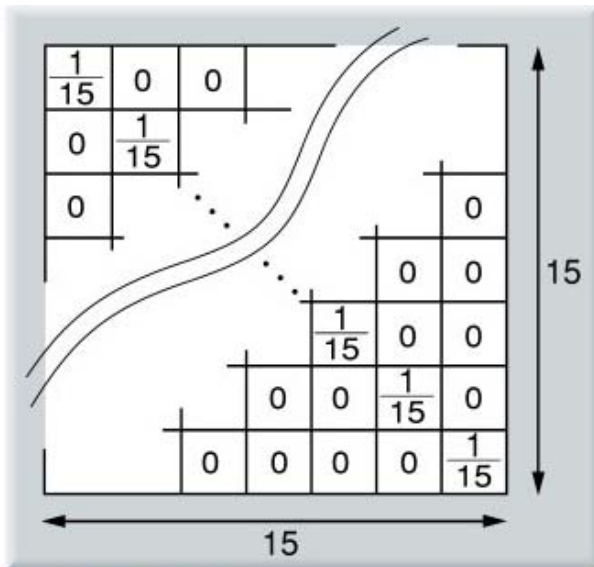
[b] 平均化フィルタ (3×3画素) の適用結果



[c] 平均化フィルタ (5×5画素) の適用結果

一方向の平滑化

- 平滑化を特定の方向に限って行うフィルタでは、一方向に流れるような効果が得られる



[a] フィルタの例



[b] 入力画像



[c] [a]の適用結果

メディアンフィルタ

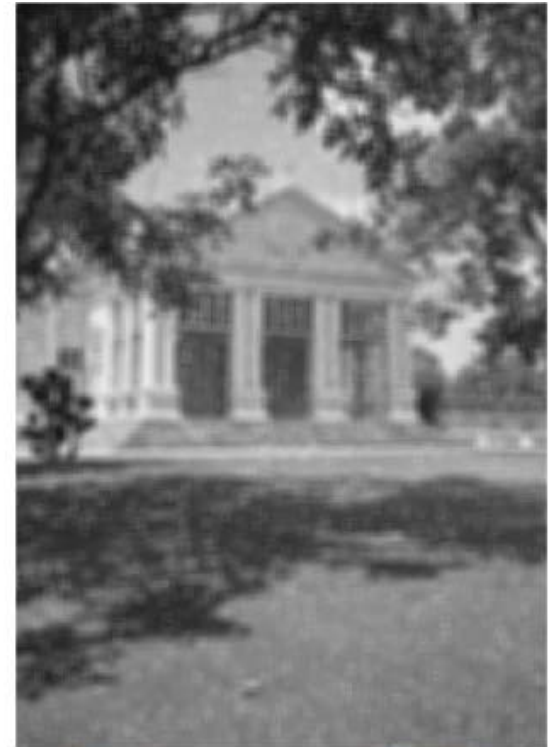
- 平均の代わりに、ある領域内の中央値を出力するものをメディアンフィルタと呼ぶ
- 平均化のようにエッジがなまることなくスパイク上のノイズが軽減される



[a] 入力画像



[b] メディアンフィルタの適用結果



[c] 平均化フィルタの適用結果

エッジ検出

- 注目画素と隣の画素の差を出力する処理は、連続関数の場合の微分に相当するため微分フィルタと呼ばれ、画像のエッジ検出に用いられる
- 横方向の差分を $\Delta_x f$ 、縦方向の差分を $\Delta_y f$ とするとき次の式で方向に依存しないエッジが得られる

$$\sqrt{(\Delta_x f)^2 + (\Delta_y f)^2}$$

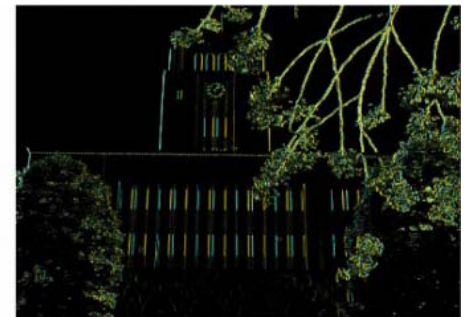
- エッジ検出にはソーベルフィルタもよく用いられる

0	0	0
0	-1	1
0	0	0

[a] 微分フィルタ横方向



[b] 入力画像



[c] [a]の適用結果

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

1	2	1
0	0	0
-1	2	-1

[a] 横方向

[b] 縦方向

0	1	0
0	-1	0
0	0	0

[d] 微分フィルタ縦方向



[e] [d]の適用結果



[f] 微分の大きさ

ソーベルフィルタ

微分フィルタの適用例

鮮鋭化

- 入力画像の濃淡を残したままエッジを強調するフィルタは鮮鋭化フィルタと呼ばれる

0	-1	0	-1	-1	-1
-1	5	-1	-1	9	-1
0	-1	0	-1	-1	-1
[a]			[b]		

