

4 画像解析

(その1) 顕微鏡粒子画像中の粒子の計数

1. 目的

画像解析は、「見つける」「数える」「形を測る」「識別する」「記号・文字を読む」など複雑かつ多様な作業を画像処理により実現する技術であり、自然観測・生産現場・医療をはじめ様々な分野で利用されている。ここでは、顕微鏡で見た粒子画像において粒子の数を「数える」ことを目的とする。

2. 画像処理手法

図1に示すような粒子の顕微鏡画像において、粒子数を計測する。本実験では、図2に示す手順により、画像処理技術を用いて粒子の自動計数を実施する。

実験(その1)では、粒子と背景を分離し(2値化処理)、粒子のある場所の個数を数えることにより粒子の計数を行う。2値化処理を行うための前処理として、モノクロ化、シェーディングの補正を行う。また、粒子の個数を数える際の誤差を小さくするために、ノイズ除去と背景と粒子の分離を行う。

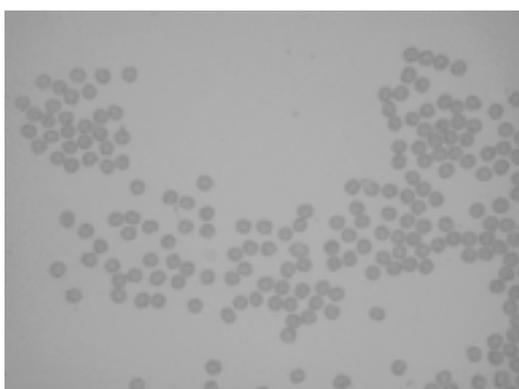


図1 粒子の顕微鏡画像の例

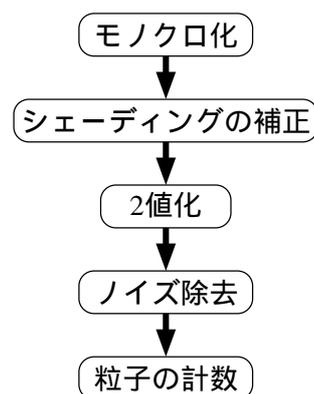


図2 粒子計数のフローチャート

モノクロ化

デジタル画像は、画像が格子状に分割され、格子点毎に数値として表現された2次元平面上の濃淡分布の情報として扱われる。1つ1つの格子点は画素(Pixel)と呼ばれ、各画素の濃淡値は数値化されている。モノクロ画像では、各画素の濃淡値は0から $2^n - 1$ まであり、 $n=8$ のとき256階調となる。カラー画像は、赤(R)・緑(G)・青(B)のRGB成分から構成されており、各色が256階調で表現されている場合、最大表示色は $1677216 (= 256^3)$ 色となる。

ここでは、カラー画像で与えられた粒子の顕微鏡画像のモノクロ化を行う。モノクロ画像の濃淡値 M は、カラー画像のRGBの値から、(1)式で算出される。

$$M = 0.299R + 0.587G + 0.114B \quad (1)$$

シェーディングの補正

普通に撮影された画像では、照明やレンズの周囲減光の影響により、画像中の場所によって明るさが均一でないことが多い。この不均一性をシェーディングと呼ぶ。シェーディングの補正を行うことによ

り、画像中の明るさは均一に近くなる。ここでは、(2)式に示すレンズのシェーディングのコサイン 4乗則により、補正を行う。

$$E = L \frac{\pi}{4} \left(\frac{D}{z} \right) \cos^4 \alpha \quad (2)$$

ただし、 E は単位面積当たりの明るさ、 L は単位立方体当りの輝度エネルギー、 D はレンズの直径、 z はレンズ中心から画像面までの距離（像距離と呼ぶことにする）、 α は入射光がレンズの光軸となす角度である。

また、画素の画像中心からの距離を d とすると、 α は(3)式で表される。

$$\alpha = \tan^{-1} \frac{d}{z} \quad (3)$$

画像には(2)式の影響があるとし、その影響を打ち消すことによりシェーディングの補正を行う。

2 値化 (図 3)

