

13章

図形と画像処理

2次元グラフィックスは、平面的な図形をデジタルデータに置き換えてコンピュータ上に表現する。3次元グラフィックスは、3次元の現実世界をさまざまな理論を使って2次元的な平面に表現する。2つのグラフィックスの相違点と共通する領域がどこにあるかに注目する。画像処理の分野では、より実際の2次元グラフィックスの生成法について解説する。

1. 2次元図形の表現

1.1 2次元座標系

平面上の点の位置を指定するためには、基準となる目印が必要である。この目印の役を果たすもっとも基本的な道具が平面に固定された (x, y) 座標系である。図13.1に示すように、まず平面に1本の水平な直線を固定し、これを x 軸とよぶ。次に1本の垂直な直線を固定し、これを y 軸とよぶ。このような x 軸と y 軸の組が (x, y) 座標系である。 x 軸と y 軸の交点 O をこの座標系の原点とよぶ。

1.2 図形の表現

平面上に表わされる図形の中で、もっとも簡単なものは点であり、点を組み合わせて、線分、折れ線、多角形、曲線などの複雑な線が表現され、各種の線を組み合わせて、いろいろな図形を表現することができる。点は座標系における座標値、線は点の配列の列によって定義することができる。

点

x_i と y_i の組 (x_i, y_i) を点 P の座標という。点 P の位置は、座標 (x_i, y_i) が与えられると、一義的に決まる。

線分

点 P_i と点 P_j を結ぶ線分を P_iP_j で表わす。線分 P_iP_j は、この両端点の座標 (x_i, y_i) 、 (x_j, y_j) によって表現できる。線分に向きをつけて考える場合は、 P_i から P_j に向かう線分に対して、 P_i を始点、 P_j を終点という。

折れ線

ひとつの線分の終点を次の線分の始点とする形で、いくつかの線分をつないだものを折れ線とよぶ。折れ線は、線分によってつながれる端点の列で表わすことができる。

多角形

n 角形は、その境界上に並んだ頂点の列 P_1, P_2, \dots, P_n を指定することによって表現できる。この多角形は、 n 本の辺 $P_1P_2, P_2P_3, \dots, P_{n-1}P_n, P_nP_1$ を持つ。

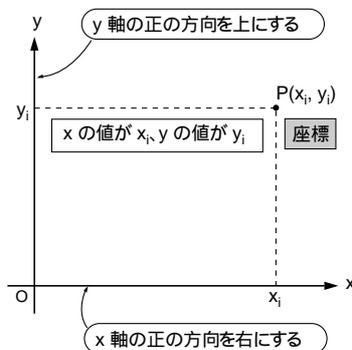


図13.1 2次元座標系

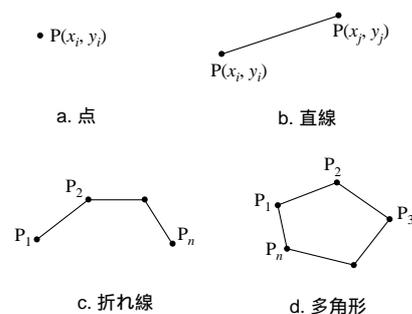


図13.2 2次元図形の定義

座標を定義された2次元図形は、平行移動、回転、拡大・縮小を始めとするさまざまな幾何変換式によって変換される。図13.3は、ドローツールによって描いた文字を、各種座標変換ツールを用いて変換した例である。

平行移動

x軸方向、y軸方向に対する2つの平行移動量を加算することで、図形の平行移動を行なうことができる。図13.3bは、移動ツールを選択してから、図形をマウスドラッグして平行移動している。

回転移動

図形に対して回転移動を行うときは、回転角度を与える前に回転の中心位置を指定する必要がある。図13.3cは、回転ツールを使用して、始めに回転の中心座標をマウスクリックで指定し、次に図形のマウスドラッグによって回転移動を行っている。

拡大・縮小(スケーリング)

拡大・縮小は、図形にx軸方向、y軸方向の倍率を掛けることで行なわれる。両方向の倍率が等しいときは、元の形のままで拡大・縮小されるが、倍率が異なっている場合は図形が変形する。拡大・縮小を行なうときも、その中心位置を指定しなければならない。図13.3dは拡大・縮小ツールを使用して、始めに移動の中心座標をマウスクリックで指定し、次に図形のマウスドラッグによって拡大・縮小を行なっている。

