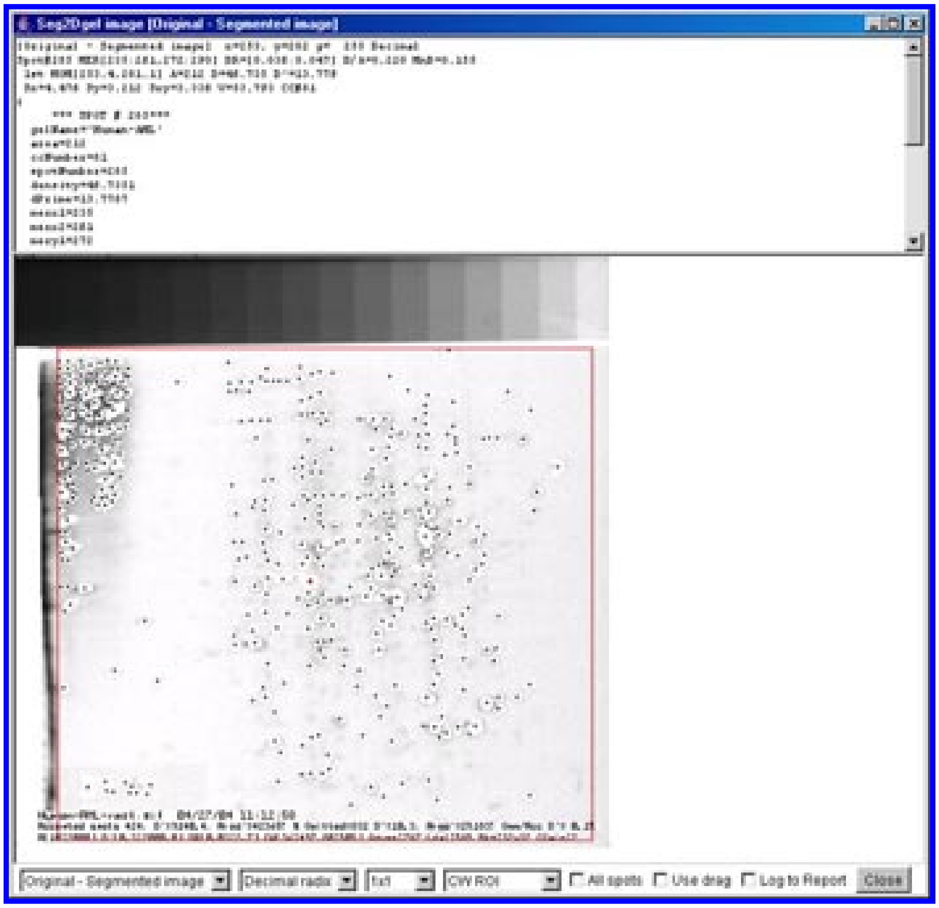


Seg2Dgel - 2D electrophoresis gel spot segmenter pre-alpha version

Welcome To Seg2Dgel

http://open2dprot.sourceforge.net/Seg2Dgel and (http://www.lecb.ncifcrf.gov/Open2Dprot/Seg2Dgel/ mirror)

```
Note: This pre-alpha level software is under construction and will
change.
Not all functionality described is fully implemented or fully
debugged.
This documentation will undergo revision as the program develops.
```

Introduction

Seg2Dgel is a Java 2D-electrophoretic gel segmentation program for finding and measuring the integrated density and position of spots in a gel or image with similar types of data. We are enhancing it to segment spots in 2D LC-MS images. It is a step [2] module in the pipeline analysis for the Open2Dprot project.
The segmentation is performed on a computing window region of interest (ROI) of a 2D gel image file. It uses the second derivative (laplacian) magnitude and direction of the gaussian-smoothed gel image as well as neighborhood connectivity properties in determining spot extents. It can handle grayscale TIFF, JPEG and GIF images. It generates a spot list of segmented spots which is saved in a Sample Spot-list File (SSF) in a variety of file formats (XML, tab-delimited, and human readable text).

The program may be run either interactively (-gui) with a graphical user interface (popup Report Window GUI) or under an OS shell command to implement batch (-nogui). If the demonstration switch is set(the default) to (-demo), it defaults to demonstration mode for some sample gels supplied with the distribution. Add the -nodemo switch to disable demo mode.

In the former GUI case (-gui), after the segmentation is finished, the user has the option of interactively viewing any of the images used by the segmenter and querying them for quantified spot data to look at small pixel windows (3x3 to 21x21 pixels) of the image data, or analyze the image or spot list data with dynamic histograms that allow limited data filtering. The latter is useful for titrating threshold sizing parameters. This is done using the Image Viewer window. The user may then modify the input switch options using the Command Options Wizard window and save the new options in a "Seg2Dgel. properties" file in the current project directory so that it may be used as the default switch options in subsequent running of the segmenter.

You can currently download the pre-alpha version and install it on your computer. Currently, Seg2Dgel is hardwired to start with the demo gel and with the -gui switch. The remainder of this home page contains links to some screen shots of the interactive GUI. The Web site contains some initial (rudimentary) documentation.

See the Reference Manual for more details on the segmenter. You can read about downloading and installing the program on your computer. The pre-alpha level source code is on the open2dprot. sourceforge.net CVS server. In addition, snapshots of the Java source code are available the SourceForge Files mirror server.

Examples - samples of screen shots

To give the flavor of running the segmenter, we provide a few screen shots of the graphical user interfaces and some images generated by the segmenter for the initial version of the segmenter. You can these images in the list below or view all of the screen shots in a single Web page.

- Initial user interface Report Window
- Initial command line options Wizard window
- Images that you can investigate with the ImageViewer window

1. original image
2. average image
3. propagated central core image
4. magnitude 2nd derivative image
5. background image
6. segmented image
7. original - segmented image
8. final user interface Window with timer

- Investigating histograms of image density or spot features with ImageViewer
    1. histogram of spot areas
    2. histogram of spot densities
    3. histogram spot density ranges
    4. histogram of image pixel densities
    5. horizonal slice of image pixel densities

Please contact us with suggestions and comments. If you make interesting changes in the source code, please send us a copy and describe your changes so we can merge them in the released version.

Contact us	Seg2Dgel is a contributed program available at Open2Dprot.sourceforge.net/Seg2Dgel Powered by SourceForge Logo	Revised: 11/07/2005

Table of Contents

Seg2Dgel

Demos & screen shots
Version
Downloads
Status
Revision history
FAQ
References

Source code
Project summary
CVS access
Browse CVS
Seg2Dgel Javadocs
O2Plib Javadocs
Files mirror

Other Web resources
SWISS-2DPAGE
WORLD-2DPAGE
2D-HUNT
LECB 2D data sets
ANL GelBank
Google search

Hosted at Open2Dprot
open2dprot.sourceforge.net

Powered by SourceForge Logo

Contact us

Revised: 11/07/2005

Seg2Dgel Program Contributors

The current Seg2Dgel program is the result of collaborative efforts by various groups and various projects. The current Java program is partly based based on some of the code refactored from the C/Unix GELLAB-II 2D gel spot segmenter program (sg2gii, 1993 version). That in turn had been refactored from a DEC-10/SAIL language version. The new Java version also includes code from the open source Flicker and MAExplorer programs.

Many individuals have been involved over the years in its development both as direct collaborators running 2D gel experiments as well as in offering suggestions for improvement including: Lew Lipkin, Eric Lester, Jim Myrick, Mary Robinson, Carl Merril, Heinz Busse, Peter Sonderegger, Pete Rogan, Ye Chung Wu, Steve Aley, Mike Alley, Rob Ashmore, Kyle Upton, Jeff Orr, Mort Schultz, George Carman, Mark Miller, Arthur Olson, Norman and Leigh Anderson, Sam Hanash, Jian Li, and many others in the electrophoresis community. Published descriptions of the algorithms for the GELLAB-II program on the Open2Dprot Web site and are in the References section of this Web site.

The core group developing the Seg2Dgel program source currently consists of

```
    * Peter Lemkin
    * Greg Thornwall
    * Jai Evans
```

Contact us	Seg2Dgel is a contributed program available at open2dprot.sourceforge.net/Seg2Dgel	Revised: 10/04/2005

Table of Contents

Seg2Dgel

License
PDFs
Tutorials
Demos
Version
Downloads
Status

- Revision history
- FAQ
- References

Source code

- Project summary
- CVS access
- Browse CVS
- Seg2Dgel Javadocs
- O2Plib Javadocs
- Files mirror

Other Web resources

- SWISS-2DPAGE
- WORLD-2DPAGE
- 2D-HUNT
- LECB 2D data sets
- ANL GelBank
- Google search

Hosted at Open2Dprot
open2dprot.sourceforge.net

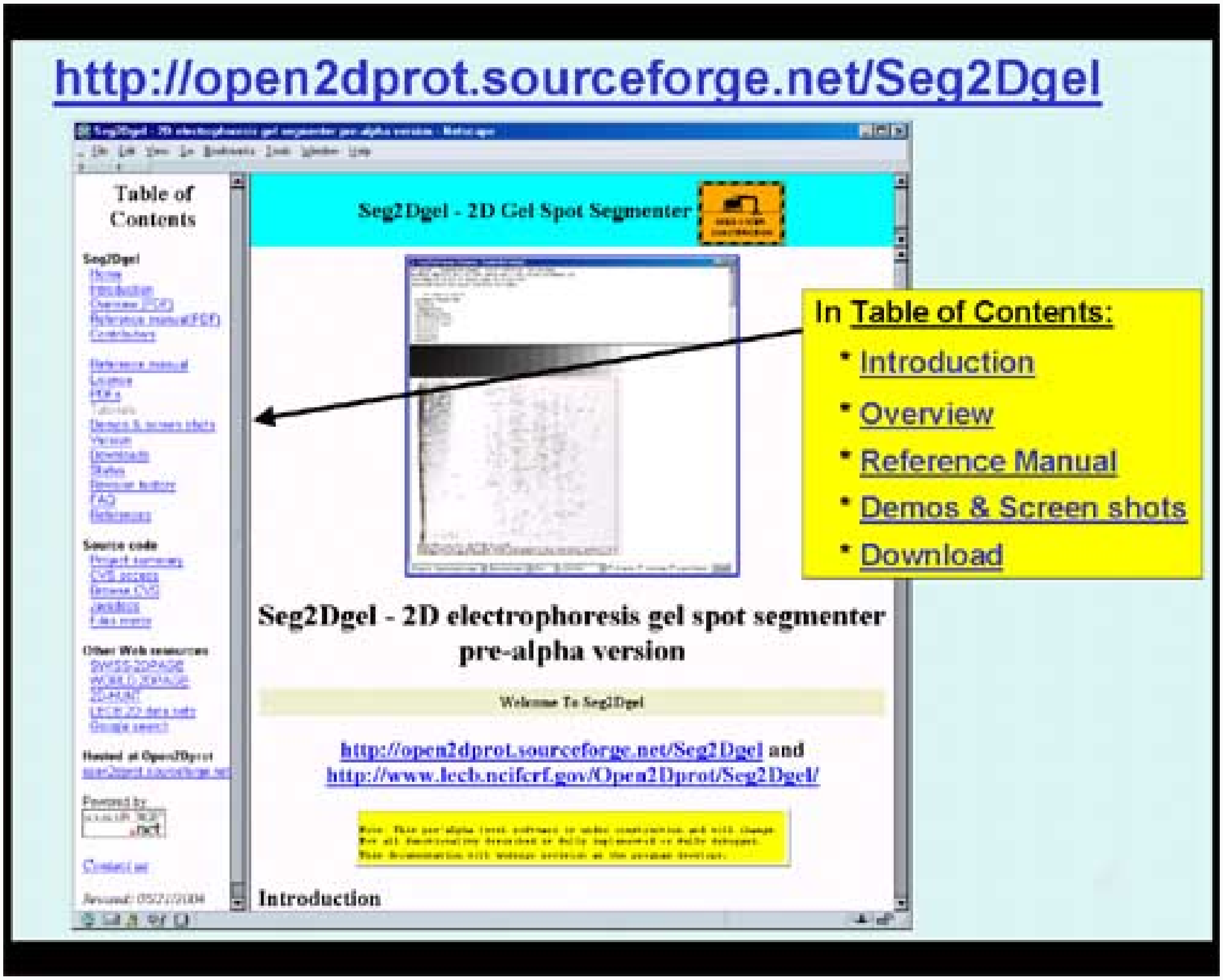
Powered by SourceForge Logo

Contact us

Revised: 11/07/2005

Seg2Dgel Reference Manual

Seg2Dgel is a Java 2D electrophoreic gel segmentation program for finding and measuring the integrated density and position of spots in a gel or similar type of data. Figure 1 shows the home page. It is a one of the modules in the Open2Dprot project.



http://open2dprot.sourceforge.net/Seg2Dgel

Seg2Dgel - 2D Gel Spot Segmenter

Seg2Dgel - 2D electrophoresis gel spot segmenter pre-alpha version

Welcome To Seg2Dgel

http://open2dprot.sourceforge.net/Seg2Dgel and http://www.lecb.ncifcrf.gov/Open2Dprot/Seg2Dgel/

Introduction

In Table of Contents:

- Introduction
- Overview
- Reference Manual
- Demos & Screen shots
- Download

Figure 1. Home page for the Seg2Dgel program. The Web site, at http://open2dprot.sourceforge.net/Seg2Dgel contains the software (with downloadable binaries and source code), as well as documentation.

The following description will tell you what the options are for the segmenter and how to run the program - both interactively and from the command line. If you interact with it through the windowing system, then Figure 2 shows a screen view of the Seg2Dgel reporting window.

```
Seg2Dgel - finished at: 08/06/04 16:41:35
Seg2Dgel $Date: 2004/08/05 20:20:17 $ - $Revision: 1.17 $ (Open2Dprot)
Today's date is 08/06/04 16:41:36
Switches:  -gui
Switches:  -gui -timer -accesssionDB:accession.txt -ssfFormat:X
           -restOfPixFile -segmentedPixFile -ctlCorePixFile -lowpassFilter:7
           -laplacian:5 -backgroundSize:32 -CCminSize:2 -thrArea:10,2000
           -thrDensity:0.3,2000 -thrRange:0.03,4.0 -projDir:demo/

Reading accession database file: demo\xml\accession.txt
Computing window region and sizing parameters
----------------------------------------
Window [14:475,74:505] (pixels) [rows,cols]=[512,512]
Spot Area sizing limits (10:2000) (pixels**2)
Integrated Density sizing limits (0.300:2000.000) (od)
Density difference sizing limits (0.0300:4.0000) (od)
Zonal Notch filter background window size: 32X32 (pixels)

Summary of segmented spot statistics
-----------------------------------
Total of 2015 accepted D spots accumulated density=61294.0, area=11752
Total of 933 accepted D' spots accumulated density=13292.0, area=76350
Total of 1082 omitted D' spots accumulated density=119.4, area=23119
Omitted(D')/Accepted(D') spots failing D' resizing= 0.9%
Total # spots failed Area sizing[areaT1,areaT2]=[1540, 0]
Total # spots failed Density sizing[densityT1,densityT2]=[3150, 0]
Total # spots failed ODrange sizing[ODrangeT1,ODrangeT2]=[3109, 0]

List of image files and generated files
----------------------------------------
Input pix file          [demo\ppx\gel-HM-019.gif]
SSF file                [demo\xml\gel-HM-019.xml]
Central Core pix file   [demo\tmp\gel-HM-019-cc.gif]
Average pix file        [demo\tmp\gel-HM-019-avg.gif]
Mag 2nd Deriv. pix file [demo\tmp\gel-HM-019-mag.gif]
Background pix file     [demo\tmp\gel-HM-019-bkg.gif]
Segmented pix file      [demo\tmp\gel-HM-019-seg.gif]
'RestOf' pix file       [demo\tmp\gel-HM-019-rest.gif]

FINISHED! The Sample Spot-list File, SSF, is demo\xml\gel-HM-019.xml
Run time =0:0:2 (H:M:S) or 2.5 seconds

Breakdown of times and memory usage for steps in the segenntation
-----------------------------------------------------------------
Step []                  0.0 seconds  532.7Mb-tot 501.5Mb-free
Step [Reading gel]       0.2 seconds  532.7Mb-tot 528.9Mb-free
Step [Init]              0.0 seconds  532.7Mb-tot 526.7Mb-free
Step [Average]           0.1 seconds  532.7Mb-tot 526.7Mb-free
Step [Ctrl core]         0.7 seconds  532.7Mb-tot 500.4Mb-free
Step [Background]        0.0 seconds  532.7Mb-tot 500.4Mb-free
Step [Estimate D']       0.0 seconds  532.7Mb-tot 500.4Mb-free
Step [Rethreshold D']    0.0 seconds  532.7Mb-tot 500.4Mb-free
Step [Save images]       1.1 seconds  532.7Mb-tot 525.7Mb-free
Step [Save SSF]          0.3 seconds  532.7Mb-tot 504.5Mb-free
Step [Done]              0.0 seconds  532.7Mb-tot 504.5Mb-free

Seg2Dgel - finished at: 08/06/04 16:41:38
```

Figure 2. Screen view of the Seg2Dgel program Graphical User Interface. This screen shot shows the status after a segmentation of the demo gel (Human-AML 512x512 8-bit pixels) has completed. The sizing parameters are shown at the top. Next the summary statistics is reported. This is followed by a list of generated files (spot list and segmented spot images). Finally, since the timer was enabled, it gives a breakdown of how much time was spent in different parts of the program. [This was run on a 1Ghz pentium-4 Windows 2000

system.] The pull-down menus at the top of the window used to invoke File operations, Edit, View, Quantify (i.e. spot segmentation), and Help commands. Although, the button commandss at the bottom of the window are also available in the menus, they are replicated as buttons for convenience. The **Clear** button will clear the window. **SaveAs** will let you save the text in a text file. The **Edit options** button pops up a window to let you edit the options including the gel name, computing window, sizing thresholds etc. The **Segment** button starts the segmenter on the current options. You can stop the segmenter by pressing the **Stop program** button. After you have segmented the image, you can review the images by pressing the **Image Viewer** to popup the image viewing window. A status area appears in the lower left corner and reports the current state of the segmenter during an analysis. It shows "Done" since the segmentation had completed.

1. Introduction

Seg2Dgel is a Java 2D electrophoresis gel segmentation program for finding and measuring the integrated density and position of spots in a sample gel. We are adding other features to enable it to segment 2D LC-MS data presented as images. Seg2Dgel is part of the Open2Dprot project (http://open2dprot.sourceforge.net/). The segmentation is performed on a computing window region of interest (ROI) of a 2D gel image file. It uses the second derivative (laplacian) magnitude and direction of the gaussian-smoothed gel image as well as neighborhood connectivity properties in determining spot extents.

The program may be run either interactively (-gui) or under an OS shell command to implement batch (-nogui). If the -gui mode is used, after the segmentation is finished, the user has the option of interactively viewing any of the images used by the segmenter and querying them for quantified spot data or look at small pixel windows (3x3 to 21x21 pixels) of the image data and the intermediate images used in the segmentation. The user may also modify the input switch options and save the new options in a "Seg2Dgel.properties" file in the current project directory so that it may be used as the default switch options in subsequent running of the segmenter.

NOTE: the "Seg2Dgel.properties" file is not currently used in the current version.

Project directory structure for Open2Dprot and Seg2Dgel

All Open2Dprot programs assume a project directory structure. This must exist for the program to proceed. You can either create the structure prior to running any of the programs or you can create it on the fly using the -projDir:*user-project-directory*. It will lookup and/or create the following sub-directories:

```
user-project-directory/batch/ - batch temporary files - [not used]
user-project-directory/cache/ - cache files - [not used]
user-project-directory/ppx/ - original input image files
user-project-directory/rdbmx/ - RDBMS CSD database files - [not used]
user-project-directory/tmp/ - generated derived image files
user-project-directory/xml/ - accession DB, generate SSF files
```

The use of these directories is discussed in the rest of this document.

Input gel image file

The sample to be segmented is specified by its image file name using the -sample switch with or without the file extension (e.g., -sample:plasma27.tif or -sample:plasma27). The file extension is determined by looking up the image in the ppx/ project subdirectory at run time.

The application looks up the sample in the accession database (if the latter exists in xml/accession.xml or as specified using the -accessionFile switch). It then gets additional information about the sample including the computing window and the grayscale calibration if they exist in the database. If no entry exists, the entire image is analyzed and no grayscale calibration is used. The sample name is the image name without the file path or file extension. The Open2Dprot http://open2dprot.sourceforge.net/Accession pipeline module is used for entering samples into the accession database.

[Status: The Open2Dprot Accession module program is not released yet. The accession database could be edited manually as either XML (accession.xml), or tab-delimited text (accession.txt) with Excel.]

Input images may be in TIFF (.tif, .tiff), JPEG (.jpg), GIF (.gif), or PPX (.ppx GELLAB-II) format. TIFF images may be 8-bits/pixel through 16-bits/pixel, whereas JPEG, GIF, and PPX are 8-bit images. Gray values in the images files have black as a 0 pixel value. Gray values are mapped to 0 for white and maximum pixel value for black. This allows us to compute integrated density/spot (if the gel is calibrated in optical density) or integrated grayscale/spot. For purposes of this documentation, assume that the input image names without the file extension is *gelName* so in the case of the example plasma27.tif file name, *gelName* would be "plasma27" would be the sample name.

For color images, the RGB data is first mapped to grayscale using the NTSC color to grayscale transform (gray = 0.33*red + 0.50*green + 0.17*blue) [FUTURE].

The input gel image file is kept in the *user-project-directory*/ppx/ sub-directory. If it is not in that directory, then it will prompt you for the name of a project directory and create it and the sub-directories if they do not already exist. Then it will copy your input file to project directory so that the database is consistent and may be used by the other Open2Dprot analysis pipeline programs.

[STATUS: currently NTSC conversion is not available.]

Output spot list file in Sample Spot-list File (SSF)

The output is a quantified spot list in various ASCII formats including XML and tab-delimited as well as the historical GELLAB-II SSF formats. Images may be generated (a the user's option) to view the segmented spots and various intermediate images used by Seg2Dgel.

The data output file is called the Sample Spot-list File (SSF) and is saved in the *user-project-directory*/ xml/ directory. The generated name the same as the base name of the image file but with a different extension depending on the output format. The possible extensions are: .ssf (for ASCII format compatible with GELLAB-II), .xml (XML format), and .txt (tab-delimited format). The format is specified by the -ssfFormat:{**F** | **G** | **T** | **X**} command line switch.

[STATUS: currently XML -ssfFormat:**X** is the default format, but the XML generated will change with new changes in MIAPE.]

Output images files

You may save various images used in segmenting the input image. These are mapped to 8-bits reguardless of the actual internal resolution and saved as 8-bit JPEG (.jpg) in the *user-project-directory*/tmp/ directory at the same directory level as the input image file.

The segmented output image is -seg.jpg. If the central core image is to be saved (by specifying the -ctlCorePixOutputFile switch), then it is saved as file tmp/*gelName*-cc.jpg. In addition, the average image is written out as *gelName*-avg.jpg, background notch filter image *gelName*-bkg.jpg, the Laplacian magnitude as *gelName*-mag.jpg. If -restOfPixOutputFile switch is specified, then an additional output image *gelName*-rest.jpg is created which is the original gel less the segmented spots (i.e., *gelName*-seg.jpg).

[STATUS: currently the output images are GIF (.gif) images.]

Spot overlays in the output image

Spots may be indicated in the 'seg' or 'restOf' output images in several ways. The -drawDot:{**O** | **Z**} switch will draw a dot, the -drawDot:{**O** | **Z**} switch will draw a plus ("+"), and the -drawBoundary:{**O** | **Z**} switch will draw a boundary. It will draw it in the default 'Z' (segmented) image rather than in a copy of the 'O'riginal image. The **-drawMinEnclRect:{P|O}** switch to draw minimum enclosing rectangle spot overlays.

Calibration grayscale data in the accession database file

If the grayscale is to be calibrated in particular units, this is done through the entry for the sample in the accession database. The accession database entry specifies 4 items: a short list (15 to 25 entries) of calibration values sorted in assending order, a corresponding list of gray values also in assending order, calibration units (e.g., "optical density"), and an abbreviation for the units (e.g., "od"). It is assumed that the image pixels when mapped to calibration units (such as Optical Density (OD), Counts-Per-Minute (CPM), etc.) will be stochiometric so that a spot's integrated density corresponds to the spot's protein concentration. How well this relationship holds helps determine the accuracy of the quantification. We use a piecewise-linear extrapolation to map this list into a gray-scale to calibration-units lookup table. This is used in all image pixel density computations so that the computations are performed in calibration-units space rather than grayscale space for more accurate quantification. The Open2Dprot http://open2dprot.sourceforge.net/Accession pipeline module is used for editing and defining this calibration data for each sample.

Use of calibration data during analysis

If there is a grayscale to optical density (or other units, counts/minute, etc.) calibration, it is assumed that it is specified in the accession database. The accession database may be read if it exists when the sample is processed. A calibration entry consists of calibration information including grayscale to optical density, microns/pixel, the computing window where the sample image is to be segmented, etc. The calibration maps gray scale pixel values to a calibrated value (such as optical density, counts/minute, etc.) that may correspond to protein concentration if the protein sample quantification method (dye, radioautography, etc.) and the scanner have a reasonably linear or know mapping. If there is no calibration specified, it does a 1:1 mapping of grayscale to "Grayscale" units.

The grayscale to calibration grayscale map is be specified by pairs of calibration (grayscale-values, calibrated-units-values). This is then used to construct a piece-wise linear calibration map to use as a table lookup to map grayscale values to calibration values. If the mapping is reasonably well behaved, for a reasonably smooth curve, 15 to 25 points will suffice. The computing window is the valid region in image that should be segmented. Regions outside of this are ignored.

[STATUS: Calibrations can be optionally be specified in the accession database file if it exists. The calibration procedure will be available in the http://open2dprot.sourceforge.net/Accession pipeline module of Open2Dprot. This will allow users to generate and add the calibration to the accession database for samples.]

2. Seg2Dgel spot segmentation algorithm

The segmentation algorithm is described in two parts. The first part describes the global image processing operations. The send part describes individual spot segmentation operations.

Global image processing algorithm

1. The segmenter first computes a smoothed average image using gaussian filtering of the original image.
2. Then it computes the Laplacian of the smoothed image and defines central core image pixels (of 'unlabeled' pixels with value UNLABELED_CC_CODE) where the laplacian pixels are < 0 in both x and y directions.
3. The next step is to find all of the spots in a sequential raster search of the central core image. Each spot is completely analyzed before it goes on to the next spot. When it finds an unlabeled pixel

with code UNLABELED_CC_CODE) it processes pixels belonging to that spot by labeling the 4-neighbor connected components as spots as the central core (CC) with a pixel value sequenced or successive spots by modulo values in the range [MIN_CC_CODE : MAX_CC_CODE]. The 8-neighbor connected background pixels have a 0 or BKGRD_CODE pixel value label. It then propagates the CC pixels to create a propagated central core (PCC). The PCC pixels are labeled with a value in the [MIN_PCC_CODE : MAX_PCC_CODE] range by adding PCC_BASE_CODE offset to the CC value for that spot. The propagation uses a variety of heuristic methods with fairly good noise immunity until the propagation cuts off where the laplacian is a maximum or noise is reached. It then smoothes the spot creating the final PCC.

4. After all spots are initially segmented, it then computes a background density image by first subtracting the spots just found from the original image. This "rest of the image" is then smoothed using a zonal notch filter with a large averaging window (e.g., 32x32 to 128x128 pixels depending on the scanner pixel resolution). It will then use the background image as a table lookup of backgrounds at the centroids of the spots.

5. Then re-test all spots to see if there new density' (density corrected for background) is < the minimum density d1 defined by the -thrDensity:*d1,d2* switch option. Spots that no longer pass the threshold test are removed from the segmented image and from the spot list.

6. Finally it writes out the spot list in the format specified by the -ssfFormat switch option. If images are to be saved in disk files, these are also written out at this time.

The following algorithm describes the processing each spot undergoes during its segmentation.

Spot finding algorithm

The steps in finding the final PCC for the current spot are enumerated as follows. This algorithm is iterated for each spot as it traverses down the image in a raster pattern.

1. Find the Central Core (CC) for the current spot.
2. If the central core size is < ccMinSize threshold, then delete the spot.
3. If the -splitSpots:{B | R | T},*minCCsplitSize* option is enabled and central core size is greater than minCCsplitSize, then try to split the spot into several spots.
4. If the -saturateSpots:*percentThreshold* option is enabled, then if the (maxOD of the spot)/(maxOD in the image) is > percentThreshold then try to fill holes in the saturated spot with CC pixels.
5. Remove concavities in the current CC spot.
6. Assign the next CC number (modulo the range [MIN_CC_CODE:MAX_CC_CODE] range to the current spot.
7. If -optimizeFBL is enabled, then remove interior CC pixels to speed up the subsequent steps. This is useful for segmenting very large spots or when using very high pixel resolution.
8. Propagate the CC to a PCC (Propagated Central Core) by looking for the maxima away from the center of the spot in all directions. The pixel value assigned is the CC value + PCC_BASE_CODE.