濃淡値に基準値（^{しきい} 閾値と呼ぶ）を設け、各画素の濃淡値が閾値よりも明るい、暗いかによってその画像を白（0）か黒（1）の 2 値へ変換することを 2 値化処理と呼ぶ。入力画像 $f(i, j)$ の閾値 t による 2 値化画像 $g_t(i, j)$ は以下の(4)式の操作によって得られる（ i, j は画像中の座標値）。

$$g_t(i, j) = \begin{cases} 1, & f(i, j) \leq t \text{ のとき} \\ 0, & f(i, j) > t \text{ のとき} \end{cases} \quad (4)$$

画像の 2 値化処理は、物体と背景とを分離するのに最も広く使用されている手法の 1 つである。計測する対象の濃淡分布が、背景の濃淡分布よりも十分に高い（もしくは低い）場合、2 値化処理により背景から対象を分離することができる。

閾値は、濃淡ヒストグラムを調べることで決定する。ヒストグラムでは、濃淡値が横軸、出現画素数が縦軸にとられており、本実験で用いる粒子画像では、図 4 に示すように谷の部分の濃淡値を閾値とすることができる（モード法）。



図 3 2 値化画像

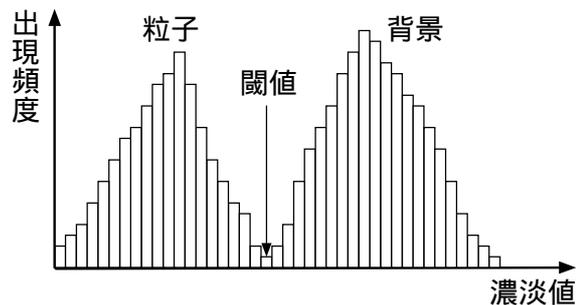


図 4 ヒストグラムと閾値

ノイズの除去 (図 5)

2 値化画像の孤立点除去によるノイズ除去を行う。ここでは、1 画素だけ孤立して存在する画素は細かいノイズであるとみなし、除去する。具体的には、黒画素（粒子部分）の 8-近傍がすべて白画素の場合には、その黒画素を白画素に変更する。また、白画素（背景部分）の 4-近傍が黒画素の場合には、黒画素に変更する。

ここで、4-近傍とは着目画素の上下左右にある4つの画素のことであり、8-近傍とは着目画素の周囲にある8つの画素のことである。ちなみに、着目画素とその4-近傍画素との連結関係を4-連結と呼び、着目画素とその8-近傍画素との連結関係を8-連結と呼ぶ。

さて、上記の処理を行っても、粒子内部に白画素領域が残ることがある。そこで、白画素領域のうち、面積が小さい部分を粒子領域に統合する。ここでは、白画素のうち統合可能な面積の閾値を設定し、閾値以下の面積を持つ白画素領域は黒画素に統合する。



図5 ノイズ除去画像



図6 4回収縮処理を行った画像

粒子の計数

黒画素領域について収縮処理（図6）を行った後、ラベリング処理を行って黒画素の連結領域（同じ色で繋がっている領域）の数を数えることにより、粒子を計数する。

収縮処理とは、同じ色が連結している領域の輪郭画素を取り除いて1層分小さくする処理のことである（図7）。画像 $f(i, j)$ を収縮した画像 $g(i, j)$ は式(5)の手順で得られる。

$$g(i, j) = \begin{cases} 0, & f(i, j) \text{ あるいはその4-近傍（または8-近傍）のいずれかが0のとき} \\ 1, & \text{その他のとき} \end{cases} \quad (5)$$

また、ラベリング処理とは、同じ連結成分に属する領域毎に番号（ラベル）を付ける処理のことである。ここで、連結成分とは4-連結あるいは8-連結で繋がっている1-画素成分（対象物画素）あるいは0-画素成分（背景画素）の集まりである。

粒子同士が接触している場合は、収縮処理を繰り返すことにより粒子が分離できる。ただし、繰り返し数が多すぎると、粒子が消滅してしまう。そこで、1回収縮処理を行う度に連結領域数と画像の変化を見て、粒子数を推定する。