その他、2次元図形の座標変換には、対称移動(リフレクト)、せん断移動(シアー)などがある。

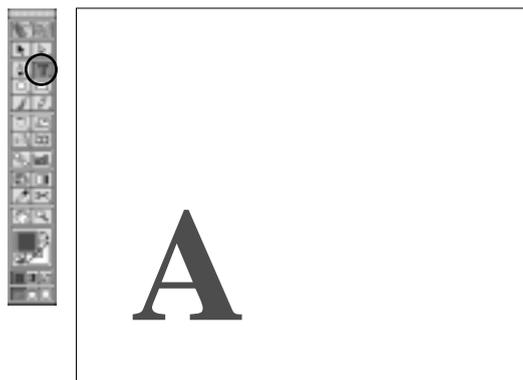


図13.3a 文字ツールで描いた図形

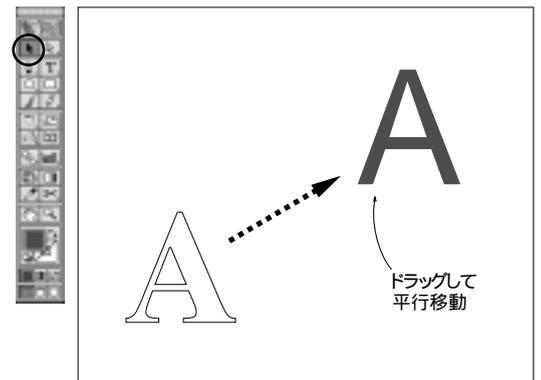


図13.3b 平行移動

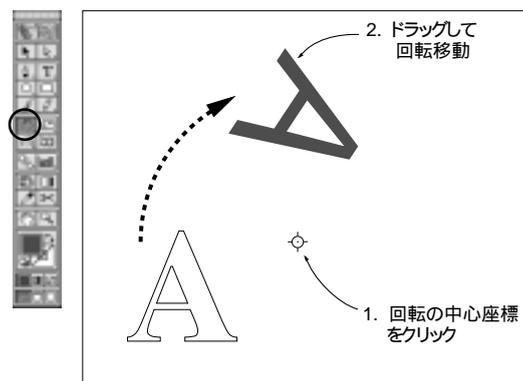


図13.3c 回転移動

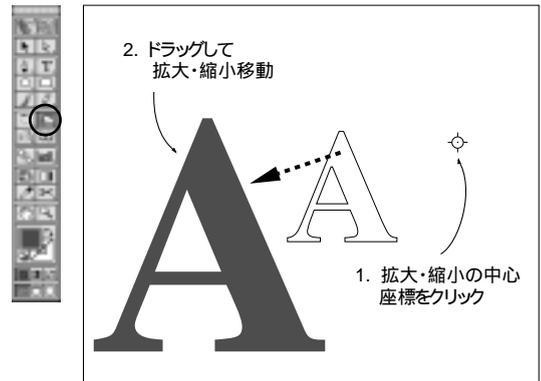


図13.3d 拡大・縮小

2. 3次元図形の表現

2.1 3次元座標系

3次元空間の点の位置を表わすためには、 (x, y, z) 座標系を用いる。図134に示すように、互いに直交し1点Oで交わる向きをついた3本の直線(これをx軸、y軸、z軸とよぶ)を固定する。これを (x, y, z) 座標系という。任意の点Pに対して、PとOを頂点に持ち、x軸、y軸、z軸に平行な直方体を考え、この直方体のx軸に平行な辺の長さを x_i 、y軸に平行な辺の長さを y_i 、z軸に平行な辺の長さを z_i とする。このとき (x_i, y_i, z_i) を点Pの座標という。

3次元空間における折れ線や多角形も、2次元図形と同様に点の配列の列によって表わすことができる。

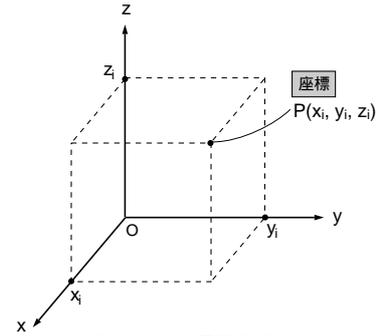


図13.4 3次元座標系

2.2 立体図形の表現方法

立体図形を表現するための代表的なモデルは、ワイヤーフレームモデル、サーフェスモデル、ソリッドモデルの3種類であり、次のような定義内容と特徴を持つ。

ワイヤーフレームモデル

直線や曲線の稜線だけを記録することによって立体の形状を表現するモデル。面の情報や立体の情報を明示的に持たないため、面の表示や立体どうしの衝突チェックなど、中身が詰まった立体としての処理はできない。

サーフェスモデル

立体の面データだけを表現したモデル。面の隠線・隠面消去表示やレンダリングが可能。体積計算や物体どうしの集合演算はできない。

ソリッドモデル

実物と同様に中身が詰まった立体としての完全な立体データを表現したモデル。レンダリングや稜線表示などの各種表示の他に、集合演算やさまざまなシミュレーションが可能である。