This propagation is terminated by various conditions including running into another spot, noise, etc. If the image is not adequately smoothed, this step will not work very well.

9. Fill corners of the PCC.
10. Fill holes in the PCC.
11. Round corners of the PCC.
12. If the -optimizeFBL was enabled, then restore the interior CC pixels saved in step [7].
13. Finally, compute the spot features using data from the original image (not the averaged image) and save the features for this spot in a list of all spots.

It uses the final PCC to compute spot features including:

```
    density weighted centroid: (xMom, yMom),
    standard deviation and covariance spot size: (Sx, Sy, Sxy)
    integrated density: density
    integrated density: meanDensity
    spot area: area
    max and min OD: (minD, maxD)
    M.E.R. (minimum enclosing rectangle): [merX1:merX2, merY1:merY2]
```

Using these features, it estimates and includes the spot volume computed as

```
    volume = 4 * Sqrt(PI) * maxD * Sx * Sy.
```

It then filters out spots that do not pass the threshold ranges test

```
   (area in [a1:a2]) and (density in [d1:d2]) and (od diff in [o1:o2])
```

The od diff is (maxD - minD). The thresholds may be set by the command line switches

```
  -thrArea:a1,a2  -thrDensity:d1,d2  -thrRange:o1,o2
```

After all spots are found, it computes a background image from the original image less the spots. This is then used to compute a zonal notch filter background image from this image. It uses the background image as a table lookup of backgrounds at the centroids of the spots. It also adds background related spot features including:

```
    mean background density: meanBkgrd
    background density: densityBkgrd
```

It then computes the density corrected for background,

```
     density' = density - densityBkgrd * area
```

for each spot and then retests Density' agains the lower density limit *t1* -thrDensity:*t1*,*t2*. Spots failing this test are then removed. It then adds the spot feature:

```
    density'
```

Image smoothing operations

The first image convolution operation performed is needed to smooth the image. If the image is not sufficiently smoothed, it will not find the edges of the spots correctly. segmentation: Gaussian lowpass filter smoothing and a Laplacian. The Gaussian smoothing size is specified by one of the following switches -lowpassFilter:{**3** | **5** | **7** | **13**} size (i.e., 3x3, 5x5, 7x7, 13x13). If you do not specify the filter, then it is defaulted from the image size.

The -averageOD switch does the image averaging computations in OD space rather than grayscale space.

The averaging filters are defined as follows:

```
  a) use -lowpassFilter:3 switch
      1 2 1
      2 4 2  divided by 16
      1 2 1

  b) for the -lowpassFilter:5 switch
      1 1 2 1 1
      1 2 4 2 1
      2 4 8 4 2   divided by 52
      1 2 4 2 1
      1 1 2 1 1

  c) for the -lowpassFilter:7 filter (Miller & Olson)
      4  -6  -12  -14  -12   -6  4
     -6   9   18   21   18    9 -6
     -12 18   36   42   36   18 -12
     -14 21   42   49   42   21 -14  /441
     -12 18   36   42   36   18 -12
     -6   9   18   21   18    9 -6
      4  -6  -12  -14  -12   -6  4

   d) -lowpassFilter:13,size.
      Gaussian filter Cij is convolved with image data Jij and
      then normalized
     j11 ... j1h      c11 ... c1h
         ...      *       ...
     jd1 ... jdh      cd1 ... cdh
```

Image Laplacian operations

Similarly, there are three symmetric Laplacian edge detection filters: 3x3, 5x5 and BusseFilter specified by -laplacian:3, -laplacian:**5**, and -laplacian:**B**,*gridSize*,*mode* where mode is **P** or **C**. In the Busse Laplacian, the gridSize defaults to 3x3. If the mode being 'P', pixel weights are used, for 'C' 3x3 averaged pixel weights are used. The default Laplacian is -laplacian:**3** is the default 3x3 laplacian. For a 250 microns/pixel image, the -lowpassFilter:7 filter is optimal while the -laplacian:**13** one is better for a 170 microns/pixel image. In addition, there are two line-filter Laplacian filters useful for detecting short line-like objects (as in some 2D LC-MS images).

The -differenceOD switch does the image Laplacian computations in OD space rather than grayscale space.

The Laplacian filters are defined as follows where Where, a CC pixel is defined as an UNLABELED_CC_CODE (i.e., 1) if both dX and dY < 0, else it is define as a background pixe BKGRD_CODE (i.e., 0). The magnitude is computed as either the city-block distance(faster) or Euclidean distance of (dX, dY).

```
 a) use -laplacian:3 switch
         0  0  0
   dX = 1 -2  1
         0  0  0

         0  1  0
   dY = 0 -2  0
         0  1  0

 b) use -laplacian:5 switch
         0  0  0  0  0
   dX = 1  1 -4  1  1
         0  0  0  0  0

         0  1  0
         0  1  0
   dY = 0 -4  0
         0  1  0
         0  1  0

 c) use -laplacian:B,mode,gridSize switch.
  This switch is like -laplace:3 but with a pixel spacing specified
  by gridSize and the mode determines if it a single pixel (P)
  or a 3x3 neighborhood average (C) at that grid spacing. This
  filter is useful for fuzzy or noisy images.
```

Use of Line Laplacian for detecting "spots" in 2D LC-MS images

There is an experimental line laplacian that we are optimizing for use with 2D LC-MS images. It is specified as **-laplace:H,*height,width*** for horizontal lines and as **-laplace:V,*height,width*** for vertical lines. The values for height and width are image dependent and are currently being titrated on a variety of images. You use either the horizontal or the vertical filter - not both. The screen shots in Figures 3 shows the use of a horizontal filter on low resolution 2D-LC-MS data (thanks to Laboratory of Proteomics and Analytical Technologies, NCI-Frederick).

```
  a) use -laplacian:V,height,width
    The vertical line detector filter using horizonal
    difference with widths (3,5,7,9) pixels
    =================================================

          0  0  0
    dX3 = 1 -2  1
          1 -2  1
            ...
          0  0  0

          0  0  0  0  0
    dX5 = 1  2 -6  2  1
          1  2 -6  2  1
              ...
          0  0  0  0  0

          0  0  0  0  0  0  0
    dX7 = 1  2  3 -12 3  2  1
          1  2  3 -12 3  2  1
                  ...
          0  0  0  0  0  0  0

          0  0  0  0  0  0  0  0  0
    dX9 = 1  2  3  4 -20 4  3  2  1
          1  2  3  4 -20 4  3  2  1
                   ...
          0  0  0  0  0  0  0  0  0

  b) use -laplacian:H,height,width
    The horizontal line detector filter using vertical
    difference with heights (3,5,7,9) pixels
    ==================================================

          0  1  1 ... 0
    dY3 = 0 -2 -2 ... 0
          0  1  1 ... 0

          0  1  1 ... 0
```

```
          0  2  2 ... 0
    dY5 = 0 -6 -6 ... 0
          0  2  2 ... 0
          0  1  1 ... 0

          0  1   1  ... 0
          0  2   2  ... 0
          0  3   3  ... 0
    dY7 = 0 -12 -12 ... 0
          0  3   3  ... 0
          0  2   2  ... 0
          0  1   1  ... 0

          0  1   1  ... 0
          0  2   2  ... 0
          0  3   3  ... 0
          0  4   4  ... 0
    dY9 = 0 -20 -20 ... 0
          0  4   4  ... 0
          0  3   3  ... 0
          0  2   2  ... 0
          0  1   1  ... 0
```

Handling saturated spots

The -saturateSpots:*percentThreshold* option will try to find saturated spots greater than percent threshold of the darkest pixel (default 0.95). Note that integrated density values for these spots will not be accurate. Normally the -saturateSpots switch is not used.

[STATUS: Not currently available - being debugged.]

Handling the splitting of spots segmented as one spot

[***REWRITE***]

There are situations where some spots are merged into single spots. Examples include post-modification charge trains of high concentration spots. Visual inspection of the segmented gels makes it easy to split these spots by matching opposing concavities.

The -splitSpots:{**B** | **R** | **T**},*minCCsplitSize* switch will try to split large spots based on an 'B'oundary analysis or 'R'egion analysis or a 'T'hreshold analysis based on Miller and Olson's spot splitting algorithm [A. Olson, M. Miller, 1989]. This is useful in some cases where spots are not resolved which should have been. A useful starting value for minCCsplitSize should be at least 15 pixels. Note that the other minimim spot sizing parameter - the primary minimum spot sizing parameter set by -ccMinSize:**nbrPixels**, with default size 2 pixels, ignores spots or subspots less than nbrPixels. Normally the -splitSpots switch is not used.

The algorithm first performs two checks to see if a spot is a candidate for splitting:

```
 (1) check if FBL CC spot meets minimum CC size criteria and
 (2) check whether split it using one of the following alternative
     heuristics:
    BOUNDARY ANALYSIS:
          look for notch pairs and break apart the spot
          by drawing white between the well-formed notch pair points
          and then find the FBL again!
    THRESHOLDING:
         if splitting a spot and the sub-spot i to be used has
         CCarea[i] < Tb, then merge all adjacent subspots j that also
         have CCarea[j] < Tb.
```

[STATUS: Not currently available - being debugged.]

Image usage within Seg2Dgel

The seven images are called pix1 through pix7 that are allocated as follows as buffer pointers. Images are generally create when needed and deleted when finished to save memory. However, if you are using the -gui option, the images are not removed so you can access them more easily with the Image Viewer. The pix5 image is not generated unless the -segmentedPixOutFile is specified. The pix7 image is not generated unless the -restOfPixOutFile is specified. The pix2, pix3, pix4 images are saved if the -ctlCorePixOutputFile switch is specified. All of the pix images are 16-bits except pix3 which is 32-bits (for now) because of 32-bit CC coding to be used in the future.

```
  pix1 is original image                          ppx/gelName.pixFileExtn>
  pix2 is magnitude 2nd derivative image          tmp/gelName-mag.jpg
  pix3 is directional derivative central core     tmp/gelName-cc.jpg
  pix4 is average of the original image           tmp/gelName-avg.jpg
  pix5 is segmented output image                  tmp/gelName-seg.jpg
  pix6 is background OD notch filter image        tmp/gelName-bkg.jpg
  pix7 is rest-of image = original - segmented    tmp/gelName-rest.jpg
```

[STATUS: note we currently generate .gif images rather than .jpg, but this will change.]

Propagated Central core image coding scheme for 8-bit images

The following is the current propagated central core (PCC) coding scheme. NOTE: This will change when we go to a more robust 16-bit encoding.

```
BKGRD_CODE          0          - background pixel
UNLABELED_CC_CODE   1          - unlabeled central core pixel
MIN_CC_CODE         2          - minimum labeled central core pixel
MAX_CC_CODE         99         - maximum labeled central core pixel
PCC_BASE_CODE       100        - guard region splitting previously
                                 merged spots
MIN_PCC_CODE        102        - min. labeled prop. central core pixel
MAX_PCC_CODE        199        - max. labeled prop. central core pixel
KILL_SPOT_CODE      254        - deleted spot in central core
                                 if NEQ sizing (0 if -noDelete).
ISOLATED_PIXEL_CODE 255        - isolated pixel or non
                                 4-ngh connected pixel
```

3. Running Seg2Dgel and specifying parameter options via the command line

The program may be run either interactively (-gui) with a graphical user interface (GUI) or under an OS shell command to implement batch (-nogui) depending on how it was started. In the former case, after the segmentation is finished, the user has the option of interactively viewing any of the images used by the segmenter and querying them for quantified spot data or look at small pixel windows (3x3 to 21x21 pixels) of the image data. The user may also modify the input switch options and save the new options in a "Seg2Dgel.properties" file in the current project directory so that it may be used as the default switch options in subsequent running of the segmenter. All options including the input image to be segmented are specified via GNU/Unix style switches on the command line (-*switch{optional :parameters*} and its negation as -no*switch*). However, if GUI mode is used, you can interactively specify the switches and their options.

The computing window ROI size

The computing window is a rectangular region in the image that is considered valid and that spots in this region should be segmented. Any region outside of this window is ignored.

For an image with nCols X nRows, the default computing window is set to [2:nCols-2 x 2:nRows-2] if it is not defined. You can set the computing window using the -cw:x1,x2,y1,y2 command line switch. In any case it is clipped to [2:pixWidth-2 x 2:pixHeight-2] where the image is of size pixWidth x pixHeight.

Local Folders and files created and used by Seg2Dgel

When Seg2Dgel is first started, it will check for the following folders and files in the installation directory and create them if they can not be found.

- **Seg2Dgel.properties** - is the default command line switch options to be used (if present). You can edit the default switch options and then save them into the Seg2Dgel.properties file.

If you specify an image to be semented, it will check whether it is in a ppx/ subdirectory. If not, it will ask you if you want to create a project directory and will then set up the following four directories and copy your image into the ppx/ directory. You can also use the -projDir:*user-project-directory* switch to specify a (possibly new) project directory.

- **batch/** directory holding temporary batch files - [NOT USED]
- **cache/** directory holding temporary CSD cache files - [NOT USED]
- **ppx/** directory holding your original gel input files
- **rdbms/** directory holding CSD database RDBMS files - [NOT USED]
- **tmp/** directory holding generated sample image files
- **xml/** directory that holds the accession files, sample segmentation image files, and SSF spot-list files

Special debugging options -debug and -pixdump switches

Two special switches are available for debugging Seg2Dgel and for investigating failure modes of particular spots to not segment correctly for some gels. This is meant only for those willing to delve into the source code.

- -debug:*bits,dbx1,dbx2,dby1,dby2* - dumps various conditional debugging parameters onto the report window as well as the output SSF file. The debugging is only active in the range [dbx1:dbx2, dby1:dby2] and ignore otherwise. The 'bits' are the debug bits specified as either octal or decimal and enable particular debugging output if the program was compiled with debugging enabled.

- -pixDump:*width,dpx1,dpy1,stepSize* - dump pictures generated during the segmentation algorithm defined by the computing window (with maximum width 'width' of 21 pixels for an 132 column screen (the default) and up to 30 for a 132 column screen) on the screen and output SSF. The step size is pixel sampling distance. The (dpx1,dpy1) specifies the ULHC of the dump window. The LRHC (dpx2,dpy2) is computed as (dpx1+width, dpy1+width).

Sg2Dgel command-line arguments switch usage

The command line arguements usage is:

```
Seg2Dgel input-image-file [< optional switches >]
```

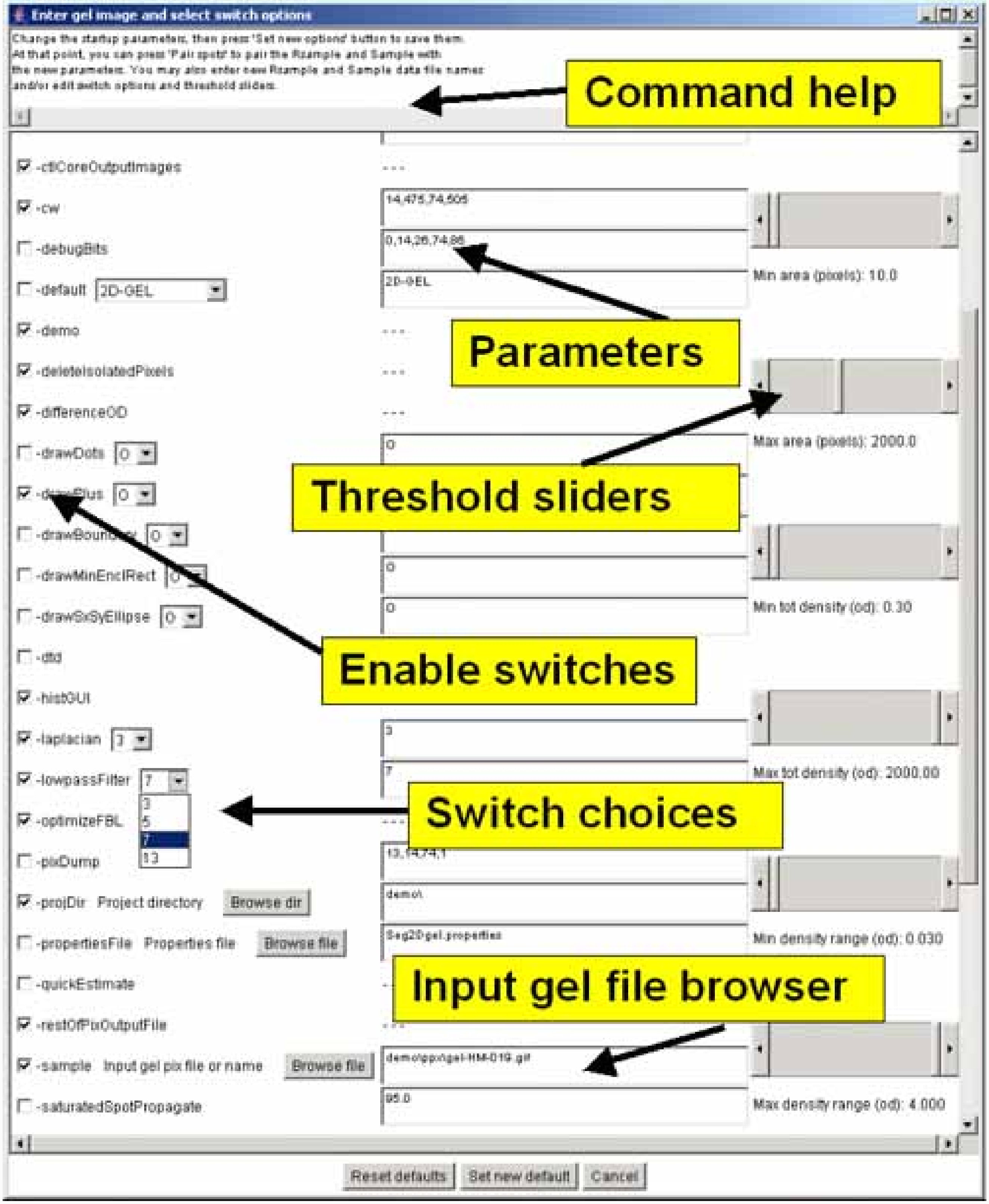
The complete list of switches is given later in this manual and as well as some examples of typical sets of switches. The user defined default switches may be specified as a resource string 'Seg2Dgel.properties' file saved in the project directory. For example:

```
-lowpassFilter:3 -laplace:B,C,3 -saturatedPropSpot:99.7 \
```

```
        -bkgrdSize:64 -bkgrdCorrection -restOfPixOutputFile \
        -ccMinSize:4 -drawPlus:0
```

Options wizard window for setting the command line switches

If you invoke the **Edit options** button in the Report window (or from the Edit menu), it will popup an options wizard shown in Figure 3.



Change the startup parameters, then press 'Set new options' button to save them.
At that point, you can press 'Pair spots' to pair the Rsample and Sample with
the new parameters. You may also enter new Rsample and Sample data file names
and/or edit switch options and threshold sliders.

☑ -ctlCoreOutputImages ---
☑ -cw 14,475,74,505
☐ -debugBits 0,14,26,74,86
☐ -default 2D-GEL
☑ -demo ---
☑ -deleteIsolatedPixels ---
☑ -differenceOD ---
☐ -drawDots 0
☑ -drawPlus 0
☐ -drawBoundary 0
☐ -drawMinEnclRect 0
☐ -drawSxSyEllipse 0
☐ -dtd
☑ -histGUI
☑ -laplacian 3
☑ -lowpassFilter 7 (3, 5, 7, 13)
☑ -optimizeFBL ---
☐ -pixDump 13,14,74,1
☑ -projDir Project directory demo\
☐ -propertiesFile Properties file Seg2Dgel.properties
☐ -quickEstimate
☑ -restOfPixOutputFile ---
☑ -sample Input gel pix file or name demo\gpix\gel-HM-019.gif
☐ -saturatedSpotPropagate 95.0

Min area (pixels): 10.0
Max area (pixels): 2000.0
Min tot density (od): 0.30
Max tot density (od): 2000.00
Min density range (od): 0.030
Max density range (od): 4.000

Figure 3. Screen view of the popup options wizard window for setting the command line switches, parameter and specifying a gel input image file. All of the command-line input switches (symbols starting with a '-') are available in the scrollable window. Switches are checked if they are enabled and if the switch requires a value, the current value is shown in the data entry window to its right. On the right are several threshold sliders for the upper (max) and lower (min) sizing ranges for several parameters including area, density, and OD range which set the values for switches -ccMinSize:*minNbrPixels*, -saturatedSpot:*percentSaturation*, -thrArea:*a1, a2*, -thrDensity:*d1*,*d2* and -thrRange:*o1*,*o2*. At the top is a **Browse file** button to use for specifying a different gel input image file. There is a **Browse dir** button to use for specifying a different project directory. Clicking on any option switchs will show a short help message associated with that switch at the top of the window. Pressing the **Set new default** button will pass the new options values back to Seg2Dgel. To use the new values, press the **Segment** button in the main Report Window to resegment the gel using the new values.

Updating Seg2Dgel from the Open2Dprot Web server using -update switch

As new versions of Seg2Dgel are developed and put on the Web server, a more efficient way of updating your version is to use the -update commands. There are four options:

```
        -update:program      to update the program jar file
        -update:demo         to update the demonstration files
        -update:doc          to update the documentation files

        -update:all          to update all of the above
```

After updating the program, it should be exited and restarted for the new program to take effect.

Increasing or decreasing the allowable memory used by Seg2Dgel

If you are working with very large images that require a lot of memory, you might want to increase the memory available at startup.

If you are using the Windows Seg2Dgel.exe file or clicking on the Seg2Dgel.jar file, you can't change the default startup memory.

If you are using the Seg2Dgel.jar in a script using the java interpreter as in the following example which uses the **-Xmx*256*M** (specifying using 256Mbytes at startup). Change 256 to a larger size if you want to increase startup memory.

```
   java -Xmx256M -jar Seg2Dgel.jar {additional command line args}
```

4. Command and Report Window - the command center

Seg2Dgel is designed to be used efficiently in a batch mode with minimal command line output. It is also designed to optionally provide a graphical user interface (GUI) which creates a Report Window that captures a report of the segmenter output as well as additional output directed to it by the user. There are a set of pull-down menus as well as a set of buttons for often used functions.

All logged output is sent to the report window in a scrollable text window that may be saved or used for cut and paste operations. A set of command buttons at the bottom of the window are replicates of commands in the menus, but are easier to access. They include the following functions:

- **Status area** - reports the current status of the segmenter as it is processing the image.
- **Clear** - clear the report screen
- **SaveAs** - save this report screen in a text file (user is prompted for the file name)
- **Stop program** - stop the segmentation analysis after the current spot being segmented is finished.
- **Edit options** - popup an Options Window to let the user set/clear switches, assign switch parameters (if any), adjust the threshold sizing limits, and specify an input gel image file.
- **Segment** - rerun the segmenter on the (possibly changed) current switch options and input gel image file
- **Image viewer** - popup the Image Viewer window to investigate details of any images created during segmentation
- **Close** - save the current SSF file if open and exit the Seg2Dgel program.

5. Pull-down menus in the Graphical User Interface (GUI)

The menu bar a the top of the Report Window contains five menus.

1. File menu - to open the gel to be segmented and other file operations.
2. Edit menu - to change various options including command line switches, computing window, and memory size
3. View menu - to invoke the image viewer used to inspect the images after segmentation.
4. Quantify menu - to run the segmenter on your gel.
5. Help menu - popup Web browser documentation on Seg2Dgel

Menu notation

In the following menus, selections that are *sub-menus* are indicated by a '▶'. Selections prefaced with a '☑'

and indicate '☐' indicate that the command is a checkbox that is enabled and disabled respectively. Selections prefaced with a '◉' and indicate '○' indicate that the command is a multiple choice "radio button" that is enabled and disabled respectively, and that only one member of the group is allowed to be on at a time. The default values set for an initial database are shown in the menus. Selections that are not currently available will be grayed out in the menus of the running program. The command short-cut notation **C-*key*** means to hold the Control key and then press the specified *key*.

5.1 File menu

These commands are used to open the gel to be segmented and other file operations. The current menus and the menu commands (non-working commands have a '*' prefix) are listed below. You can use either the "Edit options" button to popup the Options Window editor to change the gel input file or the (**File menu | Open gel image file**) command.

- **Open gel image file** - specify the gel image file to segment
- ***Set project directory** - set the new default project directory and create the new directory and (batch/ cache/ ppx/ rdbms/ tmp/ and xml/) subdirectories if needed. See the -projDir:*new_project-directory* switch that you can use to do this on startup.

- **Clear report** - clear the report text area so that you can start a new report that will be easier to cut and paste or do save to a file
- **Save report to a file** - save the report to a text file
- **Edit options to change directory to set new image** - change the project directory, image name and other parameters to assign a new image to segment. [Same as (**Edit | Edit options**) menu command.

- **Update**▶ - update Seg2Dgel programs and data from open2dprot.sourceforge.net/Seg2Dgel server
-     **Seg2Dgel all files** - to get the latest files
-     **Seg2Dgel program** - to get the latest program release (Seg2Dgel.jar) and other version dependent files from Seg2Dgel server
-     **Seg2Dgel demo files** - get latest demo project data files
-     **Seg2Dgel documentation** - to get the latest documentation (for stand-alone operation)

- **Quit** - exit the program when in GUI mode.

5.2 Edit menu

These commands are used to change various defaults. These are saved when you save the state and when you exit the program.

- **Options**▶ - change the command line options
-     **Open options file** - read a new options file (this overides the previous options file which could be Seg2Dgel.properties).

- **Save options file** - write the current options to the current options file (default Seg2Dgel. properties)
- **SaveAs options file** - write the current options to a new file you specify
- **Edit options** - popup the an Options Wizard popup window to to edit the switch options and thier parameters, threshold parameter sliders, and input image file
- ***Reset default options** - reset options to the initial default
  ----
- **Regions of interest** - popup the image viewer to let you interactively define regions of interest (ROI) for: the computing window, ND step wedge window, -debug window, -pixdump window.

5.3 View menu

This menu contains commands to invoke the Image Viewer popup window used to inspect the images after segmentation. You have the ability to modify the Region of Interest (ROI) by redefining it for the (-cw, -pixdump, and -debug regions). Note: in the Image Viewer first select the ROI as either CW, -dbug ROI, or -pixdump ROI. Then click on the upper left-hand corner of the new region and type Control-U. Then click on the upper lower left-hand corner of the new region and type Control-L. This will then redefine the selected ROI and use it the next time you run the segmenter from the Report Window. Figure 4. shows an example of the Image Viewer. See the Examples - samples of screen shots in the Demos for more examples and description of the some of the operations available in the Image Viewer.