鮮鋭化フィルタの例



[a] 入力画像



[b] 鮮鋭化フィルタの適用結果



[a] 入力画像



[b] 鮮鋭化フィルタの適用結果

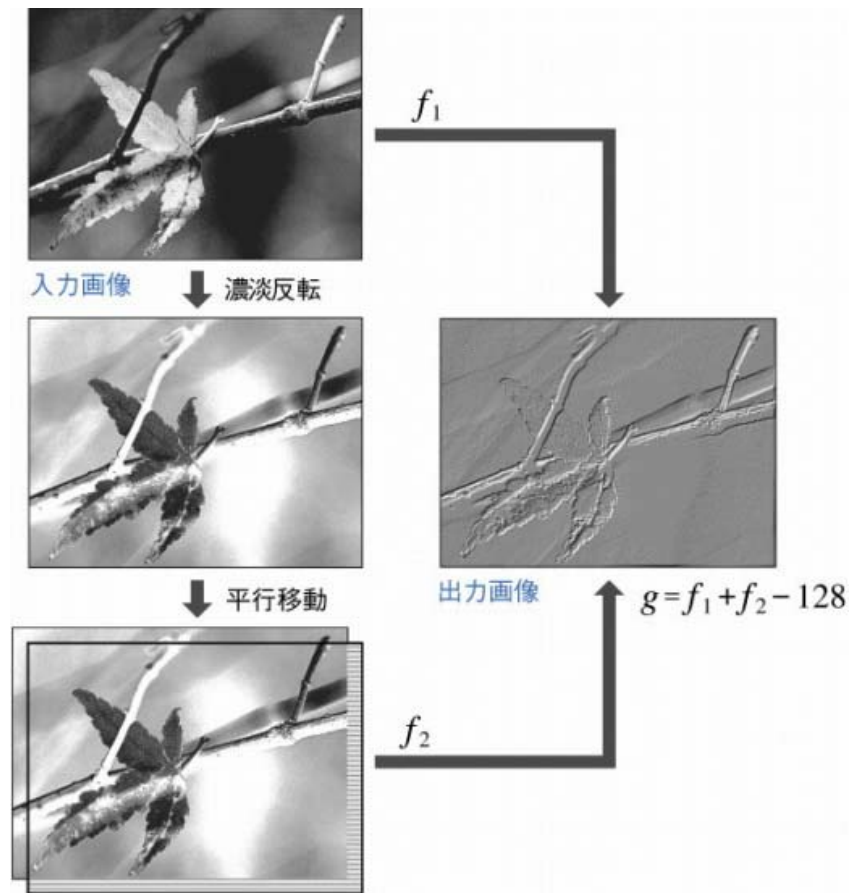
エンボス処理

- 画像の濃度差を利用して画像の一部を浮き出たせる処理
- 画素値を0~255とするとき, 入力画像 f_1 の濃淡を反転して数画素ずらした画像 f_2 から次式でエンボス画像を求める

$$g = f_1 + f_2 - 128$$

1	0	0
0	0	0
0	0	-1

エンボス処理のフィルタ



エンボス画像の生成

画像構成要素の置き換え

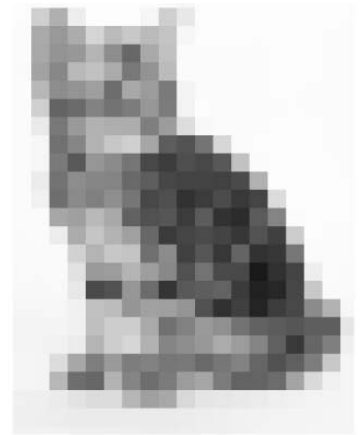
- デジタル画像を正方形や長方形，ポリゴン等の形状の構成要素に分割できる
- 画素はデジタル画像の最小構成要素
- 画像構成要素の中を画素地の平均値で置き換える処理をモザイク処理と呼ぶ



