MIATool User Manual[*]

Software Version: V1.1
Manual Version: V1.1
May 18, 2009

MIATool Development Team
Team Leader: Raimund J. Ober[†,‡]
Primary software and manual author: Jerry Chao[†,‡]
Current Members: Hongguang Xi[‡],
Jincheng Pang[†,‡], Anish Abraham[†,‡],
Sripad Ram[‡], E. Sally Ward[‡]

[†]Department of Electrical Engineering
University of Texas at Dallas
Richardson, TX 75080

[‡]Department of Immunology
University of Texas Southwestern Medical Center
6001 Forest Park Road, Room ND6.310
Dallas, TX 75390

Email: wardlab@utsouthwestern.edu

Contents

1. Software Overview

The **M**icroscopy **I**mage **A**nalysis Tool (MIATool) is a software application designed for the viewing and processing of large sets of image data. It makes efficient use of memory by working with pointers to images rather than the images themselves, and it saves disk space by storing reusable processing parameters in lieu of actual processed images on the hard drive. In addition, MIATool allows for multiple interpretations of a given image data set, realized by different, in general N-dimensional, arrangements of pointers to the images. A physical image data set can therefore be represented, viewed, and processed independently in different ways. A new arrangement of image pointers can be derived from an existing arrangement, and the current version of MIATool has available two types of such pointer array 'massaging' operations to support different interpretations of a given image data set.

MIATool supports the viewing and processing of an N-dimensional array of images. At its core is an image viewer which allows the traversal of an N-dimensional array of images. Besides the standard display as pixels of varying intensity values, options are available to view the images as mesh or contour plots. In addition, options exist that allow the concurrent viewing of multiple images, either in parallel in separate windows, or as an RGB overlay. Useful features such as the reporting of pixel intensity values, the optional display of a crosshair, and the display of pixel intensity profiles along the row and column of the crosshair, are also supported.

The current version of MIATool supports four different image editing tools which can be used to process the images displayed in the viewer. The intensity adjustment tool provides different ways to modify the pixel intensity values. The crop tool allows trimming of the images to retain only the portion that is of interest. Both the intensity adjustment tool and the crop tool offer options to apply changes to all or a subset of the images in the N-dimensional array. The two remaining tools - the segmentation tool and the label tool - can be used for manual image segmentation and image labeling.

Physically, the images as well as tool-related parameters such as intensity adjustment settings and crop settings are stored in a special directory structure on disk. Known quite simply as MIATool directories, these special directory structures serve as the standard input to, and output of, the software application.

For more details concerning the design of MIATool , please refer to [Chao et al., 2007].

2. Installation

2.1 Minimum system requirements

- PC with Pentium IV or equivalent processor suggested.

- MIATool was implemented and tested on Windows XP Service Pack 2. However, the executable provided may run on other versions of the Windows operating system as well.

- 512 MB RAM suggested.

2.2 Obtaining the software

Step 1: Fill in and submit the MIATool software request form at http://www4.utsouthwestern.edu /wardlab/miatool. An email will be sent to you in response.

Step 2: After receiving the email, access the Ward Lab software download page at http://www4.utsouthwestern.edu/wardlab/downloads.

Step 3: Login with the username and password given in the email response to your request for MIATool .

Step 4: Click on the MIATool download link and save the file **MIAToolSetup.exe**.

2.3 Installation

This section details the steps for installing MIATool V1.1 and MATLAB Component Runtime (MCR) 7.6. The MCR is required for the proper functioning of MIATool.

Step 1: Run the downloaded executable **MIAToolSetup.exe** by doing one of the following:

- If you are using a Windows user account with administrative privileges, then simply double-click **MIAToolSetup.exe**.

- If you are using a Windows user account that does not have administrative privileges, but you know the user name and password of an account with administrative privileges, then right-click on **MIAToolSetup.exe** and select **Run as...**. In the dialog box that appears, select **The following user:** and enter the user name and password of an account with administrative privileges. Click **OK**.

- If you don't know user name and password of an account with administrative privileges, then ask your administrator for help.

Step 2: In the welcome dialog box, click **Next >**.

Step 3: Read the MIATool software license agreement and select **I accept the agreement** to accept it. Click **Next >**.

Step 4: Specify the location where you would like MIATool to be installed. The default location is 'C:\Program Files', but you can change it if you want. Click **Next >**. If the folder you specify already exists, a dialog box will pop up and ask you to confirm the location. Please note the following:

Whatever location you specify, a folder called 'MIATool' will be created directly below it after the installation is completed. For example, if the location is 'C:\Program Files', then after installation completes, you will find the folder 'C:\Program Files\MIATool' created. This folder called 'MIATool' will be referred to as the **MIATool installation directory** throughout the remainder of this document.

Step 5: Optionally check the **Create a desktop icon** box if you would like to create a shortcut to MIATool on your desktop. Click **Next >**.

Step 6: Confirm the settings you've specified and click **Install** to start the installation.

Step 7: When installation is almost finished, an InstallShield Wizard will pop up to start preparing for the installation of MATLAB Component Runtime (MCR) 7.6.

Step 8: When the MCR welcome dialog box comes up, click **Next >**.

Step 9: If the same version of the MCR is already installed on your PC, then you will be asked whether you wish to modify, repair, or remove the MCR. In this case, click **Cancel**, confirm that you want to cancel the MCR installation, and continue to Step 14.

Step 10: In the dialog box that appears next, enter your information and click **Next >**.

Step 11: Specify the location where you would like the MCR to be installed. You can accept the default location or change it if you want. Click **Next >**.

Step 12: In the next dialog box, click **Install** to start the MCR installation.

Step 13: If you do not have .NET Framework installed, a warning box will pop up to allow you to cancel the MCR installation and install .NET Framework first. MIATool itself does not require .NET Framework, so unless you need .NET Framework for something else, simply click **OK** to continue.

Step 14: The MCR is installed. Click **Finish** to continue.

Step 15: The ensuing dialog box displays an important message concerning the launching of MIATool in the next and final step of the installation. Please read it and click **Next >**.

Step 16: MIATool installation is now complete. To immediately test the installation, make sure the **Launch MIATool** box is checked. Click **Finish**. If you chose to launch MIATool, the application

will take some time to initialize itself, after which you will see one of the following:

- An error message indicating that MIATool cannot start because the file mclmcrrt76.dll is missing. If you get this error, try restarting your PC. After the restart, try to launch MIATool again by following the instructions in Section 3.1.

- The MIATool main menu figure shown in Figure 3.1.1. If you see this, then the installation was successful. Go to the tutorial in this manual to begin using MIATool.

2.4   Uninstalling MIATool

MIATool can be uninstalled only with a user account that has administrative privileges. Use **Add or Remove Programs** in the Control Panel to remove **MIATool V1.1**. Alternatively, run the executable **unins000.exe** in the 'V1.1' folder under your MIATool installation directory (see Step 4 of Section 2.3 to find out where your MIATool installation directory is).

Note that the MATLAB Component Runtime (MCR) has to be uninstalled separately. If you wish to remove the MCR, use **Add or Remove Programs** in the Control Panel to remove **MATLAB Component Runtime 7.6**.

3. Tutorial

This is a hands-on tutorial that is designed to familiarize the user with the most common usage of MIATool . In addition to the step-by-step instructions, detailed information is given throughout the tutorial to clarify or reinforce the ideas behind the steps taken. Details are also provided to explain aspects of the software that are not part of the hands-on tutorial. To help the user to utilize this tutorial in the most efficient way, the text is color-coded to categorize the type of information that is provided:

The step-by-step instructions are given in black. If a quick run-through of the tutorial is desired to get an idea of how the software works, then it is sufficient to go through only the text in black. For a fuller understanding of the various aspects of the software, however, it is recommended that you go through all the text.

Text in red is meant to help you with troubleshooting in case you run into problems in the hands-on tutorial.

Alternatives to the given instructions are shown in blue. Text in blue is also used to provide suggestions to try things that are not part of the main sequence of step-by-step instructions.

Text in green is used to provide explanation, clarification, or reinforcement of the ideas behind the instructions. Equally importantly, it is also used to give details concerning parts of the software (e.g., drop-down menu options) that are not used by the hands-on tutorial, but are likely to be useful to the user. Please note, however, that not all parts of the software are discussed in this tutorial.

3.1   Starting MIATool

Step 1: *Open MIATool main menu* The MIATool main menu figure can be opened in one of three ways:

- From the Windows Start menu, select **Start → Programs → MIATool → V1.1 → MIATool**.

- If you chose to create a desktop shortcut to MIATool during installation, you can double-click the shortcut **MIATool V1.1** on your desktop.

- Double-click the file **MIATool.exe** in the 'V1.1' folder under your MIATool installation directory (see Step 4 of Section 2.3 to find out where your MIATool installation directory is).

You will see the MIATool main menu shown in Figure 3.1.1.

- If this is your first attempt at launching MIATool, you might get an error message indicating that MIATool cannot start because the file mclmcrrt76.dll is missing. If you get this error, try restarting your PC. After the restart, try to open the MIATool main menu figure again.

- If you have not successfully opened the MIATool main menu figure before, you might find that nothing happens and that there are no error messages. If you see this behavior, try starting MIATool again using a user account with administrative privileges. (Right-click, choose **Run as...**, and enter the administrative account user name and password.) You should only have to do this the first time. Subsequent launches of the program should not require an account with administrative privileges.
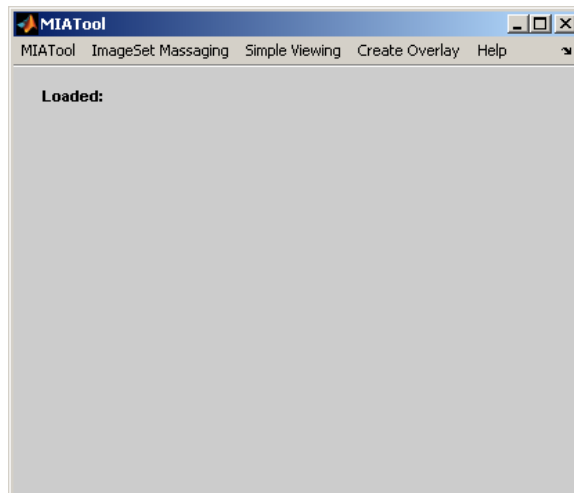

Figure 3.1.1: MIATool main menu

3.2   Importing images into MIATool

This section illustrates how to import images into what is called a MIATool directory, a special directory structure that MIATool can recognize and work with. We will import two series of sample

10

grayscale images provided as part of this MIATool distribution.

Step 1: *Choose location to create new MIATool directory* Open the MIATool main menu (see Section 3.1 if you need help). From the MIATool main menu, select **MIATool → Create New MIATool Directory**. Using the directory browser that pops up, select the location where you want to create the MIATool directory. For illustration here, we will choose the root of the C drive. Click **OK**.

- You can choose any location you like. Just keep in mind that the example here assumes that C: is chosen.

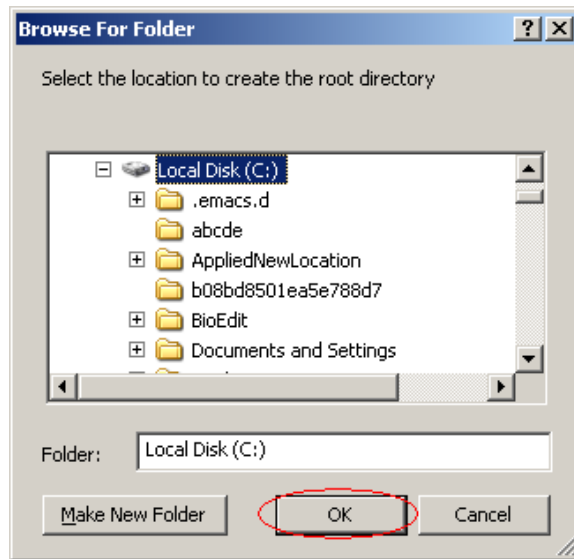
Figure 3.2.1: Selecting location to create MIATool directory

Step 2: *Provide name for root folder of new MIATool directory* Specify **miatooldemo1** as the name of the root folder of the MIATool directory. Click **OK**.

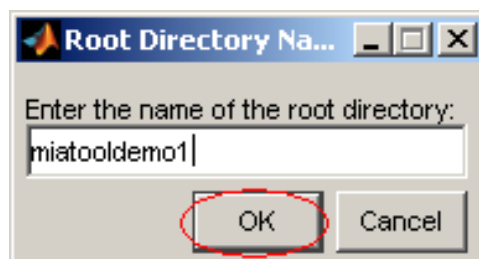- You can specify any name you like. Just keep in mind that the example here assumes the name miatooldemo1.

Figure 3.2.2: Providing name for root of MIATool directory

Step 3: *Specify first series of images to import* Select the first series of images to import into the MIATool directory by performing the following:

A. Using Windows Explorer, look in the folder '\V1.1\SampleData\grayscale_single_jpgs' under your MIATool installation directory and see for yourself that the folder contains 10 individual jpeg images named '1.jpg', '2.jpg', ..., '10.jpg'. We want to import these 10 images into the MIATool directory as the first image series.

B. Click the **Browse** button and use the directory browser to select the folder '\V1.1\SampleData\grayscale_single_jpgs' under your MIATool installation directory. Click **OK**. The 'Series Path' text box will now contain the full path to the selected folder.

C. Because the images in the selected folder are individual jpeg images with file names of the form '1.jpg', '2.jpg', ..., make sure the option **n.xxx** is selected in the 'Series Type' drop-down menu.