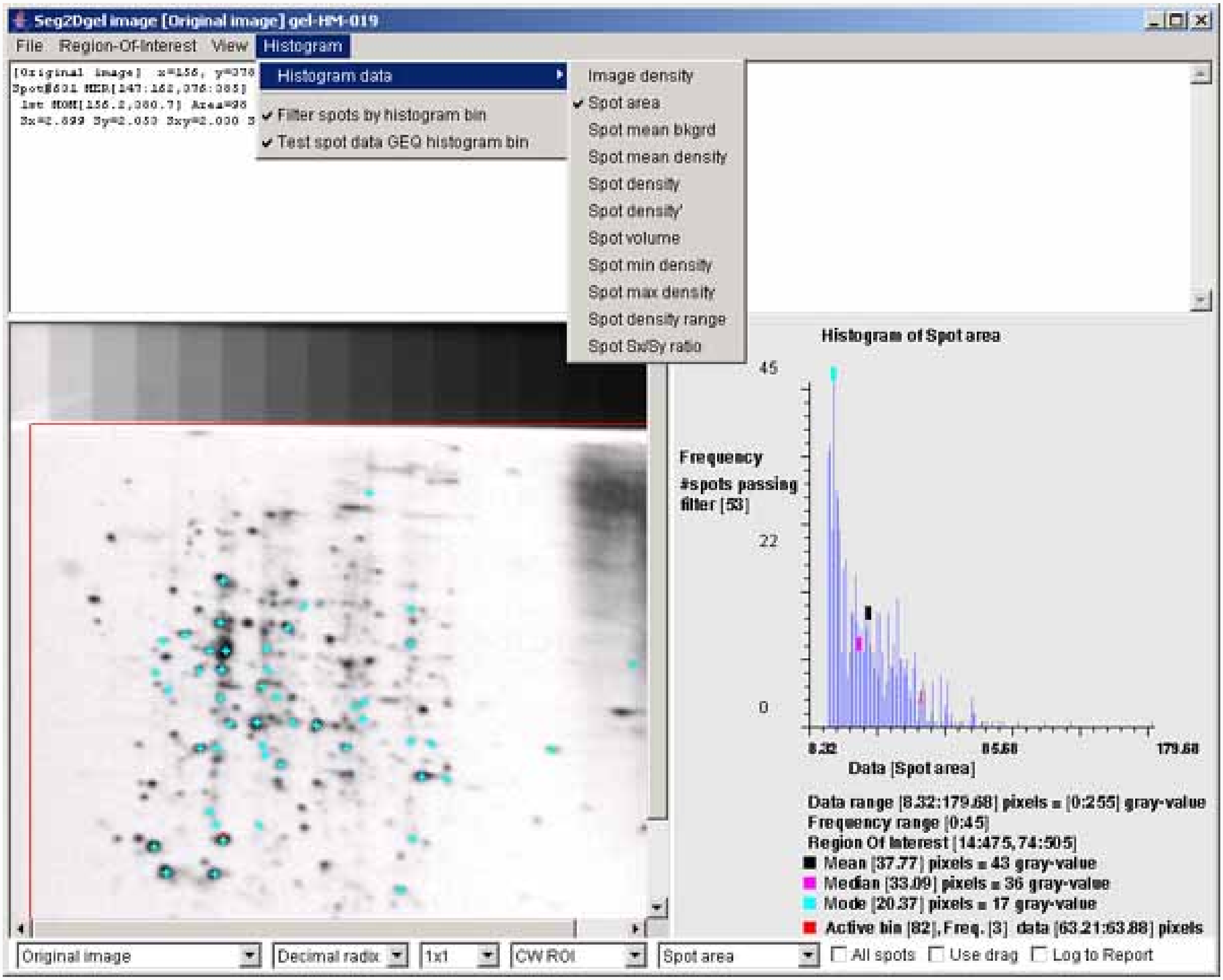
- **Image viewer** - popup the image viewer window to inspect the images created by the segmenter.
- ☑**Add histogram to Image Viewer** - add a dynamic histogram subwindow to the Image viewer when it is popped up.
- **Show SSF report** - read the Sample Spot-list File generated by the segmenter and display it in the Report Window
- **Show SSF browser** - read the Sample Spot-list File generated by the segmenter and display it in a popup browser.

5.3.1 Histogram display in the Image Viewer window

The Image Viewer, in addition the the image display, also has an optional dynamic histogram display. This may be useful for setting the threshold parameters. The user may select a feature from a list including image density under the region of interest (ROI) as well as spot features (area, mean background density/spot, mean spot density/spot, total spot density, total spot D', minD/spot, maxD/spot, spot density range (maxD - minD)). Horizontal and/or vertical image line cross-section histograms (i.e. image slices) are also available via the (**View | Horizontal slice**) and (**View | Vertical slice**) menu commands.

The "-histGUI" ("-nohistGUI) switch can be used to add (remove) it from the command line. The (**View | Add histogram to Image Viewer**) checkbox menu command or the popup options wizard window can also be used to change it.

You can do rudimentary spot data filtering (if enabled in the Histogram menu of the Image Viewer) where spots passing the filter are indicated in the image. You can click on a bin in the histogram and the filter will show spots (with a cyan "+") GEQ or LEQ the bin value you selected. Clicking on the histogram, will select that bin and display the corresponding data for that bin. Note, if you enable the ☑**Use drag**) checkbox, you can change the bin by draging the mouse. See the [Examples - samples of screen shots](#) in the Demos for more examples and descriptions for using the histograms. There is also an example of using the horizontal image density slice.



```
[Original image] x=156, y=378
Spot#631 MBR[147:162,378:385]
 1st MOM[156.2,380.7] Area=98
 Sx=2.899 Sy=2.053 Sxy=2.000 S
```

Data range [8.32:179.68] pixels = [0:255] gray-value
Frequency range [0:45]
Region Of Interest [14:475,74:505]
■ Mean [37.77] pixels = 43 gray-value
■ Median [33.09] pixels = 36 gray-value
■ Mode [20.37] pixels = 17 gray-value
■ Active bin [82], Freq. [3] data [63.21:63.88] pixels

Figure 4. Example of the Image Viewer popup window. It also shows the histogram of the spot area distribution for spots inside of the computing window. The histogram may display various spot features. This shows the spot areas under the computing window (CW ROI). You may change the ROI after the segmentation using the (**Region-Of-Interest | Set ROI type | CW ROI**) menu command in the Image Viewer. The "Spot area" was selected using the (**Histogram | Histogram data | Spot area**) menu command. You may data filter spots that are have an area greater or equal (GEQ) than the histogram bin selected by clicking on it are shown in the image

on the left as cyan colored "+" marks. The data filter is enabled by the (**Histogram** | ☑**Filter spots by histogram bin**) menu command. You set the The data filter test to GEQ (or LEQ) the bin by the (**Histogram** | ☑**Test spot data GEQ histogram bin**) menu command. Clicking on the histogram will select that bin and display the corresponding data for that bin. Note, if you enable the ☑**Use drag**) checkbox, you can change the bin by draging the mouse.

5.4 Quantify menu

This menu is used to run the segmenter to perform the analysis.

- **Segment -** start the gel segmentation on the current input image file and current option settings
- **Stop segmenter -** abort the gel segmentation after the current spot is finished being segmented

5.5 Help menu

These commands are used to invoke popup Web browser documentation on Seg2Dgel. Some of the commands will load local documentation in the the GUI report window.

The documentation is kept on the Internet at http://open2dprot.sourceforge.net/Seg2Dgel. Normally, these help commands should pop up a Web browser that directly points to the Web page on the http://open2dprot.sourceforge.net/Seg2Dgel site. If your browser is not configured correctly, it may not be able to be launched directly from the Seg2Dgel program. Instead, just go to the Web site with your Web browser and look up the information there.

- **Seg2Dgel Home -** the open2dprot.sourceforge.net/Seg2Dgel home page
- **Reference Manual**▶ - this reference manual
  - **1. Introduction**
  - **2. Algorithm**
  - **3. Running Seg2Dgel**
  - **4. Command and Report window**
  - **5. Menus**
  - **6. Installing Seg2Dgel**
  - **7. Command option switches**
  - **8. Demonstrations**
  - **9. References**
- **Sample screen shots -** examples several screen shots
- **Status -** status of the segmenter bugs and future options
- **Revision history -** show latest Seg2Dgel release and bugs that were fixed
- **FAQ -** frequently asked questions
- **Contributors -** to the segmenter sub-project
- **PDF files -** additional documentation including this manual
  --
- **Local PDF manual -** popup a browser on the local copy of the Seg2Dgel Web site PDF document

also available on the Web site. You can update this document from the Web site using the (**File | Update | Seg2dgel documentation**) command.

- **Open2Dprot home** - home page for the Open2Dprot project
- **License** - the open source license
- **About** - display information about the program

6. Downloading, installing and running Seg2Dgel

The installation packages are now available for **download** from the SourceForge Files mirror. Look for the most recent release named "Seg2Dgel-dist-V.XX.XX.zip". These releases include the program (both as Windows .exe file and a .jar file), required jar libraries, demo data, Windows batch and Unix shell scripts. Download the zip file and put the contents where you want to install the program. Note that there is a Seg2Dgel.exe (for Windows program). You might make a short-cut to this to use in more easily starting the program. Alternatively, you can use the sample .bat and .sh scripts to run the program explicitly via the java interpreter. Note that this method assumes that you have Java installed on your computer and that it is at *least* JDK (Java Development Kit) or JRE (Java Runtime Environment) version 1.5.0. If you don't have this, you can download the latest version free from the java.sun.com Website.

6.3 Running Seg2Dgel

There are several ways to run the program. On Windows, you can start Seg2Dgel by clicking on the startup icon shown in Figure 5 below. For Unix systems including MacOS-X, you can start Seg2Dgel from the command line by running the Seg2Dgel.jar file. If your computer is setup to execute jar files, just type the jar file. In both systems, you can specify additional command line arguments in Windows .bat and unix .sh scripts (see examples below.

Figure 5. Startup icon for Seg2Dgel. Clicking on the icon starts Seg2Dgel. To start Seg2Dgel, click on the startup icon shown in Figure 6 below - or you can run the `demo-Seg2Dgel.bat` script. For Unix systems including MacOS-X, you can start Seg2Dgel from the command line by clicking on the `Seg2Dgel.jar` file or using the `demo-Seg2Dgel.sh` script. These two scripts run the program in batch. There are also GUI versions of the two scripts `demo-Seg2Dgel-GUI.bat` and `demo-Seg2Dgel-GUI.bat` that will pop up a user interface. You could make short-cuts (Windows) or symbolic-links in Unix to make it easier to start.

6.4 Requirements: minimum hardware and software requirements

A Windows PC, MacIntosh with MacOS-X, a Linux computer or a Sun Solaris computer having a display resolution of at *least* 1024x768. We find that a 1024x768 is adequate, but a 1280x1024 screen size much better since you can see the Popup Report window, Options window, and Image Viewer window at the same time. At least 30 Mb of memory available for the application is required and more is desirable for comparing large images or performing transforms. If there is not enough memory, it will be unable to load the images, the transforms may crash the program or other problems may occur.

An Internet connection is required to download the program from the Open2Dprot Seg2Dgel Web site. New versions of the program and associated demo data will become available on this Web site and can be uploaded to your computer using the various (**File | Update from Web server | ...**) menu commands. If you have obtained the installer software that someone else downloaded and gave to you, then you do not need the Internet connection to install the program. We currently distribute Seg2Dgel so that it uses up to 256Mb. See discussion on increasing memory.

6.5 Files included in the download

The following files are packaged in the distribution you install. you can periodically a (File | Update from Web server | ... program) menu command to update the files from the open2dprot.sourceforge.net Web server.

- **Seg2Dgel.jar** is the Java Archive File for Seg2Dgel that is executed when you run Seg2Dgel.
- **Seg2Dgel.exe** is the Windows executable file that packages Seg2Dgel.jar with a java startup. This was packaged using the open-source launch4j (launch4j.sourceforge.net) program.
- **Open2Dprot-Accession.dtd** is the XML .dtd file that can be used with the accession database file Accession.xml.
- **Open2Dprot-SSF.dtd** is the XML .dtd file used to generate the SSF .xml file if the -ssfFormat:X switch option is used. Note: this is only used if -dtd switch is set.
- **PDF/fullSegDoc.pdf** is a copy of the Seg2Dgel Web site for use as a reference manual when using the segmenter off-line. It may not be as up to date as the one on the web site but you can do a (**File | Update from Web server | Seg2Dgel documentation**) menu command to update it.
- **jai_core.jar** is the core Java runtime from SUN's Java Advanced Imaging (JAI) at java.sun.com
- **jai_codec.jar** is the JAI tiff file reader from SUN's Java Advanced Imaging JAI at java.sun.com
- **xercesImpl.jar** is the Xerces 2 XML SAX/DOM XML library from Apache xml.apache.org
- **xml-apis.jar** is the Xerces 2 XML SAX/DOM XML library from Apache xml.apache.org
- is contains the Mozilla.1.1.html and LEGAL.txt files for Seg2Dgel and the release-license-jai_files for the Java Advanced Imaging .jar files used by the TIFF file reader.
- **demo/** is a demonstration project directory containing the data in the following subdirectories:
    1. **ppx/** is a directory holding your gel input files (4 images)
    2. **tmp/** is a directory holding generated image files
    3. **xml/** is a directory that will hold the Sample Spot-list Files output SSF files as well as the accession database file.

  You can do a (**File | Update from Web server | Seg2Dgel demo files**) menu command to update it.

- **demo/xml/accession.xml** is a demonstration project accession database file for the 4 images.

7. List of the command line switches

The command line usage is:

```
          Seg2Dgel -sample:input-gel-image-file [< optional switches >]
```

where the order of arguments is not relevant. In the following list of switches, items in **bold** within '{}' and separated by '|' are specific values which must be used (e.g., for -laplacian:{**3** | **5** | **B**},*gridSize*,*mode* the instance might be -laplacian:3). Variable values in *italics* indicate that a numeric value for that variable should be used (e.g., for -thrArea:*a1*,*a2* it might be -thrArea:20,2000). Some switches have several alternate fixed choices in which case this indicated as a list of bolded items inside of a set of '{...}' with '|' separating the items. You must pick one of the items and do not include the '{}' brackets. Also, do NOT include any extra spaces in the arguments of the switch - it will be counted as if it were another switch.

Command line switches

```
-accessionDB:accessionFile specifies that the gel image
       be looked up in an accession database. The accession database
       has the gel computing window as we as a possible ND
       calibration. (Default is -noaccessionDB:accession.xml).

-allowTouchingEdges allows spots which touch an edge to be counted
        rather than deleted. (Default is allowTouchingEdges).

-averageOD uses OD instead of grayvalue in Gaussian filters.
        (Default is -averageOD).

-bkgrdSize:n computes background using nXn zonal notch filter
        (default n is 32 that works for a 250 microns/pixel image).

-bkgrdCorrection - do background density subtraction correction
        so that estimate D' = D-A*mnBackground for each spot.
        Otherwise D' = D. (Default is -nobkgrdCorrection).

-boundaryInSSF - add the boundary chain code to each SSF file entry.
        This currently requires you use the -drawBoundary
        option. Note, it will add quite a bit to the SSF file size!
        (Default is -noboundaryInSSF).

-ccMinSize:minNbrPixels sets minimum number of pixels a
        spot's central core has to be before it is considered a
```

```
        spot. (Default -ccMinSize:2).

-ctlCorePixOutputFile saves the central core image on the picture
        disk with an gelName-cc.jpg picture file. Also save
        magnitude (gelName-mag.jpg), average (gelName-avg.jpg),
        and background (gelName-bkg.jpg) images. (Default
        is -noctlCorePixOutputFile).

-cw:cwx1,cwx2,cwy1,cwy2 is the computing window that can be
        of maximum size [k : pixWidth-k, k : pixHeight-k] pixels
        (which is the default value if the -cw option is not
        specified). The value of k is determined by the size of the
        -lowpassFilter:size (where k is that size).  (Default
        is -nocw).

-debug:bits,dbx1,dbx2,dby1,dby2 dumps various conditional
        debugging parameters onto the report window as well as the
        output SSF file.  The debugging is only active in the range
        [dbx1:dbx2, dby1:dby2] and ignore otherwise. The 'bits' are
        the debug bits specified as either octal or decimal and enable
        particular debugging output if the program was compiled with
        debugging enabled.  (Default is -nodebug).

-default:{2D-GEL | 2D-LC-MS | NO-DATABASE} sets the default
        switches to a specific configuration depending on the sample type:
        For 2D-GEL it sets the switches to:
          -nodemo -timer
          -thrArea:10,2000 -thrDensity:0.3,2000 -thrRange:0.03,4.0
          -lowpassFilter:7 -laplacian:3 -bkgrdSize:32
          -ccMinSize:2 -deleteIsolatedPixels
          -accession:accession.xml -ssfFormat:X
          -ctlCorePixFile -restOfPixFile -segmentedPixFile
        For 2D-LC-MS it sets the switches to:
          -nodemo -timer -nocw -noaccessionDB
          -thrArea:8,4000 -thrDensity:30,200000 -thrRange:2,65767
          -thrSxSyRatio:2.5,100.0 -lowpassFilter:3 -laplacian:H,3,9
          -bkgrdSize:32 -nobkgrdCorrection
          -noaverageOD -nodifferenceOD -drawMinEnclRect:Z
          -ccMinSize:4 -deleteIsolatedPixels
          -accession:accession.xml -ssfFormat:X
          -ctlCorePixFile -restOfPixFile -segmentedPixFile
        For NO-DATABASE it sets the switches to:
          -nodemo -timer -nocw -noaccessionDB
          -thrArea:10,2000 -thrDensity:0.3,400000 -thrRange:2,65767
          -thrSxSyRatio:2.5,100.0 -lowpassFilter:7 -laplacian:3
          -bkgrdSize:32 -nobkgrdCorrection
          -noaverageOD -nodifferenceOD
```

```
          -ccMinSize:2 -deleteIsolatedPixels
          -accession:accession.xml -ssfFormat:X
          -ctlCorePixFile -restOfPixFile -segmentedPixFile
        This disables -demo if it was set. (Default -nodefault:2D-GEL).

-deleteIsolatedSpot sets deleted spots (i.e. those which do not meet
        sizing criteria) to the value 255 (-noDelete) rather than the
        0 in the central core image. A value of 0 (-delete) indicates
        there are NO deleted spots in the central core image. With no
        switch it sets deleted spots to pixel value 254. (Default is
        -nodeleteIsolatedSpot).

-demo sets the default switches and sample input gel to a specific
       configuration. This may be overriden by turning off the -demo
       switch in the Options Wizard.

-differenceOD uses OD instead of gray value when computing central
        core Laplacian. (Default is -differenceOD).

-drawDot:{O | Z} indicates spots in the 'restOf' and
        'segmented' output images by a dot ('.').  Draw this in a copy
        of the 'O'riginal or segmented 'Z' image. (Default -nodrawDot).

-drawPlus:{O | Z} indicates spots in the 'restOf' and
        'segmented' output images by a plus ('+').  Draw this in a
        copy of the 'O'riginal or segmented 'Z' image. (Default
        -drawPlus:PZ).

-drawBoundary:{O | Z} indicates spots in the 'restOf'
        and 'segmented' output images by a boundary ('D'). Draw this
        in a copy of the 'O'riginal or segmented 'Z' image. (Default
        -nodrawBoundary).

-drawMinEnclRect:{O | Z} indicatess drawing spots in the 'restOf'
       and 'segmented' output images by spot minimum enclosing rectangles.
       (Default is -nodrawMinEnclRect).

-drawSxSyEllipse:{O | Z} indicates spots in the 'restOf' and
        'segmented' output images by an ellipse of size 2 * spot
        (sX,sY).  Draw this in a copy of the 'O'riginal or segmented
        'Z' image (default -nodrawSxSyEllipse). [NOT AVAILABLE YET]

-dtd adds the XML DTD file (Open2Dprot-SSF.dtd) in the output XML

-gui starts the segmenter with a popup Graphical User Interface
        rather than in batch mode. This captures messages from the
        segmenter.  You can then cut and paste the results or save it
```

to a text file. The GUI is also used to change the switch options, re-run the segmenter and view images after each segmentation. (Default is -nogui).

-info prints additional information on Seg2Dgel. (Default is -noinfo).

-laplacian:{**3** | **5** | **B** | **H** | **V**},{*height* | **P** | **C**},{*width* | *gridSize*} - use a 3x3 Laplacian (default), 5x5 Laplacian, or a Busse Laplacian to compute the central core image. The Busse Laplacian uses a grid Laplacian with gridSize sampling. The mode used with the Busse Laplacian is either **P** or **C**. If mode **C** option is used, it uses 3x3 average pixels in the Laplacian filter extrema, else **P** (single pixel) weights are used. The 'H'and 'V' is for a horizontal or vertical line filter with height and width are 3, 5, 7, or 9 filter sizes. E.g., 'H,3,9' or 'V,9,3'. (Default is -laplacian:3 where gridSize and mode are not used).

-lowpassFilter:{**3** | **5** | **7** | **13**} - pixels size of the lowpass averaging filter applied to the original image using a nXn filters. These correspond to a 3x3, 5x5, 7x7, or 13x17 filter. (Default is -lowpassFilter:7).

-optimizeFBL optimizes segmentation for gels with very large spots by removing interior pixels when propagating spots when using the FBL. (Default is -optimizeFBL).

-pixDump:*w,x,y,stepSize* dumps the pictures generated during the segmentation algorithm defined by the computing window (with maximum width w of 21 pixels for an 132 column screen (the default) and up to 30 for a 132 column screen) on the screen and output SSF. The step size is pixel sampling distance. The (x,y) specifies the ULHC of the dump window. (Default is -nopixdump).

-propertiesFile:*filepath* to specify an alternate "Seg2Dgel.properties" file used instead of the default file. The default is to look for the "Seg2Dgel.properties" file in the startup directory. If none, is found then it uses the builtin defaults.

-quickEstimate use quick estimate of D' using only central core and gaussian estimate. This does not do the additional spot improvement algorithms. (Default is -noquickEstimate).

-restOfPixOutputFile creates the output image *gelName*-rest.jpg defined as the original image less the segmented spots rather than just the segmented spots (the *gelName*-seg.jpg). This Y image is useful for finding spots that were fragmented or were missed. (Default is -norestOfPixOutputFile).

-saturatedSpotPropagate:*percentThreshold* propagates saturated spots from central cores to adjecent pixels with similar high gray values. Only apply this to spots > threshold percent of maximum spot OD seen in gel. (Default is -noSaturatedSpotPropagate:0.95). [Not available yet]

-segmentedPixOutputFile saves the segmented image in *gelName*-seg.jpg. (Default is -nosegmentedPixOutputFile).

-sizeD'remove - delete spots in CC and *gelName*-seg.jpg based on D' instead instead of just density D. (Default is -nosizeD'remove).

-splitSpots:{**B** | **R** | **T**},*minCCsplitSize* splits large spots with central core area > minCCsplitSize (default 15). **B** uses a boundary analysis method, **R** analyzes the Run-length Projection Map of the spot while **T** uses a method based on finding multiple subspot clusters using Miller and Olson's (1989) thresholded Laplacian test. (Default is -nosplitSpots:R,15). [Not available yet]

-ssfFormat:{**F** | **G** | **T** | **X**} defines the output format for the Sample Spot-list File (SSF) where mode is: **F** for full tab-delimited data that includes the list of spots but also the same information as the **G** format), **G** for GELLAB-II (.ssf), **T** for tab-delimited (.txt) with just the list of spots, and **X** for XML (.xml). (Default is -ssfFormat:X).

-thrArea:*a1,a2* specifies the spot area threshold sizing range [a1:a2] in pixels. (Default is -thrArea:10,2000).

-thrDensity:*d1,d2* specifies the spot D' threshold sizing range [d1:d2] in sum of OD (or grayscale if not calibrated). (For OD calibrated gels. (Default is -thrDensity:1.0,2000.0).

-thrRange:*o1,o2* specifies the spot OD threshold sizing range [o1:o2] in OD (or grayscale if not calibrated). (For OD calibrated gels. (Default is -thrRange:0.05,2.7).

-thrSxSyRatio:*p1,p2* specifies the spot shape by the ratio of spot density weighted X and Y variance (Sx,Sy) range [p1:p2]. (Default is -nothrSxSyRatio:0.03,30).

-timer enables a timer to capture processing times for each step in the segmentation. (Default is -notimer).

-title to display the image titles at the bottom of the images. (Default is -title).

-update:{**all | program | demo | doc**} specifies that all of the Seg2Dgel files, the program jar files, the documentation files or the demonstration files should be updated from the Open2Dprot Web server. The program should be exited and restarted after updating the program for this to take effect. (Default is -noupdate).

-usage prints the command level switch usage. (Default is -nousage).

-version prints the version of the program. (Default is -noversion).

7.1 Examples of some typical sets of switches

The following shows a few examples of useful combinations of command line switches. The '\' at end of some lines indicates that the next line is a continuation of the same line.

Any case-independent switch may be negated by preceeding it with a 'no' eg. '-noinfo'. The command line syntax used to invoke it is:

```
     Seg2Dgel -sample:input-gel-image-file [< opt.-switches >]
```

The following examples using switches might be useful:

```
     Seg2Dgel -sample:gel.tif
          # Simplest command line that implies -nogui

     Seg2Dgel -sample:gel.tif -ssfFormat:X -AllowTouchingEdges -
lowpassFilter:7 \
              -bkgrdSize:32 -bkgrdCorrection -averageOD -differenceOD \
              -lowpassFilter:3 -laplacian:3
          # These are the default switches used in the above example
          # and use the default XML SSF format.

     Seg2Dgel -sample:gel.tif -lowpassFilter:7 -noAllowTouchingEdges
          # Disallow spots which touch an edge and use more smoothing.
```

```
     Seg2Dgel -sample:gel.tif -low:7 -noAllow
          # shorter form of -lowpassFilter and -AllowTouchingEdges switches.

     Seg2Dgel -sample:gel.tif -ssfFormat:G
          # Sample Spot-list File GELLAB-II format has labeled data.

     Seg2Dgel -sample:gel.tif -ssfFormat:F
          # Sample Spot-list File full tab-delimited format has
          # has the same header and trailer as the GELLAB-II formated
          # labeled data. The spot list is between strings
          #   ""###-START-OF-SPOTLIST###"
          # and
          #   ""###-END-OF-SPOTLIST###"

     Seg2Dgel gel.tif -lowpass:3
          # Average using a 3x3 Gaussian filter.

     Seg2Dgel -sample:gel.tif -lowpass:5
          # Average using a 5x5 Gaussian filter.

     Seg2Dgel -sample:gel.tif -pixdump:18 -cw:313,330,195,212 -thrArea:4,100\
              -thrDensity:0,1000 -thrRrange:0,2.7
          # print numeric window trace for small computing window.

     Seg2Dgel -sample:gel.tif -pixdump:18 -thrA:4,100 -thrD:0,1000 -
thrR:0,2.7
          # Change some of the sizing parameters.

     Seg2Dgel -sample:gel.tif -drawDot:Z
          # Draw '.' dots in center of each segmented spot Z image.

     Seg2Dgel -sample:gel.tif -drawPlus:O
          # Draw '+' in center of each segmented spot overlayed in
          # the original gel.

     Seg2Dgel -sample:gel.tif -drawPlus:O -drawBoundary:O
          # Draw '+' and boundaries for each segmented spot overlayed in
          # the original gel.

     Seg2Dgel -sample:gel.tif -lowpass:13
          # Use larger 13x13 filter  for high resolution gels.

     Seg2Dgel -sample:gel.tif -lowpass:13 -laplacian:5
          # Use larger 13x13 filter and 5x5 laplacian for high resolution
gels.

     Seg2Dgel -sample:gel.tif -lowpass:13x17:1.0 -laplacian:B,C,3
```

```
          # Use larger 13x13 filter and 7x7 (2*3+1) Busse Laplacian for high
          # resolution gels.

     Seg2Dgel -sample:gel.tif -bkgrdSize:64 -thrA:30,100000 -
thrD:0.0005,100000 \
              -thrR:0.001,4.5 -lowpass:3 -ccMinSize:4 -laplacian:B,C,3 \
              -saturatedSpot:99.7 -drawBoundary:O -drawPlus:O -restOfPix
          # correct saturated spots and draw results in restOf image

     Seg2Dgel -sample:gel.tif -lowpass:7 -thrA:45,8000 -thrD:0.5:10000 -
thrR:0.04:4.0 \
              -restOfPix
          # For 1:2 sampled laser scanner gel 50 microns/pixel or 100
microns/pixel.
```

7.2 Debug option bits for the -debug switch

The following are the orthogonal octal -debug option code bits. This means you can add them together (in octal) and use that computed octal number (it will also accept decimal). The -debug:*bits,dbx1,dbx2,dby1, dby2* option is intended for use by programmers reading the source code and want to trace some of the algorithms on specific data. We have found it useful to conditionally enable traces of various processing steps for a small region [dbx1:dbx2,dby1:dby2].

```
         Octal code   Methods traced
         ==========   ================
bit:            01 =  dbugPixDBcode()
bit:            02 =  findSpotList()
bit:            04 =  propSaturatedSpot()
bit:           010 =  splitCheckSpot()
bit:           020 =  rmvConcavities()
bit:           040 =  numberComponents()
bit:          0100 =  maxPropagate()
bit:          0200 =  fillCorners()
bit:          0400 =  fillHolesInPCCconcavities()
bit:         01000 =  roundCorners()
bit:         02000 =  propSpotTo2ndDrivMaxima()
bit:         04000 =  print gray2OD[]
bit:        010000 =  rmvInteriorCCpixels()
bit:        020000 =  calcFeatures()
bit:        040000 =  addInteriorCCpixels
bit:       0100000 =  inner loop of findSpot()
bit:       0200000 =  splitPixByFBLboundaryAnal()
bit:       0400000 =  findBoundary(), pushXYboundary(), resetBnd()
bit:      01000000 =  cvtBoundaryToChainCode(), cvtChainCode_to_nextXY()
bit:      02000000 =  findNotches_in_chainCode()
```

```
bit:      04000000 =  pairChainCodeNotches(), findMinMaxPeaksInRLM(),
findPairedNotches_in_RLM()
bit:     010000000 =  drawLineBetweenNotchPair()
bit:     020000000 =  pushXYintoFBL()
bit:     040000000 =  draw Boundary Of Saturated Spots CC
bit:    0100000000 =  dump CC images At Key Points If Enabled In Code.
bit:    0200000000 =  trace Garbage Collection Calling Method.
```

8. Demonstrations

*** BEING REWRITTEN ***

8.1 Examples - samples of screen shots

To give the flavor of running the segmenter, we provide a few screen shots of the graphical user interfaces and some images generated by the segmenter for the initial version of the segmenter.