[a] 入力画像



[b] 8×8画素のブロック



[c] 16×16画素のブロック



[a] 入力画像



[b] 処理結果

画像の幾何学的変換

- 2次元座標変換は画像にも適用でき, 平行移動, 拡大・縮小, 回転などの幾何学的変換が可能



[a] 原画像



[b] [a] に拡大・縮小を適用

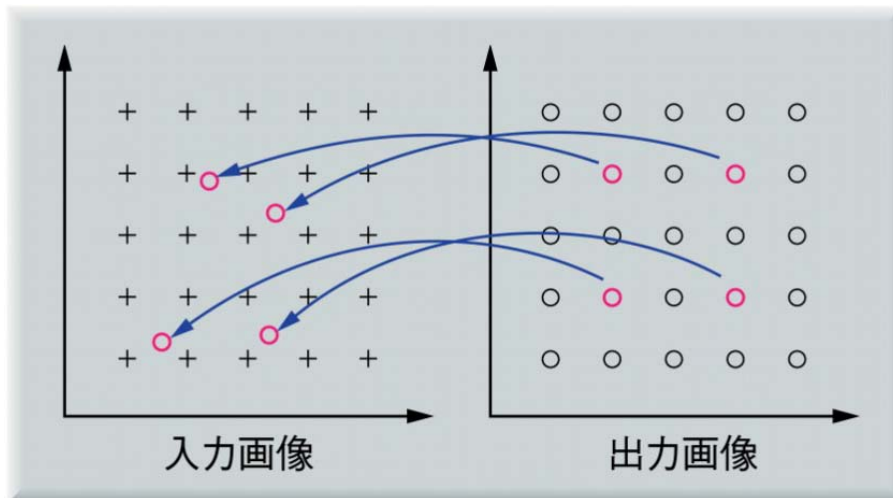


[c] [b] に回転を適用

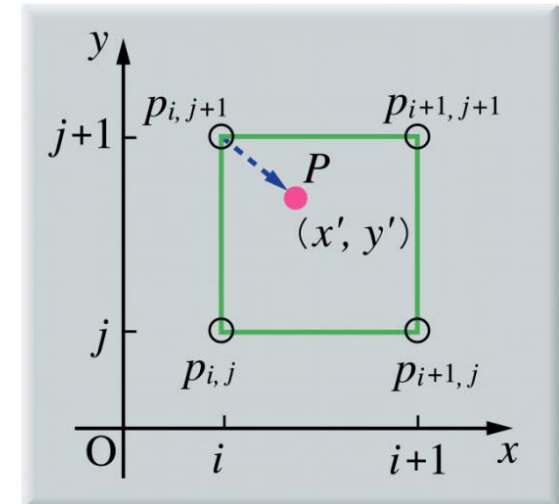
画像の際標本化と補間

- 変換後の画像の各画素値が変換前の座標系上でどこに対応するかを調べてその位置での値を求めることを画像の再標本化と呼ぶ
- 変換前の位置は一般に画素でないので、求めたい点の画素値は周辺の画素値から補完して決める
- ニアレストネイバーは補完する点に最も近い画素値とする方法

$$p = p_{i,j} \quad \text{ただし} \quad i = \lfloor x' + 0.5 \rfloor, j = \lfloor y' + 0.5 \rfloor$$



画像の際標本化



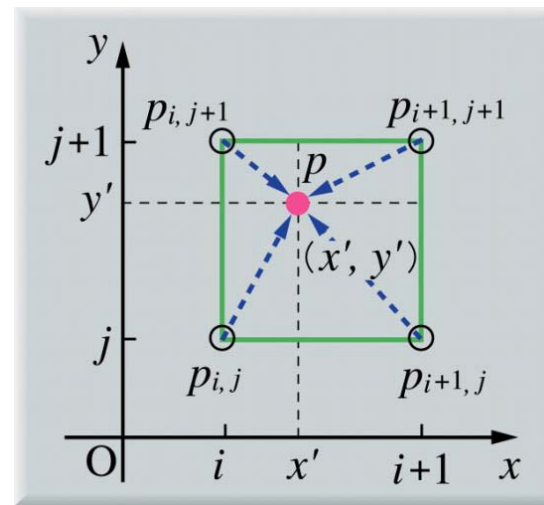
ニアレストネイバー

画像の際標本化と補間

- バイリニア補完は求めたい点の画素値をその周辺の4画素の値から双一次式によって求める

$$\begin{aligned} p = & ((i+1) - x')((j+1) - y')p_{i,j} \\ & + ((i+1) - x')(y' - j)p_{i,j+1} \\ & + (x' - i)((j+1) - y')p_{i+1,j} \\ & + (x' - i)(y' - j)p_{i+1,j+1} \end{aligned}$$

$$i = \lfloor x' \rfloor, j = \lfloor y' \rfloor$$



バイリニア補完



[a] 原画像



[b] ニアレストネイバーによる拡大画像



[c] バイリニア補完による拡大画像

イメージモザイク

- 複数の画像を接合して1枚の画像を作成する手法
- 接合処理は以下の手順で行われる
 - ① 濃度補正
 - ② 位置合わせと座標系の統一
 - ③ 濃度変換
 - ④ 接合点の探索
 - ⑤ 接合点周辺の濃度の平滑化