D. Enter **abc** in the 'File Name Prefix' text box.

E. Because we only want to import a copy of the jpeg images, make sure the option **Copy** is selected in the 'Import Method' drop-down menu.

F. Click **OK** to import another series of images.

- Refer back to Step 4 of Section 2.3 if you don't know the location of your MIATool installation directory.

- The series type **n.xxx** requires that the source images are **individual**, **8-bit or 16-bit** files of any image format that is supported by MATLAB. The images must reside in a single folder, and they must all be of the same type and format. If there are N images, then they must be named **'1.xxx', '2.xxx', ..., 'N.xxx'**. For example, if you have N 8-bit png images, then they must be named **'1.png', '2.png', ..., 'N.png'. MIATool's import process will convert all .xxx images to the 16-bit, uncompressed, single tiff format.**

- The import method **Copy** imports a copy of the source images and leaves the source images untouched.

- The file name prefix is optional. It can be used to specify a meaningful prefix in the file name of every image in the imported series. It can, for instance, be the initials of the microscopist who acquired the images, the name of the fluorophore that was imaged, an experiment identifier, etc.
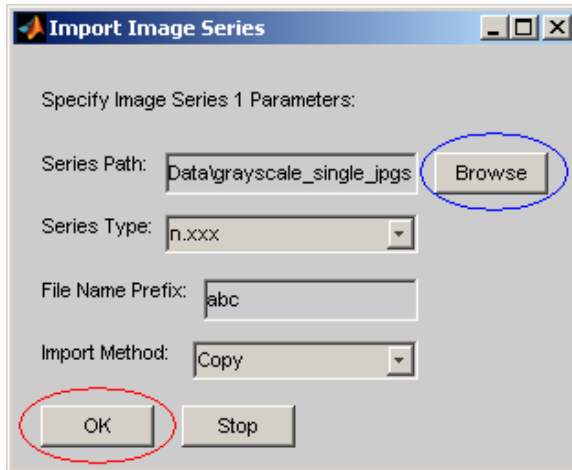
Figure 3.2.3: Importing individual tiff images

Step 4: *Specify second series of images to import* Select the second series of images to import into the MIATool directory by performing the following:

A. Using Windows Explorer, look in the folder '\V1.1\SampleData\grayscale_multiframe_tif' under your MIATool installation directory and see for yourself that the folder contains a single multi-frame tiff file called 'multiframe.tif'. This multi-frame tiff file contains 10 images. We want to import these 10 images in the form of a multi-frame tiff file into the MIATool directory as the second image series.

B. Click the **Browse** button and use the directory browser to select the folder '\V1.1\SampleData \grayscale_multiframe_tif' under your MIATool installation directory. Click **OK**. The 'Series Path' text box will now contain the full path to the selected folder.

C. Because the images in the selected folder are in the form of a single multi-frame tiff file, make sure the option **multi-frame tif** is selected in the 'Series Type' drop-down menu.

D. Leave the 'File Name Prefix' text box empty.

E. Because we only want to import a copy of the multi-frame tiff file, make sure the option **Copy** is selected in the 'Import Method' drop-down menu.

F. Click **OK**.

- The series type **multi-frame tif** requires that the source images are the frames of a **single**, **8-bit or 16-bit**, **multi-frame tiff** file. The multi-frame tiff file can have any file name of the form **'anyname.tif'**. **MIATool's import process will convert it to a 16-bit, uncompressed, multi-frame tiff file.**

- The series type **n.tif** requires that the source images are **individual**, **8-bit or 16-bit**, **tiff** files. The images must reside in a single folder, and if there are N images, then they must be named **'1.tif', '2.tif', ..., 'N.tif'**. **MIATool's import process will convert all .tif images to the 16-bit, uncompressed, single tiff format.**

13

- The series type **MIATool** requires that the source images already reside as a series in an existing MIATool directory. The application will automatically determine whether the images are individual tiff files or a single multi-frame tiff file, the two possibilities being the only valid formats for images inside a MIATool directory.
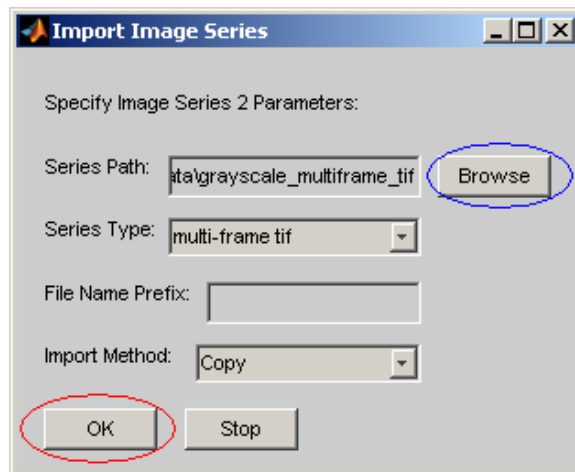
Figure 3.2.4: Importing single multi-frame tiff file

Step 5: *Terminate image series specification* We are now asked to import a third series, but because we only want to import two series of images, click **Stop** to terminate the series specification process.
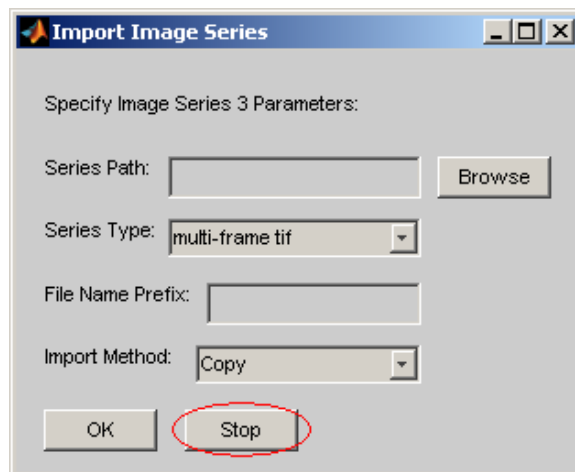
Figure 3.2.5: Terminating import series specification

Step 6: *Provide name for default ImageSet object* Enter **original** for the file name of the default ImageSet object that will be created. Click **OK**.

- You can specify any file name you like. You can also leave the text box empty, in which case

14

the name 'imageset1' will be used automatically. Just keep in mind that the example here assumes the name 'original' is specified.

- An ImageSet object is a container of pointers to images. In our present case where the images reside on disk, the pointers are simply the full file paths of the images.

- A **default** ImageSet object (which we will be creating in the present case) is one where the image pointers are arranged in a 2D array with the following properties: **1.** the rows of the 2D array correspond to the image series on disk, **2.** if a series consists of individual image files, then the pointers in its corresponding row are arranged in ascending order of the numbers in the file names of the images they reference. If a series consists of a multi-frame file, then the pointers in its corresponding row are arranged in ascending order of the frame numbers of the images they reference in the multi-frame file. In our present case, the default ImageSet object will contain a 2x10 array where the first row of pointers will reference the individual tiff images of series 1 in the order '1.tif', '2.tif', ..., '10.tif', and the second row of pointers will reference the frames in the multi-frame file in the order frame 1, frame 2, ..., frame 10.
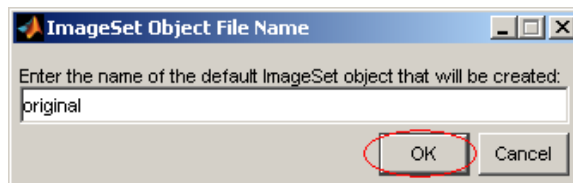
Figure 3.2.6: Specifying file name of default ImageSet object

Step 7: *New MIATool directory created* The MIATool directory has now been created and the import process is complete. In the final message box, click **OK**. Notice the message 'Loaded: C:\miatooldemo1' displayed in the MIATool main menu figure. This means that the newly created MIATool directory is automatically loaded after creation, indicating that we can start working with the images we have just imported. For now, let's close the main menu figure by clicking the close (×) button in the upper right-hand corner of the figure.
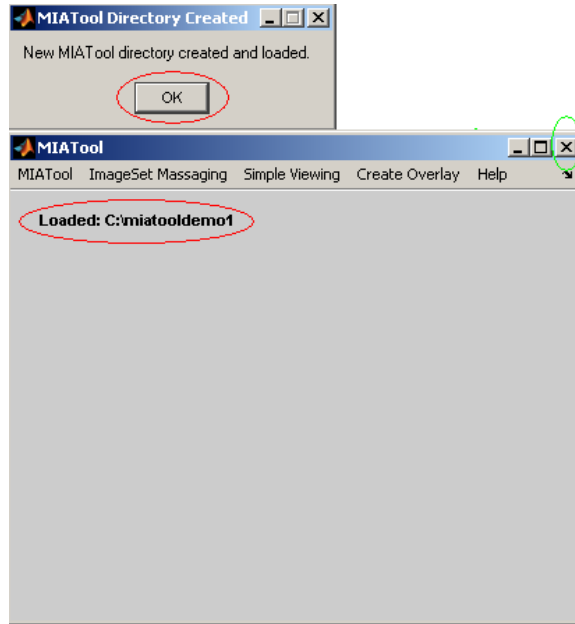
Figure 3.2.7: MIATool directory created

Step 8: ***Look at contents of root folder of MIATool directory*** Let's take a look at the MIA-Tool directory we've created. Using Windows Explorer, look in the root folder 'C:\miatooldemo1'. There you should see a subfolder called 'ImageFiles', another subfolder called 'MIAToolObjects', and a file called 'miatooldir_miatooldirectory.mia'. The file 'miatooldir_miatooldirectory.mia' contains a detailed description of the contents of the MIATool directory and provides the standard interface for MIATool to interact with the physical directory structure. A file by the same name will be found under the root folder of every MIATool directory.

- If you didn't choose 'C:\miatooldemo1' as the root folder of your MIATool directory, then make sure that you look in your root folder instead.

Figure 3.2.8: Root folder of MIATool directory

Step 9: ***Look at contents of ImageFiles subfolder of MIATool directory*** Look in the subfolder 'ImageFiles'. There you should see the folders 'Series1' and 'Series2', which contain

16

respectively the two series of images we have imported. In the 'Series1' folder, you should find ten individual tiff images with file names 'abc_s1_1.tif', 'abc_s1_2.tif', ..., 'abc_s1_10.tif'. Note that regardless of the format of the source images (jpeg in our case), the imported images will be in the tiff format. The 's1' stands for 'single tiff, series **1**', and is made part of the file name of each imported individual image automatically by MIATool . The prefix 'abc' was what we specified during the import process. In the 'Series2' folder, you should find a single multi-frame tiff file called 'm2.tif'. The 'm2' stands for 'multi-frame tiff, series **2**', and is automatically added to the file name of the imported multi-frame tiff file by MIATool . Because we did not specify a file name prefix during the import process, the file name is just 'm2.tif'. If we had provided a prefix, it would have accordingly been prepended to 'm2.tif'.
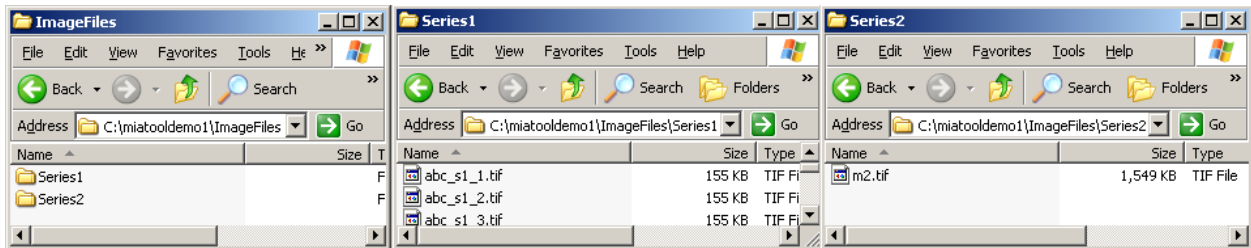
Figure 3.2.9: ImageFiles folder of MIATool directory
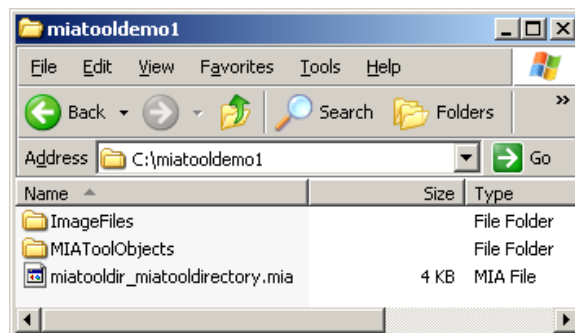
Step 10: *Look at contents of MIAToolObjects subfolder of MIATool directory* Look in the subfolder 'MIAToolObjects'. There you should see the folder 'ImageSet1', which contains the default ImageSet object we've created. The file name of the ImageSet object is 'original_imageset.mia', where 'original' is the file name we specified during the import process, and '_imageset.mia' is the standard extension used by MIATool to identify ImageSet objects.
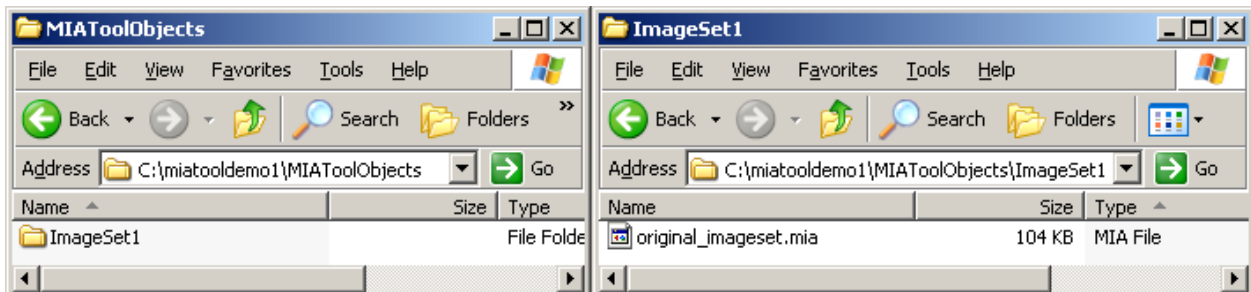
Figure 3.2.10: MIAToolObjects folder of MIATool directory

3.3   Loading a MIATool directory

Before you can view and process images in MIATool , you must first load the MIATool directory that contains the images.