You can these images in the list below or view all of the screen shots in a single Web page.

- Graphical user interface Report window
- Command line Options window
- Stages of processing - investigate with the ImageViewer window
    1. Original image
    2. Average image
    3. Propagated central core image
    4. Magnitude 2nd derivative image
    5. Background image
    6. Segmented image
    7. Original - segmented image
    8. Final user Report Window summary statistics
- Investigating histograms of image density or spot features with ImageViewer
    1. Histogram of spot areas
    2. Histogram of spot densities
    3. Histogram spot density ranges
    4. Histogram of image pixel densities
    5. Horizonal slice of image pixel densities
- Segmenting low resolution 2D-LC-MS data
    1. Original 2D LC-MS low resolution image
    2. Segmented 2D LC-MS segmented image
    3. ImageViewer of segmented 2D LC-MS image - density histogram

4. ImageViewer of segmented 2D LC-MS image - Sx/Sy histogram

8.2 Example - output of the Report Window for a segmentation

The following Report Window output was generate for the images in the above example.

```
Seg2Dgel V.0.0.5-pre-Alpha - $Date$ - $Revision$ (Open2Dprot)
Today's date is 04/27/04 10:02:06
OK: DBUG cal.extrapolateNDwedgeMap() has mapGrayToOD[]
OK: DBUG cal.extrapolateNDwedgeMap() has mapGrayToOD[]
nullComputing window region and sizing parameters
---------------------------------------------
Window [36:497,80:505] (pixels) [rows,cols]=[512,512]
Spot Area sizing limits (10:2000) (pixels**2)
Integrated Density sizing limits (0.300:2000.000) (od)
Density difference sizing limits (0.0300:2.7000) (od)
Zonal Notch filter background window size: 32X32 (pixels)

Summary of egmented spot statistics
-----------------------------------
Total of 1256 accepted D spots accumulated density=46367.0, area=9532
Total of 424 accepted D' spots accumulated density=9140.4, area=42368
Total of 832 omitted D' spots accumulated density=10.3, area=25183
Omitted(D')/Accepted(D') spots failing D' resizing= 0.1%
Total # spots failed Area sizing[areaT1,areaT2]=[1720, 0]
Total # spots failed Density sizing[densityT1,densityT2]=[3114, 0]
Total # spots failed ODrange sizing[ODrangeT1,ODrangeT2]=[4105, 0]

List image files and generated files
------------------------------------
Input pix file           [demo\ppx\Human-AML.gif]
SSF file                 [demo\xml\Human-AML.ssf]
Central Core pix file    [demo\tmp\Human-AML-cc.gif]
Average pix file         [demo\tmp\Human-AML-avg.gif]
Mag 2nd Deriv. pix file  [demo\tmp\Human-AML-mag.gif]
Background pix file      [demo\tmp\Human-AML-bkg.gif]
Segmented pix file       [demo\tmp\Human-AML-seg.gif]
'RestOf' pix file        [demo\tmp\Human-AML-rest.gif]

FINISHED! The Sample Spot-list File, SSF, is demo\xml\Human-AML.ssf
Run time =0:0:7 (H:M:S) or 7.2 seconds

Breakdown of times and memory usage for steps in the segemntation
------------------------------------------------------------------
Step [Reading gel]        0.7 seconds  6.5Mb-tot 3.0Mb-free
```

```
Step [Init]                0.3 seconds  8.4Mb-tot 0.7Mb-free
Step [Average]             0.1 seconds  8.4Mb-tot 0.7Mb-free
Step [Ctrl core]           2.0 seconds  12.2Mb-tot 4.5Mb-free
Step [Background]          0.1 seconds  12.2Mb-tot 4.5Mb-free
Step [Estimate D']         0.0 seconds  12.2Mb-tot 4.5Mb-free
Step [Rethreshold D']      0.0 seconds  12.2Mb-tot 4.5Mb-free
Step [Save SSF]            0.4 seconds  12.2Mb-tot 4.4Mb-free
Step [Save images]         2.4 seconds  14.6Mb-tot 7.3Mb-free

Seg2Dgel - finished at: 04/27/04 10:02:13
```

8.3 Examples - part of Sample Spot-list File - with default XML format

This is part of a Sample Spot-list File to illustrate the type of data available as output. This used the default -ssfFormat:X option for the default XML format. To keep this example short, the DTD is referenced as a hyperlink where you can view it with a Web browser (or save it and view it with a text editor). The DTD is normally included in the generated SSF file.

Open2Dprot-SSF.dtd (if **-dtd** switch used), else <?xml version="1.0" ?> (if **-nodtd** switch is used). <Spot_Segmentation> <Sample_parameters> <Sample_Name>"gel-HM-019"</Sample_Name> <Simple_FileName>"gel-HM-019.gif"</Simple_FileName> <Project_directory>"demo\"</Project_directory> <date>"10/21/04 10:56:26"</date> <Open2Dprot_SSF_Version>"1.2"</Open2Dprot_SSF_Version> <cwx1>14</cwx1> <cwx2>475</cwx2> <cwy1>74</cwy1> <cwy2>509</cwy2> <Pix_Height>0</Pix_Height> <Pix_Width>0</Pix_Width> <t1Area_threshold>10.0</t1Area_threshold> <t2Area_threshold>2000.0</t2Area_threshold> <t1Density_threshold>0.3</t1Density_threshold> <t2Density_threshold>2000.0</t2Density_threshold> <t1Range_threshold>0.03</t1Range_threshold> <t2Range_threshold>4.0</t2Range_threshold> <Percent_saturation_threshold>95</Percent_saturation_threshold> <ccMinSize_threshold>2</ccMinSize_threshold> <Density_units>"optical density"</Density_units> <Background_FilterSize>32</Background_FilterSize> </Sample_parameters> <Spot> <id>1</id> <area>19</area> <intensity>29.138</intensity> <normalised_volume>25.856</normalised_volume> <volume>23.228</volume> <pixel_x_coord>462.7</pixel_x_coord> <pixel_y_coord>76.0</pixel_y_coord> <local_background>0.173</local_background> <minDensity>0.239</minDensity> <maxDensity>2.350</maxDensity> <meanDensity>1.534</meanDensity> <meanBackground>0.173</meanBackground> <merX1>460</merX1> <merX2>465</merX2> <merY1>75</merY1> <merY2>79</merY2> <sxTot>1.452</sxTot> <syTot>0.960</syTot> <sxyTot>0.917</sxyTot> <ccNumber>2</ccNumber> <spotBoundaryStr>""</spotBoundaryStr> </Spot> <Spot> <id>2</id> <area>15</area> <intensity>26.323</intensity> <normalised_volume>23.936</normalised_volume> <volume>20.982</volume> <pixel_x_coord>469.2</pixel_x_coord> <pixel_y_coord>76.0</pixel_y_coord> <local_background>0.159</local_background> <minDensity>0.291</minDensity> <maxDensity>2.350</maxDensity> <meanDensity>1.755</meanDensity> <meanBackground>0.159</meanBackground> <merX1>466</merX1> <merX2>472</merX2> <merY1>75</merY1> <merY2>79</merY2> <sxTot>1.600</sxTot> <syTot>0.787</syTot> <sxyTot>0.658</sxyTot> <ccNumber>3</ccNumber> <spotBoundaryStr>""</spotBoundaryStr> </Spot> . . . <Spot> <id>1470</id> <area>23</area> <intensity>2.164</intensity> <normalised_volume>1.397</normalised_volume> <volume>1.684</volume> <pixel_x_coord>79.8</pixel_x_coord> <pixel_y_coord>503.6</pixel_y_coord> <local_background>0.033</local_background> <minDensity>0.043</minDensity> <maxDensity>0.132</maxDensity> <meanDensity>0.094</meanDensity> <meanBackground>0.033</meanBackground> <merX1>77</merX1> <merX2>83</merX2> <merY1>500</merY1> <merY2>505</merY2> <sxTot>1.567</sxTot> <syTot>1.150</syTot> <sxyTot>1.057</sxyTot> <ccNumber>37</ccNumber> <spotBoundaryStr>""</spotBoundaryStr> </Spot> <Global_segmenter_statistics> <total_D_spots_accepted> <nbr>3172</nbr> <density>72281.0</density> <area>16020</area></total_D_spots_accepted> <total_densityPrime_spots_accepted> <nbr>1470</nbr> <densityPrime>11350.8</densityPrime> <area>45912</area></total_densityPrime_spots_accepted> <total_omitted_densityPrime_spots_accepted> <nbr>1702</nbr> <densityPrime>182.2</densityPrime> <area>26369</area></total_omitted_densityPrime_spots_accepted> <Pct_omitToAccept_densityPrime_spots_failing_t1Density_resizing>1.6</

Pct_omitToAccept_densityPrime_spots_failing_t1Density_resizing> <Nbr_Spots_Failing_Area_Sizing> <nbr_below_t1Area_thr>3324</nbr_below_t1Area_thr> <nbr_above_t2Area_thr>0</nbr_above_t2Area_thr> <nbr_below_t1Density_thr>5077</nbr_below_t1Density_thr> <nbr_above_t2Density_thr>0</nbr_above_t2Density_thr> <nbr_below_t1Range_thr>3933</nbr_below_t1Range_thr> <nbr_above_t2Range_thr>0</nbr_above_t2Range_thr></Nbr_Spots_Failing_Area_Sizing> </Global_segmenter_statistics> </Spot_Segmentation>

8.4 Examples - part of Sample Spot-list File

This is part of a Sample Spot-list File to illustrate the type of data available as output. This used the default -ssfFormat:G option (this option is depricated and the XML format should be used). This is the human readable GELLAB-II ascii format.

```
Seg2Dgel V.0.0.5-pre-Alpha - $Date$ - $Revision$ (Open2Dprot)
Sample Spot-list File: demo\xml\Gel-HM-019.ssf
Computing window region and sizing parameters
---------------------------------------------
Window [36:497,80:505] (pixels) [rows,cols]=[512,512]
Spot Area sizing limits (10:2000) (pixels**2)
Integrated Density sizing limits (0.300:2000.000) (od)
Density difference sizing limits (0.0300:2.7000) (od)
Zonal Notch filter background window size: 32X32 (pixels)

Switches:  -gui -timer -restOfPixFile -segmentedPixFile -ctlCorePixFile \
           -lowpassFilter:7 -CCminSize:2 -laplace:5 -bkgrdSize:32 \
           -bkgrdCorrection -cw:36,497,80,509 -sample:Gel-HM-019\
           -thrArea:10,2000 -thrDensity:0.3,2000 -
thrRange:0.03,2.7

Background range [0.004:0.775] od
Spot#1 MER[367:380,80:86] DR=[0.036:0.047] D/A=0.143 MnB=0.024
 1st MOM[373.5,81.9] A=79 D=11.268 D'=9.337
 Sx=3.152 Sy=1.490 Sxy=1.779 V=11.888 CC#4
Spot#2 MER[36:40,89:93] DR=[0.036:0.047] D/A=0.145 MnB=0.073
 1st MOM[37.6,91.2] A=16 D=2.314 D'=1.150
 Sx=1.316 Sy=1.206 Sxy=1.050 V=2.544 CC#17
Spot#3 MER[40:51,89:94] DR=[0.036:0.047] D/A=0.182 MnB=0.085
 1st MOM[45.5,91.6] A=51 D=9.300 D'=4.942
 Sx=3.040 Sy=1.240 Sxy=1.603 V=7.785 CC#18
Spot#4 MER[53:59,89:92] DR=[0.036:0.047] D/A=0.187 MnB=0.098
 1st MOM[56.5,91.2] A=18 D=3.370 D'=1.603
 Sx=1.662 Sy=0.912 Sxy=1.003 V=2.710 CC#19
Spot#5 MER[70:78,89:96] DR=[0.036:0.047] D/A=0.367 MnB=0.111
 1st MOM[73.5,92.9] A=43 D=15.787 D'=11.018
 Sx=1.944 Sy=1.737 Sxy=1.494 V=11.210 CC#20
Spot#6 MER[78:81,89:92] DR=[0.036:0.047] D/A=0.264 MnB=0.058
 1st MOM[79.7,90.7] A=13 D=3.426 D'=2.675
 Sx=1.078 Sy=0.910 Sxy=0.822 V=2.629 CC#21
```

```
Spot#7 MER[82:93,88:94] DR=[0.036:0.047] D/A=0.241 MnB=0.051
 1st MOM[86.8,91.4] A=55 D=13.269 D'=10.458
 Sx=2.883 Sy=1.403 Sxy=1.740 V=10.638 CC#22
Spot#8 MER[60:68,89:99] DR=[0.036:0.047] D/A=0.334 MnB=0.149
 1st MOM[64.5,94.3] A=68 D=22.708 D'=12.570
             . . .
Spot#422 MER[221:234,486:494] DR=[0.036:0.047] D/A=0.107 MnB=0.051
 1st MOM[227.6,490.5] A=61 D=6.542 D'=3.424
 Sx=2.554 Sy=1.686 Sxy=1.810 V=6.902 CC#56
Spot#423 MER[358:369,488:497] DR=[0.036:0.047] D/A=0.595 MnB=0.060
 1st MOM[363.9,492.5] A=86 D=51.184 D'=46.024
 Sx=2.390 Sy=2.197 Sxy=1.931 V=43.582 CC#60
Spot#424 MER[251:261,491:500] DR=[0.036:0.047] D/A=0.065 MnB=0.053
 1st MOM[256.5,495.5] A=57 D=3.712 D'=0.672
 Sx=2.166 Sy=1.903 Sxy=1.714 V=4.356 CC#71

Summary of segmented spot statistics
------------------------------------
Total of 1256 accepted D spots accumulated density=46367.0, area=9532
Total of 424 accepted D' spots accumulated density=9140.4, area=42368
Total of 832 omitted D' spots accumulated density=10.3, area=25183
Omitted(D')/Accepted(D') spots failing D' resizing= 0.1%
Total # spots failed Area sizing[areaT1,areaT2]=[1720, 0]
Total # spots failed Density sizing[densityT1,densityT2]=[3114, 0]
Total # spots failed ODrange sizing[ODrangeT1,ODrangeT2]=[4105, 0]

List image files and generated files
------------------------------------
Input pix file           [demo\ppx\Gel-HM-019.gif]
SSF file                 [demo\xml\Gel-HM-019.ssf]
Central Core pix file    [demo\tmp\Gel-HM-019-cc.gif]
Average pix file         [demo\tmp\Gel-HM-019-avg.gif]
Mag 2nd Deriv. pix file  [demo\tmp\Gel-HM-019-mag.gif]
Background pix file      [demo\tmp\Gel-HM-019-bkg.gif]
Segmented pix file       [demo\tmp\Gel-HM-019-seg.gif]
'RestOf' pix file        [demo\tmp\Gel-HM-019-rest.gif]

FINISHED! The Sample Spot-list File, SSF, is demo\xml\Gel-HM-019.ssf
Run time =0:0:7 (H:M:S) or 8.0 seconds
```

8.5 Examples - using command line processing for batch

It is possible to run the Seg2Dgel from the command line in your operating system. We give two examples doing this. The first example shows a script for the Microsoft Windows batch (.bat) file for processing 4 images demo-Seg2Dgel.bat file (available on the Files Mirror. The third example shows the same commands in a shell script for a Unix operating system (Linux, MacOS, Solaris, etc.) is available at from the Files mirror at demo-Seg2Dgel.sh.

Command line scripts to run the GUI version on one sample are in the second example and forth example. These may be downloaded from demo-Seg2DgelGUI.bat and demo-Seg2DgelGUI.sh.

The data for these scripts is in the **demo/** directory available with the installation which also includes the scripts. Alternatively, you can download the demo data from the Files Mirror as Demo.Z.

The installation packages are available from the Files mirror under the Seg2Dgel releases.

8.5.1 Examples - batch processing under Microsoft Windows

```
REM File: demo-Seg2Dgel.bat - segment a list of samples
REM This example assumes that all .jar files listed below and demo/
directory are
REM in the current directory. Modify for other situations.
REM
REM The JDK should be installed and version 1.4 or later is required.
REM You can download the latest JDK from http://java.sun.com/
REM
REM The files needed are listed below:
REM JAR files required and mentioned in manifest:
REM    xml-apis.jar xercesImpl.jar jai_codec.jar jai_core.jar O2Plib.jar
REM
REM demo Files:
REM     demo/ppx/gel-HM-019
REM     demo/ppx/gel-HM-071
REM     demo/ppx/gel-HM-087
REM     demo/ppx/gel-HM-096
REM Accession database file is in:
REM     demo/xml/accession.xml
REM Generated Sample Spot-list Files (SSF) are saved in:
REM     demo/xml/
REM Generated images are saved in:
REM     demo/tmp/
REM
REM P. Lemkin $Date$

echo "demo-Seg2Dgel.bat"

pwd
date /T

java -Xmx256M -jar Seg2Dgel.jar -default:2D-GEL -project:demo\ -sample:gel-
```

```
HM-019
java -Xmx256M -jar Seg2Dgel.jar -default:2D-GEL -project:demo\ -sample:gel-
HM-071
java -Xmx256M -jar Seg2Dgel.jar -default:2D-GEL -project:demo\ -sample:gel-
HM-087
java -Xmx256M -jar Seg2Dgel.jar -default:2D-GEL -project:demo\ -sample:gel-
HM-096

echo "-- Finished segmenting the samples ---"
date /T
```

8.5.1.1 Examples - batch processing with GUI under Microsoft Windows

```
REM File: demo-Seg2Dgel-GUI.bat - segment a sample using the GUI
REM This example assumes that all .jar files listed below and demo/
directory are
REM in the current directory. Modify for other situations.
REM
REM The JDK should be installed and version 1.4 or later is required.
REM You can download the latest JDK from http://java.sun.com/
REM
REM The files needed are listed below:
REM JAR files required and mentioned in manifest:
REM     xml-apis.jar xercesImpl.jar jai_codec.jar jai_core.jar O2Plib.jar
REM
REM demo Files:
REM      demo/ppx/gel-HM-019
REM Accession database file is in:
REM      demo/xml/accession.xml
REM Generated Sample Spot-list Files (SSF) are saved in:
REM      demo/xml/
REM Generated images are saved in:
REM      demo/tmp/
REM
REM P. Lemkin $Date$

echo "demo-Seg2Dgel-GUI.bat"

pwd
date /T

java -Xmx256M -jar Seg2Dgel.jar -demo -gui -project:demo\ -sample:gel-HM-019

echo "-- Finished segmenting the sample ---"
```

```
date /T
```

8.5.2 Examples - batch processing under Unix (or MacOS-X)

Because java is relatively operating system independent, the same java command lines are used with the "\" changed to "/", "REM" changed to "#", and "DATE/T" to "date" from Windows to Unix script and file path conventions.

```
##!/bin/sh
# File: demo-Seg2Dgel.sh - Unix script to segment a list of samples
# This example assumes that all .jar files listed below and demo/ directory
are
# in the current directory. Modify for other situations.
#
# The JDK should be installed and version 1.4 or later is required.
# You can download the latest JDK from http://java.sun.com/
#
# The files needed are listed below:
# JAR files required and mentioned in manifest:
#     xml-apis.jar xercesImpl.jar jai_codec.jar jai_core.jar O2Plib.jar
#
# demo sample image files:
#      demo/ppx/gel-HM-019
#      demo/ppx/gel-HM-071
#      demo/ppx/gel-HM-087
#      demo/ppx/gel-HM-096
# Accession database file is in:
#      demo/xml/accession.xml
# Generated Sample Spot-list Files (SSF) are saved in:
#      demo/xml/
# Generated images are saved in:
#      demo/tmp/
#
# P. Lemkin $Date$

echo "demo-Seg2Dgel.sh"

pwd
date

java -Xmx256M -jar Seg2Dgel.jar -default:2D-GEL -project:demo/ -sample:gel-
HM-019
java -Xmx256M -jar Seg2Dgel.jar -default:2D-GEL -project:demo/ -sample:gel-
```

```
HM-071
java -Xmx256M -jar Seg2Dgel.jar -default:2D-GEL -project:demo/ -sample:gel-
HM-087
java -Xmx256M -jar Seg2Dgel.jar -default:2D-GEL -project:demo/ -sample:gel-
HM-096

echo "-- Finished segmenting the samples ---"

date
```

8.5.2.1 Examples - batch processing with GUI under Unix (or MacOS-X)

```
#!/bin/sh
# File: demo-Seg2Dgel-GUI.sh - Unix script to segment a sample using the GUI
# This example assumes that all .jar files listed below and demo/ directory
are
# in the current directory. Modify for other situations.
#
# The JDK should be installed and version 1.4 or later is required.
# You can download the latest JDK from http://java.sun.com/
#
# The files needed are listed below:
# JAR files required and mentioned in manifest:
#     xml-apis.jar xercesImpl.jar jai_codec.jar jai_core.jar O2Plib.jar
#
# demo sample image files:
#      demo/ppx/gel-HM-019
# Accession database file is in:
#      demo/xml/accession.xml
# Generated Sample Spot-list Files (SSF) are saved in:
#      demo/xml/
# Generated images are saved in:
#      demo/tmp/
#
# P. Lemkin $Date$

echo "demo-Seg2Dgel-GUI.sh"

pwd
date

java -Xmx256M -jar Seg2Dgel.jar -demo -gui -project:demo/ -sample:gel-HM-019

echo "-- Finished segmenting the samples ---"
```

```
date
```

9. Seg2Dgel References

These papers (a subset of the GELLAB-II papers), reference the GELLAB-II segmenter. The Open2Dprot Java-language Seg2Dgel program was derived from the GELLAB-II C-language program as well as from code from the MAExplorer and Flicker projects. New Java code was added as well. Although Seg2Dgel has been enhanced in many ways, the basic algorithm is similar so these papers may be useful for more details on the algorithm.

1. Lipkin, L.E., Lemkin, P.F. (1980) Database techniques for multiple PAGE (2D gel) analysis. *Clinical Chemistry* **26**, 1403-1413.
2. Lemkin, P., Lipkin, L. (1981) GELLAB: A computer system for 2D gel electrophoresis analysis. I. Segmentation and preliminaries. *Computers in Biomedical Research* **14**, 272-297.
3. Lester, E.P., Lemkin, P.F., Lipkin, L.E. (1981) New Dimensions in Protein Analysis - 2D Gels Coming of Age Through Image Processing, Invited paper, *Analytical Chemistry* **53**, 390A-397A.
4. Lemkin, P.F., Lipkin, L.E. (1983) Database Techniques for 2D Electrophoretic Gel Analysis, in Computing in Biological Science, Elsevier/North-Holland, M. Geisow, A. Barrett (eds), 181-226.
5. Lemkin, P.F., Lipkin, L.E., Lester, E.P. (1982) Extensions to the GELLAB 2D Electrophoresis Gel Analysis System. Paper given at "Clinical Applications of 2D Electrophoresis", Mayo Clinic, Nov. 15-18, 1981. *Clinical Chemistry* 840-849.
6. Lemkin, P.F., Lester, E.P. (1989) Database and Search Techniques for 2D Gel Protein Data: A Comparison of Paradigms For Exploratory Data Analysis and Prospects for Biological Modeling, *Electrophoresis* **10**(2): 122-140.
7. Lemkin, P.F. (1989) GELLAB-II, A workstation based 2D electrophoresis gel analysis system, in proceedings of Two-Dimensional Electrophoresis T. Endler, S.Hanash (Eds), Vienna Austria, Nov 8-11, 1988, VCH Press, W.Germany. pp 53-57.
8. Lemkin, P.F., Rogan, P. (1991) Automatic Detection of noisy spots in two-dimensional Southern Blots, *Applied and Theoretical Electrophoresis* **2**, 141-149.
9. Myrick, J.E., Lemkin, P.F., Robinson, M.K., Upton, K.M. (1993) Comparison of the BioImage Visage(TM) 2,000 and the GELLAB-II two-dimensional electrophoretic analysis systems. *Applied and Theoretical Electrophoresis* **3**, 335-346.

10. Wu, Y., Lemkin, P.F., Upton, K. (1993) A fast spot segmentation algorithm for 2D electrophoresis analysis. *Electrophoresis* **14**, 1350-1356.

Contact us

Seg2Dgel is a contributed program available at open2dprot.sourceforge.net/Seg2Dgel

Powered by SourceForge Logo

Revised: 12/21/2005

Sample Screen Shots for Seg2Dgel Segmenter - pre-alpha version

To give the flavor of running the segmenter, we provide a few screen shots of the graphical user interface and some images generated by the segmenter for the initial version of the segmenter.

1. Graphical user interface Report window

2. Command line Options window

3. Stages of processing - investigate with the ImageViewer window
   - 3.1 original image
   - 3.2 average image
   - 3.3 propagated central core image
   - 3.4 magnitude 2nd derivative image
   - 3.5 background image
   - 3.6 segmented image
   - 3.7 original - segmented image
   - 3.8 final user Report Window summary statistics

4. Investigating histograms of image density or spot features with ImageViewer
   - 4.1 histogram of spot areas
   - 4.2 histogram of spot densities
   - 4.3 histogram spot density ranges
   - 4.4 histogram of image pixel densities
   - 4.5 horizonal slice of image pixel densities

5. Segmenting low resolution 2D-LC-MS data
   - 5.1 Original 2D LC-MS low resolution image

1. Report window

This is the main graphical user interface when Seg2Dgel is run with the -gui switch. It contains a text logging area (center), pull-down menus, and often-used command buttons at the bottom.

```
Seg2Dgel - finished at: 08/06/04 16:41:35
Seg2Dgel $Date: 2004/08/05 20:20:17 $ - $Revision: 1.17 $ (Open2Dprot)
Today's date is 08/06/04 16:41:36
Switches:  -gui
Switches:  -gui -timer -accessionDB:accession.txt -ssfFormat:X
           -restOfPixFile -segmentedPixFile -ctlCorePixFile -lowpassFilter:7
           -laplacian:5 -backgroundSize:32 -CCminSize:2 -thrArea:10,2000
           -thrDensity:0.3,2000 -thrRange:0.03,4.0 -projDir:demo/

Reading accession database file: demo\xml\accession.txt
Computing window region and sizing parameters
---------------------------------------------
Window [14:475,74:505] (pixels) [rows,cols]=[512,512]
Spot Area sizing limits (10:2000) (pixels**2)
Integrated Density sizing limits (0.300:2000.000) (od)
Density difference sizing limits (0.0300:4.0000) (od)
Zonal Notch filter background window size: 32X32 (pixels)

Summary of segmented spot statistics
------------------------------------
Total of 2015 accepted D spots accumulated density=61294.0, area=11752
Total of 933 accepted D' spots accumulated density=13292.0, area=76350
Total of 1082 omitted D' spots accumulated density=119.4, area=23119
Omitted(D')/Accepted(D') spots failing D' resizing= 0.9%
Total # spots failed Area sizing[areaT1,areaT2]=[1540, 0]
Total # spots failed Density sizing[densityT1,densityT2]=[3150, 0]
Total # spots failed ODrange sizing[ODrangeT1,ODrangeT2]=[3109, 0]

List of image files and generated files
---------------------------------------
Input pix file           [demo\ppx\gel-HM-019.gif]
SSF file                 [demo\xml\gel-HM-019.xml]
Central Core pix file    [demo\tmp\gel-HM-019-cc.gif]
Average pix file         [demo\tmp\gel-HM-019-avg.gif]
Mag 2nd Deriv. pix file  [demo\tmp\gel-HM-019-mag.gif]
Background pix file      [demo\tmp\gel-HM-019-bkg.gif]
Segmented pix file       [demo\tmp\gel-HM-019-seg.gif]
'RestOf' pix file        [demo\tmp\gel-HM-019-rest.gif]

FINISHED! The Sample Spot-list File, SSF, is demo\xml\gel-HM-019.xml
Run time =0:0:2 (H:M:S) or 2.5 seconds

Breakdown of times and memory usage for steps in the segenntation
-----------------------------------------------------------------
Step []                    0.0 seconds  532.7Mb-tot 501.5Mb-free
Step [Reading gel]         0.2 seconds  532.7Mb-tot 528.9Mb-free
Step [Init]                0.0 seconds  532.7Mb-tot 526.7Mb-free
Step [Average]             0.1 seconds  532.7Mb-tot 526.7Mb-free
Step [Ctrl core]           0.7 seconds  532.7Mb-tot 500.4Mb-free
Step [Background]          0.0 seconds  532.7Mb-tot 500.4Mb-free
Step [Estimate D']         0.0 seconds  532.7Mb-tot 500.4Mb-free
Step [Rethreshold D']      0.0 seconds  532.7Mb-tot 500.4Mb-free
Step [Save images]         1.1 seconds  532.7Mb-tot 525.7Mb-free
Step [Save SSF]            0.3 seconds  532.7Mb-tot 504.5Mb-free
Step [Done]                0.0 seconds  532.7Mb-tot 504.5Mb-free

Seg2Dgel - finished at: 08/06/04 16:41:38
```

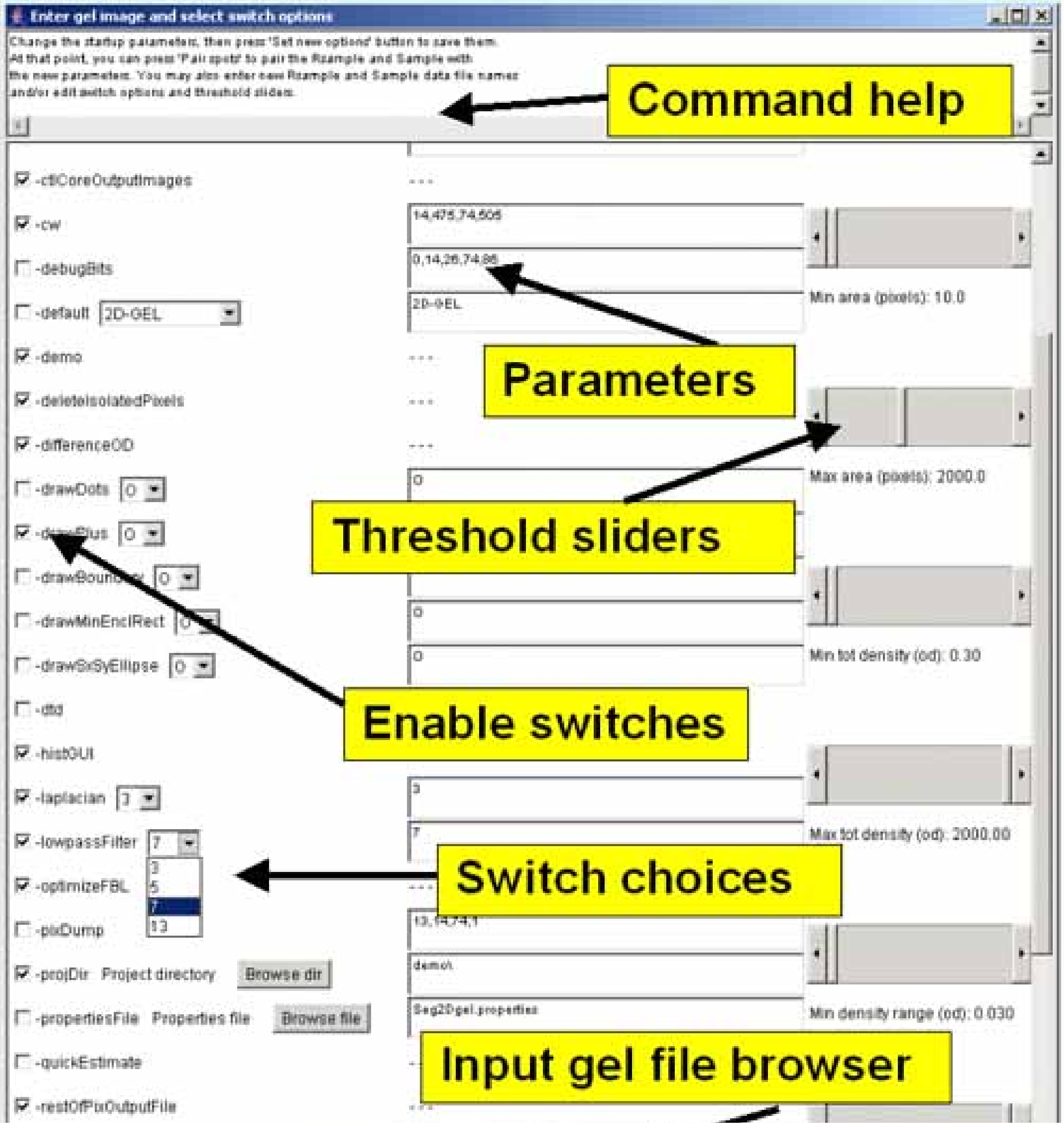
Seg2Dgel - finished at: 08/06/04 16:41:38

Done

---

2. Command line Options window

This popup window is used to editing the command line parameter options prior to resegmenting a gel. The Options window is poped up by pressing the **Image viewer** button in the Report window.