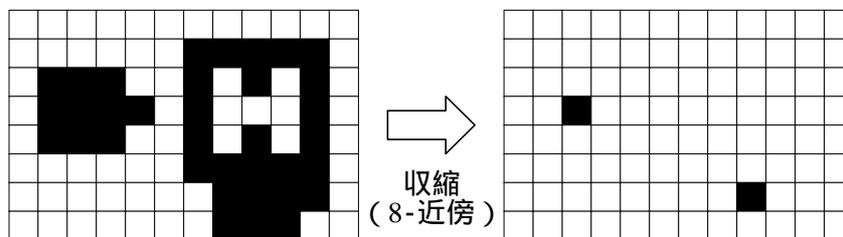


図7 収縮処理

3. 実験

次に示す手順で実験を行う。各ステップにおける処理結果は、画面への出力、及び生成された画像ファイル（拡張子は bmp）やテキストファイル（拡張子は txt）を画面上に表示することにより確認する。

モノクロ化

原画像（粒子の顕微鏡画像，カラー）をモノクロ化する．

プログラム：monochrome.exe

キーボード入力：カラー原画像ファイル名，モノクロ画像ファイル名

必要な画像ファイル：カラー原画像ファイル

生成される画像ファイル：モノクロ画像ファイル

シェーディングの補正

2-1：光量参照領域の選択

モノクロ画像ファイルを開き，背景成分のみからなる矩形領域を2つ選択して，それぞれの端点座標対を求める（ペイントのカーソル座標を読み取る）．この2つの領域のうち，1つはなるべく画像中心に近い部分を，もう1つはなるべく周辺の部分を選択する．

プログラム：Windowsのアクセサリのペイント

2-2：像距離の算出

選択した2つの矩形領域の平均画素レベルをそれぞれ求め，像距離 z を算出する．

プログラム：focus.exe

キーボード入力：モノクロ画像ファイル名，2つの矩形領域の端点座標対（左上と右下）

矩形領域を色表示した画像ファイル名

必要な画像ファイル：モノクロ画像ファイル

生成される画像ファイル：矩形領域を色表示した画像ファイル

画面への出力：像距離 z の値

2-3：シェーディングの補正

計算した像距離 z を用いて，シェーディングを補正する．

プログラム：shading.exe

キーボード入力：モノクロ画像ファイル名，推定像距離，補正モノクロ画像ファイル名

必要な画像ファイル：モノクロ画像ファイル

生成される画像ファイル：補正モノクロ画像

2値化

シェーディング補正を行ったモノクロ画像を2値化して，背景部分と粒子部分に分離する．

3-1：濃淡ヒストグラムの計測

シェーディング補正を行ったモノクロ画像のヒストグラムを求める．

プログラム：histogram.exe

キーボード入力：補正モノクロ画像ファイル名，ヒストグラムテキストファイル名

ヒストグラム画像ファイル名

必要な画像ファイル：補正モノクロ画像ファイル

生成される画像ファイル：ヒストグラム画像ファイル

生成されるテキストファイル：ヒストグラムのテキストファイル

シェーディング補正をしていないモノクロ画像に対してヒストグラムをとり比較してみる．

3-2：2 値化

ヒストグラムの画像およびヒストグラムのテキストファイル（Windows のアクセサリのメモ帳で開く）を参考に 2 値化の閾値を決定して，補正モノクロ画像の 2 値画像を得る．

プログラム：binary.exe

キーボード入力：補正モノクロ画像ファイル名，2 値化閾値，2 値画像ファイル名

必要な画像ファイル：補正モノクロ画像ファイル

生成される画像ファイル：2 値画像ファイル

ノイズ除去

4-1：ノイズ除去

2 値化画像の孤立点除去によるノイズ除去を行う．

プログラム：isolation.exe

キーボード入力：2 値画像ファイル名，孤立点除去画像ファイル名

必要な画像ファイル：2 値画像ファイル

生成される画像ファイル：孤立点除去画像ファイル

4-2：白画素領域のラベリング

白画素領域のラベリングにより，白画素の連結領域と各領域の面積を求める．

プログラム：labeling.exe

キーボード入力：孤立点除去画像ファイル名，ラベリングモード（1=白画素）

白画素ラベリング画像ファイル名，ラベリング領域面積のテキストファイル名

必要な画像ファイル：孤立点除去画像ファイル名

生成される画像ファイル：白画素ラベリング画像ファイル

生成されるテキストファイル：ラベリング領域面積のテキストファイル

4-3：小面積白画素領域の粒子部分への統合

粒子内部に存在する小面積の白画素領域を粒子部分に統合する．白画素ラベリング領域面積リストファイルの内容を参考にする（Windows のアクセサリのメモ帳で開く）．

