Segmentation by Weighted Aggregation [SWA] - User Guide

Written by Ayelet Akselrod- Ballin

Department of Applied Math and Computer Science

Weizmann Institute of Science

Rehovot, 76100 Israel

Updated October 2005

The SWA algorithm was developed at the Weizmann Institute of Science by Eitan Sharon, Meirav Galun, Achi Brandt and Ronen Basri. The software was written by Yoav Karnieli.

References

E. Sharon, A. Brandt and R. Basri, Fast multiscale image segmentation, CVPR, 2000.

E. Sharon, A. Brandt and R. Basri, Segmentation and boundary detection using multiscale intensity measurements, CVPR, 2001.

M. Galun and E. Sharon and R. Basri and A. Brandt, Texture segmentation by multiscale aggregation of filter responses and shape elements, ICCV, 2003.

1. Index

2. Overview

The SWA is an independent software tool for segmentation running on a windows XP platform. This software implements the image segmentation method of Sharon et al, and Galun et al (see references). The tool provides the capabilities of a fast multi-scale segmentation algorithm which offers state-of-the-art results on challenging images. The algorithm constructs a pyramid of graphs from fine (bottom) to coarse (top). During the construction of the pyramid various statistics of the aggregates are computed including shape, intensity variability, filter responses and more, these measurements influence the construction of higher levels in the pyramid. The software tool enables the user to modify the parameters considered throughout the segmentation according to a specific image's requirements. Additionally it allows efficient viewing and exporting of segmentation results. The exhibit tool presents the graph pyramid facilitated and reveals its rich, multiscale representation of the image. It can export matlab formats as well for post processing ad further analysis. Users familiar with other Windows programs will find that this software is fairly straightforward to use. This tutorial describes its basic features

3. Installation Instructions

SWA was compiled and most completely tested under Windows XP. Winzip must be installed on your computer before beginning.

i) System Requirements

System requirements	PC running Windows XP. MATLAB 6.5 is required only for post processing of the tool output.
Current version	22 – May – 2004
License	http://www.cs.weizmann.ac.il/~vision/SWA/index.html

ii) Installation Steps

This section describes how to install SWA on a computer with the Windows XP operating system.

1. Download SWA onto your computer.
2. Double-click on the zip files. Please make sure that there are no spaces in the path to where you have extracted the tool, for example, put it in "C:\SWA".
3. To run SWA, double click on **gui.exe** (for example click on C:\SWA\Gui\Release\gui.exe).

4. Quick Start

1. To run SWA, double click on the **gui.exe** (for example click on C:\SWA\Gui\Release\gui.exe). The **Basic Properties** window will appear:

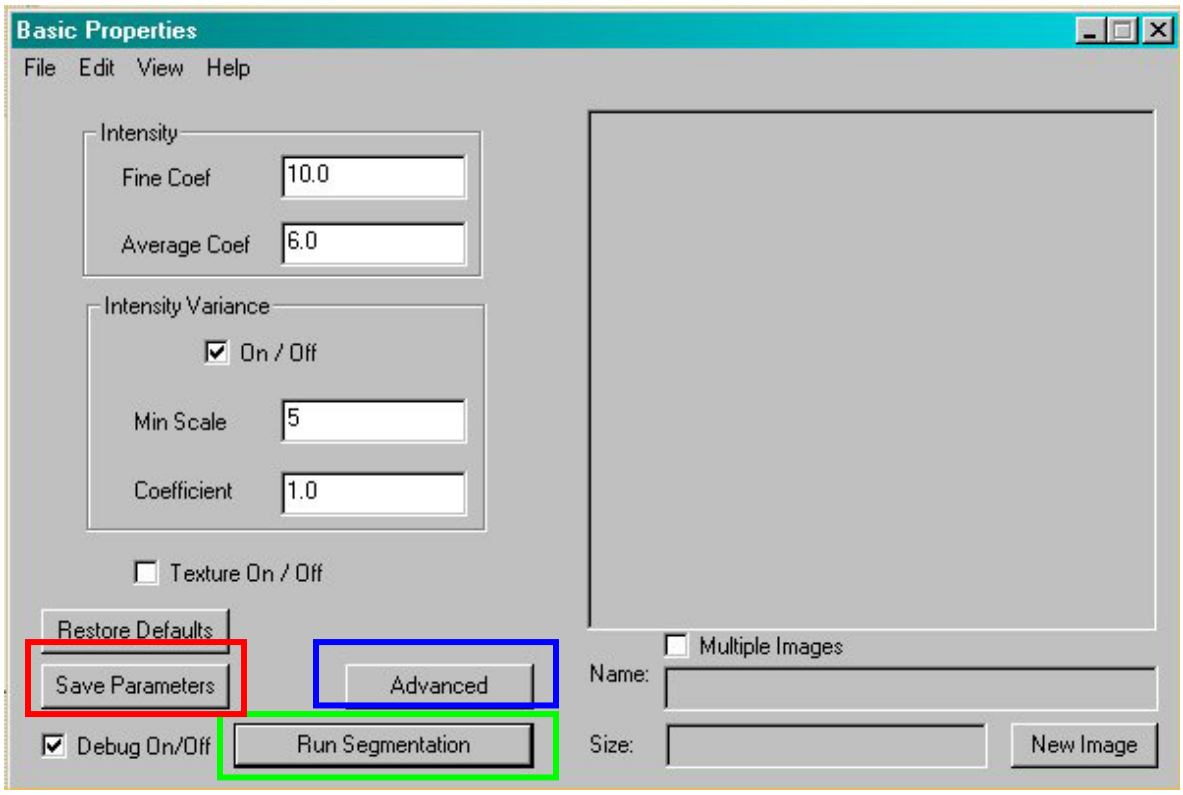

Figure 1: Basic Properties Gui

2. Press the **Advanced** button (see figure 1). In the window that appears select the **Presentation** tab. A dialog will appear that will enable you to set the paths for segmentation.exe, and results. After setting the path close the window. Make sure that you press defined make sure that you press **Save Parameters** to save the paths defined.

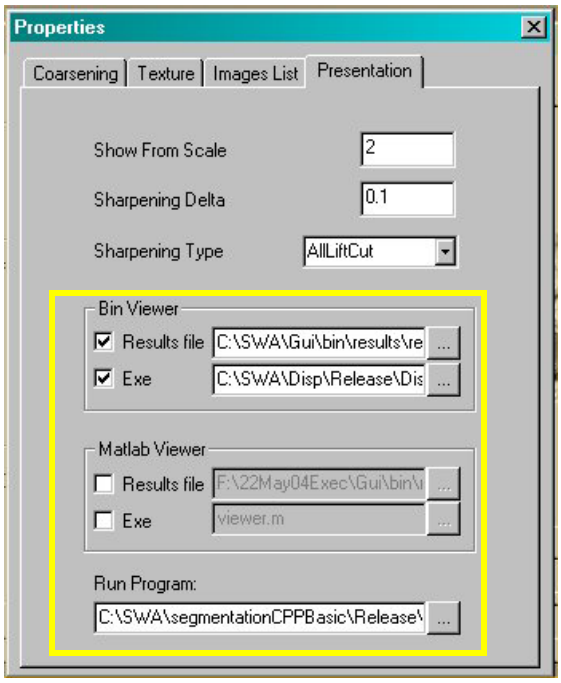

Figure 2: presentation tab

3. To load an image, go to the **basic properties** page, click the **new image button** and select the image for segmentation in the browse dialog opened.