Step 1: *Open MIATool main menu* Start MIATool by opening the MIATool main menu figure. See Section 3.1 if you need help.

Step 2: *Load MIATool directory* Select **MIATool → Load MIATool Directory**. Using the directory browser that pops up, select the MIATool directory 'C:\miatooldemo1', or whichever MIATool directory you created in Section 3.2. Click **OK**. When loading is finished, the message 'Loaded: C:\miatooldemo1' will be displayed in the main menu figure.

- A main menu figure can have only one MIATool directory loaded at any given time. To load a different MIATool directory, you simply need to perform this step again to select it. The currently loaded MIATool directory will be replaced with the newly selected one. There is no need to close and reopen the main menu figure.
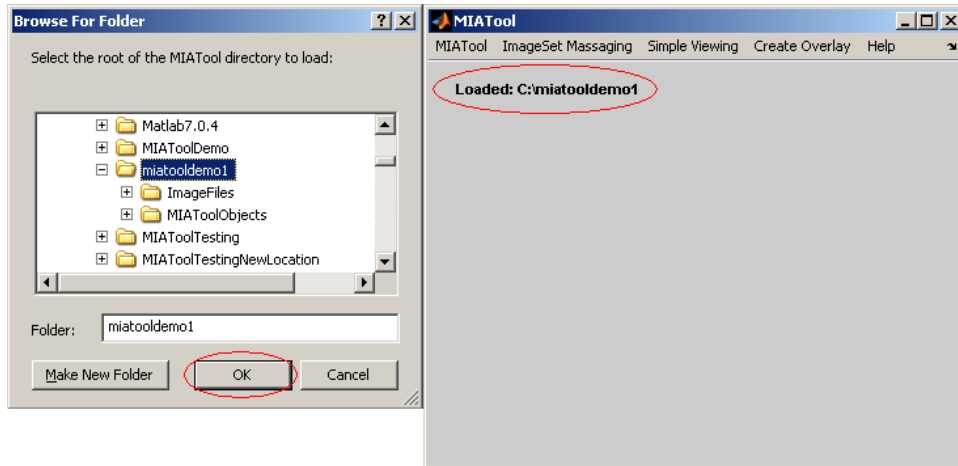
Figure 3.3.1: Loading existing MIATool directory

3.4   Image viewing and intensity adjustment

In this section, we demonstrate MIATool's image viewer and pixel intensity adjustment tool using the two series of grayscale images in the MIATool directory created in Section 3.2. Subsections follow which introduce the viewer's crosshair and its associated display of pixel intensity profiles, the viewer's mesh and contour display modes, and the concurrent viewing of multiple images, which includes the viewing of RGB overlays of the grayscale images.

Step 1: *Initiate opening of MIATool viewer* After you have loaded the MIATool directory 'C:\miatooldemo1' (see Section 3.3), select **MIATool → Open DisplayTool**.

Step 2: *Choose ImageSet object to view and intensity-adjust* Select 'original_imageset.mia', the default ImageSet object we created in Section 3.2 and the only choice we have right now. Click **OK**. (See Figure 3.4.1, left panel.)

Step 3: *Choose a default DisplaySet object* Select 'DisplaySet' from the 'Object Type' drop-down menu. Highlight 'default DisplaySet' (the only option) in the 'Available Objects' list box and click the **Add** button to add it to the 'Selected Objects' list box. Click **Done**. (See Figure 3.4.1, middle panel.)

18

- To be able to adjust the pixel intensity values of the images you are viewing, you must select a DisplaySet object at this step.

- A 'DisplaySet' object is a container of pixel intensity settings for the images referred to by an ImageSet object. In our present case where there are a total of 20 images, the DisplaySet object will be automatically configured to store the pixel intensity settings for 20 images.

- A **default** DisplaySet object is one such that the pixel intensity settings it contains do not alter the pixel values of the images they act on. In other words, it performs no intensity adjustments on the images.

- Whereas the **Add** button is used to select an object by placing it in the 'Selected Objects' list box, the **Delete** button removes a selected object by deleting it from the 'Selected Objects' list box. To use the 'Delete' button, simply highlight the object in the 'Selected Objects' list box which you want to remove, then click the **Delete** button.

Step 4: *Decline LabelSet and SegmentationSet object selection* In the next figure that pops up, simply click **Done** since we won't be doing any image labeling or segmentation in this part of the tutorial. (See Figure 3.4.1, right panel.)
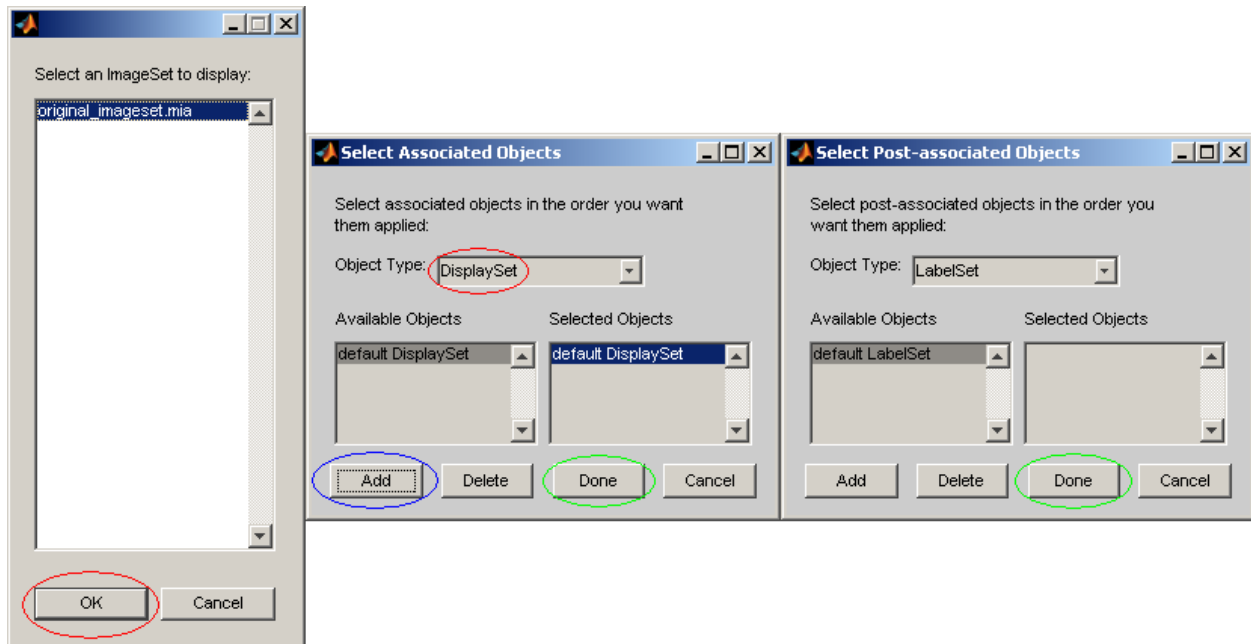
Figure 3.4.1: Opening image viewer

Step 5: *MIATool viewer opened* The MIATool viewer is now opened. You should see two figures, one that contains sliders for navigation through all the images, and one that displays whatever image is current according to the values of the sliders. Drag the figures apart so that you can see them side by side.
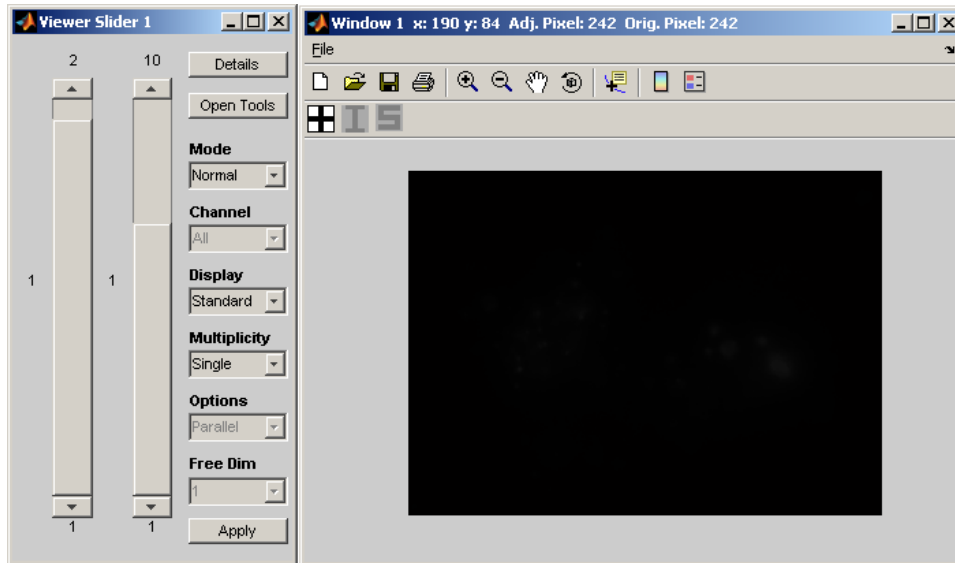
Figure 3.4.2: MIATool viewer

Step 6: _Verify currently displayed image_ There are two sliders in our current example because
we are viewing with a default ImageSet object that references two series of images on disk. (Refer
to Step 6 of Section 3.2 for a detailed description of a default ImageSet object.) The value of the
first slider (either 1 or 2) corresponds to the series number on disk, and the value of the second
slider (integer between 1 and 10) corresponds to the image number within a series. Notice that
initially the current image defaults to image 1 of series 1. To verify this, click the **Details** button
in the figure with the sliders to see in a separate window the details concerning the current image.
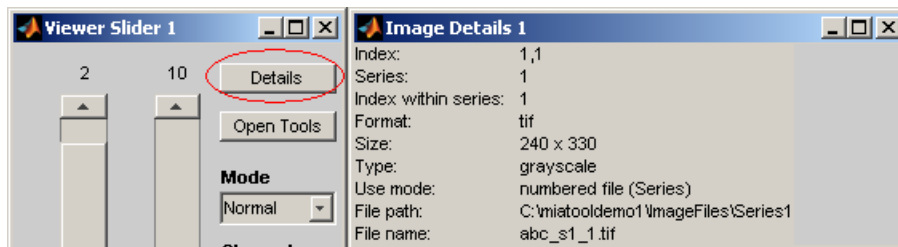
Figure 3.4.3: Details window showing information for image 1 of series 1

Step 7: _Change current image_ Change the value of the first slider to **2**, and with the image
details window from the previous step still open, click the **Details** button again. See that the
information in the image details window is now updated for the new current image, i.e., image 1
of series 2.

- The image details window does not update when the current image changes. Press the
'Details' button again to force the update.

20
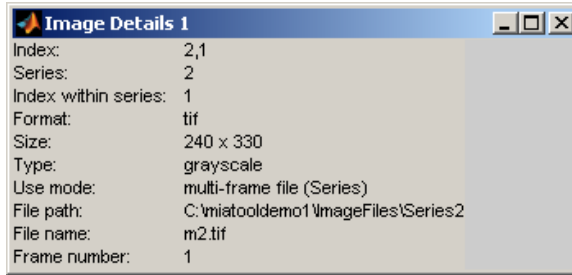
Figure 3.4.4: Details window showing information for image 1 of series 2

Step 8: *Scroll viewer sliders and observe low intensity images* Scroll the sliders and observe that you are not able to see anything in any of the 20 images. To understand why, move your mouse over the displayed image and at the same time look at the continuously changing numbers shown in the image window's title bar. For the image pixel your mouse pointer is currently pointing to, the title bar shows the x and y pixel coordinates and two pixel intensity values: adjusted and original. Note that on a scale of 0 to 65535 (i.e., a 16-bit scale, the MIATool standard), the reported pixel values are low intensity values, and hence we are not able to see anything in the image.

- The adjusted pixel values are identical to the original pixel values (see Figure 3.4.5 for an example) because we are viewing with a default DisplaySet object. Refer to Step 3 for an explanation of a default DisplaySet object.
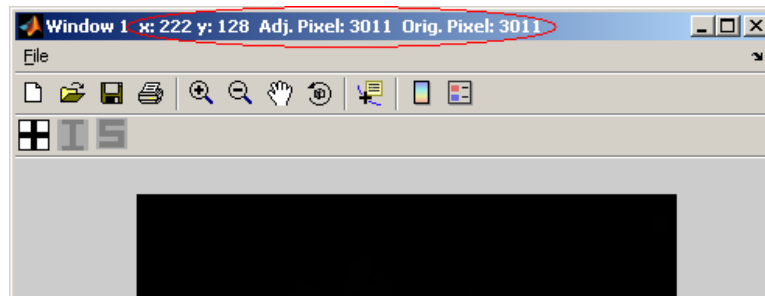
Figure 3.4.5: Title bar of image window shows pixel values of low intensity

Step 9: *Open pixel intensity adjustment tool* Let's now try some pixel intensity adjustments so that we can see something in these images. Click the **Open Tools** button in the figure with the sliders (see Figure 3.4.6, left panel). In the list box that pops up, select '1: displayset'. (It is the only choice we have since we selected only a DisplaySet object during the viewer opening process.) Click **OK** (see Figure 3.4.6, middle panel). The next figure that appears is the pixel intensity adjustment tool. For a default DisplaySet object, you will find the initial adjustment method to be 'Imadjust', which works with five sliders/text boxes: 'LowIn', 'HighIn', 'LowOut', 'HighOut', and 'Gamma' (see Figure 3.4.6, right panel).