プログラム：segmentation.exe

キーボード入力：白画素ラベリング画像ファイル名，面積閾値，白画素領域統合画像ファイル名

必要な画像ファイル：白画素ラベリング画像ファイル

生成される画像ファイル：白画素領域統合画像ファイル

粒子の計数

以下の手順を繰り返すことにより，粒子の計数を行う．

5-1：画像の収縮

背景粒子分離画像の黒画素領域について収縮処理を行う．

プログラム：contraction.exe

キーボード入力：被収縮画像ファイル名，収縮画像ファイル名，収縮処理の回数

必要な画像ファイル：被収縮画像ファイル

生成される画像ファイル：収縮画像ファイル

5-2：黒画素の連結領域の計数

ラベリング処理により，収縮した画像における黒画素の連結領域数を数える．

プログラム：labeling.exe

キーボード入力：収縮画像ファイル名，ラベリングモード（0=黒画素）

黒画素ラベリング画像ファイル名，ラベリング領域面積のテキストファイル名

必要な画像ファイル：収縮画像ファイル

算出される画像ファイル：黒画素ラベリング画像ファイル

算出されるテキストファイル：ラベリング領域面積のテキストファイル

画面への表示：ラベル数

4. 検討

- (1) カラー画像の粒子数を各自数え，実験で得られた個数との比較を行え．また，結果が異なる場合は，その原因を考えよ．
- (2) シェーディングの補正時において，中心から d_1 の距離での明るさが E_1 ， d_2 の距離での明るさが E_2 であったとする． d_1 ， E_1 ， d_2 ， E_2 を用い，(2)式，(3)式から z を求めよ．

(その2) 顕微鏡粒子画像中の粒子形状特徴の計測

1. 目的

顕微鏡で見た粒子画像において粒子の「形を測る」ことを目的とする．

2. 画像処理手法

図8に示すような粒子の顕微鏡画像において，粒子形状を計測する．本実験では，図9に示す手順により，画像処理技術を用いて粒子の形状特徴を自動的に計測する．

実験（その2）では，実験（その1）での～を行った後，連結している粒子を分離するために収縮・膨張処理を繰り返す．その後，エッジ検出を行い，エッジを2次曲線の方程式に当てはめて形状特徴を計測するとともに，粒子の周囲長・面積からも形状特徴を求める．

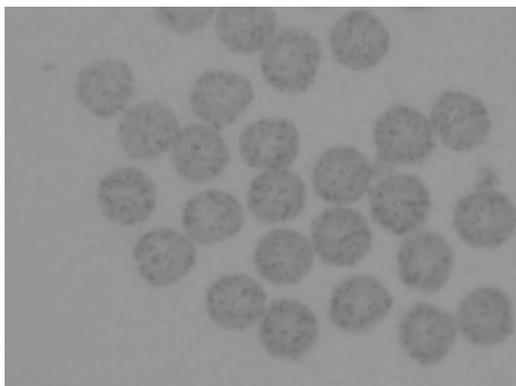


図8 粒子の顕微鏡画像の例

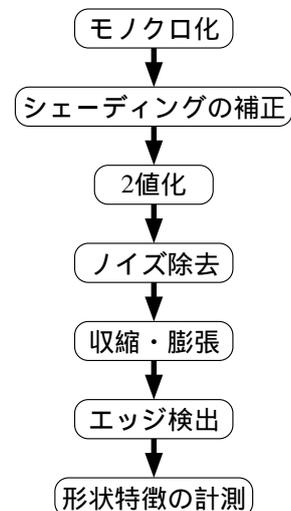


図9 粒子特徴計測のフローチャート

モノクロ化, シェーディングの補正, 2値化, ノイズ除去
 実験(その1)と同様.

収縮・膨張

収縮・膨張処理を行うことにより, 画像中の小さな孤立領域や細い線分を除去するとともに, 連結している粒子を分離する(図10, 図11).