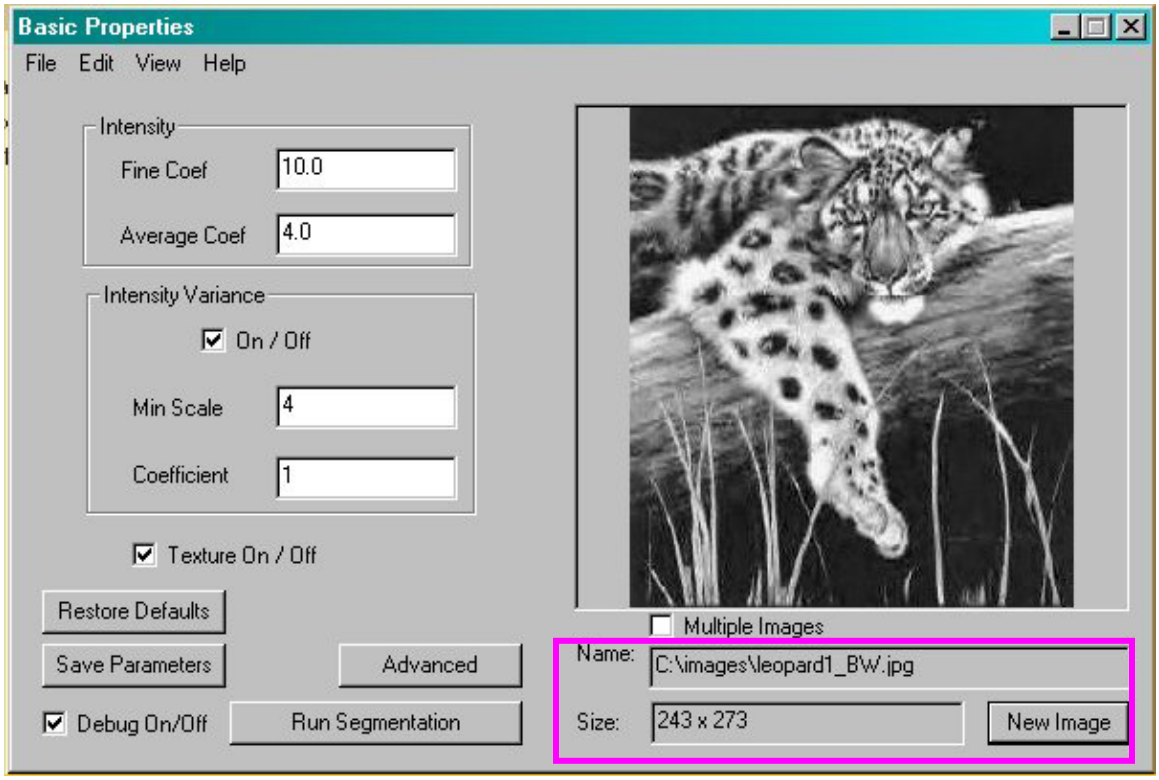

Figure 3: Load image

4. Press the **Run Segmentation** button (see figure 1).

5. View results in the **Display** window

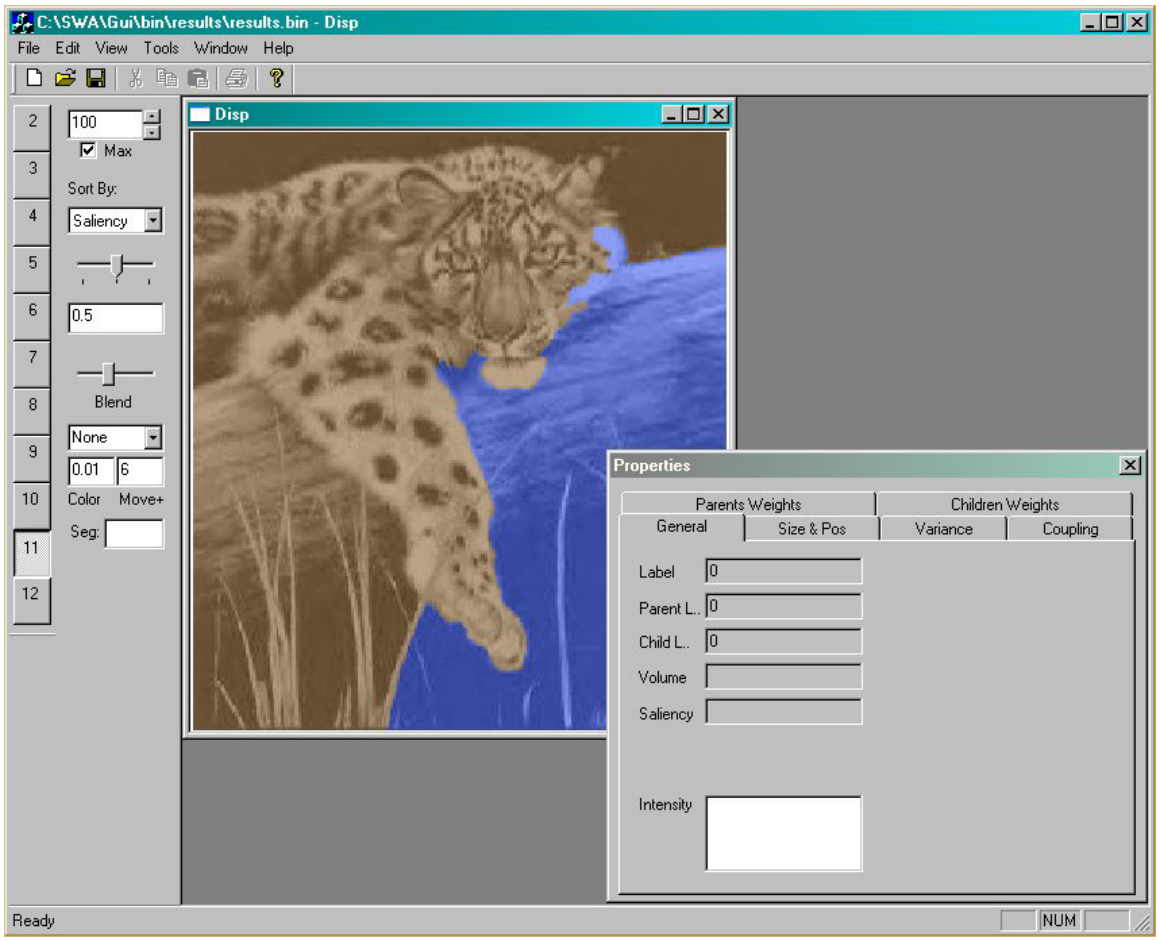

Figure 4: Display results

6. To save the Gui settings defined make sure that you Press **Save Parameters** (see figure 1), before closing the main Gui window.