- Try changing the values of the 'LowIn', 'HighIn', 'LowOut', 'HighOut', and 'Gamma' sliders/text boxes and see if you can see something in the current image.

- The 'Imadjust' method corresponds to the MATLAB Image Processing Toolbox's 'imadjust' function. Details about the 'imadjust' function can be found on www.mathworks.com.

- In addition to the 'Imadjust' method, four other pixel intensity adjustment methods are available: 'Multiply', 'Divide', 'Subtract', 'Add'. 'Multiply' multiplies every pixel by some constant, 'Divide' divides every pixel by some constant, 'Subtract' subtracts some constant from every pixel, and 'Add' adds some constant to every pixel. They each work with a text box by the same name. **Only one adjustment method is active at any given time.**

- Try the other adjustment methods by selecting them from the intensity adjustment tool's **Method** drop-down menu, and then changing the values in their respective text boxes.
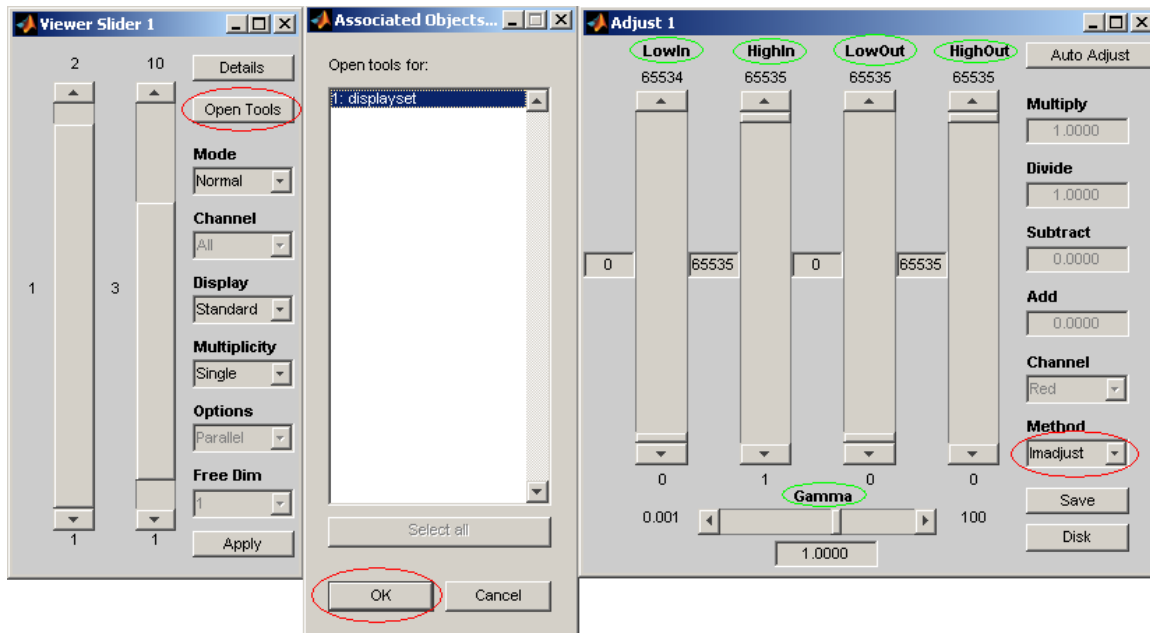
Figure 3.4.6: Opening pixel intensity adjustment tool

Step 10: *Auto-adjust pixel intensities* So that you'll see the exact same results as shown in this part of the tutorial, make sure to set the value of each viewer slider back to 1, and also set the 'Methods' drop-down menu in the intensity adjustment tool figure to 'Imadjust'. Then click the **Auto Adjust** button in the intensity adjustment tool figure. You should see a change in the values of the 'LowIn' and 'HighIn' sliders/text boxes, and should now be able to see something in the displayed image.

- The **Auto Adjust** button works with all five different adjustment methods. When the method is 'Imadjust', it maps the smallest and largest pixel values in the image to 0 and 65535, respectively, and all the values in between linearly to the range [0, 65535]. When the method is 'Multiply', 'Divide', 'Subtract', or 'Add', a constant is selected such that when applied using the respective operation, the maximum pixel value in the image becomes 65535.

22
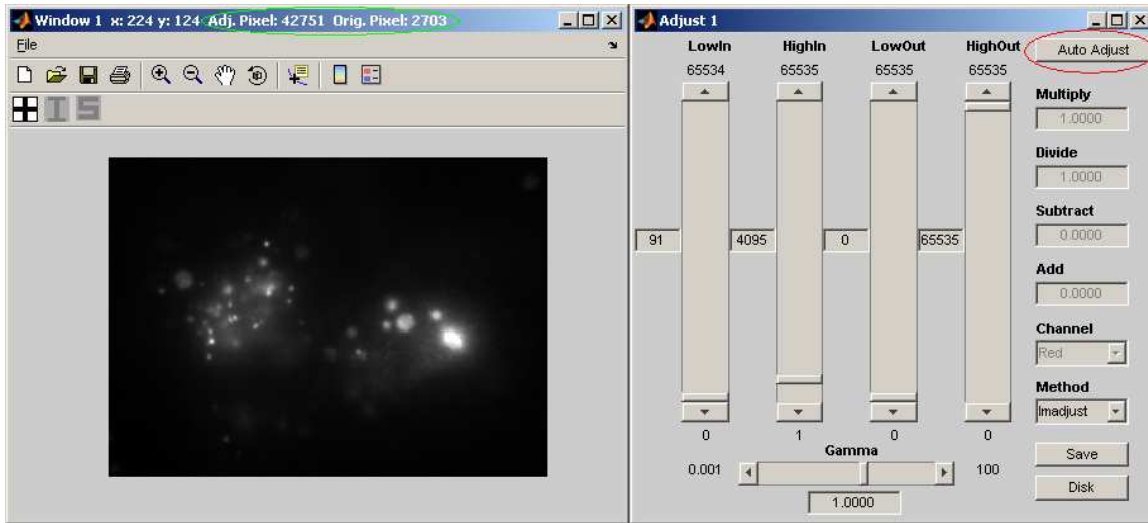
Figure 3.4.7: Auto-adjusting image 1 of series 1

Step 11: *Verify intensity adjustment has taken place* Move your mouse over the displayed image and see for yourself that because an intensity adjustment has taken place, the adjusted and original values of each pixel reported in the title bar of the image window are now different. See Figure 3.4.7, left panel, for an example.

Step 12: *Verify intensity settings are not saved* Change the current image to something other than image 1 of series 1 using the viewer sliders. Whatever the new current image you choose, notice that as before, nothing can be seen. This is because the adjustment we made was applied only to image 1 of series 1, the then-current image. Now change the current image back to image 1 of series 1. You should find that it has lost the adjustment that was applied to it. The reason for this is because we never actually saved the adjustment settings obtained from using the 'Auto Adjust' button.

Step 13: Perform the following.

A. ***Apply intensity settings to current image only*** With the current image set to image 1 of series 1, click the **Auto Adjust** button again to adjust the image. Click the **Save** button in the intensity adjustment tool. Select **Apply to Current Image** and click **OK** to save to memory the adjustment settings for the current image only. Click **OK** in the message box that pops up. (The message box reminds us that the adjustment settings have been saved to memory, but not to disk yet! More about this in the next step.) Now scroll the viewer sliders and then come back to image 1 of series 1. You will find that because we have saved them to memory, the adjustment settings are remembered for image 1 of series 1.

B. ***Apply intensity settings to all images of series 1*** We can apply the same adjustment settings to a subset of images. To apply, for example, the adjustment settings for image 1 of series 1 to all 10 images of series 1, first make sure that image 1 of series 1 is the current image. Click the **Save** button in the intensity adjustment tool. Select **Apply to Subset of Images** and click **OK**. In the 'Subset Selection' figure that appears, you will find two text

23

boxes. The left text box corresponds to viewer slider 1, and the right text box corresponds to viewer slider 2. Because we want to apply the adjustment settings to all images in series 1, enter **1** in the left text box for series 1 and **all** in the right text box for all images. (Instead of 'all', you can enter '1-10' in the right text box to get the same result.) Click **Done**. Click **OK** in the message box that pops up. Now with the first viewer slider set to 1, scroll the second viewer slider to find that the same intensity adjustment has been applied to all 10 images of series 1. With the first viewer slider set to 2, you'll find with no surprise by scrolling the second viewer slider that none of the images of series 2 are intensity-adjusted.

C. ***Apply intensity settings to all images*** We can apply the same adjustment settings to all images. Make any intensity-adjusted image, i.e., any series 1 image, the current image. Click the **Save** button in the intensity adjustment tool. Select **Apply to All Images** and click **OK**. Click **OK** in the message box that pops up. Now you should be able to see something in all images when you scroll through them.

D. ***Apply different intensity settings to images of series 2*** We can apply different adjustment settings to different images. For example, it looks like the images of series 2 are still quite a bit too dim. To use a more appropriate adjustment, first make image 1 of series 2 the current image (i.e., set first viewer slider to 2, second viewer slider to 1). Click the **Auto Adjust** button in the intensity adjustment tool to adjust the image. You should now see the object in the image significantly brighter. Click the **Save** button in the intensity adjustment tool. Select **Apply to Subset of Images** and click **OK**. In the 'Subset Selection' figure, enter **2** in the left text box for series 2 and **1-10** in the right text box to specify all 10 images. (You can instead enter 'all' in the right text box to get the same result.) Click **Done**. Click **OK** in the message box that pops up. Now with the first viewer slider set to 2, scroll the second viewer slider to find that the brighter intensity setting has been applied to all 10 images of series 2.

- To apply the adjustment settings of an image to other images, first make the image with the desired adjustment settings the current image. Then use the 'Save' button in the intensity adjustment tool to apply the adjustment settings to other images.

- When specifying a subset of images to apply adjustment settings to, any subrange of the possible values of a slider may be entered. For example, for the second viewer slider which takes on values between 1 and 10, all of the following are valid subranges: **2-7**, **1-10** (same as **all**), **4-4** (same as just **4**), **3-6**.
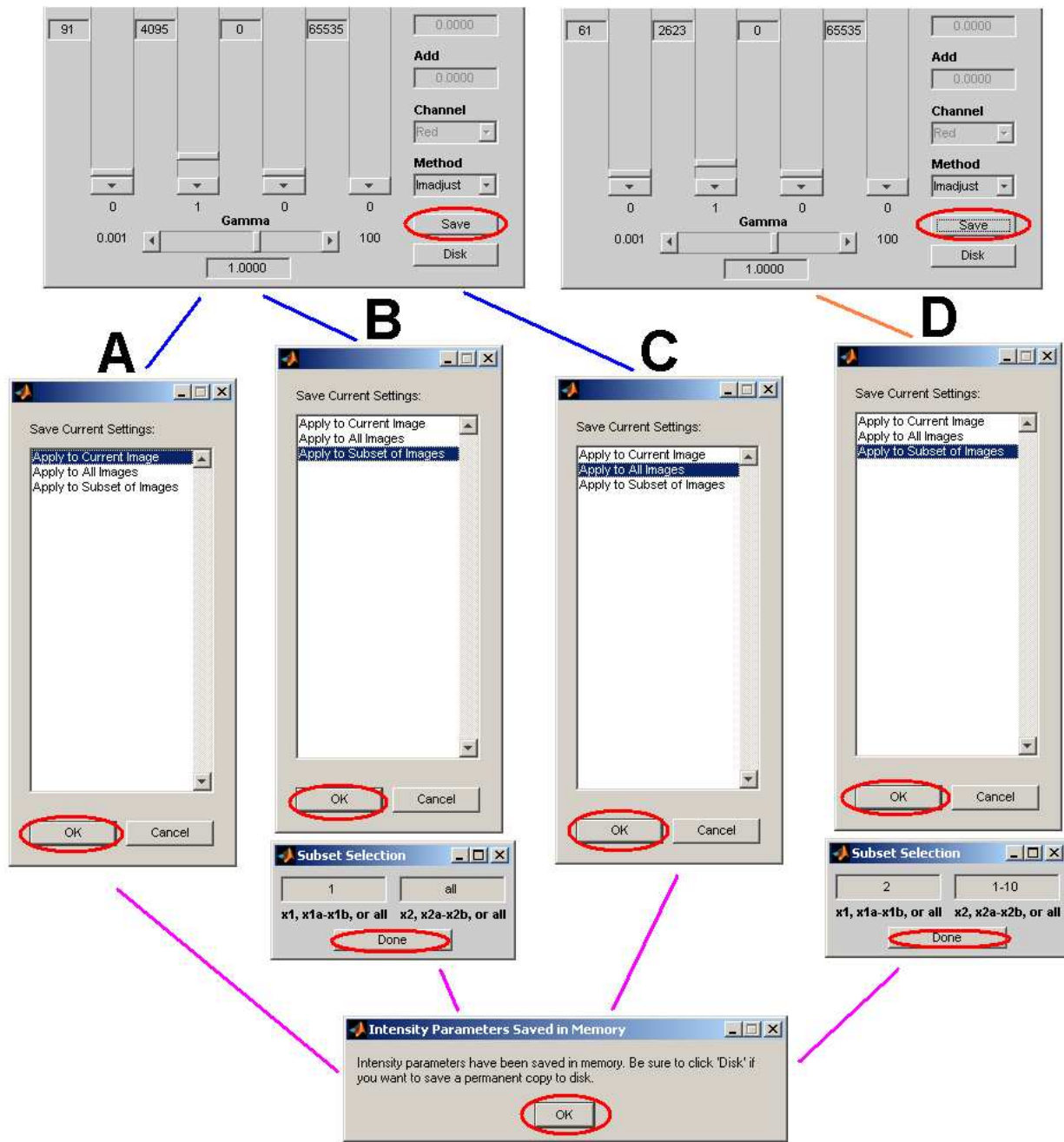
Figure 3.4.8: Saving pixel intensity settings to memory

Step 14: *Save intensity settings to disk in a DisplaySet object* As indicated by the message box that appears at the end of each 'Save' operation (see message box at bottom of Figure 3.4.8), we have so far only saved the adjustment settings to memory. This means that once we close the MIATool viewer, all the adjustment settings will be lost. To keep the adjustment settings for the next time we view these same images, we need to save them to disk in the form of a DisplaySet object under our MIATool directory. Click the **Disk** button in the intensity adjustment tool. In

the box that pops up, enter **autoadjust** for the file name of the DisplaySet object. Click **OK**. In the confirmation box that appears, click **OK**.