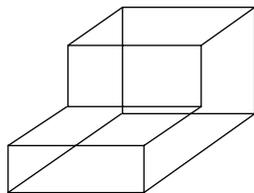


図13.5a ワイヤーフレームモデル

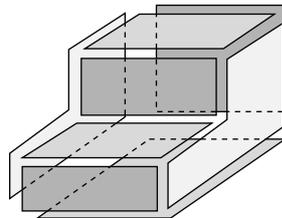


図13.5b サーフェスモデル

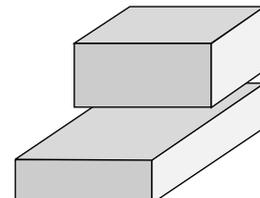


図13.5c ソリッドモデル

2.3 レンダリングと隠線・隠面消去

立体を表現するためには、視点から見えない領域を表示しないようにする処理が必要であり、この処理を隠線消去もしくは隠面消去とよぶ。立体を構成する平面(ポリゴン)を消去する隠面消去法は、幾何モデルをレンダリングする際に、視点から見て不可視面である隠面領域を検出して表示の対象から除外し、可視領域だけを表示する手法である。隠面消去法は、そのアルゴリズムによって、zバッファ法、スキャンライン法、レイトレーシング法などに分類される。

隠面消去をも含めて、視点から見た立体の状況を、透視図法的に正しく表示することをレンダリングという。

2.4 シェーディング

3次元物体が「立体的に見える」ということは、立体の表面に当たった光が反射して視点へ届くという物理現象の結果である。光の反射をもっとも単純化したモデルは、拡散反射光と鏡面反射光および環境光を考慮するものであり、一般の反射現象はこの3つの反射光が混合されたものとして近似的に表わすことができる。

拡散反射光

拡散反射光は、物体表面に入射した光が物質内で多重乱射した後、あらゆる方向に同じ強さで反射する光であり、物体全体が持つ固有の色と立体感を表わす光となる。図13.6aの物体表面上の位置Pにおける拡散反射光の強さは、入射角 θ の大きさ、すなわちPから垂直に立つ法線ベクトルNとPから光源に向かう光線ベクトルLとの内積によって表わすことができる。

鏡面反射光

鏡面反射光は、物質の表面での直接反射によって生じる。よく磨かれた物体が照射されると表面の一部にハイライトを生じるが、これが鏡面反射によって生じる光である。鏡面反射光は、入射角と反射角が等しい正反射の方向に光がもっとも強く反射され、完全鏡面反射でないものは、正反射方向の近傍にも反射される。鏡面反射光の強さは、図13.6bの角度 γ の大きさで決定される。

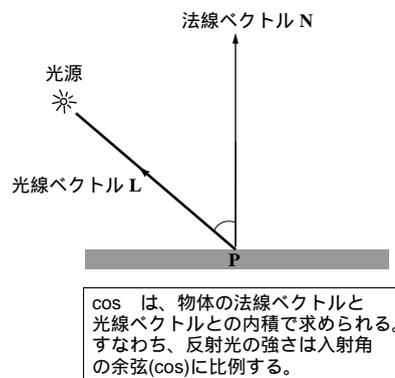


図13.6a 拡散反射光

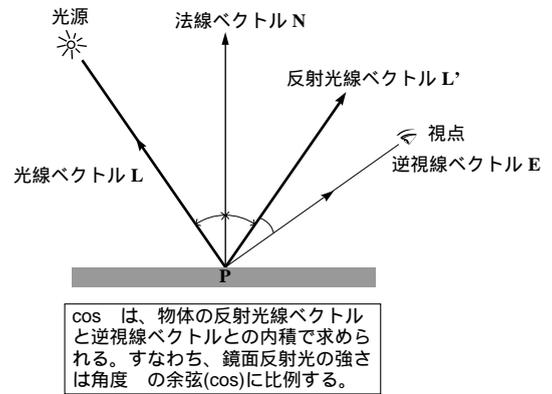


図13.6b 鏡面反射光

2.5 マッピング

2次元の平面上にある模様や凸凹データなどを、3次元曲面や平面の上に貼り付けることをマッピングといい、写実性の高いCG画像を作成するためには欠かせない技法である。マッピングには次のような手法がある。

テクスチャマッピング

2次元テクスチャを3次元曲面の表面に写像するマッピング(図13.7b)。

バンプマッピング

テクスチャの濃淡データを物体表面の高さデータに置き換える。その高さデータに基づいて法線ベクトルを傾かせ、物体表面上の擬似的な凸凹感を表現するマッピング(図13.7c)。

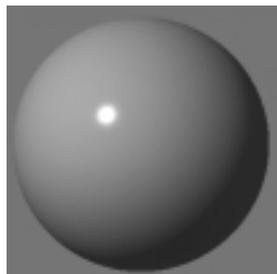


図13.7a 拡散反射と鏡面反射によるレンダリング



図13.7b テクスチャマッピング

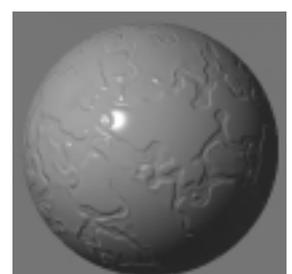


図13.7c バンプマッピング

3. 画像処理

3.1 デジタル画像の表現

(1) 画像の標本化と量子化

コンピュータにアナログ画像を取り込むためには、画像をデジタルからアナログに変換するA/D変換が必要である。画像のA/D変換は、座標 (x, y) を離散的な値に変換する空間的離散化と、濃淡/離散的な値に変換する信号強度の離散化の2段階に分けられる。前者を画像の標本化、後者を画像の量子化という。