Enter gel image and select switch options

Change the startup parameters, then press 'Set new options' button to save them.
At that point, you can press 'Pair spots' to pair the Rsample and Sample with
the new parameters. You may also enter new Rsample and Sample data file names
and/or edit switch options and threshold sliders.

☑ -ctlCoreOutputImages ...
☑ -cw 14,475,74,505
☐ -debugBits 0,14,26,74,86
☐ -default 2D-GEL 2D-GEL Min area (pixels): 10.0
☑ -demo ...
☑ -deleteIsolatedPixels ...
☑ -differenceOD ...
☐ -drawDots 0 0 Max area (pixels): 2000.0
☑ -drawPlus 0
☐ -drawBoundary 0
☐ -drawMinEnclRect 0 0
☐ -drawSxSyEllipse 0 0 Min tot density (od): 0.30
☐ -dtd
☑ -histGUI
☑ -laplacian 3 3
☑ -lowpassFilter 7 7 Max tot density (od): 2000.00
3
5
7
13
☑ -optimizeFBL ...
☐ -pixDump 13,14,74,1
☑ -projDir Project directory demo\
☐ -propertiesFile Properties file Seg2Dgel.properties Min density range (od): 0.030
☐ -quickEstimate
☑ -restOfPixOutputFile ...

☑ -restOfPixOutputFile

☑ -sample Input gel pix file or name: demo/ppn/gel-HM-019.gif

☐ -saturatedSpotPropagate: 95.0

Max density range (od): 4.000

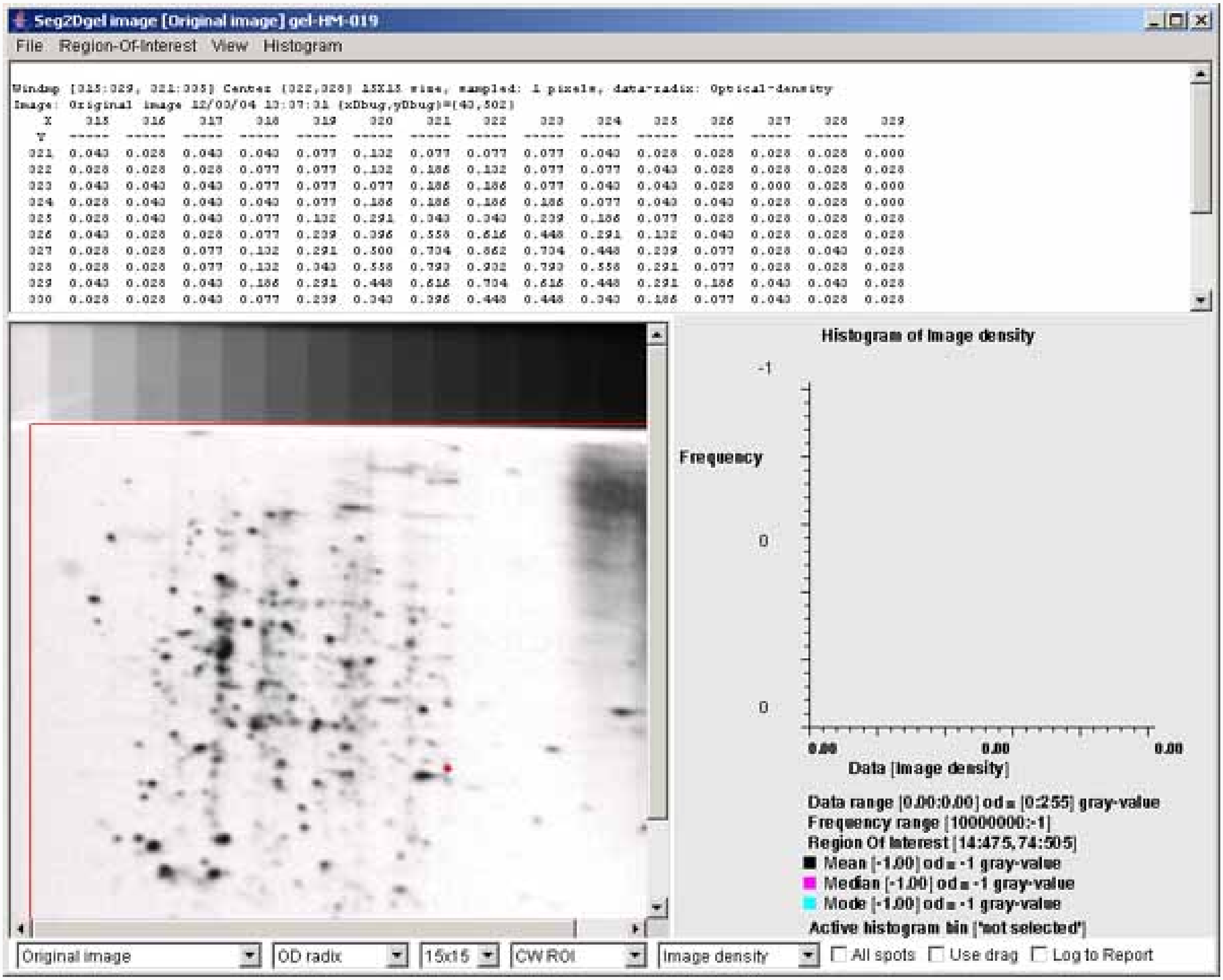
---

3. Stages of processing - images generated during the segmentation

If you enable the segmenter to generate images, you can view the various stages of processing using the ImageViewer. These are enabled using the **-ctlCorePixOutputFile**, **-restOfPixOutputFile**, and **-segmentedPixOutputFile** switches.

3.1 Original image shown in the Image viewer

This is an example of the original gel image (used in this example and included in the download of Seg2Dgel). The image is displayed in the Image viewer window that is poped up by pressing the **Image viewer** button in the Report window. A 15x15 pixel window of numeric data is shown at the top of the image (the spot has a red "+"). Because the radix was set to "OD radix", the data is in optical density values rather than gray values.



```
Window [315:329, 321:335] Center (322,328) 15X15 size, sampled: 1 pixels, data-radix: Optical-density
Image: Original image 12/03/04 13:37:31 (xDbug,yDbug)=(43,502)
   X    315    316    317    318    319    320    321    322    323    324    325    326    327    328    329
   Y  -----  -----  -----  -----  -----  -----  -----  -----  -----  -----  -----  -----  -----  -----  -----
 321  0.043  0.028  0.043  0.043  0.077  0.132  0.077  0.077  0.077  0.043  0.028  0.028  0.028  0.028  0.000
 322  0.028  0.028  0.028  0.077  0.077  0.132  0.186  0.132  0.077  0.077  0.043  0.028  0.028  0.028  0.028
 323  0.043  0.043  0.043  0.077  0.077  0.077  0.186  0.186  0.077  0.043  0.043  0.028  0.000  0.028  0.000
 324  0.028  0.043  0.043  0.043  0.077  0.186  0.186  0.186  0.186  0.077  0.043  0.043  0.028  0.028  0.000
 325  0.028  0.043  0.043  0.077  0.132  0.291  0.343  0.343  0.239  0.186  0.077  0.028  0.028  0.028  0.028
 326  0.043  0.028  0.028  0.077  0.239  0.396  0.558  0.616  0.448  0.291  0.132  0.043  0.028  0.028  0.028
 327  0.028  0.028  0.077  0.132  0.291  0.500  0.704  0.862  0.704  0.448  0.239  0.077  0.028  0.043  0.028
 328  0.028  0.028  0.077  0.132  0.343  0.558  0.793  0.932  0.793  0.558  0.291  0.077  0.028  0.028  0.028
 329  0.043  0.028  0.043  0.186  0.291  0.448  0.616  0.704  0.616  0.448  0.291  0.186  0.043  0.043  0.028
 330  0.028  0.028  0.043  0.077  0.239  0.343  0.396  0.448  0.448  0.343  0.186  0.077  0.043  0.028  0.028
```

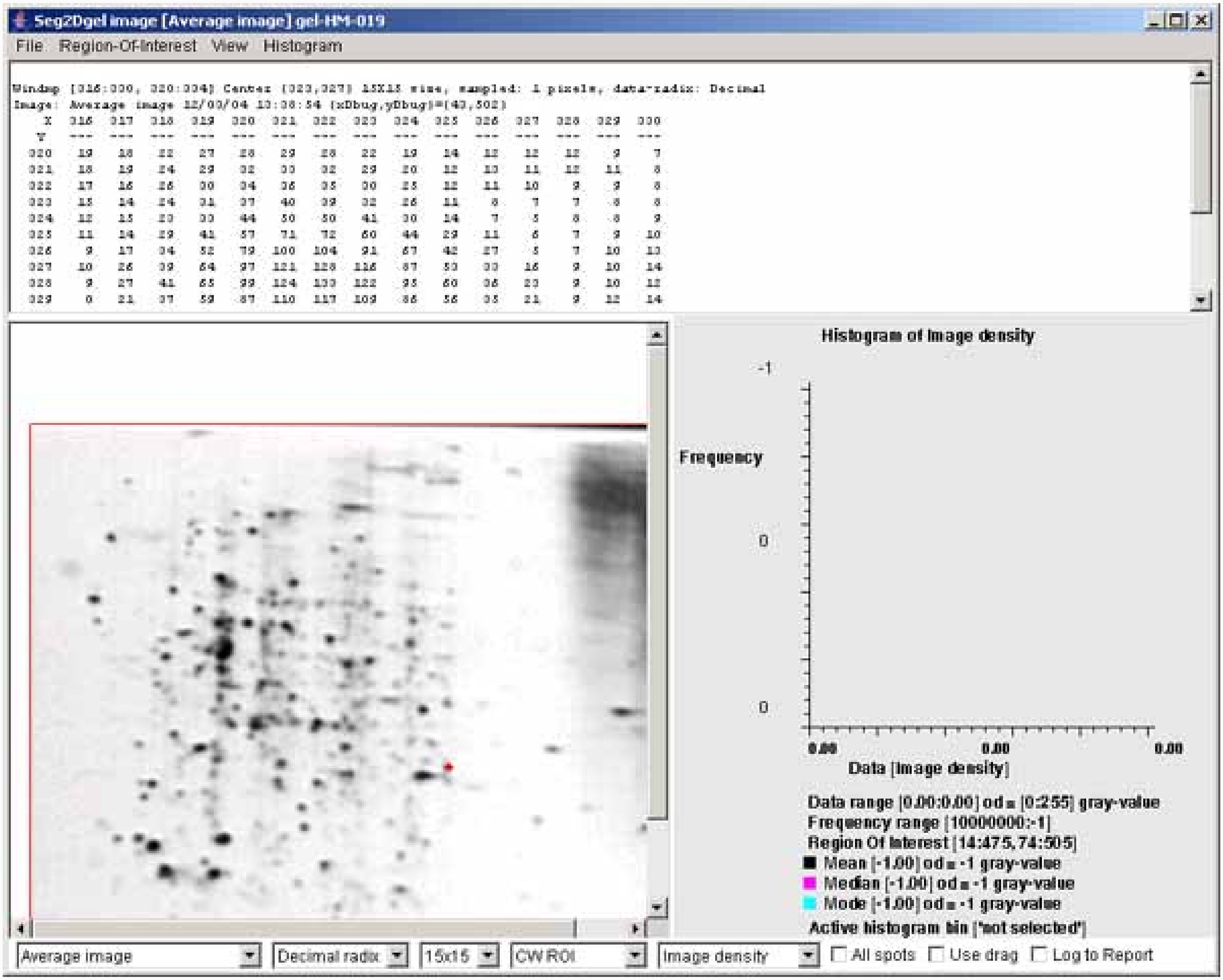
Data range [0.00:0.00] od = [0:255] gray-value
Frequency range [10000000:-1]
Region Of Interest [14:475,74:505]
■ Mean [-1.00] od = -1 gray-value
■ Median [-1.00] od = -1 gray-value
■ Mode [-1.00] od = -1 gray-value
Active histogram bin ['not selected']

Original image | OD radix | 15x15 | CW ROI | Image density | ☐ All spots ☐ Use drag ☐ Log to Report

---

3.2 Average image of the original image using a gaussian filter

During the segmentation procedure, the origial image is first smoothed using a gaussian filter using the `-lowPass:7` 7x7 pixel filter. A 15x15 pixel window of decimal numeric data is shown at the top of the image (the spot has a red "+").

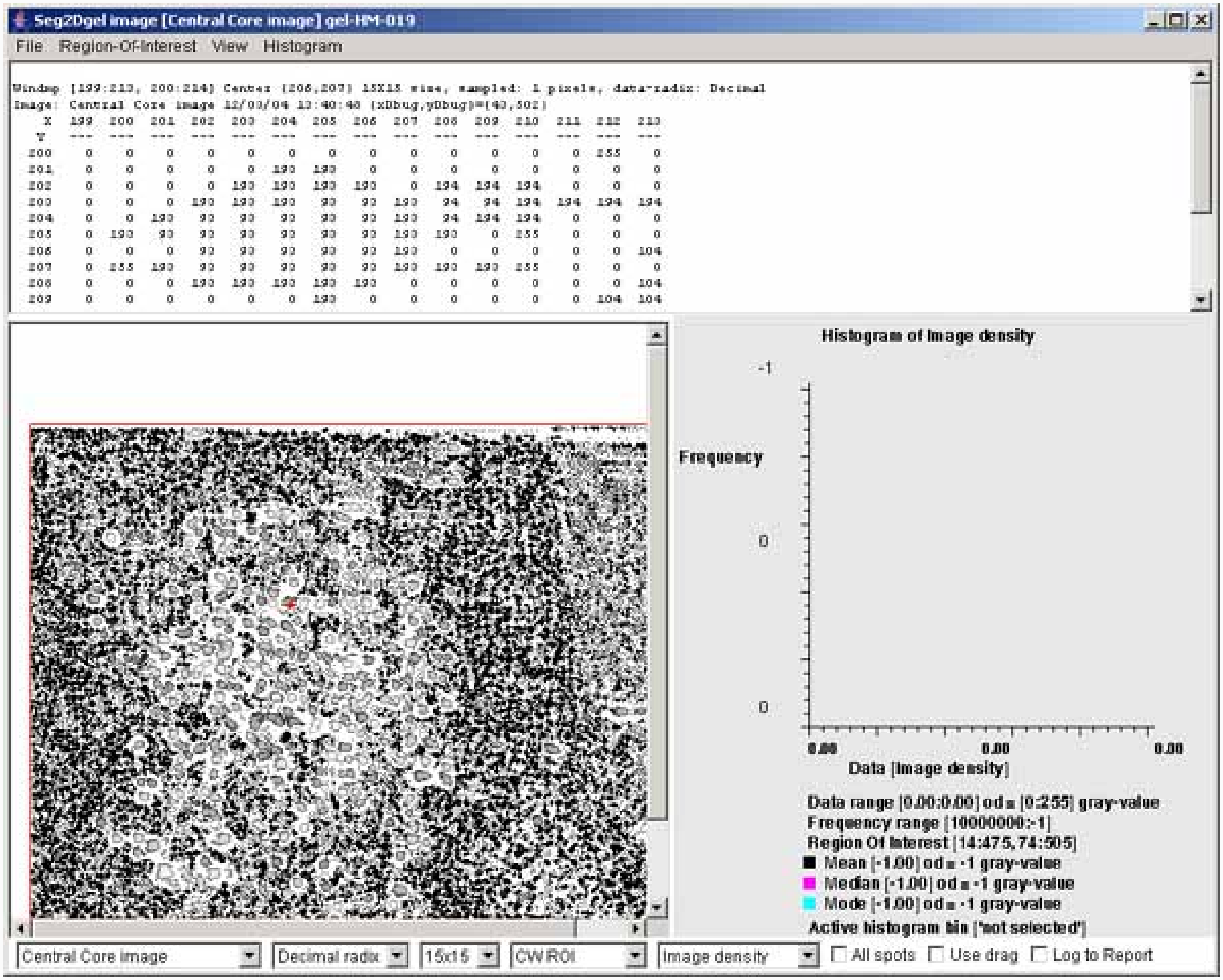


```
Window [316:330, 320:334] Center [323,327] 15X15 size, sampled: 1 pixels, data-radix: Decimal
Image: Average image 12/03/04 13:38:54 (xDbug,yDbug)=(43,502)
    X  316  317  318  319  320  321  322  323  324  325  326  327  328  329  330
    Y  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---
  320   19   18   22   27   28   29   28   22   19   14   12   12   12    9    7
  321   18   19   24   29   32   33   32   29   20   12   13   11   12   11    8
  322   17   16   26   30   34   36   35   30   25   12   11   10    9    9    8
  323   15   14   24   31   37   40   39   32   26   11    8    7    7    8    8
  324   12   15   23   33   44   50   50   41   30   14    7    5    8    8    9
  325   11   14   29   41   57   71   72   60   44   29   11    6    7    9   10
  326    9   17   34   52   79  100  104   91   67   42   27    5    7   10   13
  327   10   26   39   64   97  121  128  116   87   53   33   16    9   10   14
  328    9   27   41   65   99  124  133  122   95   60   38   23    9   10   12
  329    8   21   37   59   87  110  117  109   88   56   35   21    9   12   14
```

Data range [0.00:0.00] od = [0:255] gray-value
Frequency range [10000000:-1]
Region Of Interest [14:475,74:505]
■ Mean [-1.00] od = -1 gray-value
■ Median [-1.00] od = -1 gray-value
■ Mode [-1.00] od = -1 gray-value
Active histogram bin ['not selected']

3.3 Propagated central core image

Then, the propagated central core image computing by segmenting the average image using a laplacian and spot finding algorithm described in the reference manual using the `-laplace:5` 7x7 pixel filter.. A 15x15 pixel window of decimal numeric data is shown at the top of the image (the spot has a red "+"). Pixels within the central core have values in the range [2:99] whereas pixels in the propagated central core have values in the range [102:199].



```
Window [199:213, 200:214] Center [206,207] 15X15 size, sampled: 1 pixels, data-radix: Decimal
Image: Central Core image 12/03/04 13:40:48 (xDbug,yDbug)=[43,502]
    X  199  200  201  202  203  204  205  206  207  208  209  210  211  212  213
    Y  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---
  200    0    0    0    0    0    0    0    0    0    0    0    0    0  255    0
  201    0    0    0    0    0  193  193    0    0    0    0    0    0    0    0
  202    0    0    0    0  193  193  193  193    0  194  194  194    0    0    0
  203    0    0    0  193  193  193   93   93  193   94   94  194  194  194  194
  204    0    0  193   93   93   93   93   93  193   94  194  194    0    0    0
  205    0  193   93   93   93   93   93   93  193  193    0  255    0    0    0
  206    0    0    0   93   93   93   93   93  193    0    0    0    0    0  104
  207    0  255  193   93   93   93   93   93  193  193  193  255    0    0    0
  208    0    0    0  193  193  193  193  193    0    0    0    0    0    0  104
  209    0    0    0    0    0    0  193    0    0    0    0    0    0  104  104
```

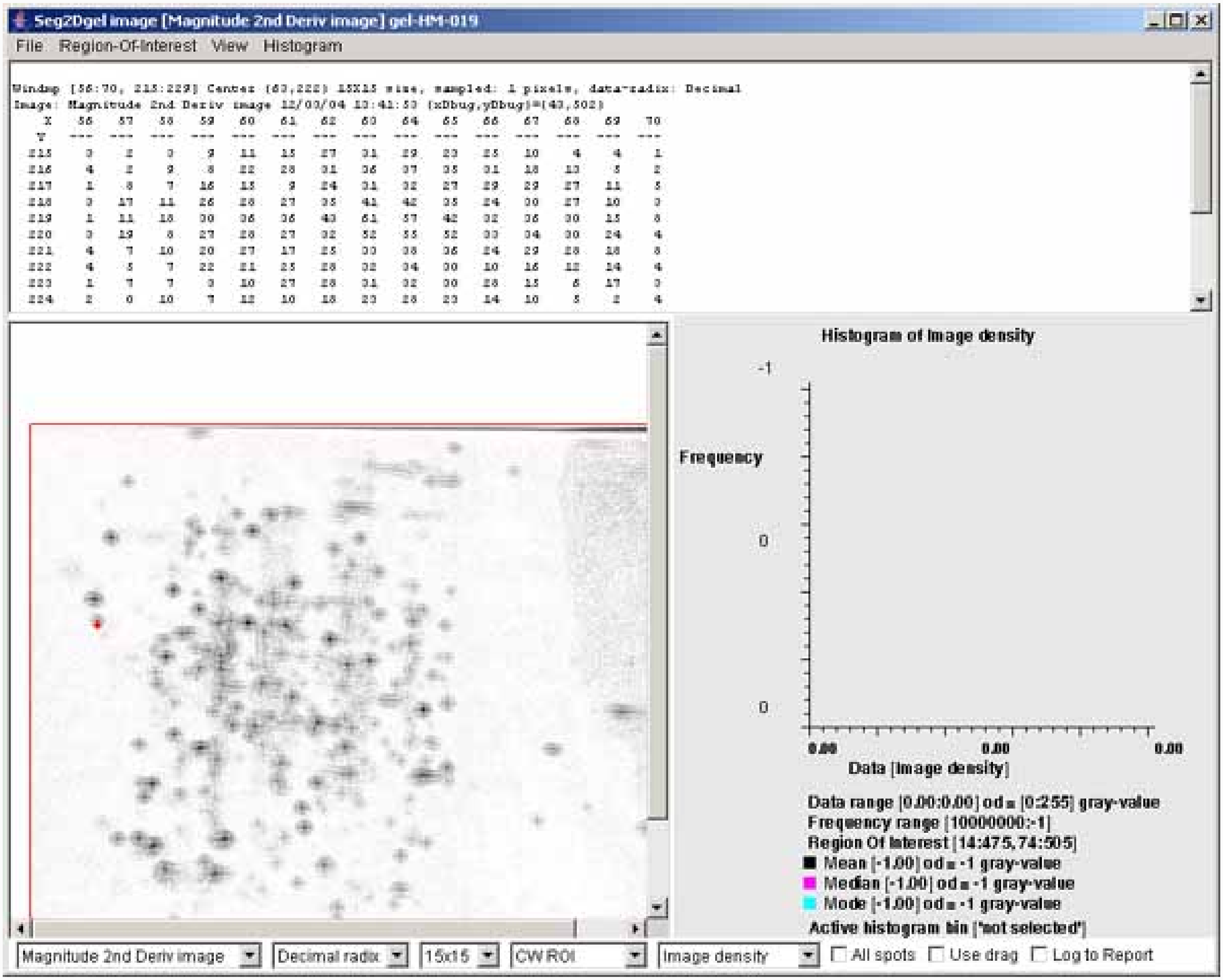
Data range [0.00:0.00] od = [0:255] gray-value
Frequency range [10000000:-1]
Region Of Interest [14:475,74:505]
■ Mean [-1.00] od = -1 gray-value
■ Median [-1.00] od = -1 gray-value
■ Mode [-1.00] od = -1 gray-value
Active histogram bin ['not selected']

Central Core image | Decimal radix | 15x15 | CW ROI | Image density | ☐ All spots | ☐ Use drag | ☐ Log to Report

---

3.4 Magnitude 2nd derivative image

Part of the computation of the propagated central core image involves computing the magnitude of the laplacian image described in the reference manual. A 15x15 pixel window of decimal numeric data is shown at the top of the image (the spot has a red "+").



