



確認テストの解答

{ 其の一 }

【問題】

- (1) 自動的な閾値の設定の際にアルゴリズムMeanを使うとどのような違いが生じるか説明せよ.
- (2) 練習のため、あえて核を除外し「小さなゴミ」だけを測定するプロトコルを作成せよ.
- (3) マクロを書くことができる読者は、(2)をマクロにせよ.

【答え】

(1) Max Entropyによる下側閾値は389であるが、Meanによる下側閾値は366となる。この結果、核は若干大きめに分節化され、またバックグラウンドの余計なシグナルも分節化される。

(2) ①[Threshold...]でMax Entropyによる自動閾値設定を行う。②[Set Measurements...]で測定項目を本文と同じように設定する。③[Analyze Particles...]で、Sizeの設定を0-1にして行う。

(3) コードはほぼ本文にあるものと同じであるが、1カ所だけ書き換えれば良い。10行目のsize = 1-Infinityをsize = 0-1にする。

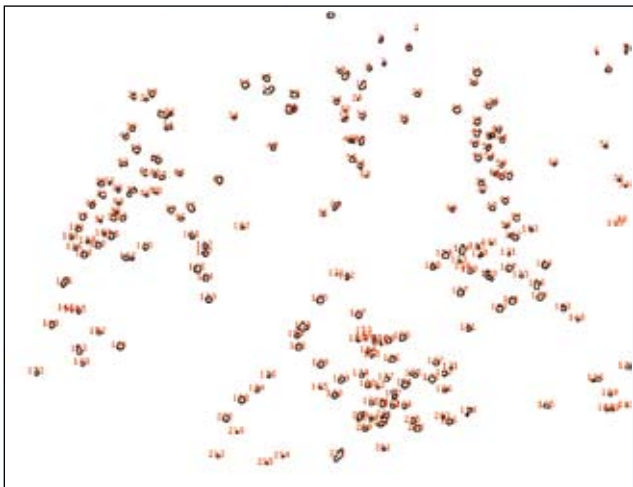
```
1   run("HeLa Cells (1.3M, 48-bit RGB)");
2   title = getTitle();
3   run("Split Channels");
4   selectWindow("C3-" + title);
5   setAutoThreshold("MaxEntropy dark");
6   run("Set Measurements...",
7       "area mean standard min centroid" +
8       " integrated limit redirect=" + "C1-" + title + " decimal=2");
9   run("Analyze Particles...",
10      "size=0-1 circularity=0.00-1.00 " +
11      "show=Outlines display exclude clear include");
```

{ 其の二 }

【問題】 サンプル画像[Hela Cells]の1番目のチャンネルはリソソームの画像である。ドット状のシグナルを確認できるだろう。[Trainable Weka Segmentation]と[Analyze Particles...]を用いてリソソームの数を数えよ。

【答え】

[Hela Cells]のチャンネルを分割し、赤のチャンネルに関して解析を行う。プラグインTrainable Weka Segmentationを用いて、ドット状のシグナルをclass1に、背景をclass2に指定する。ドット同士がつながって分節化されることがしばしばあるが、その間に背景を指定するように、何度か指定領域の追加を行う。満足できる分節化が行えたら、Create resultsで白黒の分節化画像を出力する。この二値化画像で[Analyze Particle...]を行い、リストされるオブジェクトの数は215個であった。なお、人



によってシグナルと領域の指定の仕方は異なってしまうのが通常であり、結果、リソソームの数は多少異なることが予想される。この場合にはどうしたらよいのだろうか？

領域を指定して機械学習のアルゴリズムはある法則性を学習する。この法則性は“分類モデル”と呼ばれ、ファイルとして保存することが可能である。この同一のモデルを使用すれば、まず再現性は確保される。また、モデルはブラックボックスではない。モデルは人間が理解できるような形で記述し直すことも可能であり（ただしもの

すごく長い記述になる）、この点で分類の詳しいプロセスも確認可能である。

なお、私がアノテートして作成した分類モデルは下記のリンク先からダウンロードできる。モデルを使用する際には、“Load classifier”でファイルをロードした後、“Apply classifier”で赤チャンネル、すなわちリソソームのシグナルの画像ファイルを選ばばよい。“Create probability maps instead of segmentation?”と聞かれるので、Noをクリックする。215個のリソソームがカウントされるか試してみよう。

<https://sites.google.com/site/imagejpp/articles/session5/classifierysi.model?attredirects=0&d=1>

誰が行っても同じ結果になることから、再現性という点で好ましい手段であると言える。