(2) 標本化(sampling)

アナログ画像上に等間隔に置かれた標本点における濃淡、もしくは格子で区切られた小区画における濃淡の平均値や最大値を取り出す作業を行う。画面を画素に分割する方法としては、通常は正方形格子が用いられる。標本化の際に正方形格子を用いて $M \times N$ の区画に分割したとすると、画像は $M \times N$ 個の画素からなることになる。格子間隔が狭い(画素数が多い)ほど解像度が高くなり、微細なパターンまで読み取れることとなる反面、画像データサイズは大きくなる。

画像の入出力を行う場合に、単位長さあたりに決める画素数が画像解像度である。長さの単位を1インチとし、単位長に含まれるデジタル画像の画素数をppi(pixels per inch)という。プリンタやスキャナなど、画像入出力装置を使用する際の解像度には、1インチあたりの点(dot)数を表わしたdpi(dot per inch)が用いられる。

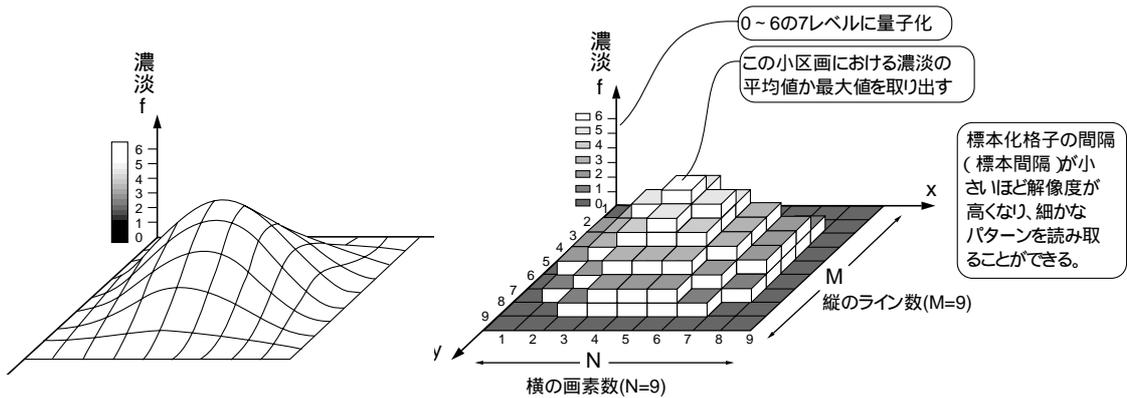


図13.8a アナログ画像の濃淡分布 $f(x, y)$

図13.8b デジタル画像の濃淡分布 $L(i, j)$



図13.9a アナログ画像の例

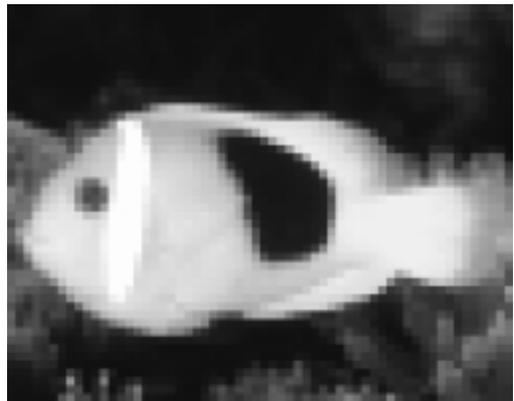


図13.9a デジタル画像の例

濃淡レベル	2^n	量子化レベル	使用される用途
1024	2^{10}	10ビット量子化	胸部X線写真など、微妙な濃淡のレベルの違いが重要な意味を持つ場合
512	2^9	9ビット量子化	
256	2^8	8ビット量子化	一般の写真などの場合。この階調があれば人間の目では不自然さはほとんど感じられない。
128	2^7	7ビット量子化	
64	2^6	6ビット量子化	
32	2^5	5ビット量子化	
16	2^4	4ビット量子化	手書き図面など、黒と白の2値が表現されればよい場合。
8	2^3	3ビット量子化	
4	2^2	2ビット量子化	
2	2^1	1ビット量子化	

表13.1 濃淡レベルと量子化レベル

(3) 量子化 (quantization)

アナログ画像上で連続的に変化する濃淡を離散的な有限個の値に変換する作業が量子化である。1つの代表値をnビット(量子化レベル)に離散化する場合、濃淡レベル数は 2^n となる。一般の写真などを扱う場合、256段階の階調があれば人間の目では不自然さはほとんど感じられないといわれる。

