FAQ: What is the algorithm used in automatic thresholding?

What is the algorithm used by ImageJ to find the ideal threshold value?

The automatic thresholding function used by **Image>Adjust>Threshold** and **Process>Binary>Threshold** is an iterative procedure based on the *isodata* algorithm [T.W. Ridler, S. Calvard, Picture thresholding using an iterative selection method, IEEE Trans. System, Man and Cybernetics, SMC-8 (1978) 630-632.]

The online Image Processing Fundamentals tutorial discusses the isodata algorithm among several others.
Briefly, the procedure divides the image into *objects* and *background* by taking an initial threshold, then the averages of the pixels **at or below the threshold** and pixels **above** are computed. The averages of those two values are computed, the threshold is incremented and the process is repeated until the threshold is larger than the composite average. That is,

```
threshold = (average background + average objects)/2
```

The code in ImageJ that implements this function is the **getAutoThreshold()** method in the **ImageProcessor** class.

Note that there are many more methods for setting an automatic threshold (such as Otsu's method, entropy method, triangle method, k-means clustering, etc).

From:
http://imagejdocu.tudor.lu/ - **ImageJ Documentation Wiki**

Permanent link:
http://imagejdocu.tudor.lu/doku.php?id=faq:technical:what_is_the_algorithm_used_in_automatic_thresholding

Last update: **2010/01/26 11:07**