5. Setting Segmentation Parameters

The following section describes the role of the different parameters. The Gui enables you to adjust them according to your image requirements. A detailed description may be found in the references listed. Note that for getting started the default parameters are definitely sufficient and no changes are necessary.

i) Main Panel

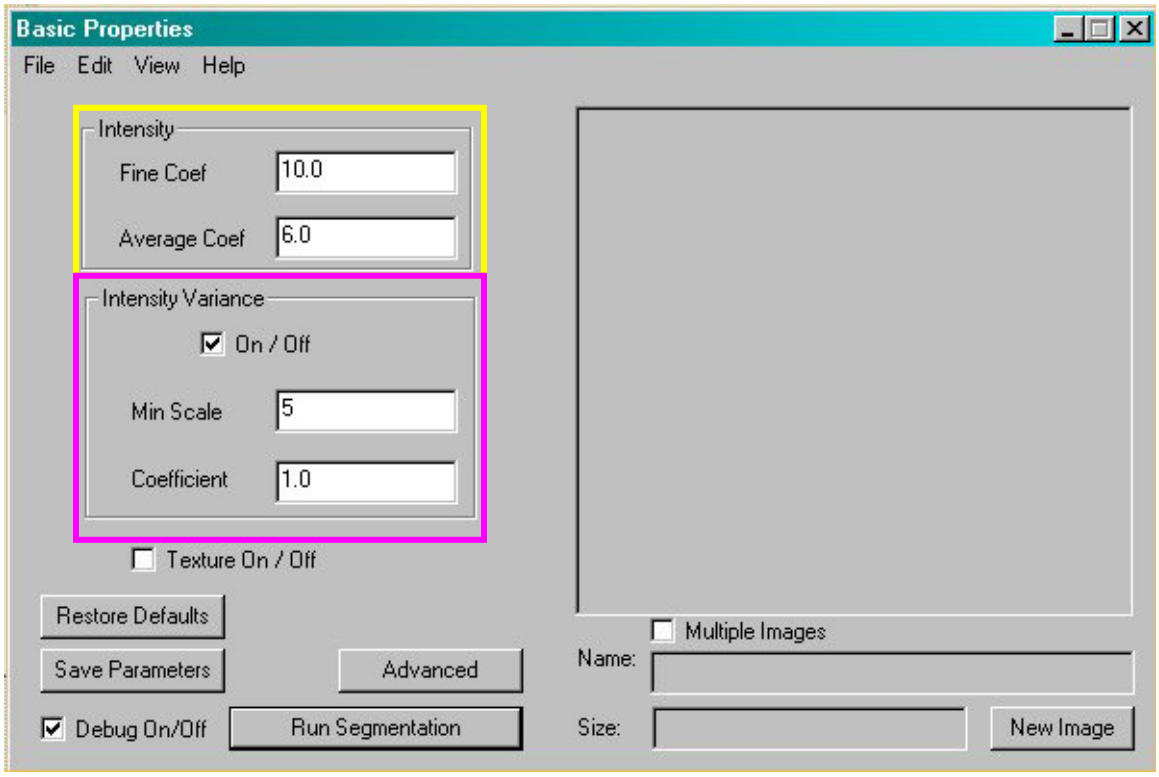

Figure 5 - Main panel

Intensity parameters:

- **Fine coefficient:** This is the edge coupling coefficient defined between two neighboring nodes. It is used in the finest level only when transforming the image to a graph. We initialize the graph with weights $W_{ij} = \exp(-\alpha|I_i-I_j|)$ between every two neighboring pixels i and j. This is the constant α for the exponential transformation.
- **Average coefficient:** This coefficient is used not for building the graph but for updating it. $\mu_k^{[s]}$ denotes the average intensity of the aggregate k at scale s. We modify the couplings to account for the intensity contrast between aggregates k and l by multiplying it by

exp(-γ| $\mu_k^{[s]} - \mu_l^{[s]}$|). This is the constant γ for exponential decreasing the coupling according to differences in average intensity levels of the two aggregates.

Intensity Variance:

- Click the check box to include intensity variance consideration in the segmentation
- Texture min scale determines the starting level for texture incorporation in similarity calculation.
- **Texture coefficient:** This is the texture coefficient β for the exponent similarity calculation. We multiply the coupling by exp(-$\beta_s D_{kl}^{[s]}$), which is the distance between the multi scale variance vector for aggregates k and l.

ii) Coarsening Tab

The following set of parameters is used during the coarsening process in the course of blocks selection for the coarser level.