元の画像の濃淡と量子化された濃淡との間で生じる差のことを量子化誤差といい、量子化される対象の輝度範囲をダイナミックレンジという。

ダイナミックレンジが一定の場合、量子化レベルを増加させると、量子化レベルの間隔が小さくなるため、量子化誤差は小さくなり、結果的に得られるデジタル画像には元の画像の濃淡がより忠実に反映される。

量子化レベルが十分でないときに、ある濃淡レベルと1だけ異なる濃淡レベルが隣り合うところで生じる境界を疑似輪郭または疑似エッジとよぶ。

(4) 色の表し方とカラーモデル

他の2つの色を混色してもう1つの色を作ることができないような、互いに独立した3つの色を三原色(三原刺激)とよび、この三原色を用いた色の混色法には、次の加法混色と減法混色がある。

加法混色

ディスプレイなどのように、光で色を表現する場合、赤(R)、緑(G)、青(B)各色の光を混色して別の色を表現する。RGBすべての色が同じ割合で重なりと無彩色(灰色)になり、各色の強さを強くしていくと白に近づく(図13.10a)。

減法混色

印刷物などのように、それ自体では光を發せず反射光によって色を表現する場合、シアン(C)、マゼンタ(M)、イエロー(Y)各色の色の濃さで色が表現される。すべての色が同じ割合で重なりと無彩色(灰色)になり、すべての色がなくなると白になる(図13.10b)。CMYのすべてが最大になると黒に近づく。

実際の画像出力装置を加法混色装置あるいは減法混色装置にモデル化すれば、次式が成り立つ。

$$C = 1 - R, \quad M = 1 - G, \quad Y = 1 - B$$

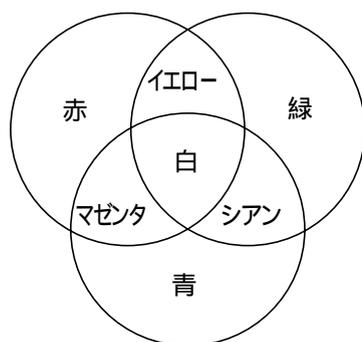


図13.10a 加法混色

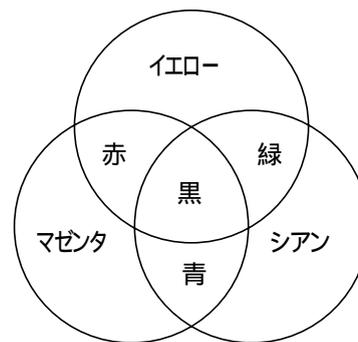


図13.10b 減法混色

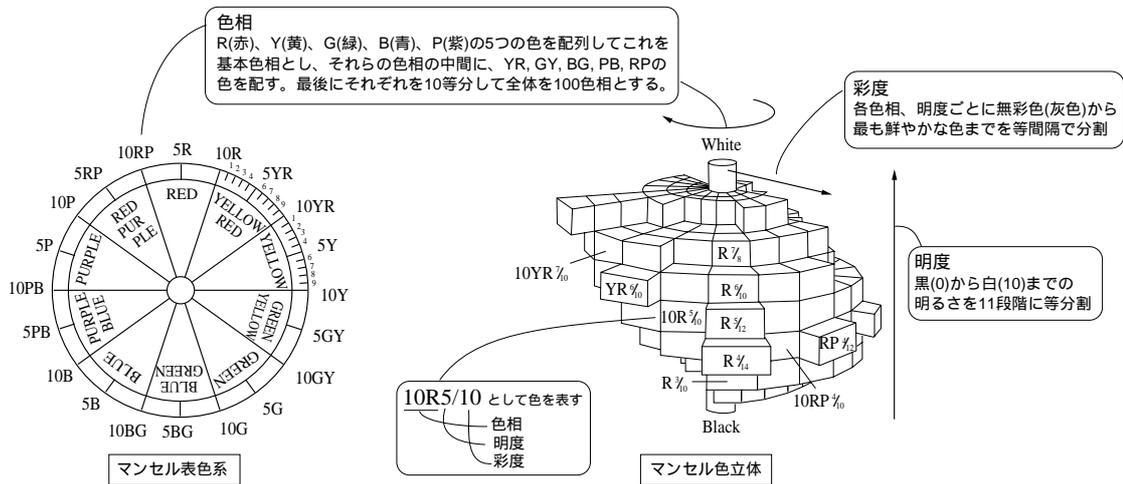


図13.11 マンセル表色系と色立体

(5) マンセル表色法

色を心理的な観点から、色相(色の違いを示す属性)、明度(明るさを示す属性)、彩度(鮮やかさの程度を示す属性)の3つの属性で定義したものがマンセル表色系であり、それを立体的に表したものがマンセル色立体である(図13.11)。

マンセル表色系の書式は、[色相][明度]/[彩度]で表現される。たとえば「7.5YR5/6」は、色相が黄色に近い橙色で、明度が5、彩度が6であることを示している。色相は1~10の10段階、もしくは2.5/5/7.5/10の4段階で表現される(5が色相の中心を表わす)。