- You can specify any file name you like for the DisplaySet object. You can also leave the text box empty, in which case the name 'displayset1' will be used automatically. Just keep in mind that the example here assumes the name 'autoadjust' is specified.

- The **Disk** button will save to disk settings that are currently saved in memory, i.e., settings most recently saved to memory using the 'Save' button.
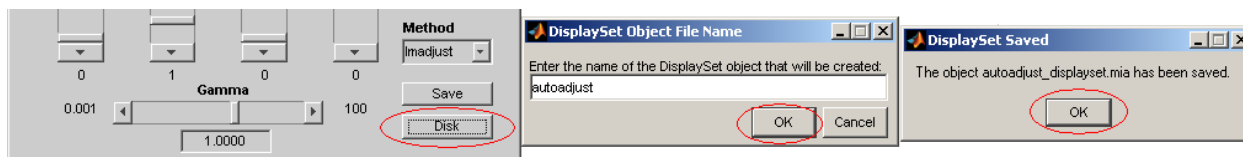
Figure 3.4.9: Saving pixel intensity settings to disk

Step 15: *Verify DisplaySet object has been saved to disk* Let's verify that the Display-Set object has been saved to our MIATool directory. Using Windows Explorer, navigate to 'C:\miatooldemo1\MIAToolObjects\ImageSet1'. There you should find a subfolder called 'display-set objects'. Inside that subfolder, you should see the DisplaySet object 'autoadjust_displayset.mia'. The file name 'autoadjust' is what we specified in the previous step, and '_displayset.mia' is the standard extension used by MIATool to identify DisplaySet objects.

- If you didn't choose 'C:\miatooldemo1' as the root folder of your MIATool directory, then make sure that you look in your root folder instead.
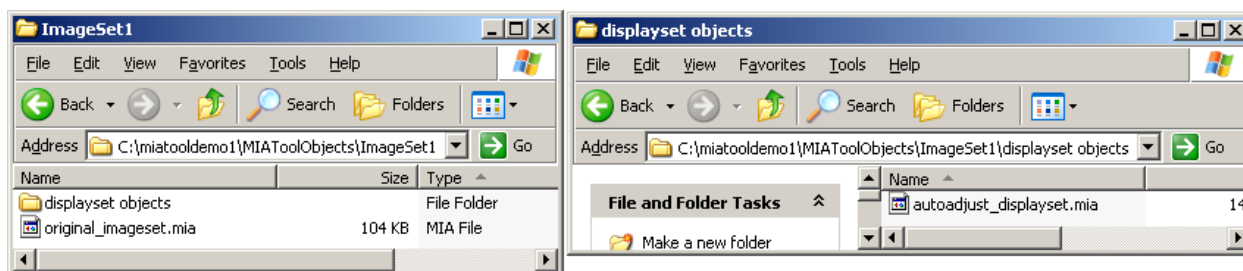
Figure 3.4.10: DisplaySet object saved to MIATool directory

3.4.1 Crosshair and pixel intensity profiles

The MIATool viewer supports a crosshair to help the user keep focus on a particular area in the image while images are scrolled through. It is useful, for example, for keeping focus on a region of dynamic interactions between subcellular objects when scrolling through a time sequence of live

26

cell images.

Step 16: *Activate and use crosshair* To activate the crosshair, click its toolbar button (black cross in white background) in the image window. When first activated, the crosshair is centered at the top left corner of the image. (If you look carefully, you'll see a white horizontal line across the top of the image and a white vertical line spanning the left edge of the image.) Click anywhere inside the image to see the crosshair better, then scroll the second viewer slider and see for yourself that the crosshair remains in the same position even when the image is changing. To change the crosshair to a new position, simply click the new position.
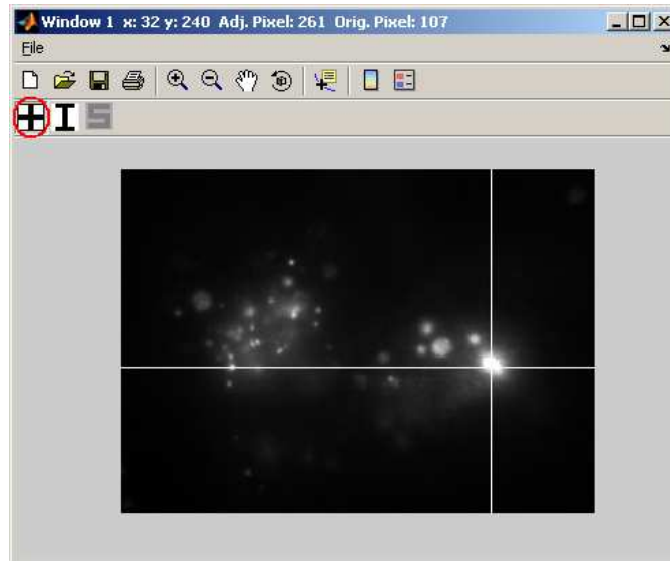
Figure 3.4.11: Crosshair activation and positioning

Step 17: *Activate and view pixel intensity profiles* Once the crosshair is activated in an image window, you can choose to see the pixel intensity profiles along the horizontal and vertical lines that comprise the crosshair. To activate the pixel intensity profiles, click their toolbar button (black 'I' in white background) in the image window. You will see pixel intensity profiles in the form of a blue curve in the left margin of the image window (for the vertical crosshair line) and a blue curve in the bottom margin of the image window (for the horizontal crosshair line). If you make image 1 of series 1 your current image and position the crosshair as shown in Figure 3.4.11, then you will see intensity profiles as shown in Figure 3.4.12. Scroll the second viewer slider to see the intensity profiles change.
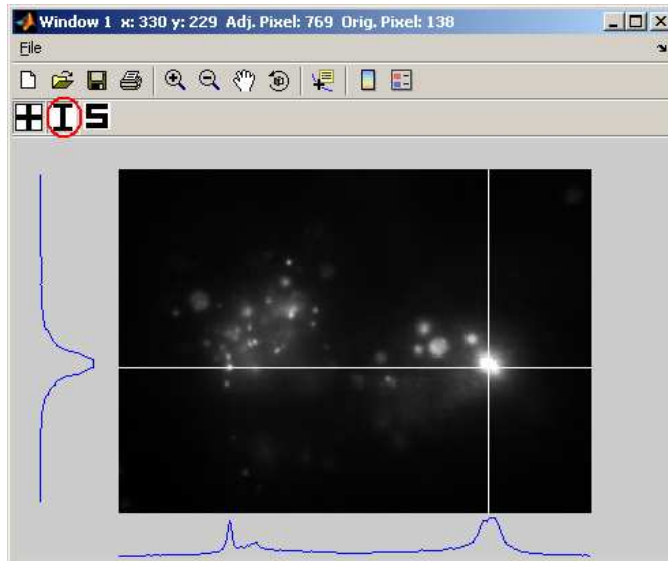
Figure 3.4.12: Pixel intensity profiles along the lines of the crosshair

Step 18: *Scale pixel intensity profiles* Sometimes the pixel intensity profiles either cannot be seen or appear as straight lines. In these situations, some manual scaling is necessary. To perform scaling, click its toolbar button (black 'S' in white background) in the image window. In the figure that pops up, you will find two text boxes in which you can enter scale factors for the horizontal and vertical intensity profiles. If you're seeing either no profiles or straight-line profiles, then you want to increase the scale factors. On the other hand, if the profiles exceed their margins and overlap with too much of the image, then you want to decrease the scale factors. For the intensity profiles shown in Figure 3.4.12, increasing both the horizontal profile scale factor and the vertical profile scale factor to 0.002 and clicking **OK** will give the profiles shown in Figure 3.4.13. Notice that the profiles now exceed their margins and obscure the image a bit, but are significantly better resolved in terms of discernible variations in the curves. Again, scroll the second viewer slider to see the intensity profiles change.
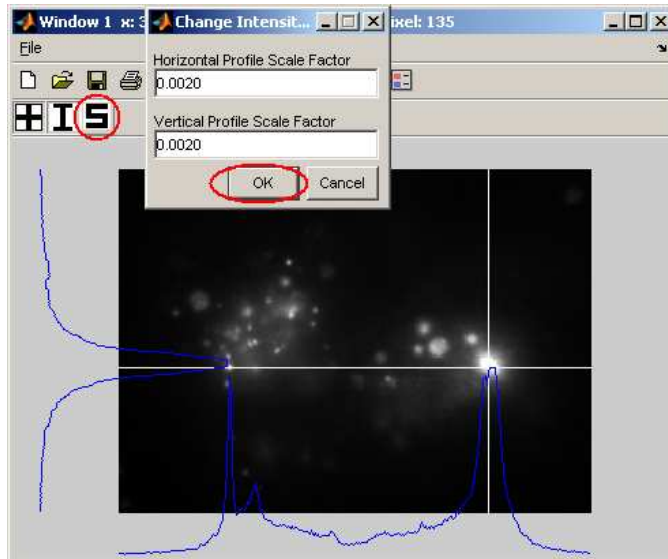
Figure 3.4.13: Manually scaled pixel intensity profiles

Step 19: *Deactivate crosshair and pixel intensity profiles* Remove the pixel intensity profiles by clicking their toolbar button (black 'I' in white background) again. Remove the crosshair by clicking its toolbar button (black cross in white background) again. (Alternatively, if you want to remove both the profiles and the crosshair in one go, clicking just the crosshair toolbar button will do the trick.)

3.4.2   Viewing images as mesh or contour plots

The MIATool viewer supports three different display modes for images: the standard mode which we have seen so far, the mesh mode, and the contour mode.

Step 20: *View images as mesh plots* To view the images as mesh plots, select **Mesh** from the **Display** drop-down menu of the figure containing the viewer sliders. You should find that the current image is now displayed as a mesh plot. Scroll the viewer sliders to see the other images displayed as mesh plots.

- The crosshair and the pixel intensity profiles are supported only in the standard mode. Notice that in the mesh mode, the crosshair, intensity profile, and profile scaling toolbar buttons are all disabled.
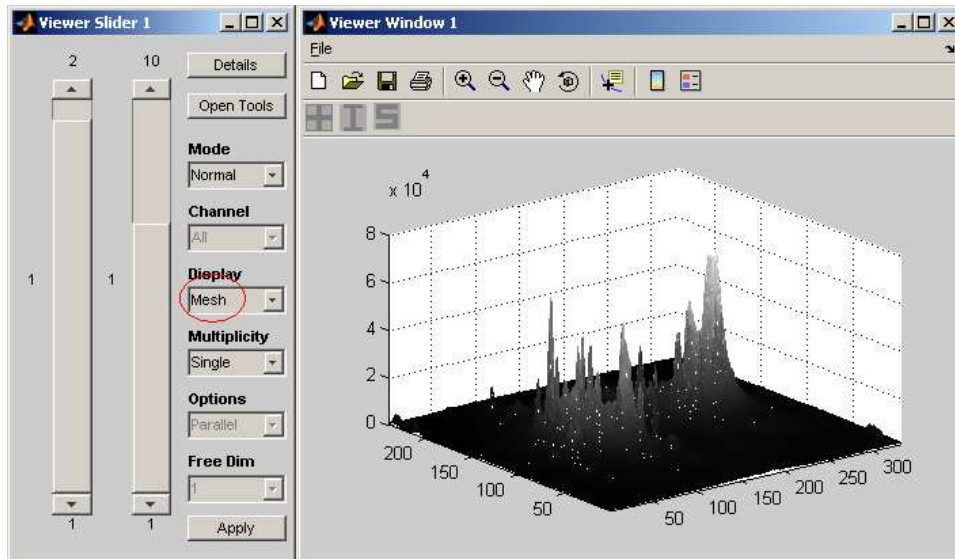
Figure 3.4.14: MIATool viewer in mesh display mode

Step 21: *View images as contour plots* Similarly, to view the images as contour plots, select **Contour** from the **Display** drop-down menu of the figure containing the viewer sliders. You should find that the current image is now displayed as a contour plot. Scroll the viewer sliders to see the other images displayed as contour plots.
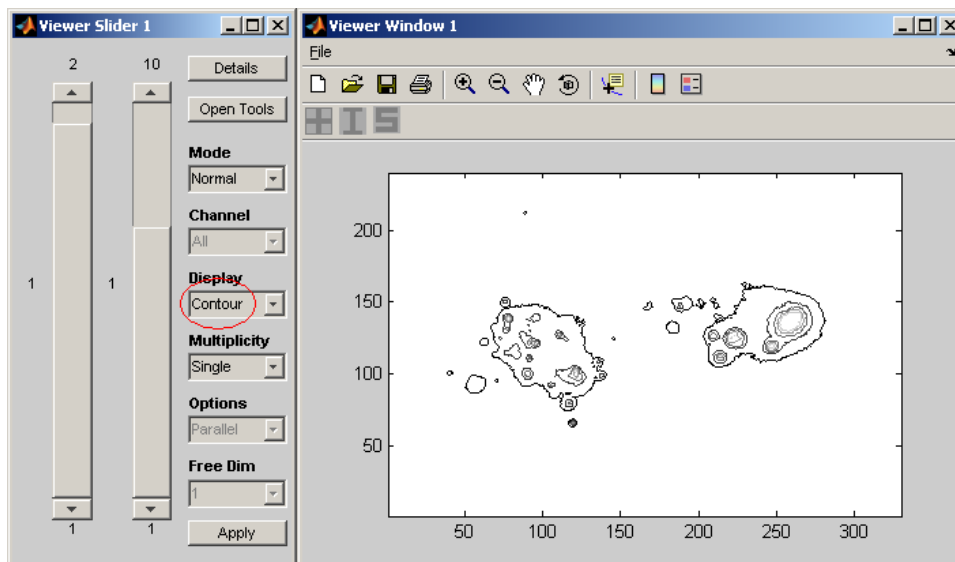
Figure 3.4.15: MIATool viewer in contour display mode

Step 22: *Return viewer to standard display mode* Return the viewer to standard display mode by selecting **Standard** from the **Display** drop-down menu of the figure containing the viewer sliders.