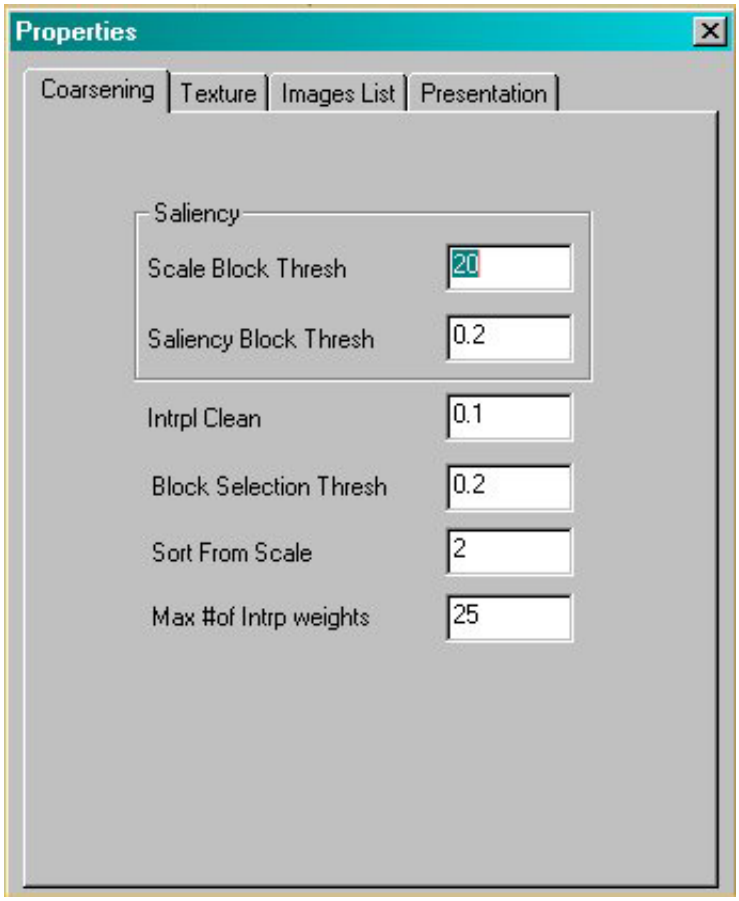

Figure 6 - Coarsening Tab

Saliency:

- **Scale Block Threshold**: Defines the starting level for assignment of segments that appear with saliency lower than the threshold as blocks in the coarse level.
- **Saliency Block Threshold**: The saliency of segment is defined as the energy of its cut normalized by the sum of its internal weights. Segments that yield saliency values lower than the threshold will be selected as blocks for the coarser level in the bottom up process.

Coarsening:

- **Interpolation cleaning**: Threshold for throwing away small weights from the graph, given their sum is not more than this proportion of its sum of weights. Allows removing weak and insignificant interpolation weights.

- **Block selection threshold:** The percentage for dependence of a fine- segment on the already chosen block- segments that will lead to not choosing it as a block- segment as well. If the already chosen block representation of a segment in the coarse level divided by the sum of its total couplings is larger than the threshold, the fine segment won't be in the coarse level, since it has sufficient representation.

- **Sort from scale:** The level to activate the sorting of blocks by volume prior to the selection of blocks for the coarser level.

- **Max number of interpolation weights:** Limits the number of coarse segments that can match a fine segment. To avoid over representation of the fine level. Works like a veto, not allowing the number of coarse blocks selected for a node to exceed this limit.

iii) Presentation Tab:

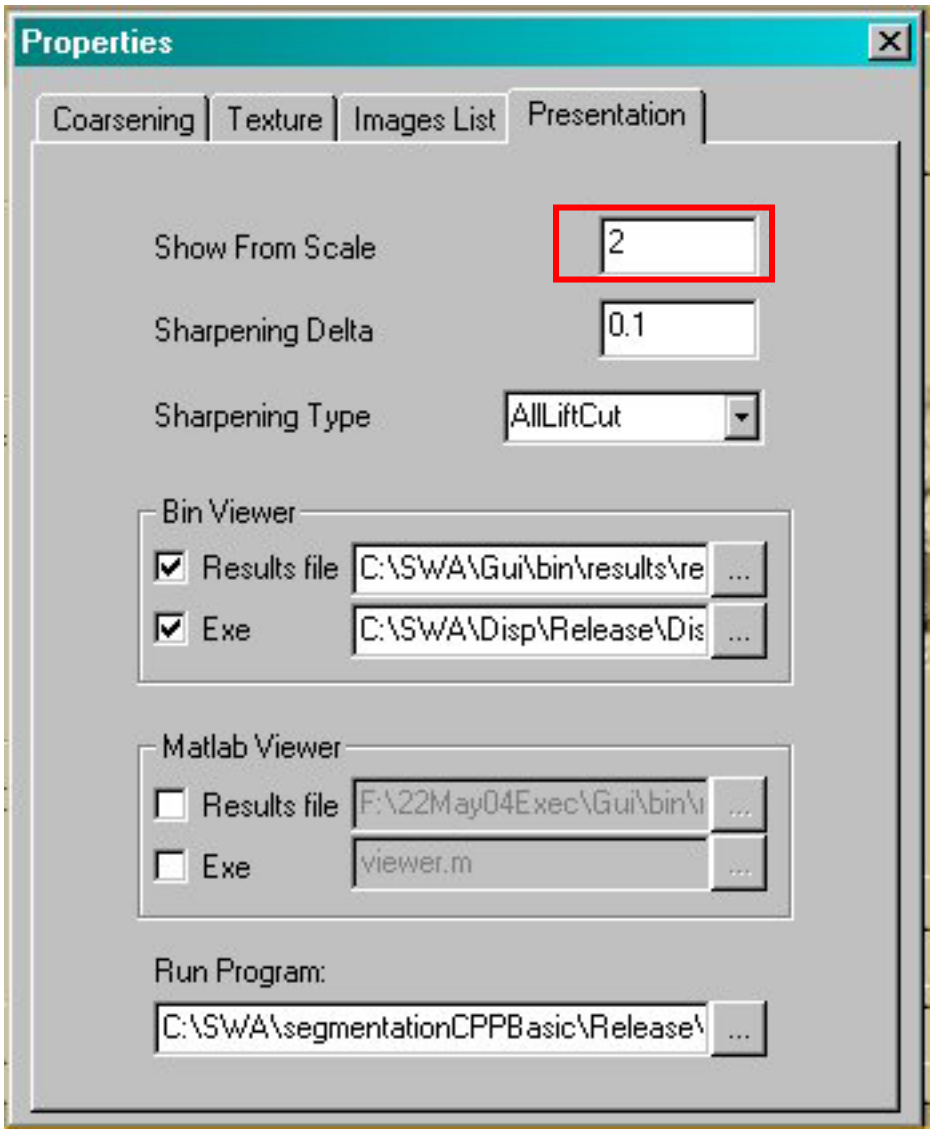

Figure 7 - Presentation tab

Show from Scale: This parameter is used for presentation. After running the segmentation the output result file will include scales from this level up. Namely this will be the finest segment presented in the display.

6. Viewing Results

The possible outputs of the SWA algorithm are binary output files and MATLAB 6.5 MAT files. Both the path to the result file and the path to result exe are determined in the presentation page (see quick tour section 2). After running the algorithm (see quick tour step 5) the results will be automatically displayed in the following window. Additionally you may view the display independently by running the disp.exe directly (its default location is .\Disp\Release\disp.exe). In this case you need to select the segmentation result file you are interested to display.