3.2 画像の変換

(1) コントラスト変換

画像が明るすぎたり暗すぎような場合や、微妙な濃淡の変化を強調したいときに行なう変換。コントラストだけではなく、階調性をも変化させることができるため、階調変換ともよばれる。

1) コントラスト変換関数によるコントラスト変換

画像の濃淡を目的に応じた適当な関数で変換する方法であり、原画像の濃淡レベルを x とすると、 x をある関数 f を用いて、 $y=f(x)$ なる y に変換することによって画像の濃淡を変換する。

図13.12はフォトタッチソフトを使用して、入力原画像をいろいろな変換関数で変換した出力例である。トーンカーブとよばれる図のグラフは、入力画像の濃淡レベルを横軸(x)とし、出力画像の濃淡レベルを縦軸(y)とした $y=ax+b$ の式で表されるグラフである。直線の傾き a の値(ゲイン)がコントラストを表し、 y 切片 b の値(オフセットあるいはバイアス)で階調レベルを表す。

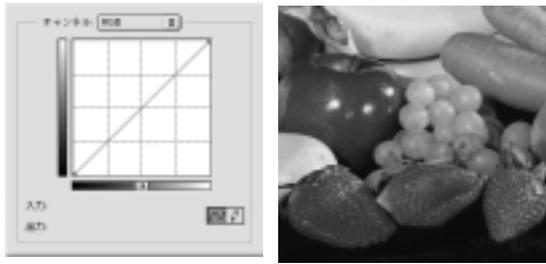
すなわち、傾き a が1のときはコントラストの変化はなく、 y 切片 b が0のときは階調の変化はない(図13.12(a))。 a が1より大きいとコントラストは高くなり(図13.12(d))、 a が1より小さいとコントラストは低くなる(図13.12(e))。 b が0より大きいときは明るさが増し(図13.12(b))、小さいときは暗さが増す(図13.12(c))。

その他にも、コントラスト強調を部分的に行う区線形変換や、グラフが曲線を描く対数変換関数など、さまざまな目的に応じた変換画像を得る方法が考えられている。

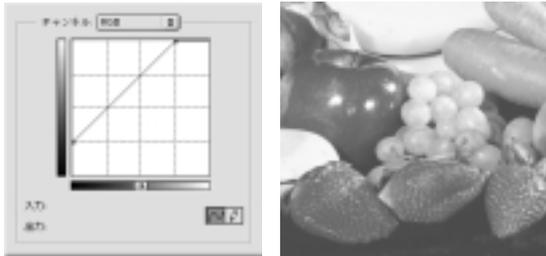
2) ヒストグラム変換関数によるコントラスト変換

画像の各濃淡レベルの画素数を、横軸に濃淡レベル、縦軸にその濃淡レベルの画素数を取ったグラフであらわしたものがヒストグラムである。ヒストグラム変換によるコントラスト変換は、この濃淡レベルのヒストグラムの形を変更することによって行なう。