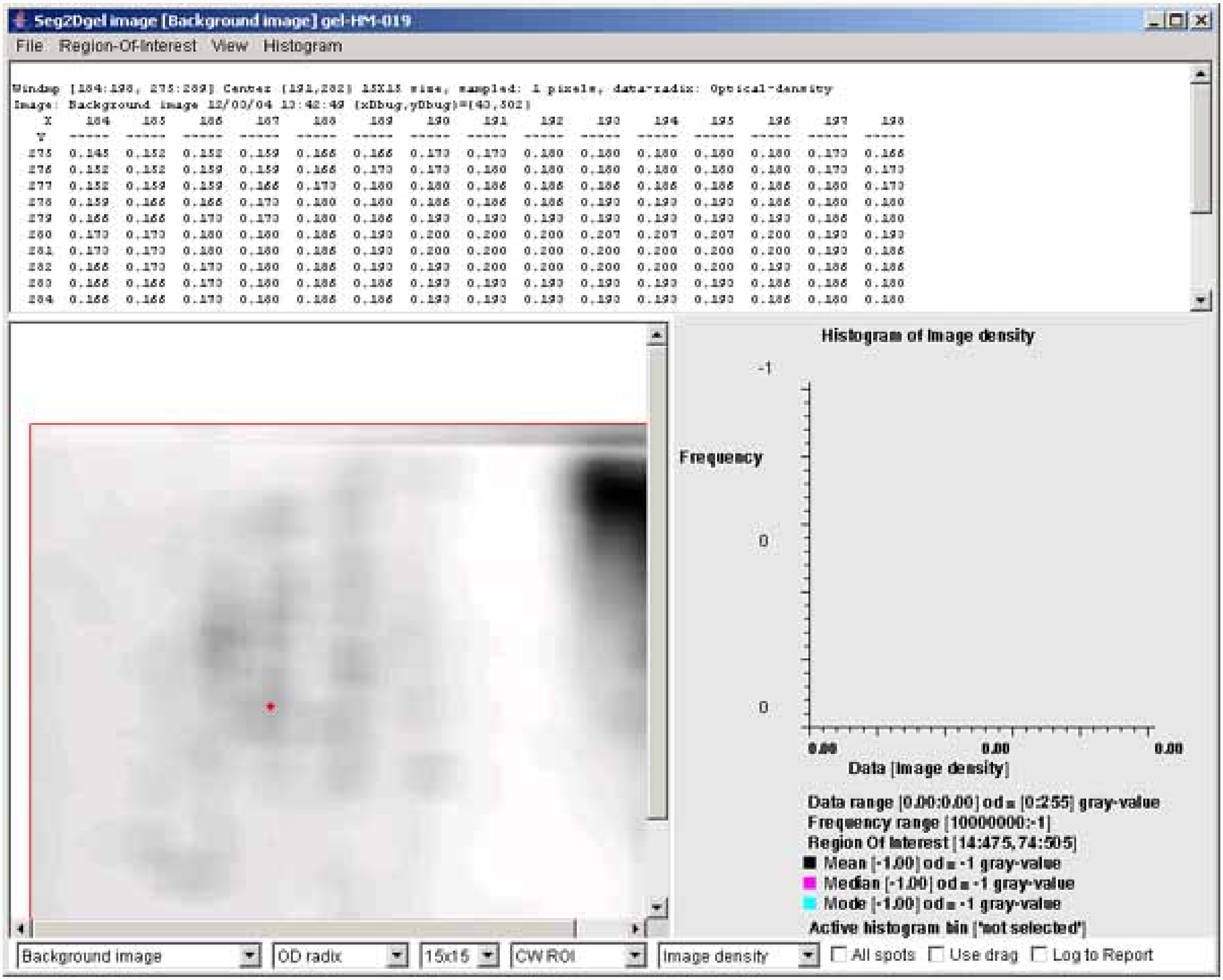
```
Window [56:70, 215:229] Center (63,222) 15X15 size, sampled: 1 pixels, data-radix: Decimal
Image: Magnitude 2nd Deriv image 12/03/04 13:41:53 (xDbug,yDbug)=(43,502)
   X   56   57   58   59   60   61   62   63   64   65   66   67   68   69   70
   Y  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---
 215    3    2    3    9   11   15   27   31   29   23   25   10    4    4    1
 216    4    2    9    8   22   28   31   36   37   35   31   18   13    5    2
 217    1    8    7   16   15    9   24   31   32   27   29   29   27   11    5
 218    3   17   11   26   28   27   35   41   42   35   24   30   27   10    3
 219    1   11   18   30   36   36   43   61   57   42   32   36   30   15    8
 220    3   19    8   27   28   27   32   52   55   52   33   34   30   24    4
 221    4    7   10   20   27   17   25   33   38   36   24   29   28   18    8
 222    4    5    7   22   21   25   28   32   34   30   10   16   12   24    4
 223    1    7    7    3   10   27   28   31   32   30   28   15    6   17    3
 224    2    0   10    7   12   10   18   23   28   23   14   10    5    2    4
```

Data range [0.00:0.00] od = [0:255] gray-value
Frequency range [10000000:-1]
Region Of Interest [14:475,74:505]
■ Mean [-1.00] od = -1 gray-value
■ Median [-1.00] od = -1 gray-value
■ Mode [-1.00] od = -1 gray-value
Active histogram bin ['not selected']

3.5 Background image

After the spots are initially segmented, the spots are removed from a copy of the original image. This image is essentially the background of the gel. This is then smoothed using a zonal notch filter described in the reference manual. This background image is then used as a table-lookup to estimate the background at the centroids of spots. A 15x15 pixel window of numeric data is shown at the top of the image (the spot has a red "+"). Because the radix was set to "OD radix", the background data is in optical density values rather than gray values. The density corrected for background is computed as

```
density' = density - (area * meanBackgroundDensity).
```

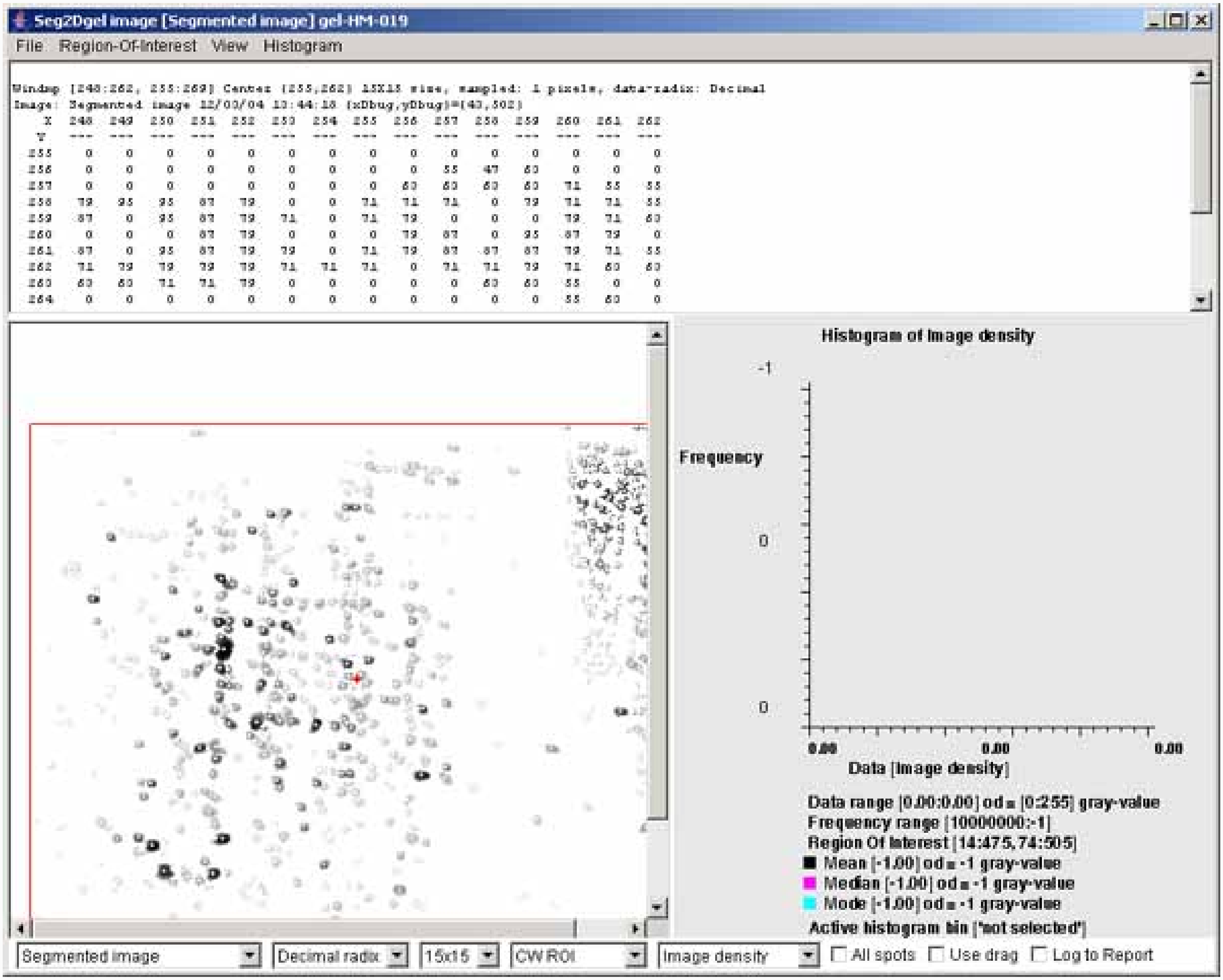


```
Window [184:198, 275:289] Center [191,282] 15X15 size, sampled: 1 pixels, data-radix: Optical-density
Image: Background image 12/03/04 13:42:49 (xDbug,yDbug)=(43,502)
   X    184    185    186    187    188    189    190    191    192    193    194    195    196    197    198
   Y  -----  -----  -----  -----  -----  -----  -----  -----  -----  -----  -----  -----  -----  -----  -----
 275  0.145  0.152  0.152  0.159  0.166  0.166  0.173  0.173  0.180  0.180  0.180  0.180  0.180  0.173  0.166
 276  0.152  0.152  0.159  0.159  0.166  0.173  0.173  0.180  0.180  0.180  0.180  0.180  0.180  0.173  0.173
 277  0.152  0.159  0.159  0.166  0.173  0.180  0.180  0.186  0.186  0.186  0.186  0.186  0.186  0.180  0.173
 278  0.159  0.166  0.166  0.173  0.180  0.180  0.186  0.186  0.186  0.193  0.193  0.193  0.186  0.180  0.180
 279  0.166  0.166  0.173  0.173  0.180  0.186  0.193  0.193  0.193  0.193  0.193  0.193  0.193  0.186  0.180
 280  0.173  0.173  0.180  0.180  0.186  0.193  0.200  0.200  0.200  0.207  0.207  0.207  0.200  0.193  0.193
 281  0.173  0.173  0.180  0.180  0.186  0.193  0.200  0.200  0.200  0.200  0.200  0.200  0.200  0.193  0.186
 282  0.166  0.173  0.173  0.180  0.186  0.193  0.193  0.200  0.200  0.200  0.200  0.200  0.193  0.186  0.186
 283  0.166  0.166  0.173  0.180  0.186  0.186  0.193  0.193  0.193  0.193  0.193  0.193  0.186  0.186  0.180
 284  0.166  0.166  0.173  0.180  0.186  0.186  0.193  0.193  0.193  0.193  0.193  0.193  0.186  0.180  0.180
```

```
Data range [0.00:0.00] od = [0:255] gray-value
Frequency range [10000000:-1]
Region Of Interest [14:475,74:505]
■ Mean [-1.00] od = -1 gray-value
■ Median [-1.00] od = -1 gray-value
■ Mode [-1.00] od = -1 gray-value
Active histogram bin ["not selected"]
```

3.6 Segmented image

The segmented image image shows the spots extracted from the original image. The centers of spots are indicated with a white "+". The data for the selected spot (red "+") is shown at the top. (When debugging, additional information about the spot is available.)

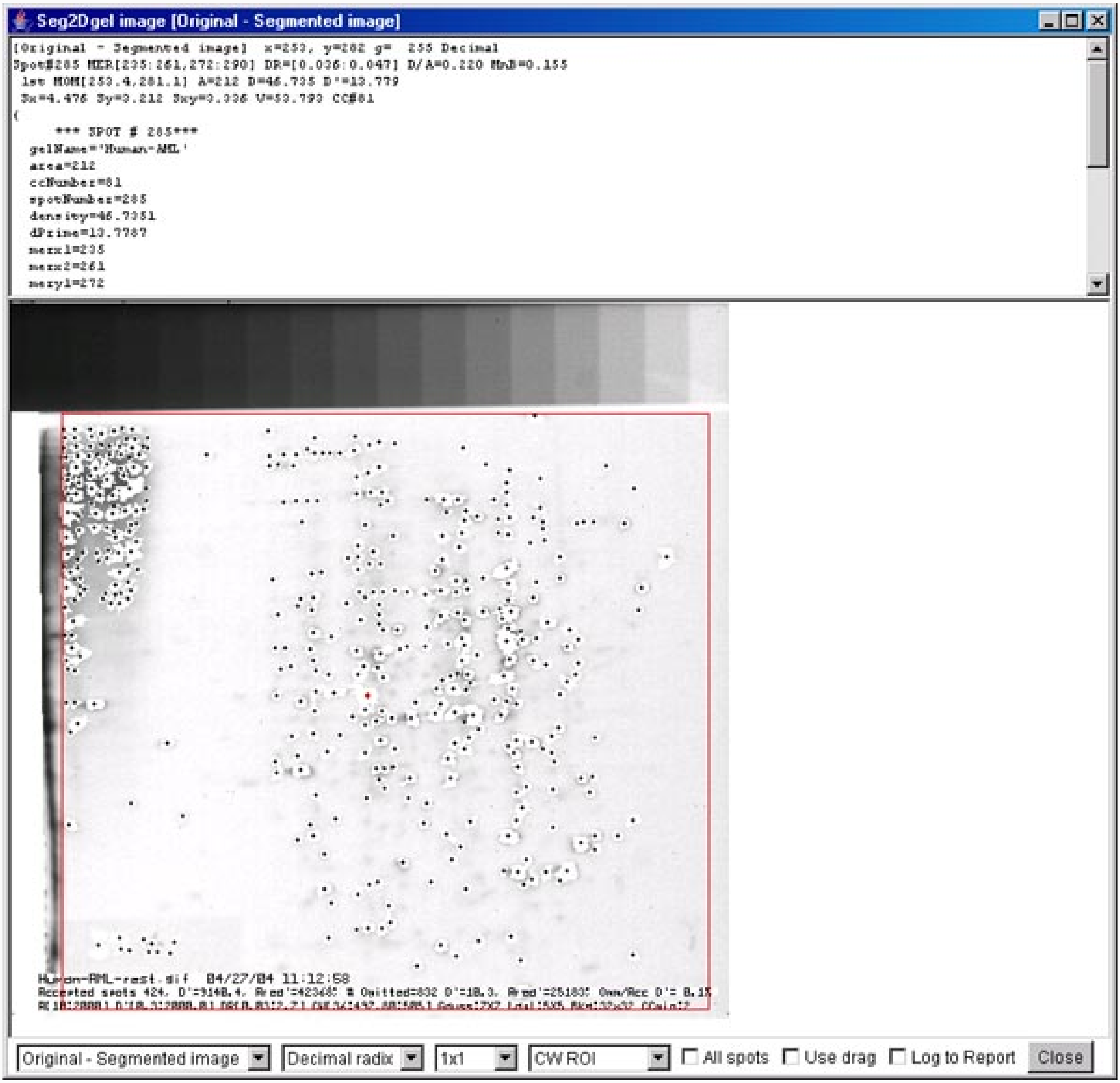


```
Window [248:262, 255:269] Center (255,262) 15X15 size, sampled: 1 pixels, data-radix: Decimal
Image: Segmented image 12/03/04 13:44:18 (xDbug,yDbug)=(43,502)
    X  248  249  250  251  252  253  254  255  256  257  258  259  260  261  262
   Y  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---
  255    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
  256    0    0    0    0    0    0    0    0    0   55   47   63    0    0    0
  257    0    0    0    0    0    0    0    0   63   63   63   63   71   55   55
  258   79   95   95   87   79    0    0   71   71   71    0   79   71   71   55
  259   87    0   95   87   79   71    0   71   79    0    0    0   79   71   63
  260    0    0    0   87   79    0    0    0   79   87    0   95   87   79    0
  261   87    0   95   87   79   79    0   71   79   87   87   87   79   71   55
  262   71   79   79   79   79   71   71   71    0   71   71   79   71   63   63
  263   63   63   71   71   79    0    0    0    0    0   63   63   55    0    0
  264    0    0    0    0    0    0    0    0    0    0    0    0   55   63    0
```

Data range [0.00:0.00] od = [0:255] gray-value
Frequency range [10000000:-1]
Region Of Interest [14:475,74:505]
■ Mean [-1.00] od = -1 gray-value
■ Median [-1.00] od = -1 gray-value
■ Mode [-1.00] od = -1 gray-value
Active histogram bin ['not selected']

3.7 The "RestOf" image or (original image - segmented image)

The "RestOf" image is computated as the original image less the segmented image. This shows any spots or parts of spots that the segmenter may have missed and can be used to help judge the quality of the segmentation. The centers of spots are indicated with a black "+".



```
[Original - Segmented image]  x=253, y=282 g=  255 Decimal
Spot#285 MER[235:261,272:290] DR=[0.036:0.047] D/A=0.220 MnB=0.155
 1st MOM[253.4,281.1] A=212 D=46.735 D'=13.779
 Sx=4.476 Sy=3.212 Sxy=3.336 V=53.793 CC#81
{
     *** SPOT # 285***
  gelName='Human-AML'
  area=212
  ccNumber=81
  spotNumber=285
  density=46.7351
  dPrime=13.7787
  merx1=235
  merx2=261
  mery1=272
```

3.8 final user Report Window with showing summary statistics

The Report Window shows the sumary results of the segmentation. This includes the numbers of spots segmented, and omitted because of failing the threshold sizing (by area, density, and density range). It also shows the names of the images that were generated in the *tmp* directory. Finally, it shows the amount of time the segmenter spent in the various stages of processing.

```
Seg2Dgel - finished at: 08/06/04 16:41:35
Seg2Dgel $Date: 2004/08/05 20:20:17 $ - $Revision: 1.17 $ (Open2Dprot)
Today's date is 08/06/04 16:41:36
Switches:  -gui
Switches:  -gui -timer -accesssionDB:accession.txt -ssfFormat:X
           -restOfPixFile -segmentedPixFile -ctlCorePixFile -lowpassFilter:7
           -laplacian:5 -backgroundSize:32 -CCminSize:2 -thrArea:10,2000
           -thrDensity:0.3,2000 -thrRange:0.03,4.0 -projDir:demo/

Reading accession database file: demo\xml\accession.txt
Computing window region and sizing parameters
---------------------------------------------
Window [14:475,74:505] (pixels) [rows,cols]=[512,512]
Spot Area sizing limits (10:2000) (pixels**2)
Integrated Density sizing limits (0.300:2000.000) (od)
Density difference sizing limits (0.0300:4.0000) (od)
Zonal Notch filter background window size: 32X32 (pixels)

Summary of segmented spot statistics
------------------------------------
Total of 2015 accepted D spots accumulated density=61294.0, area=11752
Total of 933 accepted D' spots accumulated density=13292.0, area=76350
Total of 1082 omitted D' spots accumulated density=119.4, area=23119
Omitted(D')/Accepted(D') spots failing D' resizing= 0.9%
Total # spots failed Area sizing[areaT1,areaT2]=[1540, 0]
Total # spots failed Density sizing[densityT1,densityT2]=[3150, 0]
Total # spots failed ODrange sizing[ODrangeT1,ODrangeT2]=[3109, 0]

List of image files and generated files
---------------------------------------
Input pix file          [demo\ppx\gel-HM-019.gif]
SSF file                [demo\xml\gel-HM-019.xml]
Central Core pix file   [demo\tmp\gel-HM-019-cc.gif]
Average pix file        [demo\tmp\gel-HM-019-avg.gif]
Mag 2nd Deriv. pix file [demo\tmp\gel-HM-019-mag.gif]
Background pix file     [demo\tmp\gel-HM-019-bkg.gif]
Segmented pix file      [demo\tmp\gel-HM-019-seg.gif]
'RestOf' pix file       [demo\tmp\gel-HM-019-rest.gif]

FINISHED! The Sample Spot-list File, SSF, is demo\xml\gel-HM-019.xml
Run time =0:0:2 (H:M:S) or 2.5 seconds

Breakdown of times and memory usage for steps in the segemntation
-----------------------------------------------------------------
Step []                    0.0 seconds  532.7Mb-tot 501.5Mb-free
Step [Reading gel]         0.2 seconds  532.7Mb-tot 528.9Mb-free
Step [Init]                0.0 seconds  532.7Mb-tot 526.7Mb-free
Step [Average]             0.1 seconds  532.7Mb-tot 526.7Mb-free
Step [Ctrl core]           0.7 seconds  532.7Mb-tot 500.4Mb-free
Step [Background]          0.0 seconds  532.7Mb-tot 500.4Mb-free
Step [Estimate D']         0.0 seconds  532.7Mb-tot 500.4Mb-free
Step [Rethreshold D']      0.0 seconds  532.7Mb-tot 500.4Mb-free
Step [Save images]         1.1 seconds  532.7Mb-tot 525.7Mb-free
Step [Save SSF]            0.3 seconds  532.7Mb-tot 504.5Mb-free
Step [Done]                0.0 seconds  532.7Mb-tot 504.5Mb-free

Seg2Dgel - finished at: 08/06/04 16:41:38
```

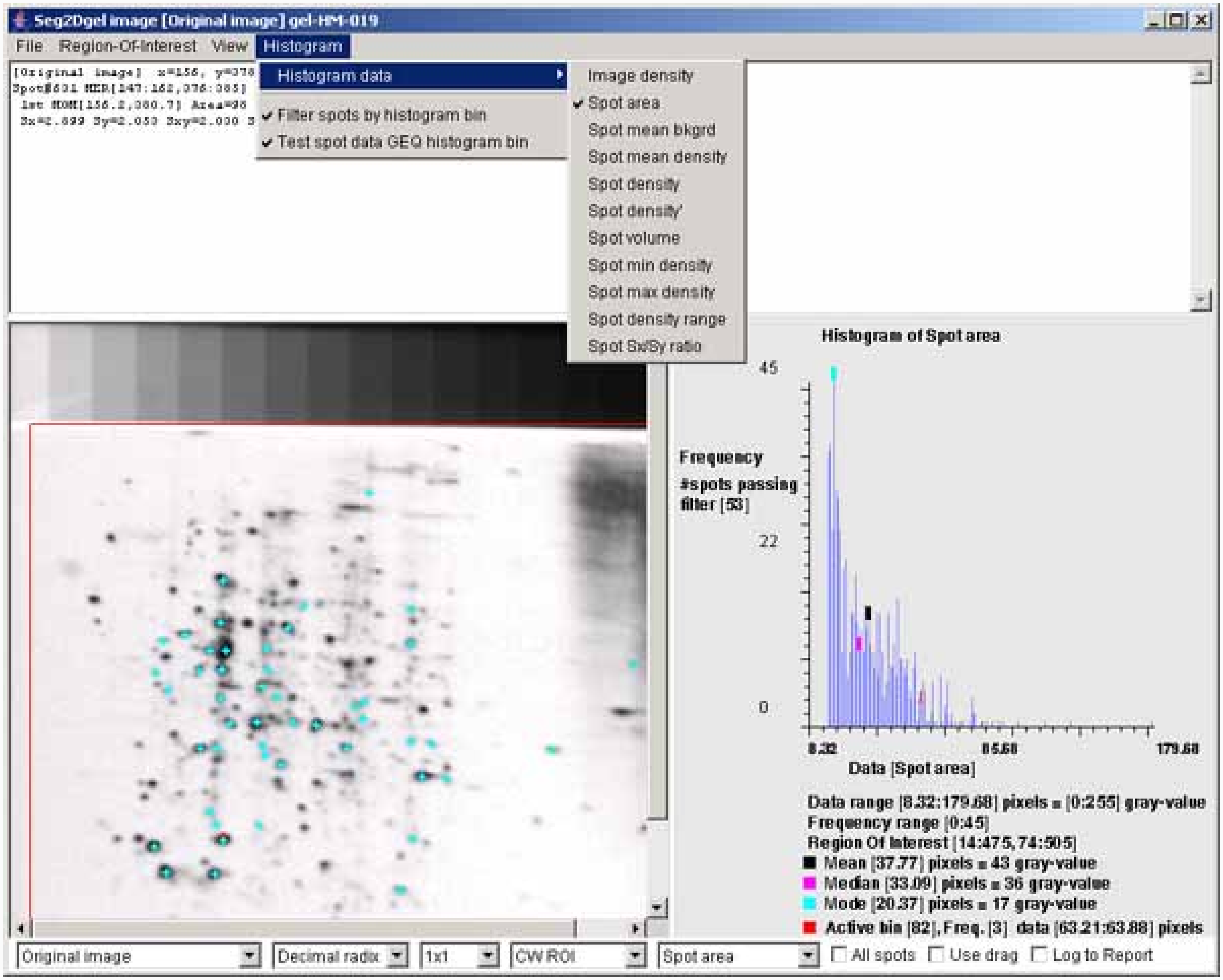
---

4. Investigating histograms of image density or spot features with ImageViewer

It is also possible to investigate distributions of spot density and other spot properties using histograms. It may also be used for data filtering subsets of spots by interacting with the histogram.

4.1 The histogram of spot areas within the computing window ROI of segmented image

The histogram may display various spot features. This shows the spot areas under the computing window (CW ROI). You may change the ROI after the segmentation using the (**Region-Of-Interest | Set ROI type | CW ROI**) menu command in the Image Viewer. The "Spot area" was selected using the (**Histogram | Histogram data | Spot area**) menu command. You may data filter spots that are have an area greater or equal (GEQ) than the histogram bin selected by clicking on it are shown in the image on the left as cyan colored "+" marks. The data filter is enabled by the **(Histogram | ☑Filter spots by histogram bin**) menu command. You set the The data filter test to GEQ (or LEQ) the bin by the (**Histogram | ☑Test spot data GEQ histogram bin**) menu command. Clicking on the histogram will select that bin and display the corresponding data for that bin. Note, if you enable the ☑**Use drag**) checkbox, you can change the bin by draging the mouse.



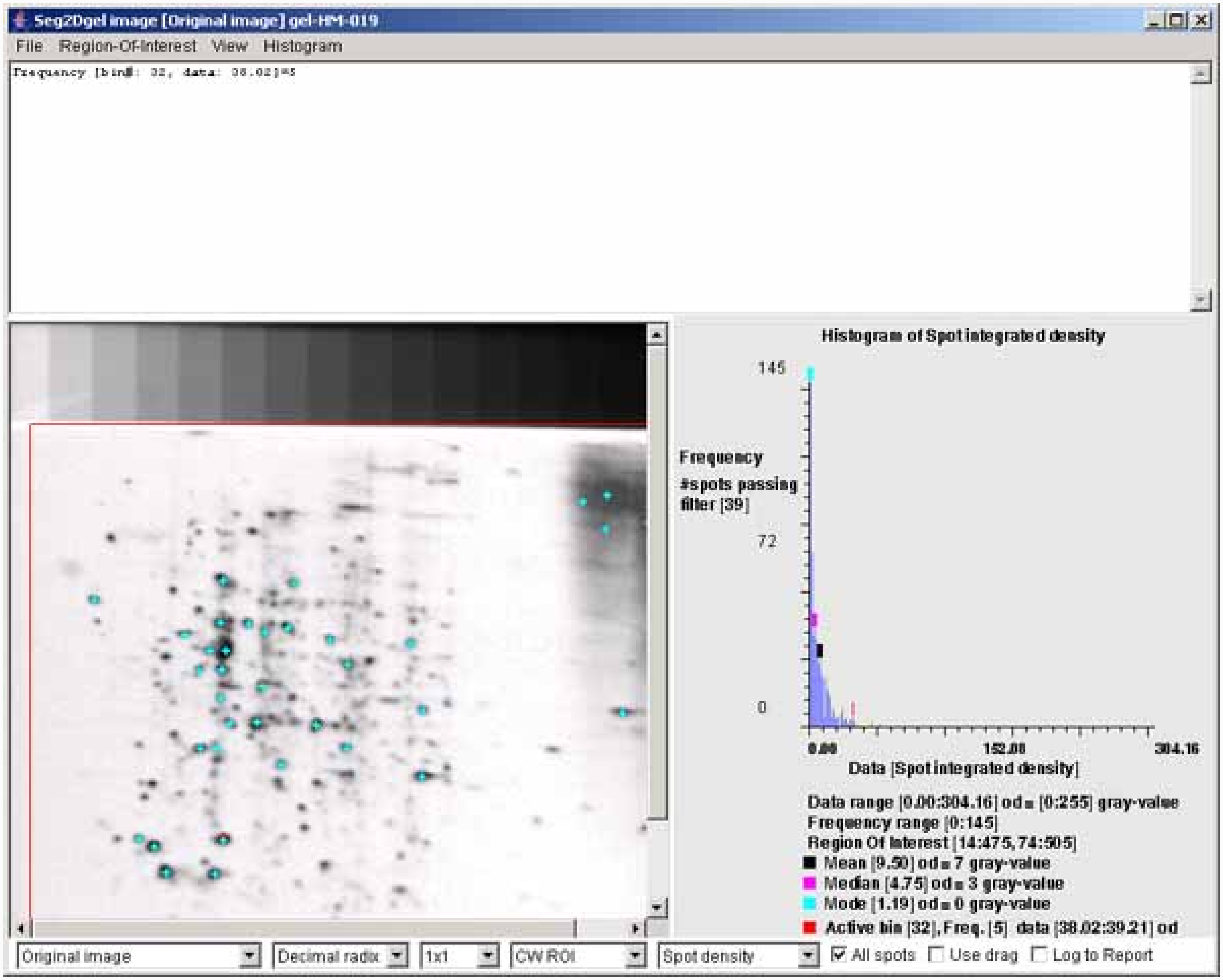
```
[Original image] x=156, y=378
Spot#631 MBR[147:162,376:385]
 1st MOM[156.2,380.7] Area=96
 Sx=2.899 Sy=2.053 Sxy=2.000 S
```

Data range [8.32:179.68] pixels = [0:255] gray-value
Frequency range [0:45]
Region Of Interest [14:475,74:505]
■ Mean [37.77] pixels = 43 gray-value
■ Median [33.09] pixels = 36 gray-value
■ Mode [20.37] pixels = 17 gray-value
■ Active bin [82], Freq. [3] data [63.21:63.88] pixels

4.2 The histogram of spot densities within the computing window ROI of segmented image

The histogram may display various spot features. This shows the spot integrated densities under the computing window (CW ROI). You may change the ROI after the segmentation using the (**Region-Of-Interest | Set ROI type | CW ROI**) menu command in the Image Viewer. The "Spot density" was selected using the (**Histogram | Histogram data | Spot density**) menu command. You may data filter spots that are have a total spot density greater or equal (GEQ) than the histogram bin selected by clicking on it are shown in the image on the left as cyan colored "+" marks. The data filter is enabled by the

(**Histogram | ☑Filter spots by histogram bin**) menu command. You set the The data filter test to GEQ (or LEQ) the bin by the (**Histogram | ☑Test spot data GEQ histogram bin**) menu command. Clicking on the histogram will select that bin and display the corresponding data for that bin. Note, if you enable the ☑**Use drag**) checkbox, you can change the bin by draging the mouse.