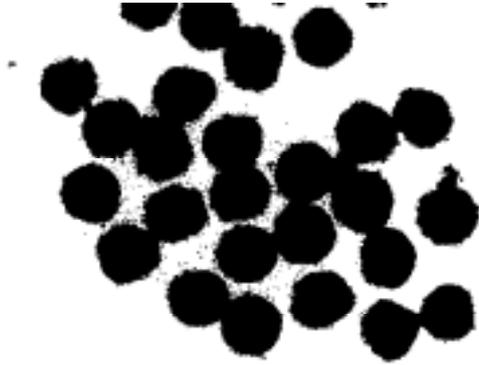


図10 収縮・膨張前の画像



図11 収縮・膨張後の画像

ここで, 膨張処理とは, 実験(その1)の で触れた収縮処理と逆の処理で, 同じ色が連結している領域の輪郭画素を外側に1層分増やし, 厚くする処理のことである(図12). 画像 $f(i, j)$ を収縮した画像 $g(i, j)$ は式(6)の手順で得られる.

$$g(i, j) = \begin{cases} 1, & f(i, j) \text{ あるいはその4-近傍 (または8-近傍) のいずれかが1のとき} \\ 0, & \text{その他のとき} \end{cases} \quad (6)$$

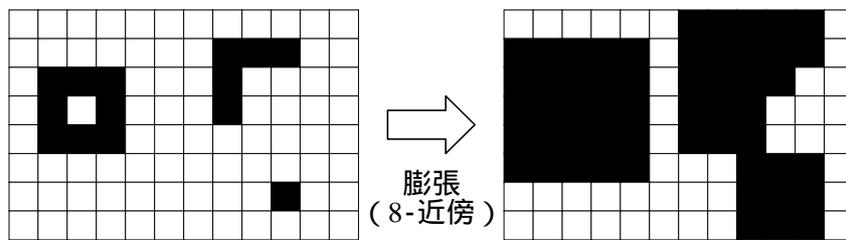


図12 膨張処理

エッジ検出

粒子の輪郭(エッジ)の検出を行う(図13).

形状特徴の計測

対象の形状がどの程度円に近いかを表す円形度を用い, 形状特徴を計測する. 円形度 e は(7)式で定義される.

$$e = \frac{4\pi S}{L^2} \quad (7)$$

ただし, S は対象の面積, L は周囲長であり, 真円のとき e は1となり, 複雑な形状ほどその値は小

さくなる。

まず、得られたエッジの2次曲線への当てはめを行う(図14)。一般に、2次曲線の方程式は(8)式で表される。

$$Ax^2 + Bxy + Cx^2 + Dx + Ey + F = 0 \quad (8)$$

(8)式において係数は6個であるが、5係数は残りの1係数の定数倍とすることができるので、ある係数を1とおき、残りの5係数を求めれば良い。従って、エッジ上の5点の (x, y) 座標を画像から求め、(8)式に代入し、連立方程式を解くことで、各係数を求めることができる。得られた2次曲線の係数を用いて周囲長・面積を求め、粒子の円形度を求める。

また、2次曲線への当てはめを行わずに、粒子のエッジの画素数と粒子1個の全画素数を数えることにより周囲長・面積を求め、同様に円形度を求める。



図13 エッジ検出結果

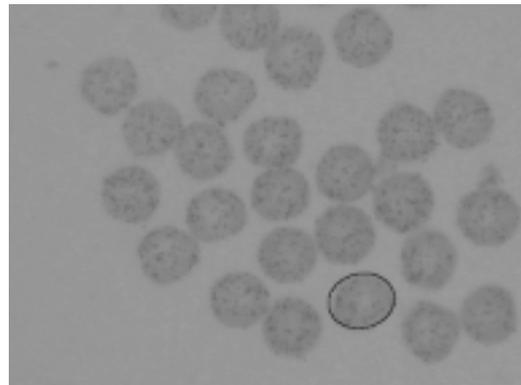


図14 2次曲線の当てはめ

3. 実験

モノクロ化, シェーディングの補正, 2値化, ノイズ除去
実験(その1)と同様。

収縮・膨張

手順5-1を n 回行った後、手順5-2を n 回行って、エッジを鮮明化する(n は1~5程度)。

5-1: 画像の収縮

実験(その1)と同様。

5-2: 画面の膨張

背景粒子分離画像の黒画素領域について膨張処理を行う。

プログラム: expansion.exe

キーボード入力: 被膨張画像ファイル(5-1で収縮を n 回行った画像ファイル)名
膨張処理の回数, 膨張画像ファイル名

必要な画像ファイル: 被膨張画像ファイル(5-1で収縮を n 回行った画像ファイル)