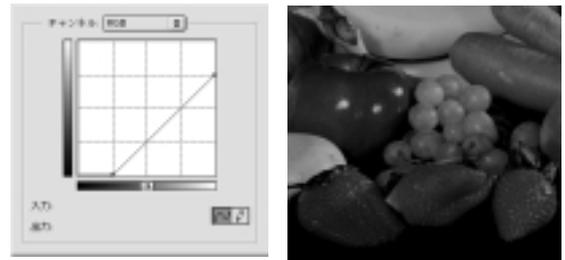
a. 原画像とトーンカーブ



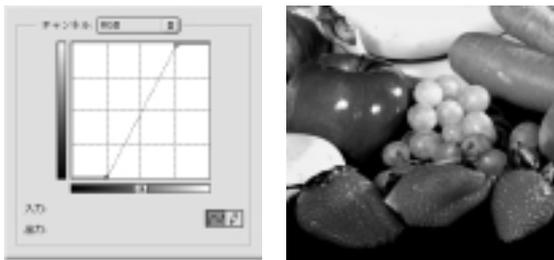
b. 階調を上げる



c. 階調を下げる



d. コントラストを上げる



e. コントラストを下げる

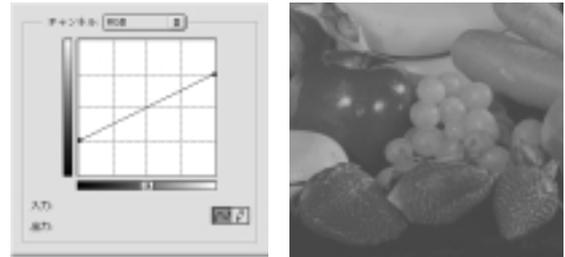


図13.12 コントラスト変換関数によるコントラスト変換

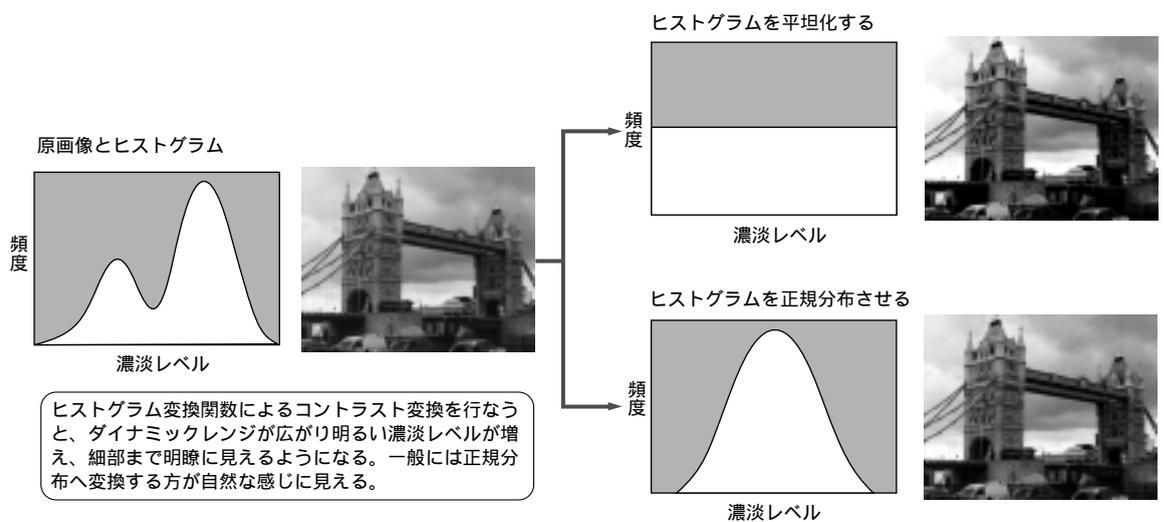


図13.13 ヒストグラム変換関数によるコントラスト変換

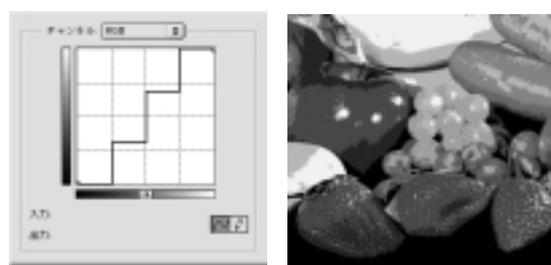
(2) 画像効果

グラフィックデザインにおける画像効果を求めるための画像変換方法には次のようなものがある。

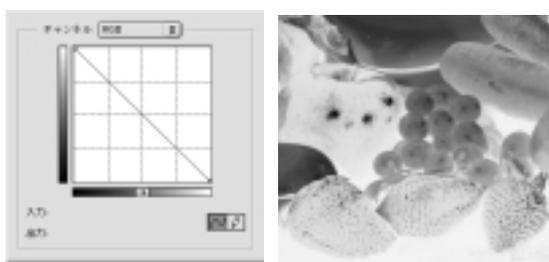
ハイコントラスト効果
原画像の濃度レベルを、しきい値を境に白と黒の2色にする処理によって得る効果。



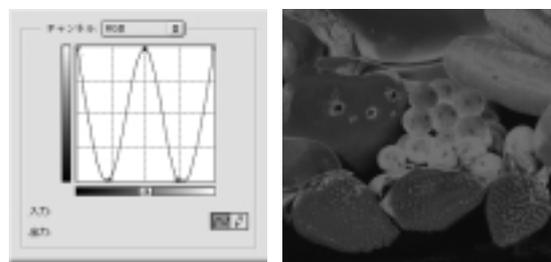
ボスタリゼーション効果
多階調の画像をいくつかの階調に制限して表示する効果。



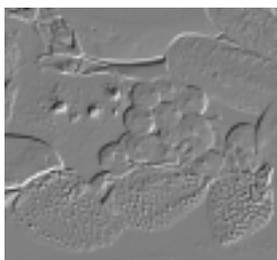
ネガ効果
原画像の濃度を反転させることによって得られる効果。



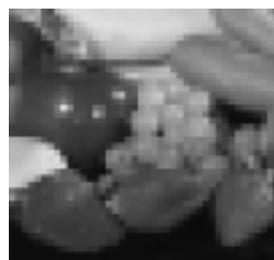
ソラリゼーション効果
ポジ画像とネガ画像が混ざり合ったような画像効果。



エンボス効果
画像の濃度差を利用して画像の一部を浮き出たように見せる効果。もともとは写真処理で用いられた手法の応用であり、写真処理ではラインプロセスとよばれる。写真処理の場合は、ネガ像とポジ像をわずかにずらして重ね合わせて印画紙に露光し、その効果を得る。



モザイク効果
画像をモザイク状にする処理によって得る効果。複数の画素からなる小さな正方形や長方形(サブマトリックス)を1単位とし、サブマトリックス内の画素の濃度の平均値で、そのサブマトリックスを塗りつぶす。