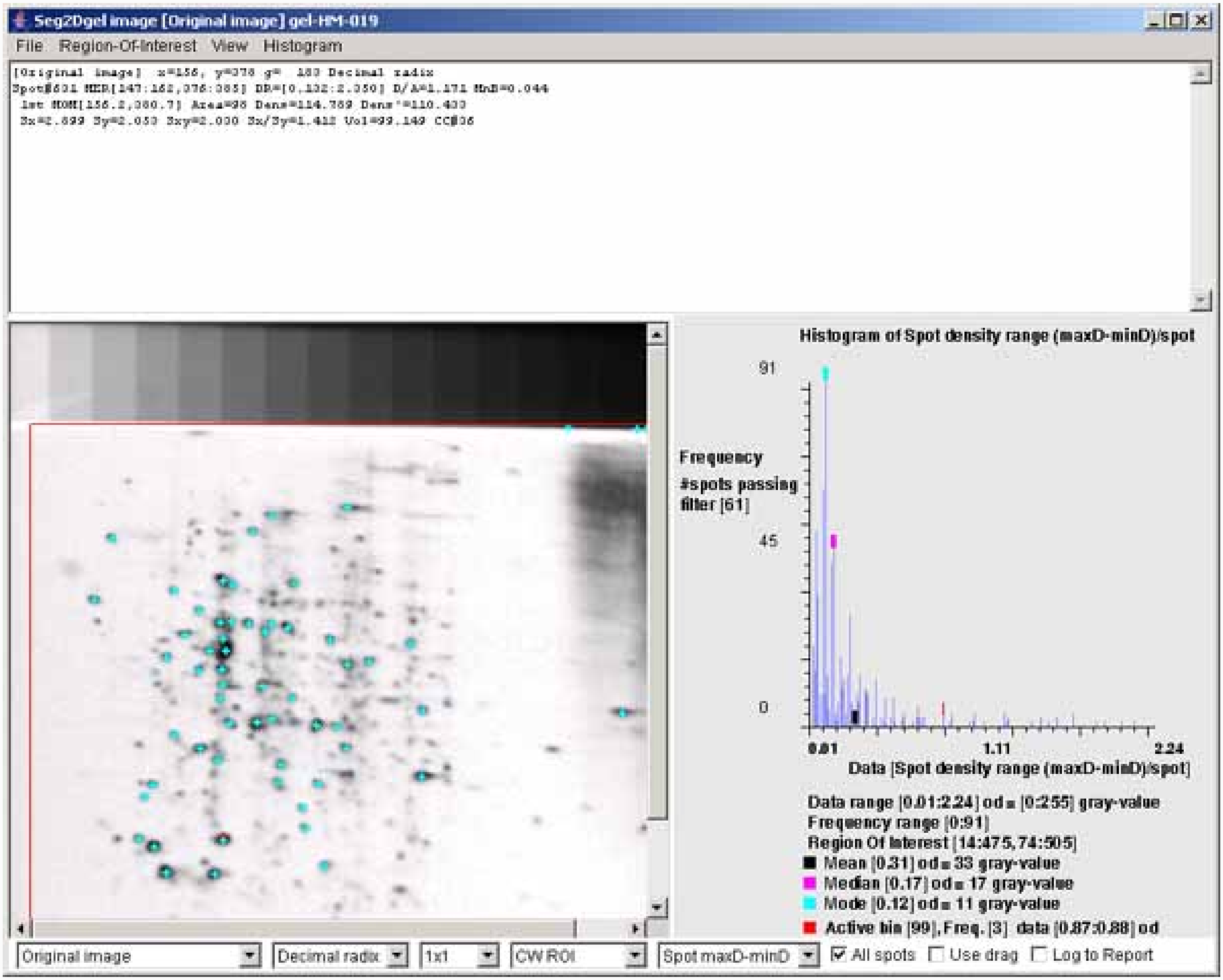
```
Frequency [bin#: 32, data: 38.02]=5
```

```
Data range [0.00:304.16] od = [0:255] gray-value
Frequency range [0:145]
Region Of Interest [14:475,74:505]
■ Mean [9.50] od = 7 gray-value
■ Median [4.75] od = 3 gray-value
■ Mode [1.19] od = 0 gray-value
■ Active bin [32], Freq. [5]  data [38.02:39.21] od
```

4.3 The histogram of spot density range within the computing window ROI of segmented image

The histogram may display various spot features. This shows the spots density ranges (i.e., max density/

pixel - min density/pixel) under the computing window (CW ROI). You may change the ROI after the segmentation using the (**Region-Of-Interest | Set ROI type | CW ROI**) menu command in the Image Viewer. The "Spot density range" was selected using the (**Histogram | Histogram data | Spot density range**) menu command. You may data filter spots that are have a total spot density range (maxD - minD) greater or equal (GEQ) than the histogram bin selected by clicking on it are shown in the image on the left as cyan colored "+" marks. The data filter is enabled by the (**Histogram | ☑Filter spots by histogram bin**) menu command. You set the The data filter test to GEQ (or LEQ) the bin by the (**Histogram | ☑Test spot data GEQ histogram bin**) menu command. Clicking on the histogram will select that bin and display the corresponding data for that bin. Note, if you enable the ☑**Use drag**) checkbox, you can change the bin by draging the mouse.



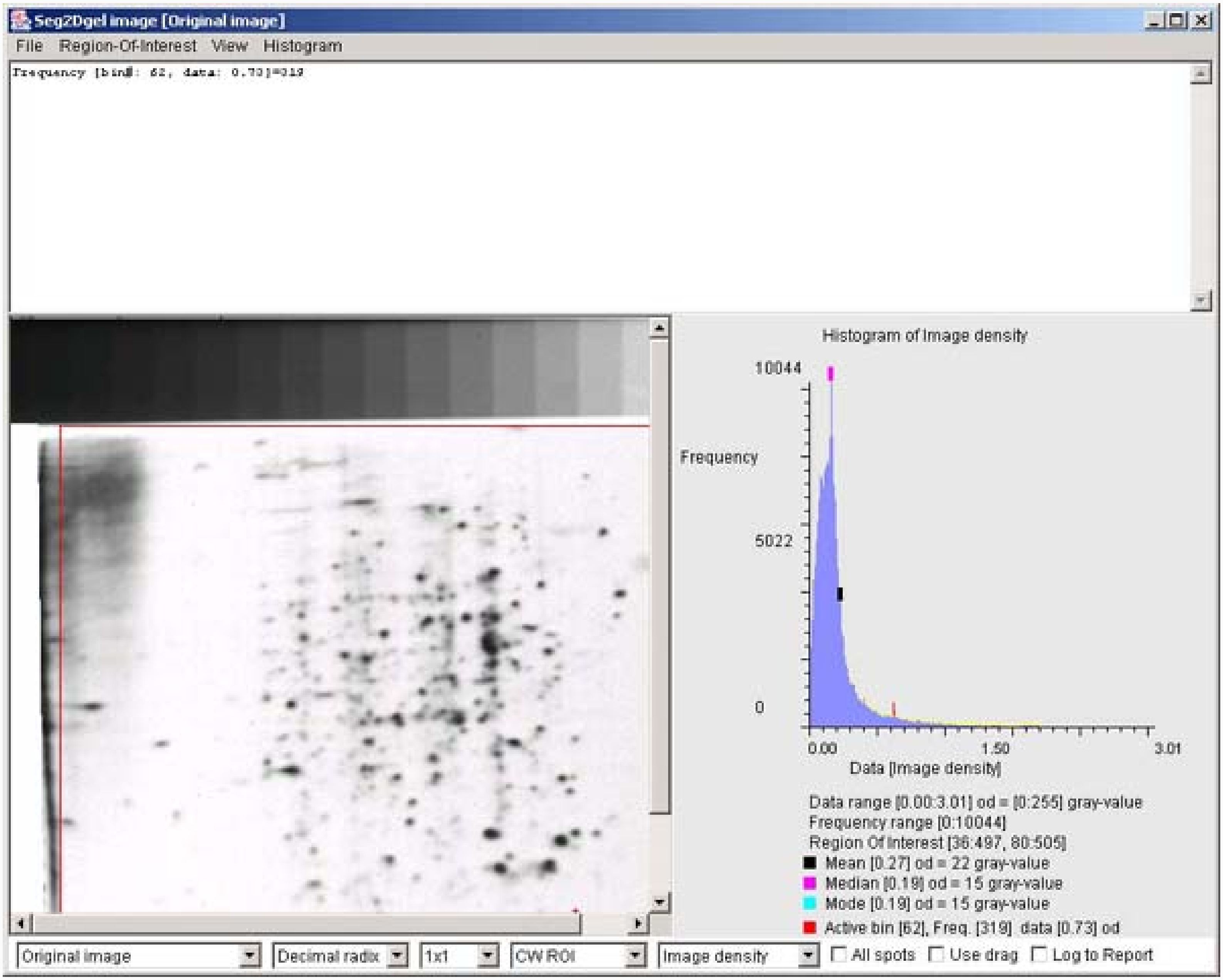
```
[Original image]  x=156, y=378 g=  183 Decimal radix
Spot#631 MBR[147:162,376:385] DR=[0.132:2.350] D/A=1.171 MnD=0.044
 1st MOM[156.2,380.7] Area=96 Dens=114.759 Dens'=110.433
 Sx=2.899 Sy=2.053 Sxy=2.030 Sx/Sy=1.412 Vol=99.149 CC#36
```

```
Data range [0.01:2.24] od = [0:255] gray-value
Frequency range [0:91]
Region Of Interest [14:475,74:505]
■ Mean [0.31] od = 33 gray-value
■ Median [0.17] od = 17 gray-value
■ Mode [0.12] od = 11 gray-value
■ Active bin [99], Freq. [3]  data [0.87:0.88] od
```

Original image | Decimal radix | 1x1 | CW ROI | Spot maxD-minD | ☑ All spots ☐ Use drag ☐ Log to Report

4.4 The histogram of image pixel density range within the computing window ROI of segmented image

The histogram may display various spot features. This shows the spots density ranges (i.e., max density/ pixel - min density/pixel) under the computing window (CW ROI). You may change the ROI after the segmentation using the (**Region-Of-Interest | Set ROI type | CW ROI**) menu command in the Image Viewer. The "Image density" was selected using the (**Histogram | Histogram data | Image density**) menu command. Clicking on the histogram will select that bin and display the corresponding data for that bin. Note, if you enable the ☑**Use drag**) checkbox, you can change the bin by draging the mouse.

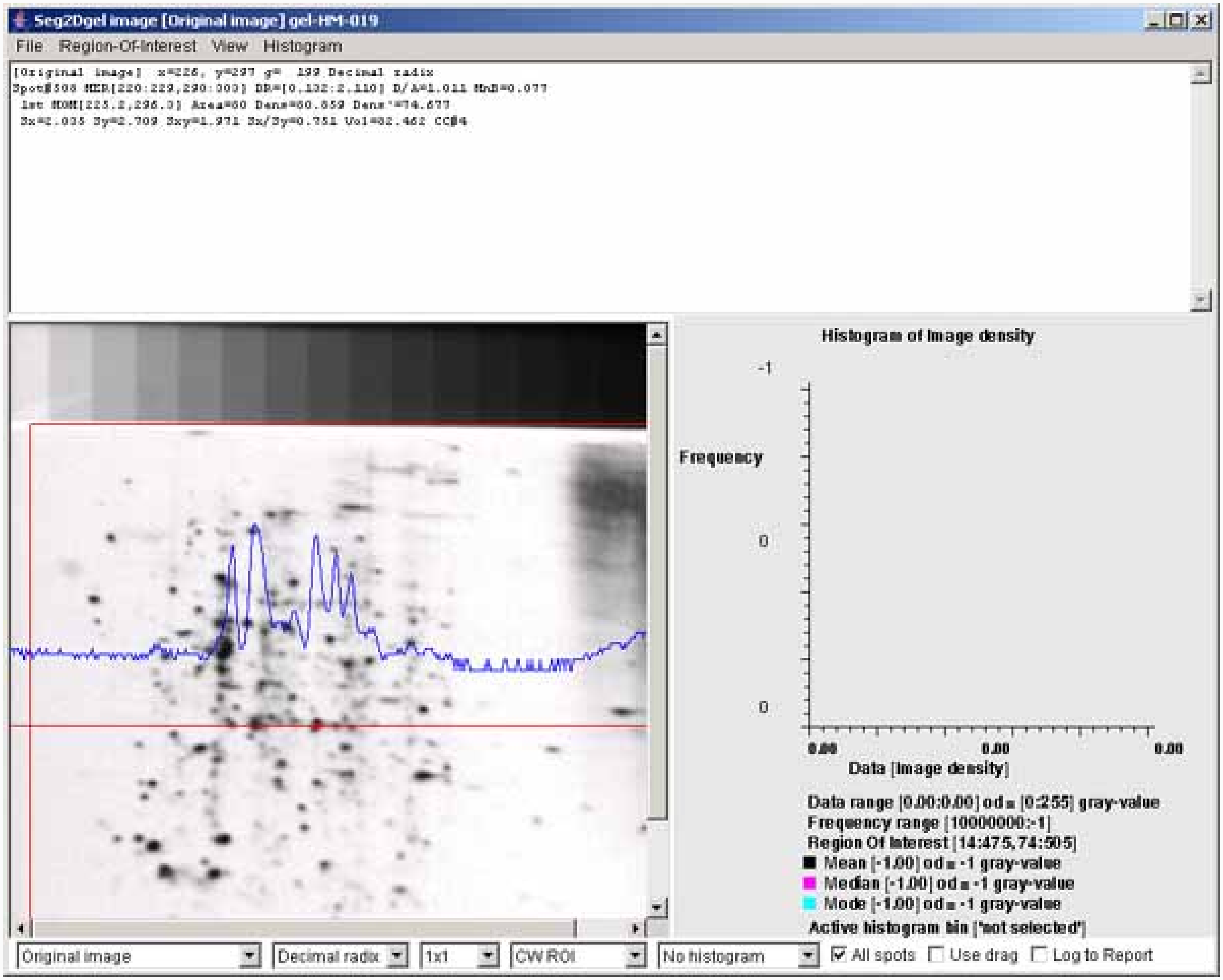


```
Frequency [bin#: 62, data: 0.73]=319
```

Data range [0.00:3.01] od = [0:255] gray-value
Frequency range [0:10044]
Region Of Interest [36:497, 80:505]
■ Mean [0.27] od = 22 gray-value
■ Median [0.19] od = 15 gray-value
■ Mode [0.19] od = 15 gray-value
■ Active bin [62], Freq. [319] data [0.73] od

4.5 Horizonal slice of image pixel densities

You can view a horizontal and/or a vertical pixel density slice through the image. This figure shows the horizontal slice. You enable the slice display by the (**View | ☑Horizontal slice**) menu checkbox command in the Image Viewer. You may simultaneously enable the slice display by the (**View | ☑Vertical slice**) menu checkbox command. Note, if you enable the ☑**Use drag**) checkbox, you can move the slice by draging the mouse.



```
[Original image]  x=226, y=297 g=  199 Decimal radix
Spot#508 MBR[220:229,290:303] DR=[0.132:2.110] D/A=1.011 MnB=0.077
 1st MOM[225.2,296.3] Area=60 Dens=60.659 Dens'=74.677
 Sx=2.035 Sy=2.709 Sxy=1.971 Sx/Sy=0.751 Vol=82.462 CC#4
```

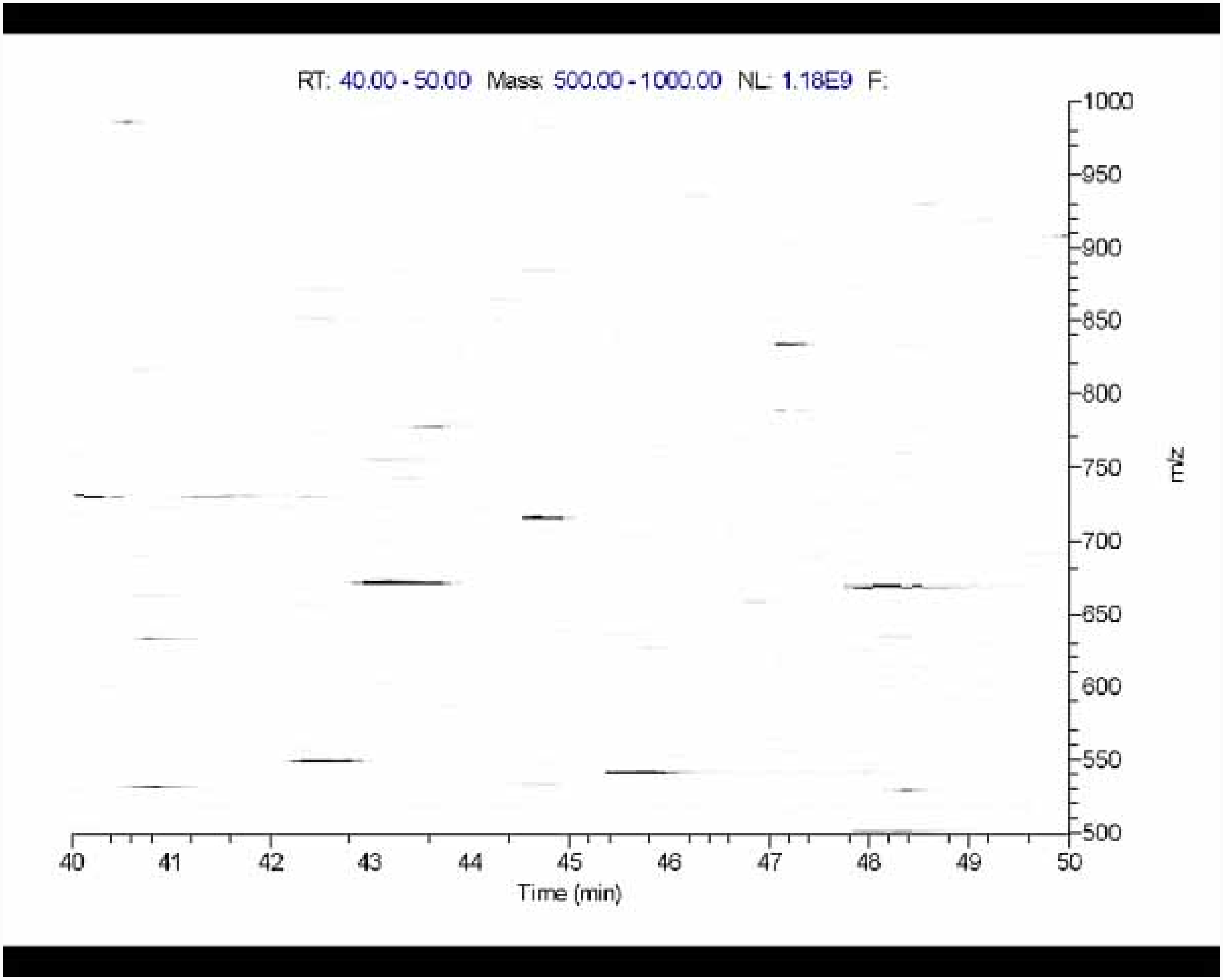
Data range [0.00:0.00] od = [0:255] gray-value
Frequency range [10000000:-1]
Region Of Interest [14:475,74:505]
■ Mean [-1.00] od = -1 gray-value
■ Median [-1.00] od = -1 gray-value
■ Mode [-1.00] od = -1 gray-value
Active histogram bin ['not selected']

5. Segmenting low resolution 2D-LC-MS data

The segmenter may be used for segmenting low resolution 2D LC-MS image data.

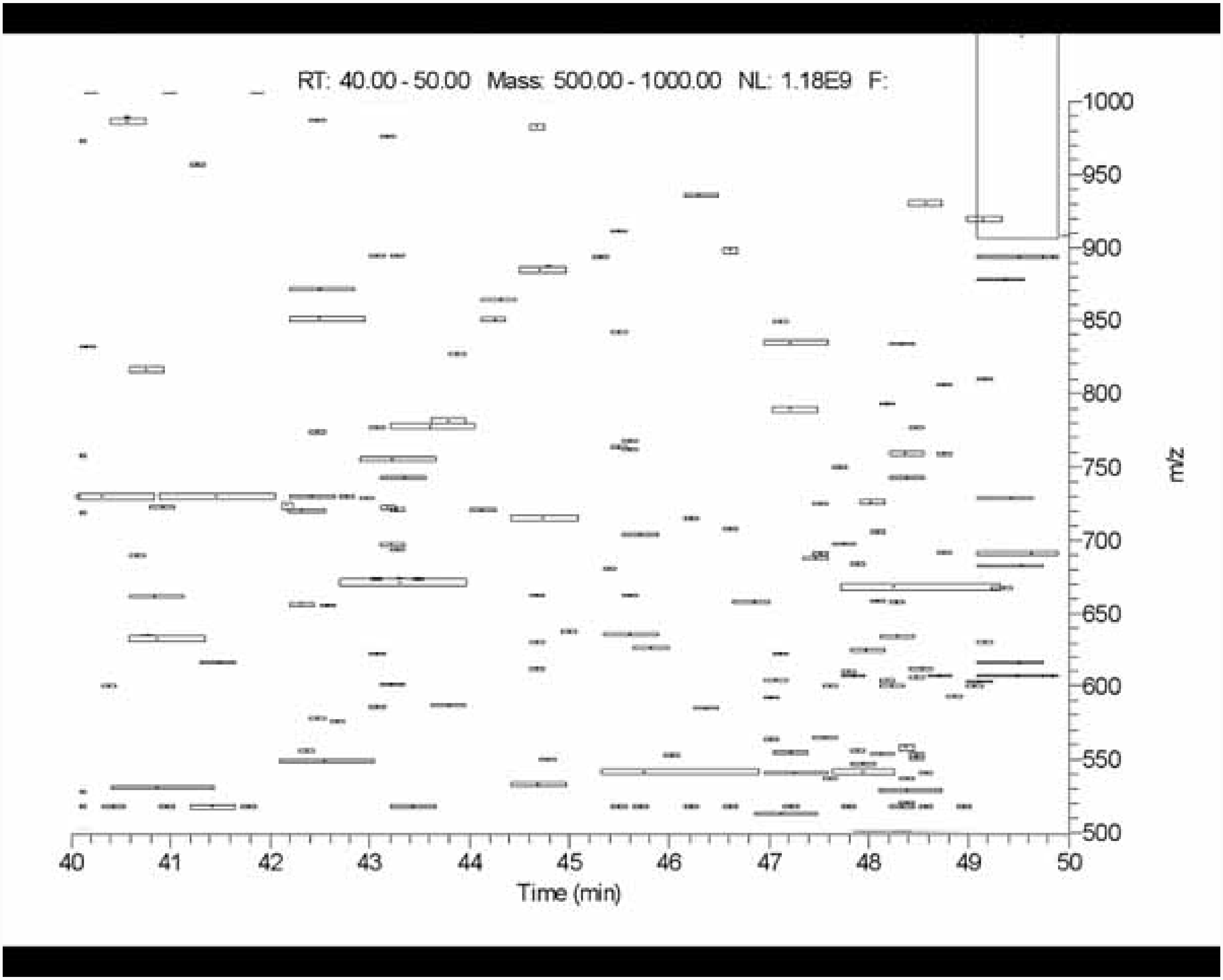
5.1 Original 2D LC-MS low resolution image image to be segmented

Original 2D LC-MS image. This shows the low resolution image data input into the spot segmenter (thanks to Laboratory of Proteomics and Analytical Technologies, NCI-Frederick).



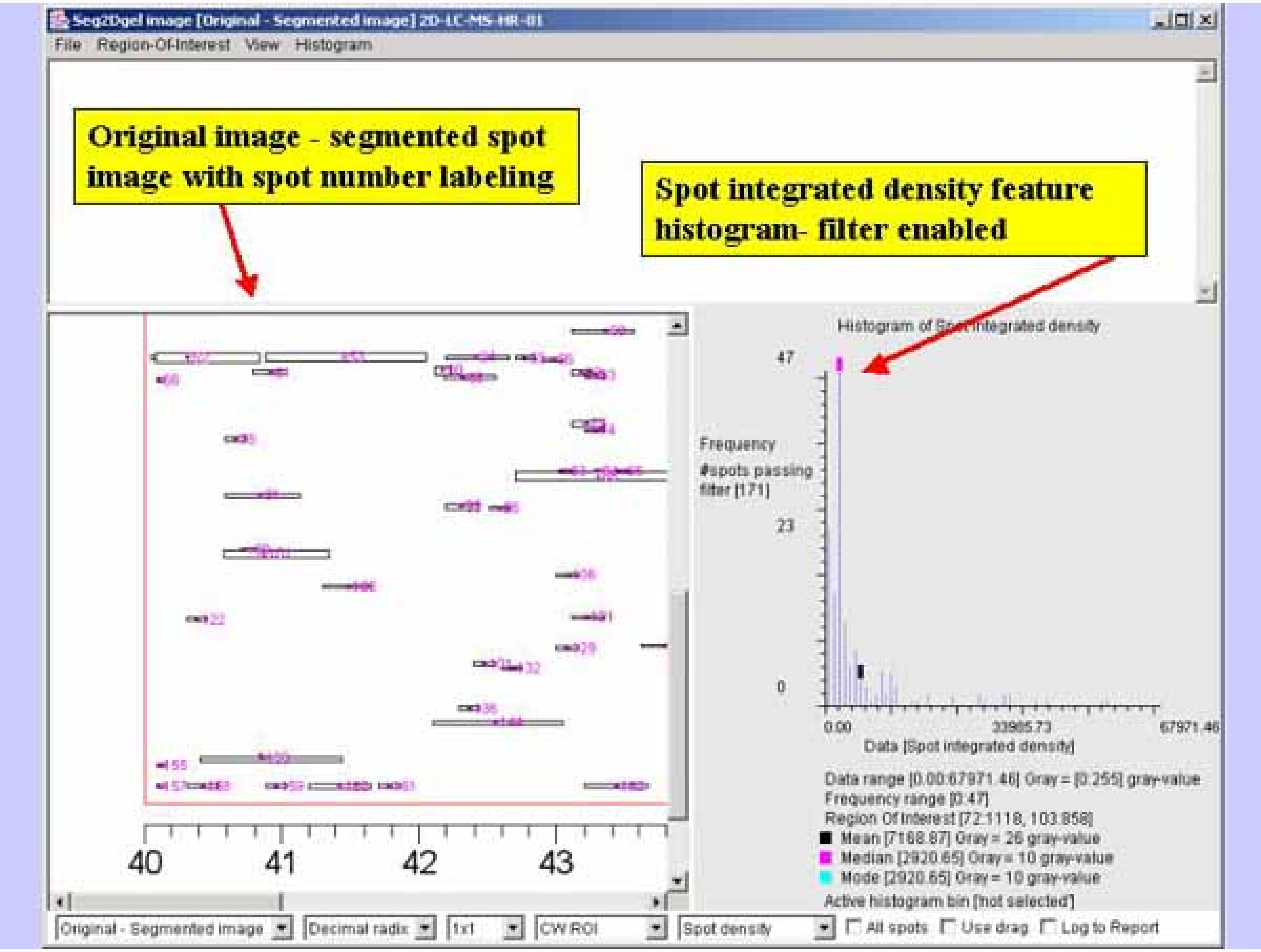
5.2 Segmented 2D LC-MS image resulting from spot segmentation

Segmented 2D LC-MS image resulting from spot segmentation. This shows the use of a horizontal filter on low resolution 2D-LC-MS image data (thanks to Laboratory of Proteomics and Analytical Technologies, NCI-Frederick). It uses a 3x9 pixel horizontal laplacian filter specified as **-laplace:H,3,9**. The ROI is about 800x600 in the center of the image (excluding the labeling), with **-drawMinEnclosingRect** enabled to draw boxes around each detected spot. The **-thrSxSxRatio:2.5,100.0** threshold sizing was was used to filter out small noisy line segments that uses a density weighted spot X and Y deviation ratio Sx/Sy >= 2.5. Thanks to Laboratory of Proteomics and Analytical Technologies, NCI-Frederick.