生成される画像ファイル: 膨張画像ファイル

収縮・膨張処理の回数を増やして(n を大きくして)結果を比較する。

エッジの検出

6-1：エッジの検出

エッジの検出を行う。

プログラム：edge.exe

キーボード入力：収縮・膨張処理画像ファイル名，エッジ検出画像ファイル名

必要な画像ファイル：収縮・膨張処理画像ファイル

生成される画像ファイル：エッジ検出画像ファイル

6-2：エッジ点位置の指定

エッジ検出済画像ファイルを開き，1つの粒子について，エッジ部分の座標を5箇所大まかに求め，座標値を得る。また，このとき粒子内部の点の座標（7-3で利用）を1箇所調べる。

プログラム：Windowsのアクセサリのペイント

6-3：エッジ点の位置自動補正

大まかに求めた座標値を用い，エッジ上の位置を正確に自動検出する。

プログラム：fitting.exe

キーボード入力：エッジ検出画像ファイル名，エッジ点の座標，検出確認用画像ファイル名

必要な画像ファイル：エッジ検出画像ファイル

生成される画像ファイル：検出確認用画像ファイル

画面への出力：正確なエッジ点の座標

形状特徴の計測

7-1：2次曲線方程式の算出

検出された5つのエッジ点の座標により，2次曲線の方程式の係数を求める。

プログラム：equation.exe

キーボード入力：エッジ点の座標

画面への出力：2次曲線の方程式の係数

7-2：2次曲線の重畳表示

得られた方程式の係数を入力して描画し，原画像に重畳表示して，一致度を確認する。

プログラム：ellipse.exe

キーボード入力：モノクロ画像ファイル名，2次曲線の係数，2次曲線重畳画像ファイル名

必要な画像ファイル：モノクロ画像ファイル

生成される画像ファイル：2次曲線重畳画像ファイル

7-3：粒子の周囲長と面積の算出

2次曲線の当てはめを行わず，画像から粒子の周囲長と面積を算出する。

プログラム：size.exe

キーボード入力：エッジ検出画像ファイル名，(6-3で求めた)正確なエッジ点の座標(1点)

(6-2で求めた)内部点の座標，エッジ及び内部点確認用画像ファイル名

必要な画像ファイル：エッジ検出済画像ファイル

生成される画像ファイル：エッジ及び内部点確認用画像ファイル

画面への出力：粒子の周囲長と面積

4. 検討

- (1) 収縮・膨張処理の回数が多いと、どのような現象が起こるか考察せよ。また、その原因は何か述べよ。
- (2) エッジ当てはめを行った 2 次曲線 $Ax^2 + Bxy + Cy^2 + Dx + Ey + F = 0$ を角度 θ 回転させることにより、 $\frac{(x-p)^2}{a^2} + \frac{(x-q)^2}{b^2} = 1$ (楕円) の形に変形せよ。ヒント: $\theta = \frac{1}{2} \tan^{-1} \frac{B}{A-C}$
- (3) 楕円の面積は πab , 周囲長は近似的に $2\pi\sqrt{\frac{a^2+b^2}{2}}$ となることから, (2)の結果を用い, 粒子の周囲長と面積を求め円形度を計算せよ。同様に, 楕円の方程式を用いずに得られた粒子の周囲長と面積から円形度を計算せよ。また, これらの結果について比較・考察せよ。

参考文献

- (1) 田村秀行監修: コンピュータ画像処理入門 (総研出版, 1985)。
- (2) 村上伸一著: 画像処理工学 (東京電機大学出版局, 1996)。
- (3) 谷口慶治編: 画像処理工学 - 基礎編 - (共立出版株式会社, 1996)。