ぼかし効果
近傍の画素どうしの濃度差を小さくして、ぼかしのかった画像を得る効果。



図13.14 画像効果を求めるいろいろな画像変換

(1) マスク処理

画像Aと画像Bを合成する際に、画像Bのどの部分を画像Aに挿入するかを明示したマスクを作成し、それを元に2つの画像を合成する方法である。図13.15のマスク画像は、画像Bの特定部分を1とし、それ以外を0に2値化したものである。これを基にして、マスク画像の1の部分には画像Bを貼め込むことにより、AとBの画像を合成することができる。

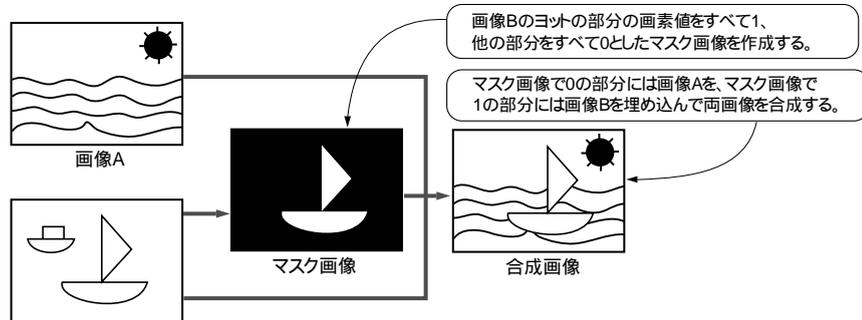


図13.15 マスク処理

(2) オーバーレイ処理

複数の画像の部分部分を排他的に合成するのではなく、ある画像の上に別の画像を重ねて表示するような処理がオーバーレイ処理である。オーバーレイでは、それぞれの画像の濃淡レベルに重みづけを行ない合成する。

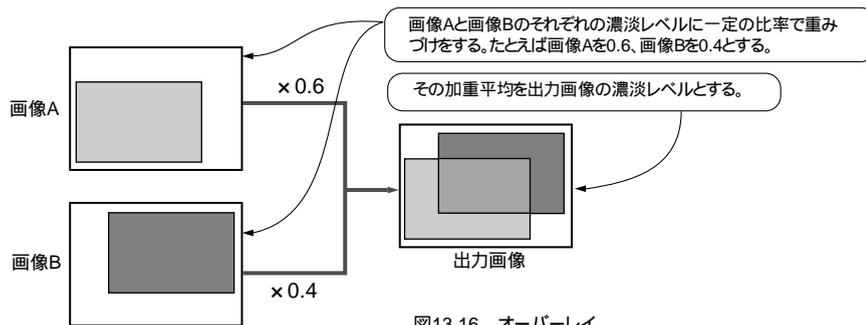


図13.16 オーバーレイ

(3) クロマキー処理

クロマキー処理は画像の中からある特定の色(クロマ)を持つ領域を抜き出し、そこに他の画像を埋め込む手法である。はめ込みたい部分の画像の色を特定することが可能ならば有効な手法である。

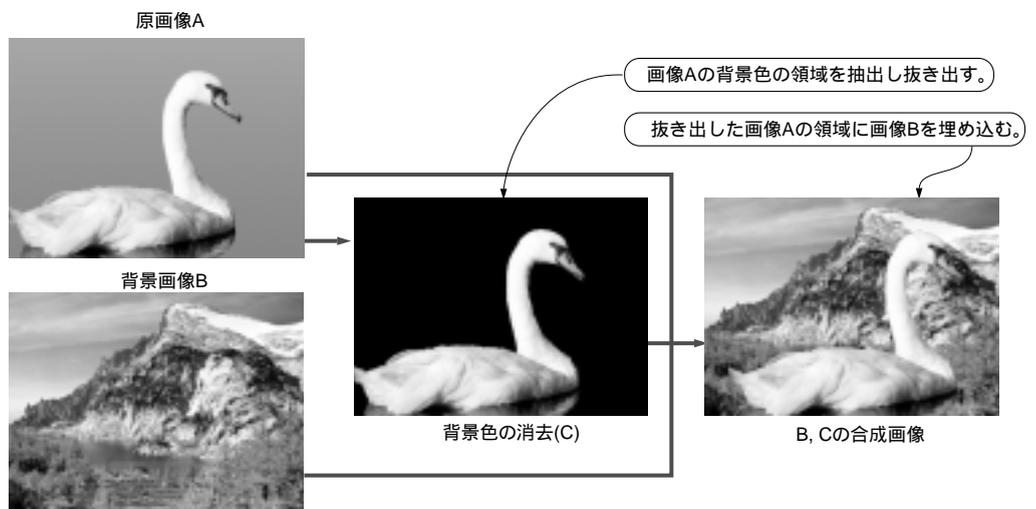


図13.17 クロマキー処理