5.3 ImageViewer of segmented 2D LC-MS image with spot density histogram

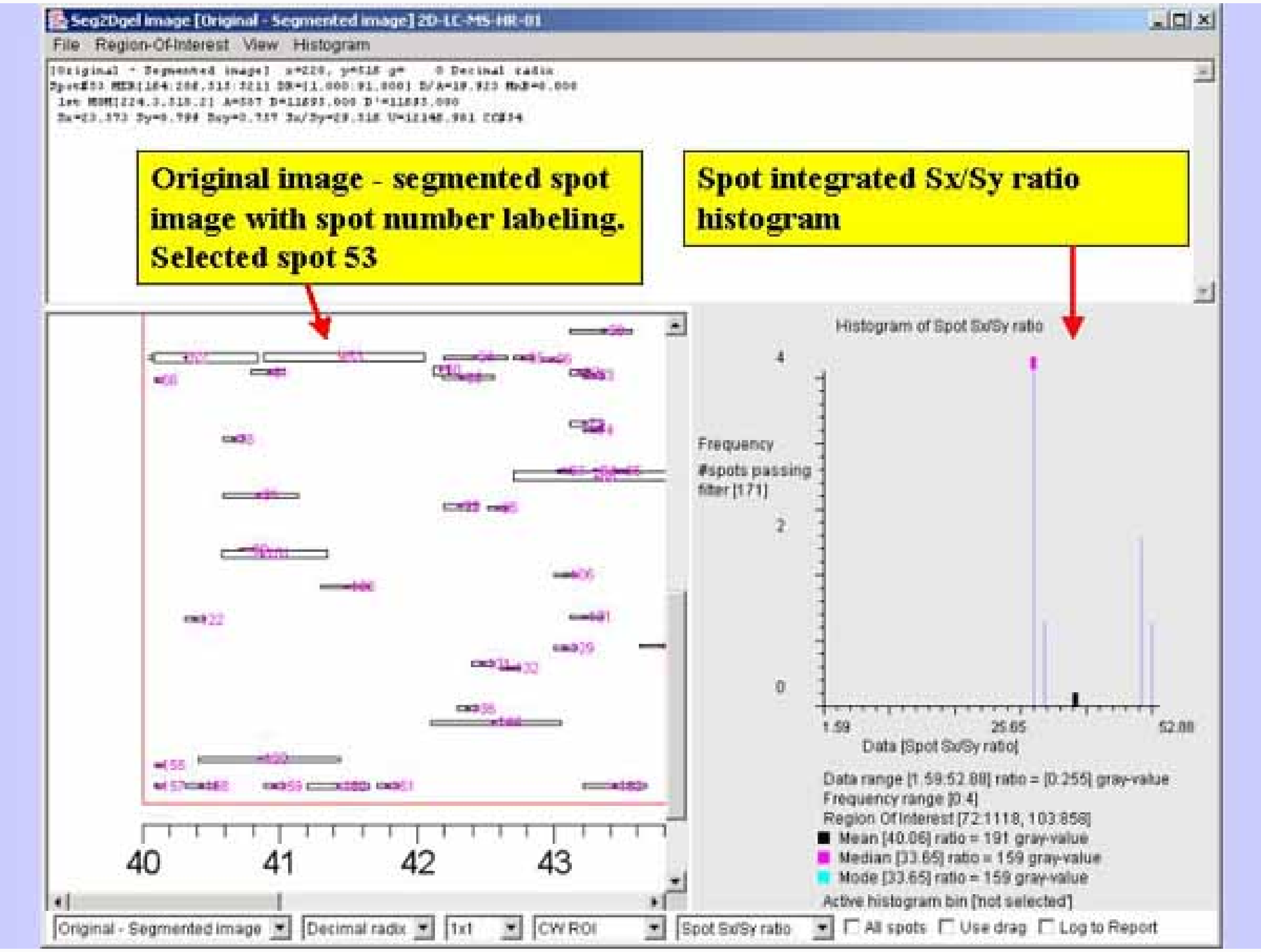
The segmented 2D LC-MS image resulting from spot segmentation using the Image viewer. The histogram of spot densities is shown on the right. Spot labels have been enabled (purple spot numbers). The data for the selected spot is shown at the top of the ImageViewer. This shows the use of a horizontal filter on low resolution 2D-LC-MS image data (thanks to Laboratory of Proteomics and Analytical Technologies, NCI-Frederick). It used a 3x9 pixel horizontal laplacian filter specified as **-laplace:H,3,9**. The ROI is about 800x600 in the center of the image (excluding the labeling), with **-drawMinEnclosingRect** enabled to draw boxes around each detected spot. The **-thrSxSxRatio:2.5,100.0** threshold sizing was was used to filter out small noisy line segments that uses a density weighted spot X and Y deviation ratio Sx/Sy >= 2.5. Thanks to Laboratory of Proteomics and Analytical Technologies, NCI-Frederick.



Data range [0.00:67971.46] Gray = [0:255] gray-value
Frequency range [0:47]
Region Of Interest [72:1118, 103:858]
■ Mean [7168.87] Gray = 26 gray-value
■ Median [2920.65] Gray = 10 gray-value
■ Mode [2920.65] Gray = 10 gray-value
Active histogram bin [not selected]

5.3 ImageViewer of segmented 2D LC-MS image with spot Sx/Sy histogram

The segmented 2D LC-MS image resulting from spot segmentation using the Image viewer. The histogram of spot densities is shown on the right. Spot labels have been enabled (purple spot numbers). The data for the selected spot is shown at the top of the ImageViewer. This shows the use of a horizontal filter on low resolution 2D-LC-MS image data (thanks to Laboratory of Proteomics and Analytical Technologies, NCI-Frederick). It used a 3x9 pixel horizontal laplacian filter specified as **-laplace:H,3,9**. The ROI is about 800x600 in the center of the image (excluding the labeling), with **-drawMinEnclosingRect** enabled to draw boxes around each detected spot. The **-thrSxSxRatio:2.5,100.0** threshold sizing was was used to filter out small noisy line segments that uses a density weighted spot X and Y deviation ratio Sx/Sy >= 2.5. Thanks to Laboratory of Proteomics and Analytical Technologies, NCI-Frederick.



Original image - segmented spot image with spot number labeling. Selected spot 53

Spot integrated Sx/Sy ratio histogram

Data range [1.59:52.88] ratio = [0:255] gray-value
Frequency range [0:4]
Region Of Interest [72:1118, 103:858]
■ Mean [40.06] ratio = 191 gray-value
■ Median [33.65] ratio = 159 gray-value
■ Mode [33.65] ratio = 159 gray-value
Active histogram bin [not selected]

Contact us Seg2Dgel is a contributed program available at open2dprot.sourceforge.net/Seg2Dgel Revised: 10/04/2005

Seg2Dgel - (PDF) documents

There are a number of Adobe Acrobat PDF formated documents and slide shows available for the Seg2Dgel project.

If you do not have Adobe Acrobat, you can download it for free from Adobe. The following lists some of the PDF documents you may download. As otheres are created, they will be added to this list.

1. Seg2Dgel Overview slide show for use as a printable document (1 slide/page)
2. Seg2Dgel Overview slide show for use as a printable document (2 slides/page)
3. Seg2Dgel Overview slide show for use as a printable document (6 slides/page)
4. This Seg2Dgel Web site as a PDF document for use as a printable document

---

Contact us Seg2Dgel is a contributed program available at open2dprot.sourceforge.net/Seg2Dgel Revised: 10/04/2005

Seg2Dgel Jar Version

```
The Seg2Dgel program is available on the Open2Dprot server
     http://open2dprot.sourceforge.net/Seg2Dgel

or from the mirror server at
     http://www.lecb.ncifcrf.gov/Open2Dprot/Seg2Dgel

After you have downloaded and installed the program, you can update
```

```
it quickly by having Seg2Dgel copy the Seg2Dgel.jar file from the Web
server using the (File | Update from Web server | Seg2Dgel program)
menu command. After you do the update, you need to restart the
program to use the new version.

Until futher notice, Seg2Dgel is to be considered pre-alpha-level
code until it is officially released. This means that there may be
some functionality not fully implemented, that works incorrectly,
or that has changed. New commands and functionality are in the
process of being added. Please report problems and suggestions to us.

1. Full download installation Seg2Dgel version 0.5.12
Revised: 12-21-2005

2. Update Seg2Dgel program (Seg2Dgel.jar) version 0.5.12
Revised: 12-21-2005
```

Seg2Dgel Revision History

This describes the revision history of the Seg2Dgel program of released or soon to be released versions. Generally, only the most recent versions are kept on the Web site (see Version). Documentation for unresolved bugs and requested functionality is found in the status Web page.

- **V.0.5.11 11-07-2005**: Changed installer paradigm to use launch4j. Also, new manifest for Seg2Dgel.jar file that now includes the Class-Path so the demo .bat and .sh scripts were simplified in this release. The (**Edit menu | Resize memory limits**) command was depricated - see Increasing or decreasing the allowable memory used by Seg2Dgel in the Reference manual.

- **V.0.5.10 05-11-2005**: Rebuilt after refactored O2Plibrary.

- **V.0.5.9 04-20-2005**: Fixed bugs in library and synchronized with O2Plibrary.

- **V.0.5.7 03-21-2005**: Fixed bug in -usage and -version switches. Changed SSF output files so it generates <sample-name>-SSL.xml rather than <sample-name>.xml.

- **V.0.5.4 01-19-2005:** Fixed bug that failed for some combinations of segmenting a new image if the prior image used the -accessionDB option.

- **V.0.5.3 01-18-2005:** Changed the Image Viewer so it now has a zoom function (1X, 2X, 3X, or 4X). Fixed bug histogram display. Fixed bug when switch input images. Changed menu. Now you must specify (in -gui mode) changes in the Edit Options popup rather than from the File menu.

- **V.0.4.7 12-08-2004:** Cleanup and using new O2Plib methods.
- **V.0.4.5 11-29-2004:** Updated and edited documentation in Reference Manual.
- **V.0.4.5 11-23-2004:** Added documentation in Reference Manual to show how to segmenter under batch processing with either Windows or Unix.
- **V.0.4.5 11-22-2004:** Fixed bug with histogram display. Cleanup and refactored code so uses new O2Plib methods. Added new switch **-dtd** (default is -nodtd) to add (omit) Open2Dprot-SSF.dtd when generate the SSF files.
- **V.0.4.4 11-09-2004:** Cleanup and refactored code into O2Plib. Fixed histogram problem when used with data filtering that was introduced when went to the O2Plib. Removed cal/ calibration directory. Added consistent O2Plib project subdirectory code to test and create (batch/ cache/ ppx/ rdbms/ tmp/ and xml/) subdirectories.
- **V.0.4.3 11-03-2004:** Fixed a bug where it would not write out images that were not square (introduced with new library). Fixed a few other minor bugs. Added ImageViewer (**View |Show spot numbers**) as an image overlay and (**Histogram | Sx/Sy ratio**) histogram filter.
- **V.0.4.1 11-01-2004:** Added new **-laplacian:{H|V},{height},{width}** line filter switch for locating flat Horizontal or Vertical spots more easily. Also changed some switch names for consistency as well as other cleanup. Added switches **-thrSxSyRatio:{s1,s2}** switch to do additional spot thresholding by the ratio of the density-weighted spot sizes Sx and Sy, Sx/Sy. This is useful for better separating flat horizontal or vertical spots. Added switches **-drawMinEnclRect:{P|O}** switch to draw minimum enclosing rectangle spot overlays.
- **V.0.3.3 10-21-2004:** Problem with zoom code. Was sometimes not accessing correct image in Image Viewer after segmentation. Stepped back and disabled zoom code (left it in the code) until the zoom functionality is finished.
- **V.0.3.1 10-13-2004:** Moved code from Seg2Dgel and other applications into the new Open2Dprot common library O2Plib.
- **V.0.2.5 08-25-2004:** enabled the command line processing using the Seg2Dgel.exe file for running it in batch. Run the command with "-nodemo" and other switches you need as well as the gel image or gel sample name (if using the -accession file option). A new switch -default was added to give you the -demo options but with -nogui and with no gel name.
- **V.0.2.3 08-10-2004:** Enabled the -accessionDB:*accession file* option. If a gel is not in the accession file, then (currently) use the -demo defaults and continue processing.

- **V.0.2.2 08-10-2004:** Changed accession database input file format so that the sample name is now "Sample" instead of "PubPixFileName".

- **V.0.1.3 08-1-2004:** Changed (Gel Segmentation File, GSF, .gsf) to (Sample Spot-list File, SSF, .ssf) to be compatible with the rest of Open2Dprot. **NOTE: Version V.0.1.X is NOT compatible with Version 0.0.XX. You must download a new version of the installer. The (File | Update from Web server | *) will not do the full update.**

- **V.0.0.25 07-28-2004:** Change some of the tab-delimited (-gsfFormat:F) GSF Epilogue section to more descriptive correct symbols.

- **V.0.0.24 07-27-2004:** Change some of the XML (-gsfFormat:X) GSF Epilogue section to more descriptive correct symbols.

- **V.0.0.22 07-06-2004:** Refactored code and fixed bug in -gsfFormat:F option.

- **V.0.0.21 07-06-2004:** Added support code for -accessionDB switch (working but disabled for now).

- **V.0.0.20 07-02-2004:** Added support code (disabled for now) for -accessionDB switch.

- **V.0.0.18 06-10-2004:** Fixed bug with -laplacian:B,P,gridSize - however the -laplacian:B,C, gridSize is not working. Fixed bug with Options wizard window so now the text fields and the pulldown choices are synchronized.

- **V.0.0.17 06-10-2004:** Optimized memory some more. Now draw interpolation lines in horizontal and vertical slices so it is easier to see. Added spot finder abort on ill-formed spots if the size of the central core is > the upper area sizing threshold. This helps avoid hanging up the segmenter on impossible spots that may be created if the gel image is saturated in regions. Saturation, results in the laplacian not being accurate with the result that spots might be merged together. This will result in some spots missed and bogus spots being formed from the remnants of these aborted spots. However, the real problem is that the spot images are saturated. Other algorithms could be added (such as the -splitSpot that is being debugged) to deal with this problem. Garbage in - garbage out.

- **V.0.0.16 06-09-2004:** Optimized memory some more. Fixed ImageViewer list of images when switched paradigm in Version V.0.0.14.

- **V.0.0.15 06-08-2004:** Cleanup.

- **V.0.0.14 06-07-2004:** Added ability to add/remove the histogram display in the Image Viewer (when in -gui mode). The "-histGUI" ("-nohistGUI) switch can be used to add (remove) it from the command line. The (**View | Add histogram to Image Viewer**) checkbox menu command or the popup options wizard window can also be used to change it.

- **V.0.0.12 06-04-2004:** Added pull-down menus to Image Viewer window. Added a histogram display in the Image Viewer. This may be useful for setting the threshold parameters. The user may select a feature from a list including image density under the region of interest (ROI) as well as spot features (area, mean background density/spot, mean spot density/spot, total spot density, total spot D', minD/spot, maxD/spot, spot density range (maxD - minD)). Also added horizontal or vertical image line slice histograms (i.e., image slices). You can do rudimentary spot data filtering (if enabled in the Histogram menu of the Image Viewer) where spots passing the selected histogram-bin filter are indicated in the image. If filtering is enabled, you click on a bin in the histogram and the filter will show spots (with a cyan "+") GEQ or LEQ the bin value you selected.

- **V.0.0.9 05-05-2004:** Fixed bug where Option Window would lockup if you used it more 2 times in a row. Added ability to modify the Region of Interest by redefining it for the (-cw, -pixdump, and -debug regions). Note: in the Image Viewer first select the ROI as either CW, -dbug ROI, or -pixdump ROI. Then click on the upper left-hand corner of the new region and type Control-U. Then click on the upper lower left-hand corner of the new region and type Control-L. This will then redefine the selected ROI and use it the next time you run the segmenter from the Report Window.

- **V.0.0.8 05-05-2004:** The -gsfFormat:mode options have been refined so that Full tab-delimited mode packages the tab-delimited spot list within searchable delimiters strings ###START-OF-SPOTLIST### and ###END-OF-SPOTLIST###. It includes the preface (gel accession and parameters data) and an epilog (statistics on spots segmented). The Options Window has been refactored so that it now has explicit choices for switches like -gsfFormat:{**F** | **G** | **T** | **X**}, and the -projDir:*directory-path* and -propertiesFile:*alternate-properties-file*.

- **V.0.0.7 05-02-2004:** It can now generate XML with an internal DTD. Added a (View menu | Show GSF report) to look at the generated GSF file in the Report Window. Added a (View menu | Show GSF browser) to look at the generated GSF file in a popup browser. Changed the -gsfFormat:mode options so that they are now {**F** | **G** | **T** | **X**}. They define the output format for the Gel Segmentation File (GSF) where mode is: 'F' for full tab-delimited (default .txt), 'G' for GELLAB-II (.gsf), 'T' for tab-delimited (.txt) spots-only, and 'X' for XML (.xml). (Default is -gsfFormat:X).

- **V.0.0.6 04-29-2004:** Moved the local documentation files to the Web site so that all documentation now comes from the Web site where it is easier to update. Various cleanups including adding -ccMinSize:*nbrPixels* and -saturateSpots:*percentThreshold* options threshold sliders to the Option Window.

- **V.0.0.5 04-27-2004:** This version will handle moderate resolution TIFF, JPEG, GIF images. The DBUG_TEST_IMAGE option is enabled in this release which means it will reset the switches to a known state and ignore the Seg2Dgel.properties file for now. You must explicitly change the options using the (Edit menu | Options | Edit Options) or press the **Edit options** button to change options for now. Not all options work. See the Status page for more details.

Contact us

Seg2Dgel is a contributed program available at open2dprot.sourceforge.net/Seg2Dgel

Revised: 11/07/2005

Seg2Dgel Program Status

The following are known bugs, suggested features, functions that are being debugged or being developed. Documentation for resolved bugs is found in the revision history Web page.

- Revision history
- FAQ
- Known bugs
- Suggested Features/FAQ

Known Bugs

The neumonic *GGGR* is shorthand for "Grep, Get Gestault and Refactor".

- **V.0.0.12 06-04-2004:** Note that a lot of memory is required for very large images.

- **V.0.0.12 06-04-2004:** Background image is sometimes not being computed correctly for images without a calibration.

- **V.0.0.8 05-05-2004:** There is no bounds checking (yet) when using the threshold sliders in pairs in the Options Window so that for a paired switch such as -thrArea:*a1*,*a2*, it is possible for the user to set a2 < a1 slider values. So be careful in using the sliders until bounds checking is added.

- **V.0.0.5 04-27-2004:** The DBUG_TEST_IMAGE option is currently enabled in this release which means it will reset the switches to a known state and ignore the Seg2Dgel.properties file for now when you start Seg2Dgel. You must explicitly change the options using the (Edit menu | Options | Edit Options) command to change options for now. Comand-line switches passed through the operating system are ignored while the DBUG_TEST_IMAGE is on.

- **V.0.0.5 04-27-2004:** The spot splitting method for -splitSpot is not finished being refactored and debugged. It is disabled until then.

- **V.0.0.5 04-27-2004:** The saturated spot method for -saturatedSpot is not finished being refactored and debugged. It is disabled until then.

- **V.0.0.5 04-27-2004:** The method for -drawBoundary is not finished being refactored and debugged. Need to GGGR use of the FBL and hashFBL for the boundary bnd objects and check its reuse (is it unique?). Until then, boundaries are disabled switches until debugged and validated including -drawBoundary, -saturatedSpotPropagate:%threshold, and -splitSpot:(B | R | T),minCC

- **V.0.0.5 04-27-2004:** Need to GGGR to optimize use of the FBL and and refactor its use in boundary analysis methods.

- **V.0.0.5 04-27-2004:** Need to GGGR the Sg2rlm. Why do we need readRLM_CWfile() to do RLM I/O? Is this strictly for debugging?

- **V.0.0.5 04-27-2004:** Need to GGGR Sg2bkg: in bkg.calcBackground(useBkgdAndIsolatedPixelFlag) use of the Sg2glb.bnd and multiple instances; also check if need to G.C. bnd.xxx in deleteBnd(). [GGGR] cvtFBLdataToBoundary() - what is interaction with global FBL and does it need separate FBL?

- **V.0.0.5 04-27-2004:** Need to GGGR interaction of mer[xy][12] and merp[xy][12] variables.

Suggested Features

- **V.0.0.9 05-07-2004:** [TODO] Allow resizing of text area view in the Image viewer window.

- **V.0.0.6 04-29-2004:** [TODO] When finished debugging-pre-alpha version, remove the default data setup DBUG_TEST_IMAGE so there are no constraints on using any input gel image file data.

- **V.0.0.5 04-27-2004:** For TIFF, JPEG or GIF color images, the RGB data is first mapped to grayscale using the NTSC color to grayscale transform (gray = 0.33*red + 0.50*green + 0.17*blue).

- **V.0.0.5 04-27-2004:** Need to generate JPEG (.jpg) output images in place of current GIF (.gif) files.

- **V.0.0.5 04-27-2004:** Need to GGGR sg2bkg, evaluate two ways of computing the background image. Currently only consider BKGRD_CODE pxiels. Also may want to also include ISOLATED_PIXEL_CODE pixels.

- **V.0.0.5 04-27-2004:** Recode CC codes after more robust, do conversion coding scheme to 32-bit CC codes. This gets around possible (low liklihood) of CC code clashes. There are two alternate methods being considered:

```
These PCC code values will change once we get the basic segmenter
working so we can take advantage of 16-bit image data.  Model 1 is the
current
```

8-bit model documented in the Reference Manual The new scheme might be: [Proposed model 2] PCC image coding scheme for 16-bit images --- BKGRD_CODE 0 - background pixel UNLABELED_CC_CODE 1 - unlabeled central core pixel MIN_CC_CODE 2 - minimum labeled central core pixel MAX_CC_CODE 29999 - maximum labeled central core pixel PCC_BASE_CODE 30000 - guard region splitting previously merged spots MIN_PCC_CODE 30002 - min. labeled prop. central core pixel MAX_PCC_CODE 59999 - max. labeled prop. central core pixel KILL_SPOT_CODE 65534 - deleted spot in central core if NEQ sizing (0 if -noDelete). ISOLATED_PIXEL_CODE 65535 - isolated pixel or non 4-ngh connected pixel [Proposed model 3] PCC Design using a 32-bit fbl.cc value --- This segmented PCC encoding model can handle more than 32K spots that Model 2 handles. If we will be using input data with more than 32K spots this may be a better model. Instead of coding the CC and PCC by an offset, we allocate specific bits for setting/testing ISOLATED_PIXEL_MASK 01000 for isolated pixels (i.e. spots) KILL_SPOT_MASK 0400 for killed spots PCC_MASK 0200 for PCC pixels Then we use a segmented cc counter consisting of ccSubId 0177 LSB This cycles in the range of either 2:99 or 1:127 (the former is easier to map to the CC image). This is added to the generation number ccGenNbr 22 most significant bits of 32-bit CC int word The ccGenNbr is incremented when the ccSubId cycles from 99 back to 2. 2**22 is 4,194,304 so that should be large enough for any gel! GGGR all code to make the changes. Given a (int)cc, the three fields are decoded as: ccSubId = (cc & 0177); ccGenNbr = (cc >> 10); ccFullNbr = (ccGenNbr << 7) | ccSubId; We can also continue using BKGRD_CODE 0 - background pixel UNLABELED_CC_CODE 1 - unlabeled central core pixel where we just set all other bits to 0. -- | ccGenNbr | isolated | killed | PCC | ccSubID | | | pixel | spot | | | -- 22-bits 1-bit 1-bit 1-bit 7-bits

---

Contact us

Seg2Dgel is a contributed program available at open2dprot.sourceforge.net/Seg2Dgel

Revised: 10/04/2005

Seg2Dgel Program Frequently asked questions (FAQ)

The following are some frequently asked questions. This is a list of some of the questions that we will be answering. Please send us your questions about Seg2Dgel so we can add them and the answers to this list.

- Revision history
- Known bugs

- Suggested Features/FAQ

FAQ

- How do I specify the gel to segment? You can specify the gel to segment using the (**File | Open gel file)** menu command or using the **Browse file** button in the Options window that you bring up using the **Edit options** button in the Report window.

- What is a project and a project directory? What goes into this directory and can I have more than one? The segmenter (and other Open2Dprot programs) work in the context of a project. A project is a set of data to be analyzed together - typically a set of gel experiments. The projectcontains several subdirecties (ppx/ - containing the original images, cal/ - containing optional calibration files for each original gel, tmp/ - containing any output images generated by the segmenter, and xml/ - containing the spot list gel segmentation file and other text files).

- What is an experiment? We define an experiment as a set of gels that may include different conditions

- What is a command line switch? The Open2Dprot programs, of which Seg2Dgel is one, are designed to be run either under a batch mode or a graphical user interface window mode. Command line switches are the common control mechanism. We have adopted the Unix convention that switches start with a "-" and may have additional parameters. For example, `-thrArea:10,2000` specifies that only spots with a minimum area of 10 pixels through 2000 pixels should be accepted. See the list of command line switches for details.
- How are switches used? See the command line examples.

- How do I change the command line switch options? Bring up the Options window using the **Edit options** button in the Report window.

- How do I adjust the spot sizing parameters (area, density, OD range)? Bring up the Options window using the **Edit options** button in the Report window. Adjust the threshold sliders to the parameters you desire. Then press the **Set new default** button. This returns control back to the Report window. Then press the **Segment** button to resegment the image. After it finishes, use the **Image viewer** to see how well it segmented the gel using the new parameters. You can repeat this process to titrate various parameters to arrive a a "good" set of parameters to use.

- How do I handle requiring different sizing parameters for different gels? If the gels are similar, then you may be able to titrate an examplar gel and use it on the rest of the gels. Otherwise, you might have to "tweek" the parameters for some of the outlier gels.

- Can I save the options that I have just changed and apply them to other gels? Yes. When you exit the segmenter, it saves the new parameters in a `{project directory}/Seg2Dgel.properties` file. [Status: disabled while in pre-alpha testing level.]

- What is an OD calibration and where do I get it? A sample image calibration defines the mapping of the image pixel grayvalues in terms of another quantity such as optical density, counts/minute, etc. The calibration is found in the accession database entry for the sample. The Open2Dprot sample accessioning program Accession will contain the sample calibration procedure.

- How do I see the image results of a segmentation I am done segmenting? Press the **Image viewer** button in the Report window or use the (**View | Image viewer** menu command.

- What is a Sample Segmentation File (SSF). It is a disk file that contains the numeric data for the list of spots just segmented. The format depends on the -ssfFormat:{**F** | **G** | **T** | **X**} command line switch.

- How do I see the list of spots numeric data results of a segmentation I am done segmenting? Use the (**View | Show SSF report** menu command to display the data in the Report window. Alternatively, you can use the (**View | Show SSF browser** menu command to display the data in another browser (depends on how you computer displays .txt and .xml files).

- How do I see use the Image Viewer to look at the numeric spot feature data for a particular spot? Bring up the Image viewer and click on a spot. The data appears at top of the Image viewer. Hint: set the subwindow size to 1x1 as the window data is distracting. If you set the ☑**Use drag** checkbox, then as you drag the mouse over the image, it will update the spot data. If you set the ☑**Log to Report** checkbox, then as you click on spots, it will copy that data to the Report window for a more permanent record.

- How do I see use the Image Viewer to look at a subwindow of numeric pixel data both in decimal and in optical density (OD if calibration)? Bring up the Image viewer. Then set the subwindow size using a pull-down chooser at the bottom of the window (1x1, 3x3, ..., 21x21) pixels. Then click on the spot you are interested in the image. To view the data as decimal, set the "Decimal radix" choice. If the image is calibrated in optical density, you can view the OD values by setting the radix to "OD radix". If you set the ☑**Use drag** checkbox, then as you drag the mouse over the image, it will update the window data. If you set the ☑**Log to Report** checkbox, then as you click on spots, it will copy that data to the Report window for a more permanent record.

- When segmenting a sample, how do I stop the segmenter when it seems to have gotten stuck or is taking too long? Press the **Stop segmentation button** in the Report window.

- How do I generate an XML output file? Set the command option to the `-ssfFormat:X` using the Options window that you bring up using the **Edit options** button in the Report window.

Contact us	Seg2Dgel is a contributed program available at open2dprot.sourceforge.net/Seg2Dgel	Revised: 10/04/2005