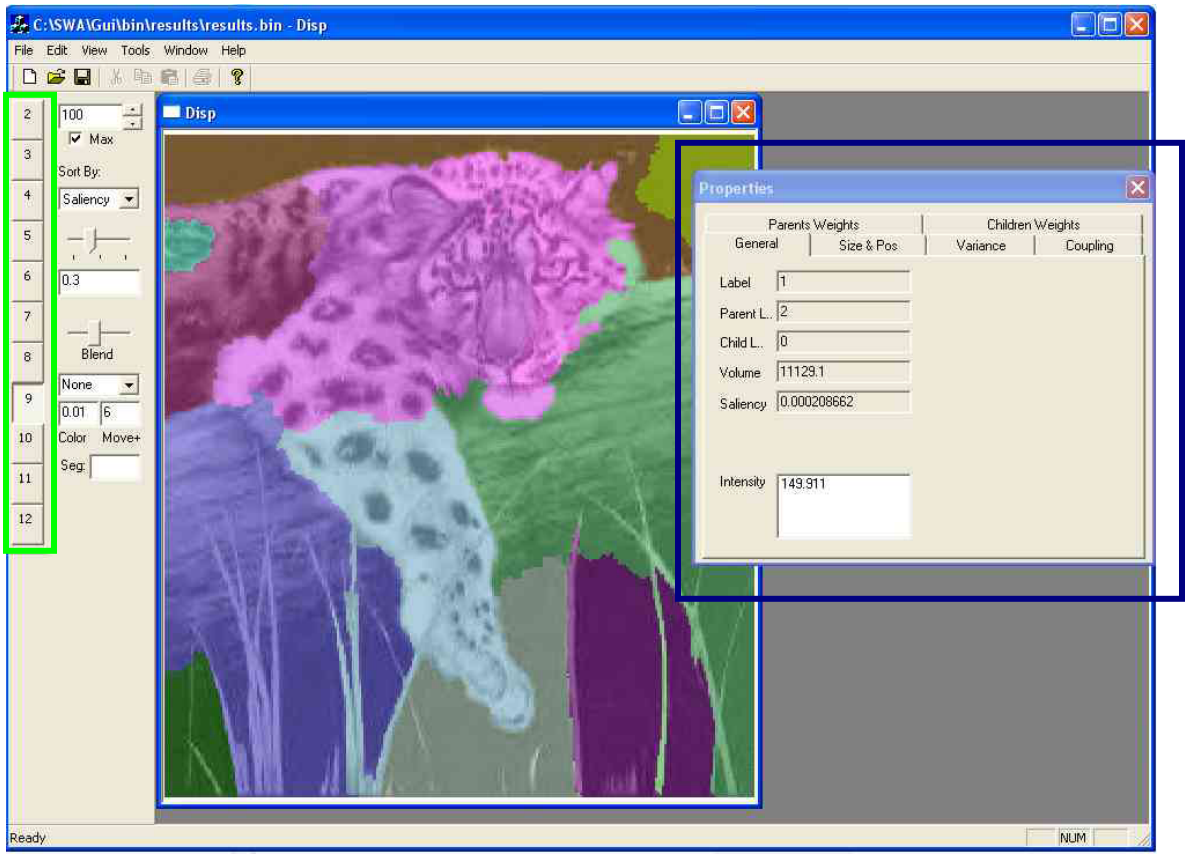

Figure 8 - Display Window

- The window includes a set of **scale buttons** sorted from coarse to fine. The scale button determines the graph level of the pyramid presented. For examples the figure presents the ninth level of the pyramid.
- The **segment number list box** represents the number of segments presented. If the **Max check box** is checked then all the segments for this scale will be presented.

- The **sort pop-down list** sorts the segments according to saliency or volume. Therefore the number of segments selected to be viewed will be the ones most salient or with largest volume.
- By adjusting the **blend horizontal slider** you can optimally represent the blending you want between the original image layer and the segmentation results layer.
- In the **segment edit box** you may select a segment number you wish to be highlighted.

Segment properties

- The properties window is opened automatically together with the display however if you closed the window, to present the properties a selected segment, click on Window and in the pull down menu click on properties.
- By clicking the mouse on a specific segment you will set its properties in the properties menu. For example in the upper figure segment number 1 was selected.
- The window includes six panels which provide a detailed description of the segments properties, including general information, size and position, variance, list of couplings, parent weights and children weights.

7. Loading Images

The input includes all standard image file types (e.g., TIF, JPEG, BMP etc.), however the tool does not support indexed images.

To load an image, go to the **basic properties** page, click the **new image button** and select the image for segmentation in the browse dialog opened.

It is also possible to batch process images by selecting multiple images in the basic properties window (see figure 1).