3.4.3 Parallel and overlay viewing

The MIATool viewer supports the concurrent viewing of multiple images, either in parallel in separate windows, or as an RGB overlay.

Step 23: *View images in parallel viewing mode* Suppose we want to simultaneously view in separate windows the correspondingly numbered images of series 1 and series 2. That is, we wish to view image 1 of series 1 together with image 1 of series 2, image 2 of series 1 together with image 2 of series 2, and so on and so forth. To do so, first switch the viewer to multiple viewing mode by selecting **Multiple** from the **Multiplicity** drop-down menu of the figure containing the viewer sliders. This activates the **Options** and **Free Dim** drop-down menus in the same figure, which apply only to the multiple viewing mode. Make sure that **Parallel** is selected in the **Options** drop-down menu, and that **1** is selected in the **Free Dim** drop-down menu, both of which should already be the case. You should now see two images, one from series 1 and one from series 2, displayed in separate windows. (At first one window will sit directly on top of the other, so simply drag them apart and position them side by side.) Use the second viewer slider to scroll through the 10 pairs of images.

- The **Parallel** option in the **Options** drop-down menu specifies that multiple images should be displayed simultaneously, but separately in different windows.

- **Free Dim** stands for 'free dimension'. Its value corresponds to the viewer slider that is 'freed'. A value of 1 means the first viewer slider is 'freed' (notice it is disabled in Figure 3.4.16), meaning that it takes on all possible values. In the present example, this means that for any image number specified by the second viewer slider, the image identified by that number in series 1 and the same in series 2 are displayed in parallel.
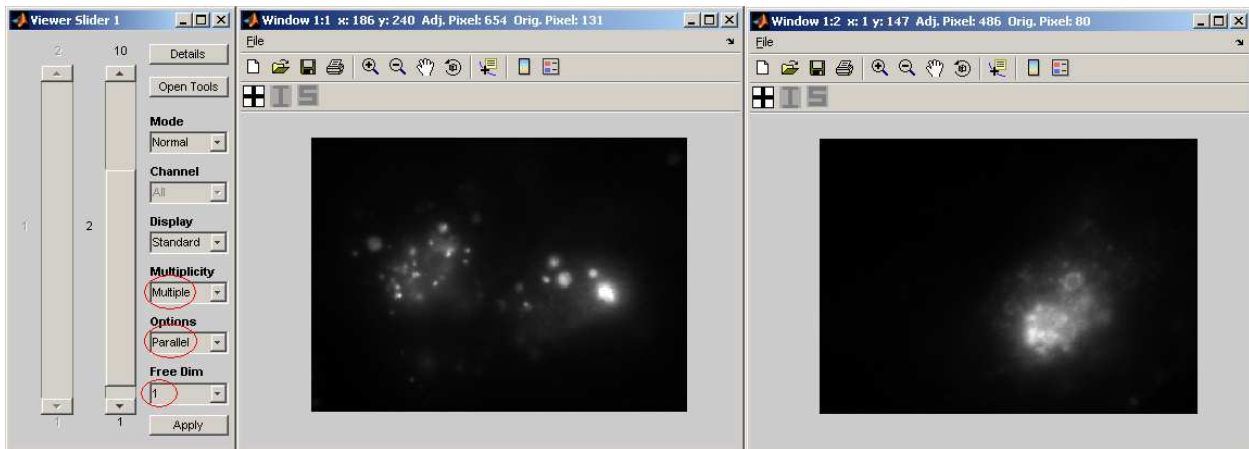
Figure 3.4.16: MIATool viewer in parallel viewing mode

Step 24: *Change active image in parallel viewing mode* In multiple viewing mode, even though more than one image is displayed at a time, only one of the displayed images is the active image. The active image is the image that the details report and tools like the intensity adjustment

tool act on. The active image is specified by the current values of the viewer sliders. For example, the state of the viewer shown in Figure 3.4.16 indicates that image 2 of series 1 is the active image. (The value 1 of the first viewer slider may be grayed out, but it is still used to convey the active image.) We can change the active image by simply clicking inside the image we want to make active. Alternately click inside the two currently displayed images and see for yourself that the value of the first viewer slider changes back and forth between 1 and 2.

- Try using the intensity adjustment tool and/or viewing the details report in parallel viewing mode. Change the active image a couple of times and convince yourself that at any given time the intensity adjustment tool and/or details report apply to the active image only.

Step 25: *View images in overlay viewing mode* Suppose we want to view the correspondingly numbered images of series 1 and series 2 as RGB overlays. To do so, with the viewer still in multiple viewing mode, select **Overlay** from the **Options** drop-down menu of the figure containing the viewer sliders. You should now see a single RGB overlay with the image from series 1 in the red channel and the image from series 2 in the green channel. Use the second viewer slider to scroll through the 10 RGB overlays.

- The **Overlay** option in the **Options** drop-down menu specifies that multiple images should be displayed simultaneously in different channels of an RGB image.

- The overlay viewing mode will work properly only with grayscale images. MIATool can display RGB images, but only in single viewing mode or parallel viewing mode.

- In overlay viewing mode, the 'freed' slider must be able to take on no more than three values. This is because we are limited to three channels, and hence three grayscale images, in creating an RGB overlay.

- As shown in Figure 3.4.17, in overlay viewing mode the pixel intensity values of all three channels of the RGB overlay are displayed in the title bar of the image window.
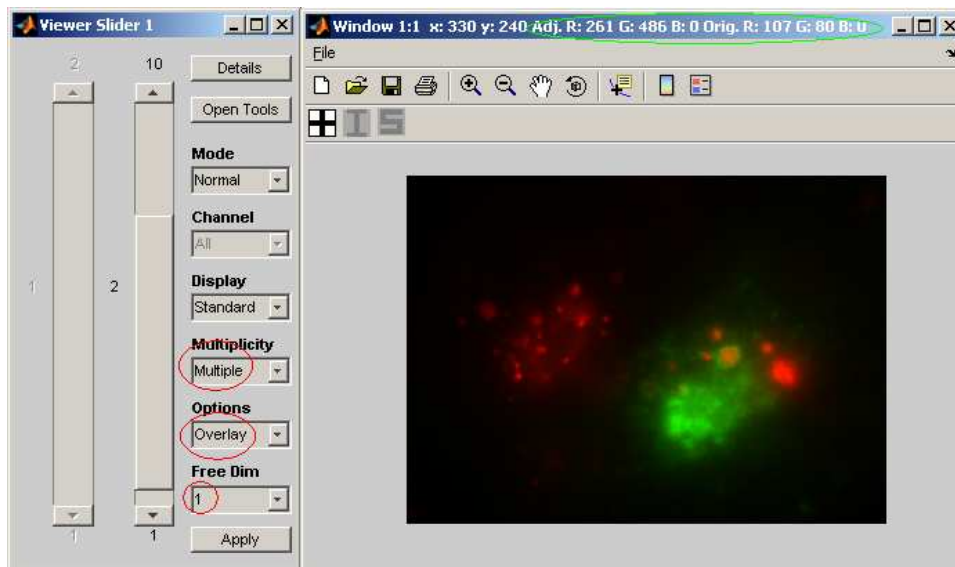
Figure 3.4.17: MIATool viewer in overlay viewing mode

Step 26: *Change active image in overlay viewing mode* In overlay viewing mode, only one of the RGB channels is the active image. For example, the state of the viewer shown in Figure 3.4.17 indicates that image 2 of series 1 (i.e., the red channel) is the active image. We can change the active image by right-clicking the figure margin around the overlay image and choosing a value for the 'freed' viewer slider from the context menu that pops up. To make the image from series 1 (i.e., the red channel) active, right-click the margin of the image window and select **Make 1 Current** from the context menu that appears. To make the image from series 2 (i.e., the green channel) active, do the same thing but select **Make 2 Current** from the context menu. Change the active image a few times and see for yourself that the value of the first viewer slider changes back and forth between 1 and 2, depending on which image you make active.

- Try using the intensity adjustment tool and/or viewing the details report in overlay viewing mode. Change the active image a couple of times and convince yourself that at any given time the intensity adjustment tool and/or details report apply to the active image only.

33
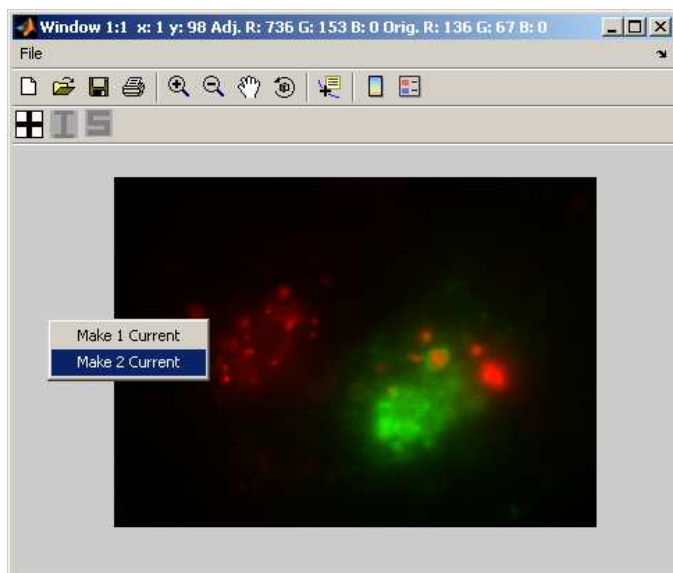
Figure 3.4.18: Changing active image in overlay viewing mode

Step 27: ***View images in all viewing mode*** In multiple viewing mode, there exists another viewing option called **All** in the **Options** drop-down menu. It is a combination of the parallel and overlay viewing modes. In our present example, this means that you will see three image windows: two with grayscale images exactly as in the parallel viewing mode, and one with an RGB overlay of the two grayscale images. Select the **All** option from the **Options** drop-down menu and see for yourself. Use the second viewer slider to scroll through the 10 pairs of images. In the all viewing mode, you can change the active image the same way you do it in the parallel viewing mode.

- The mesh and contour display modes also work in multiple viewing mode. The only exception is in the display of an RGB overlay, which is always displayed in the standard mode regardless of whether you've selected mesh or contour mode. Play with it and see for yourself!

Step 28: ***Close MIATool viewer and main menu*** We are done with this portion of the tutorial. Close the MIATool viewer by clicking the close (×) button in the upper right-hand corner of the figure containing the viewer sliders. Note that this will close everything except the MIATool main menu figure. If you like, you can close the main menu figure as well by clicking its close (×) button.

3.5 Image cropping

In this section, we demonstrate MIATool's crop tool using the two series of grayscale images in the MIATool directory created in Section 3.2.

Step 1: ***Open MIATool main menu and load MIATool directory*** Make sure that you have the MIATool main menu figure opened and the MIATool directory 'C:\miatooldemo1' loaded.

34

- Refer to Section 3.1 for help on opening the main menu figure, and Section 3.3 for help on loading the MIATool directory.

Step 2: *Initiate opening of MIATool viewer* Select **MIATool → Open DisplayTool**.

Step 3: *Choose ImageSet object to view and crop* Select the ImageSet object 'original_imageset.mia'. Click **OK**.

Step 4: *Choose a default CropSet object* Select 'CropSet' from the 'Object Type' drop-down menu. Highlight 'default CropSet' (the only option) in the 'Available Objects' list box and click the **Add** button to add it to the 'Selected Objects' list box. (See Figure 3.5.1, left panel.)

- To be able to crop the images you are viewing, you must select a CropSet object at this step.

- A 'CropSet' object is a container of crop settings for the images referred to by an ImageSet object. In our present case where there are a total of 20 images, the CropSet object will be automatically configured to store the crop settings for 20 images.