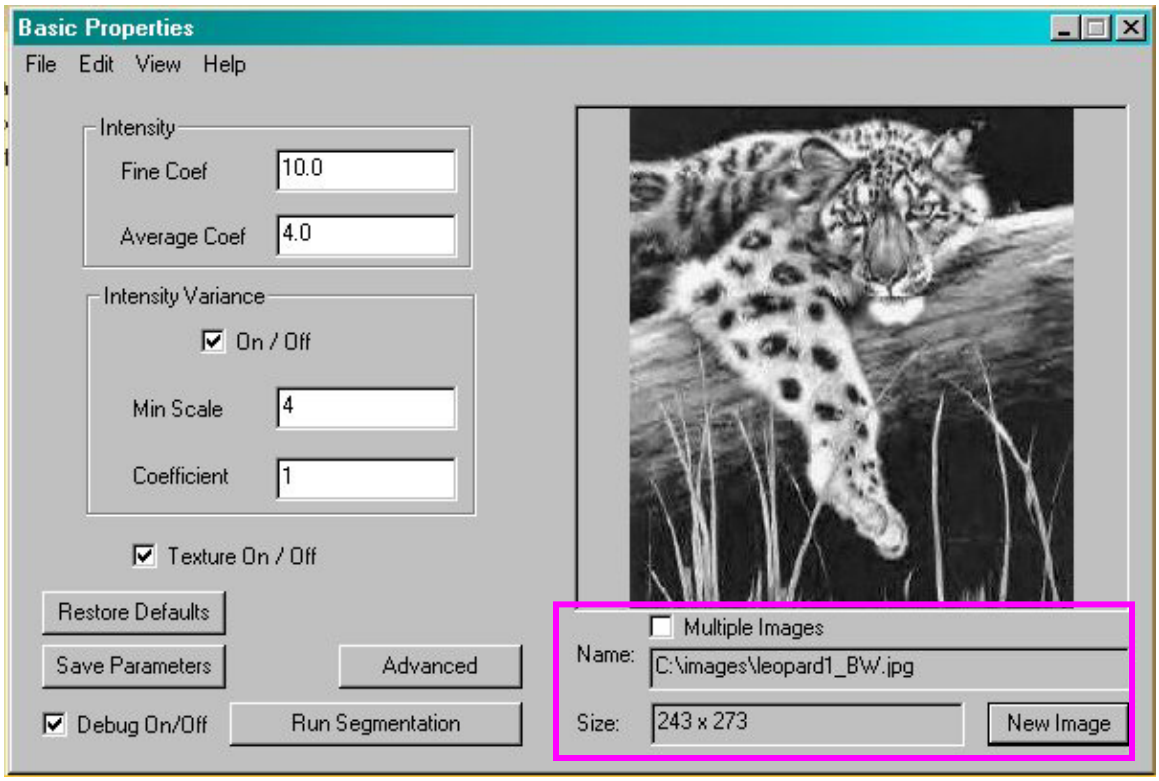

Figure 9 - Loading images

8. Batch Multiple Images

To execute the software on several images make sure the multiple check box is checked on the **Basic Properties** page and click the **Advanced** button. Then select the **Image list** tab. Here you can generate the list of images by pressing the **Add** button and selecting the images to segment from their directory.

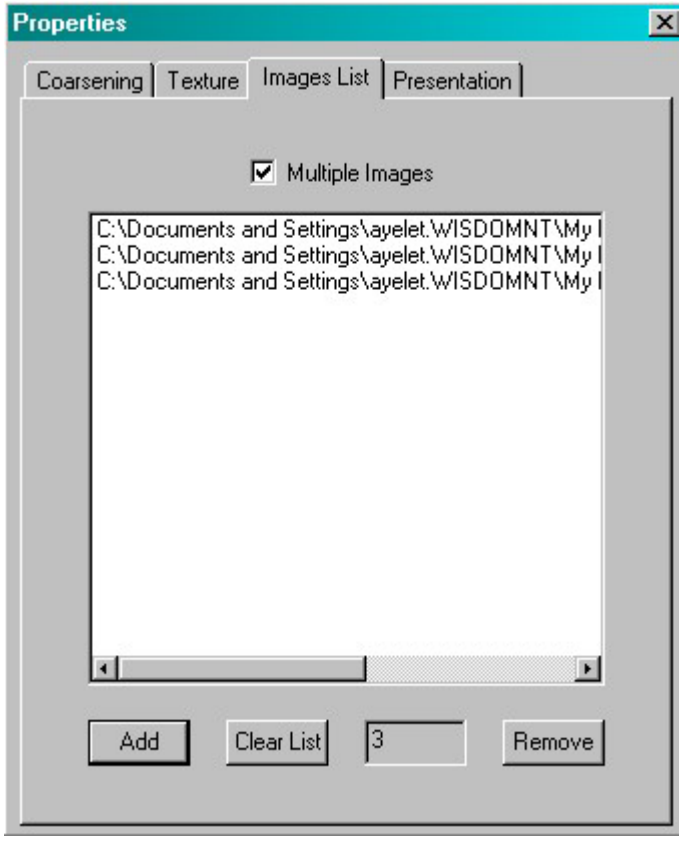

Figure 10 - Image list

The images selected will be added to the list. It is also possible to **remove** Images from the list or to **clear** the entire list by pressing the corresponding button.

9. Command Line Execution

The segmentation application can be executed using the following command line format:

segmentation.exe path segParameters.txt

Where the path indicates the location of segParameters.txt , and the parameter file (segParameters.txt) contains all the relevant parameters including the path of the input image.

Example: segmentation.exe F:\segmentation\Gui\Release\ segParameters.txt

Table 1: List of parameter index in segParameters.txt

Parameter category	Parameter name	Letter index in file	Default value
Intensity			
	Fine coef	-c	10.0
	Average coef	-a	4.0
Intensity variance			
	Min scale	-7	4
	Coefficient	-6	1
Image name		-i	E:\ \images\leopard1.jpg
Coarsening			
	Saliency-scale block threshold		20
	Saliency block threshold	-b	0.2
	Intrpl clean	-p	0.1
	Block selection thresh	-b	0.2
	Sort from scale	-m	2
	Max # of intrp weights	-u	25
presentation	Show from scale	-f	2
	Bin viewer – result file	-Xa -Xb	1 F:\22May04\Gui\bin\results\results.bin
	Bin viewer – exe	-Xc -Xd	1 F:\22May04\Disp\Release\Disp.exe
	Matlab viewer – result file	-Xg -Xh	0 F:\22May04\Gui\bin\results\segments.mat
	Matlab viewer – exe	-Xi -Xj	0 viewer.m

10. Save Parameters

To save / load specific segmentation parameters, clock on the File menu and in the pull down menu select export / import correspondingly.

11. References

M. Galun and E. Sharon and R. Basri and A. Brandt, Texture segmentation by multiscale aggregation of filter responses and shape elements, ICCV, 2003, I, 716-723, Nice.

E. Sharon, A. Brandt and R. Basri, Fast multiscale image segmentation, CVPR, I:70--77, South Carolina, 2000.

E. Sharon, A. Brandt and R. Basri, Segmentation and boundary detection using multiscale intensity measurements, CVPR,Hawaii, 2001.