- A **default** CropSet object is one such that the crop settings it contains do not alter the images they act on. In other words, it does not crop the images.

Step 5: *Choose previously saved DisplaySet object* Select 'DisplaySet' from the 'Object Type' drop-down menu. Here we want to see the images using the intensity settings we came up with and saved to the MIATool directory in Section 3.4. Therefore, highlight 'autoadjust_displayset.mia' in the 'Available Objects' list box and click the **Add** button to add it to the 'Selected Objects' list box. Click **Done**. (See Figure 3.5.1, middle panel.)

- We could have selected the DisplaySet object first, then the CropSet object. However, for viewing efficiency, it is typically better to select the CropSet object first.

Step 6: *Decline LabelSet and SegmentationSet object selection* In the next figure that pops up, simply click **Done** since we won't be doing any image labeling or segmentation in this part of the tutorial. (See Figure 3.5.1, right panel.)
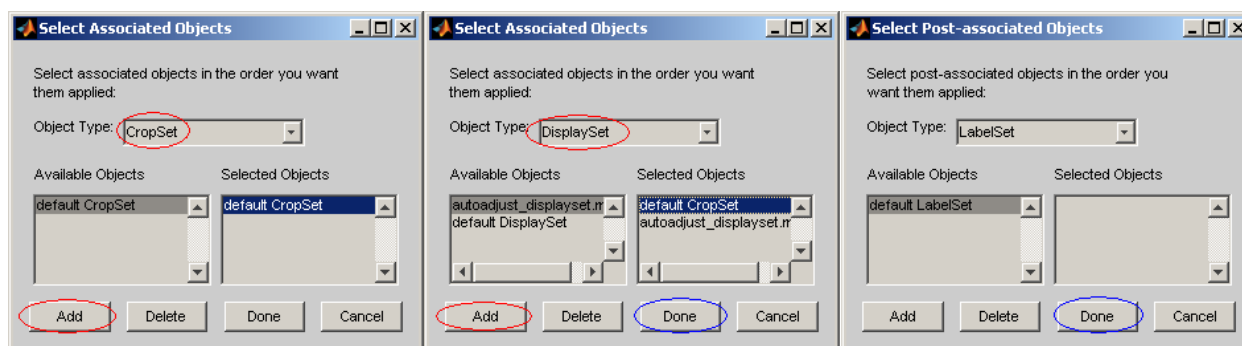
Figure 3.5.1: Opening image viewer with default CropSet and existing DisplaySet

Step 7: *Scroll viewer sliders and view images* Scroll through the images and see for yourself that they are intensity-adjusted exactly as before. This is because we are viewing with the DisplaySet object we saved previously in Section 3.4.

Step 8: *Open crop tool* Now let's try to do some cropping. Click the **Open Tools** button in the figure with the viewer sliders (see Figure 3.5.2, left panel). In the list box that pops up, select '1: cropset'. Click **OK** (see Figure 3.5.2, middle panel). The next figure that appears is the crop tool. For a default CropSet object, you will find that the initial crop settings are such that the image is uncropped. That is, the **Upper Left Pixel X** and **Upper Left Pixel Y** text boxes will each have the value 1, and the **Crop Height** and **Crop Width** values will correspond exactly to the height and width in pixels of the image (see Figure 3.5.2, right panel). Taken together, the four values describe a rectangular region that encompasses the image exactly, and hence the image is essentially uncropped.
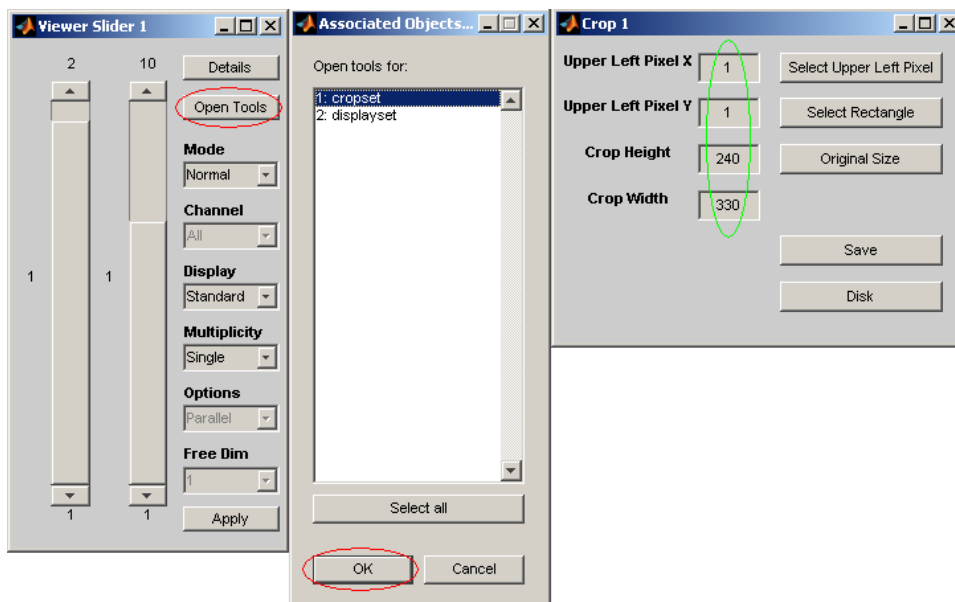
Figure 3.5.2: Opening crop tool

Step 9: *Crop image by directly specifying crop settings* The four text boxes **Upper Left Pixel X**, **Upper Left Pixel Y**, **Crop Height** and **Crop Width** together indicate the rectangular region (in units of pixels) from the original image that is currently being displayed in the image window. If you know the precise rectangular region that you want to extract from the original image, then the best way to crop the image is to directly enter values into these four text boxes. For example, suppose from image 1 of series 1 we want the height 170 by width 290 rectangular region with the upper left corner pixel at X = 20 and Y = 10. First make sure that the currently displayed image is image 1 of series 1, and then do the following **in the order given**: enter **170** in the **Crop Height** text box, enter **290** in the **Crop Width** text box, enter **20** in the **Upper Left Pixel X** text box, and enter **10** in the **Upper Left Pixel Y** text box. Note that as you enter each number, you can actually see the displayed image getting cropped. After you enter all four numbers, the image displayed is the desired result.

- The order in which you enter the numbers into the text boxes is important because at any time the current values of the four text boxes must describe a valid rectangular region inside the original image. If you had, for example, entered the X value of 20 first, then because the crop width was at the maximum value of 330 (i.e., the width of the original image), the resulting rectangular region would have exceeded the boundaries of the original image, and you would have received a warning from the crop tool.
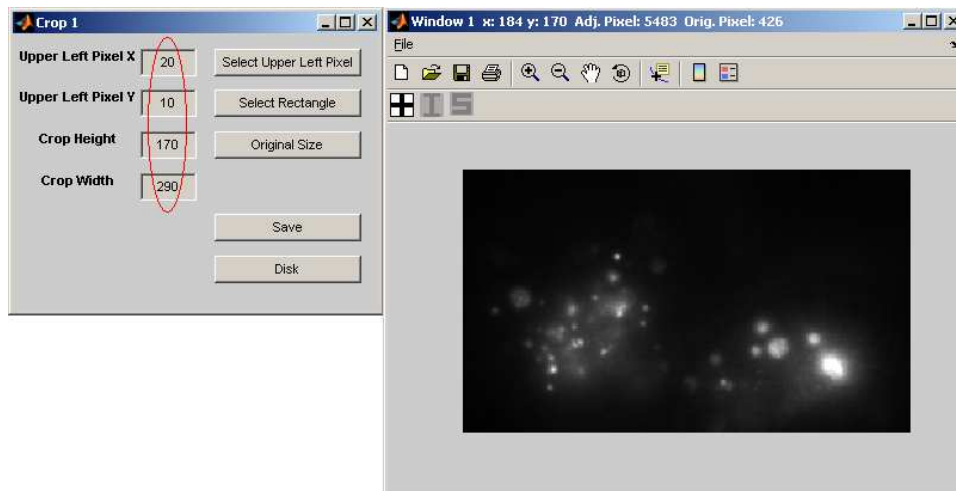
Figure 3.5.3: Exact cropping using text boxes

Step 10: *Verify crop settings are not saved* As in the case of pixel intensity adjustments, your crop settings are lost unless you save them. Change the current image to another image and then come back to image 1 of series 1. You will find the crop settings gone and the full image displayed.

Step 11: *Crop image by interactively specifying upper left corner of crop region* Another way to crop an image is by using the **Select Upper Left Pixel** button in the crop tool figure. This button will allow you to interactively pick the upper left corner of the rectangular region, and then automatically take the lower right corner of the displayed image as the lower right corner of the rectangular region. Perform the following:

A. Click the **Select Upper Left Pixel** button in the crop tool figure.

B. A box shows up that asks you to decide on the upper left pixel of the rectangular region. It suggests that you can zoom in on the image (see Section 3.14 for more on zooming) if you need to pick a pixel precisely, but asks that you deactivate the zooming before proceeding. We will not do any zooming, so just click **OK**.

C. Another box appears to tell you to select the upper left pixel and to press the <return> key once you are done. Click **OK**.

D. Click inside the displayed image at a position close to where the yellow circle is in Figure 3.5.4, panel D. **Click only ONCE!** After clicking you will see a star at the position you clicked. Press the <**return**> key.

E. The result of the cropping is now displayed in the image window.
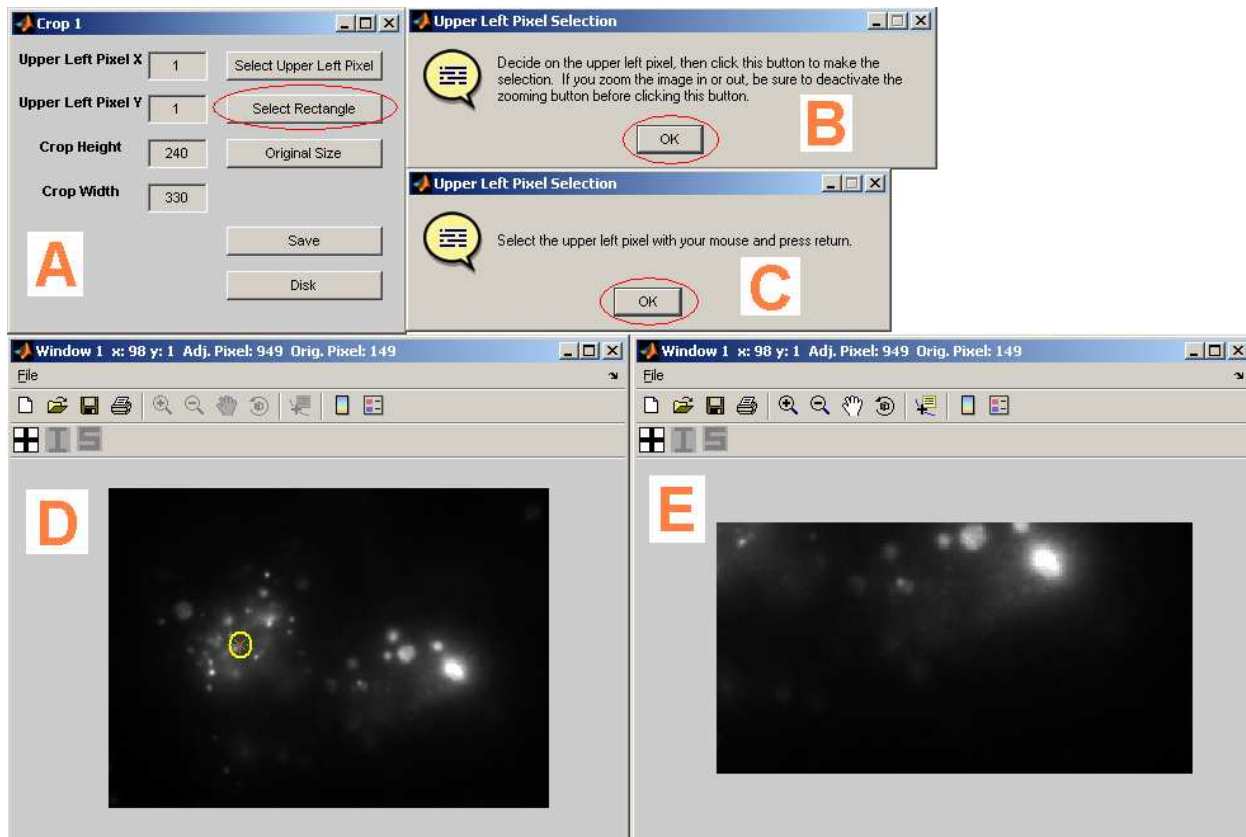
Figure 3.5.4: Cropping using the Select Upper Left Pixel button

Step 12: *Restore original image size* Before we look at the final method for cropping an image, let's first undo the cropping we just did. To do so, simply click the **Original Size** button in the crop tool figure. The displayed image should now be the original image. (Since we have not saved the crop settings, you can, of course, achieve the same thing by, again, changing the current image and then coming back to it.)
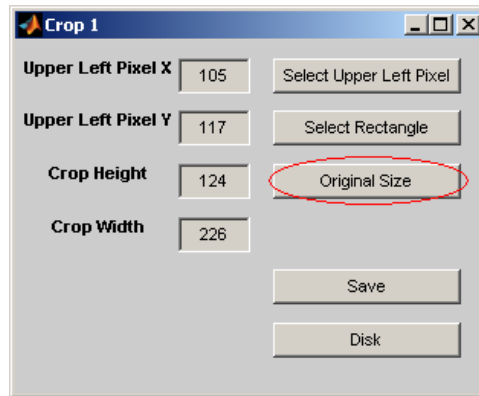
Figure 3.5.5: Restoring the original image size

Step 13: *Crop image by dragging rectangular region with mouse* A third way to crop an image is to use the **Select Rectangle** button in the crop tool figure. This button will allow you to interactively select the rectangular region by dragging your mouse. Perform the following:

A. Click the **Select Rectangle** button in the crop tool figure.

B. A box shows up that asks you to deactivate zooming before proceeding (see Section 3.14 for more on zooming). We are not doing any zooming, so just click **OK**.

C. Another box appears to tell you to go ahead and drag out a rectangle on the image. Click **OK**.

D. Drag a rectangle on the displayed image similar to that shown in Figure 3.5.6, panel D.

E. The result of the cropping is now displayed in the image window.

F. To make sure what you see on your monitor agrees with the remainder of this part of the tutorial, enter the following numbers in the crop tool text boxes to ensure that you are looking at the exact same rectangular region: **Upper Left Pixel X** = 63, **Upper Left Pixel Y** = 57, **Crop Height** = 103, **Crop Width** = 214.
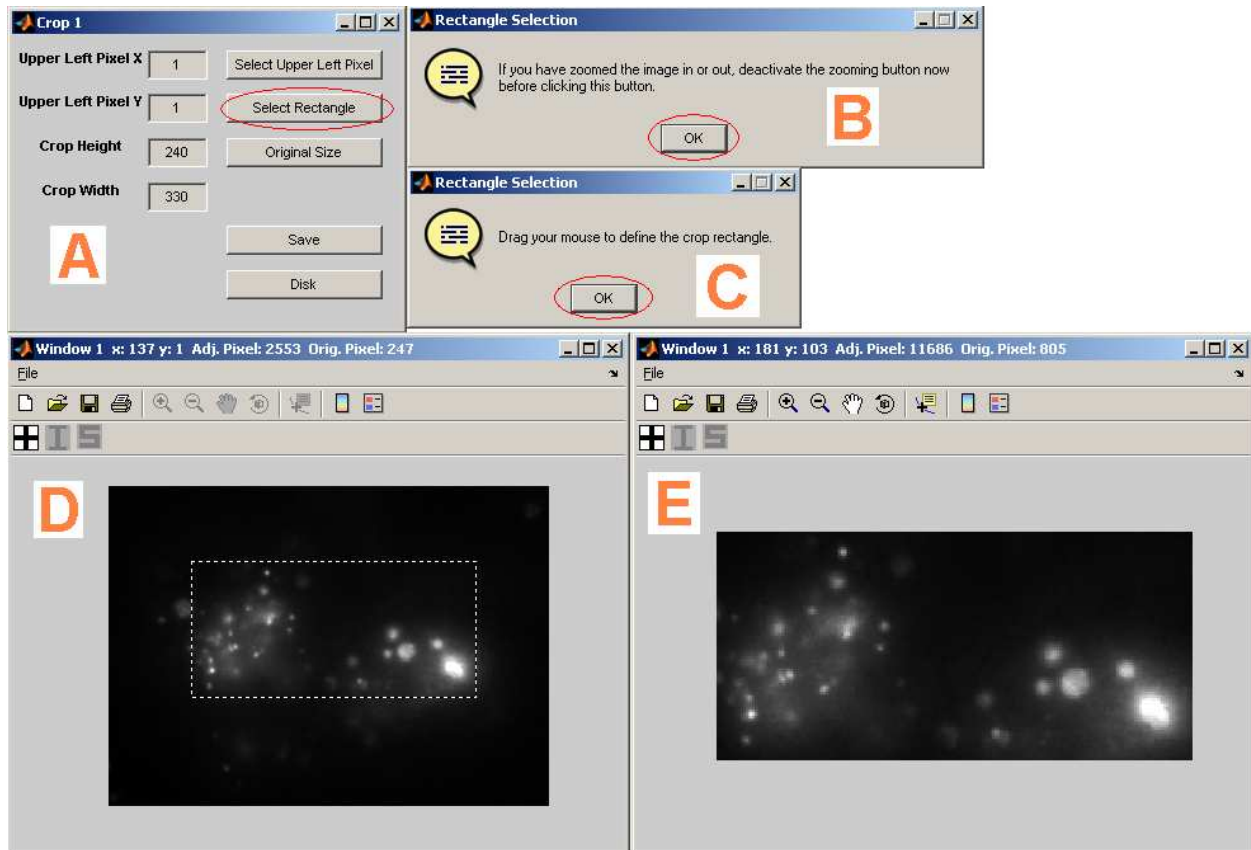
Figure 3.5.6: Cropping using the Select Rectangle button

Step 14: *Apply crop settings to all images* Let's save to memory the cropping we just did to all images. Click the **Save** button in the crop tool figure. In the list box that appears, select **Apply to All Images** and click **OK**. In the confirmation box that pops up, click **OK**. Scroll through all 20 images to see all of them cropped.

- The **Save** button of the crop tool supports exactly the same options as the **Save** button of the intensity adjustment tool. Refer to Step 13 of Section 3.4 for more about the **Save** button.

- Caution must be used when applying crop settings for the current image to other images. Since in general images can be of different dimensions, crop settings that work for the current image may not be valid for a different image.
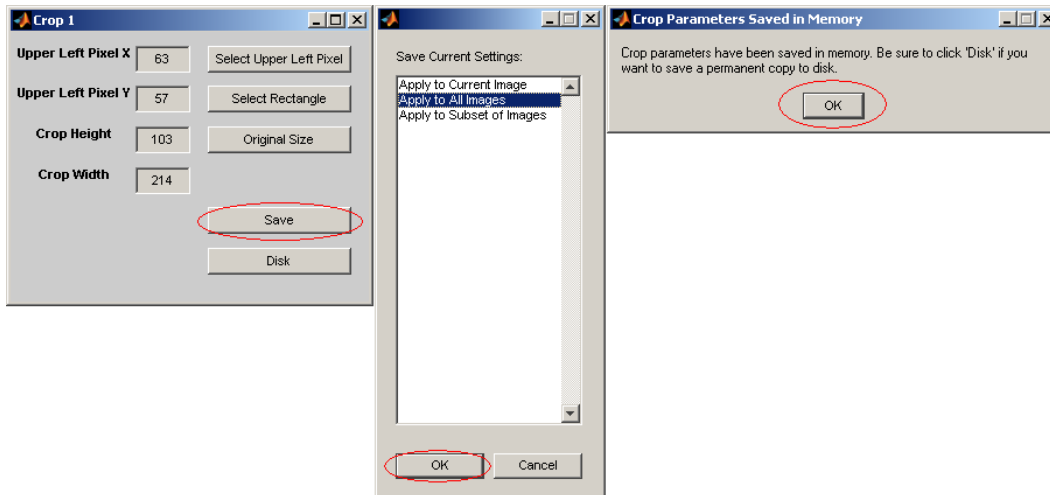
Figure 3.5.7: Applying the same crop settings to all images

Step 15: *Save crop settings to disk in a CropSet object* Let's save the crop settings to our MIATool directory on disk so that we can use them again at a later time. Click the **Disk** button in the crop tool figure. In the input box that appears, enter **small** for the file name of the CropSet object. Click **OK**. In the confirmation box that appears, click **OK**.

- You can specify any file name you like for the CropSet object. You can also leave the text box empty, in which case the name 'cropset1' will be used automatically. Just keep in mind that this tutorial assumes the name 'small' is specified.
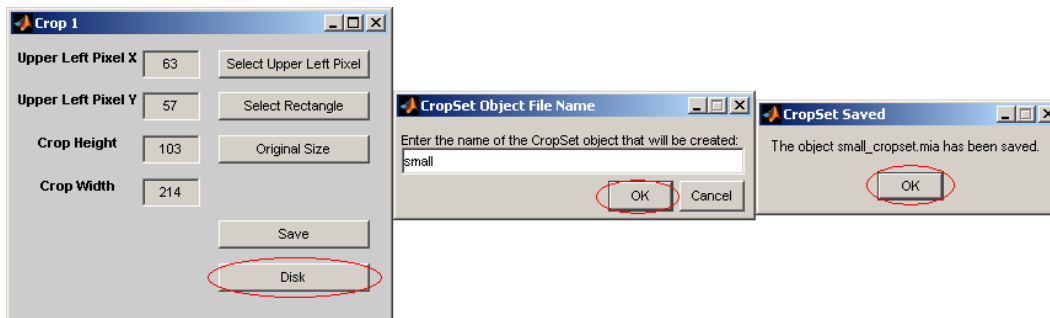
Figure 3.5.8: Saving crop settings to MIATool directory on disk

Step 16: *Verify CropSet object has been saved to disk* Let's verify that the CropSet object has been saved to our MIATool directory. Using Windows Explorer, navigate to 'C:\miatooldemo1\MIAToolObjects\ImageSet1'. There you should find a subfolder called 'cropset objects'. Inside that subfolder, you should see the CropSet object 'small_cropset.mia'. The file name 'small' is what we specified in the previous step, and '_cropset.mia' is the standard extension used by MIATool to identify CropSet objects.

- If you didn't choose 'C:\miatooldemo1' as the root folder of your MIATool directory, then

41

make sure that you look in your root folder instead.
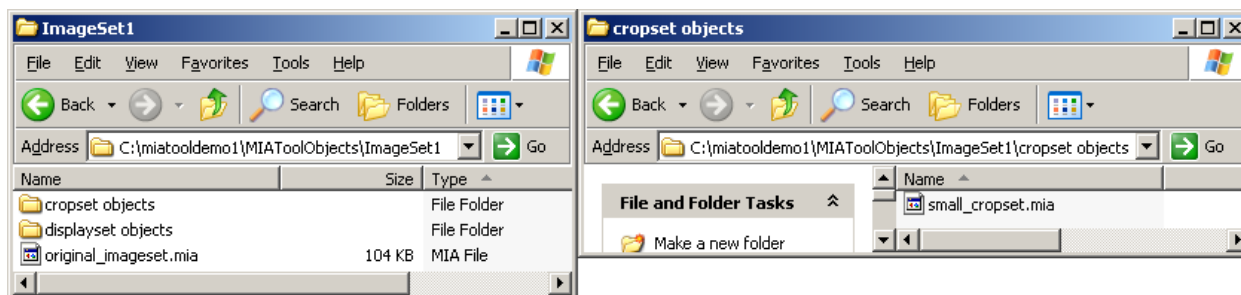
Figure 3.5.9: CropSet object saved to MIATool directory

Step 17: *View cropped images in overlay viewing mode* Switch the MIATool viewer to overlay viewing mode by selecting **Multiple** from the **Multiplicity** drop-down menu and **Overlay** from the **Options** drop-down menu. Make sure the **Free Dim** drop-down menu has the value **1** selected. The displayed image should now be an RGB overlay of cropped images. Scroll the second viewer slider to see all 10 overlays.
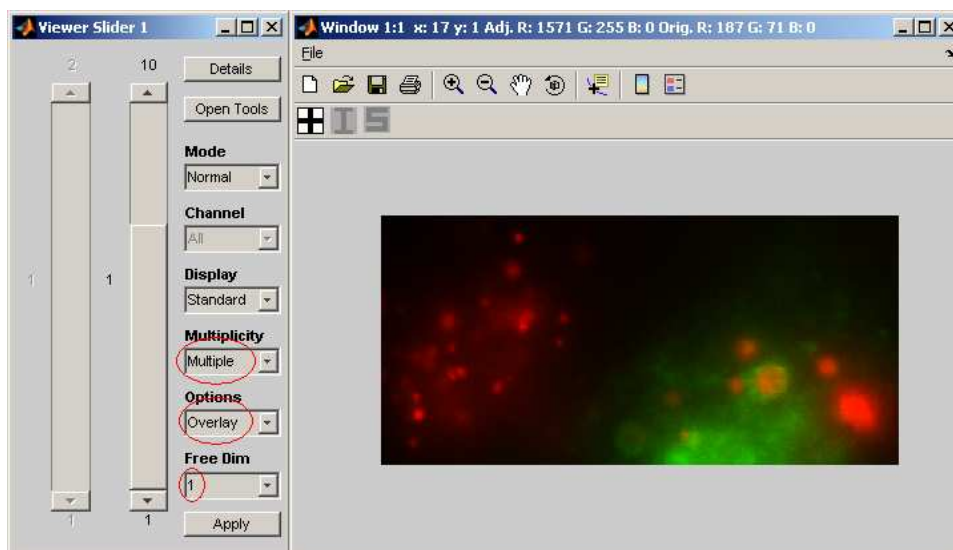
Figure 3.5.10: Overlay of cropped images

Step 18: *Change crop settings while in overlay viewing mode* From the overlays we notice that the object in the green channel is quite a bit too low and needs to be moved up to achieve a better overlay with the objects in the red channel. We want to do the moving in overlay mode so that the red channel can be used as a reference. Perform the following.

  A. Right-click the margin around the image and select **Make 2 Current** from the context menu to make the green channel the active image.

B. Move the green object up by entering the value **95** in the **Upper Left Pixel Y** text box of the crop tool figure. The displayed RGB overlay should be updated immediately with the green object moved up.

C. Click the **Save** button in the crop tool figure.

D. Select **Apply to Subset of Images** and click **OK**.

E. In the subset selection figure, enter the value **2** (for series 2) in the left text box and the word **all** in the right text box (for all images). Click **Done**. This applies the new crop settings to all images of series 2.

F. Click **OK** in the confirmation box.

G. Scroll the second viewer slider to see the green object moved up in all 10 overlays.
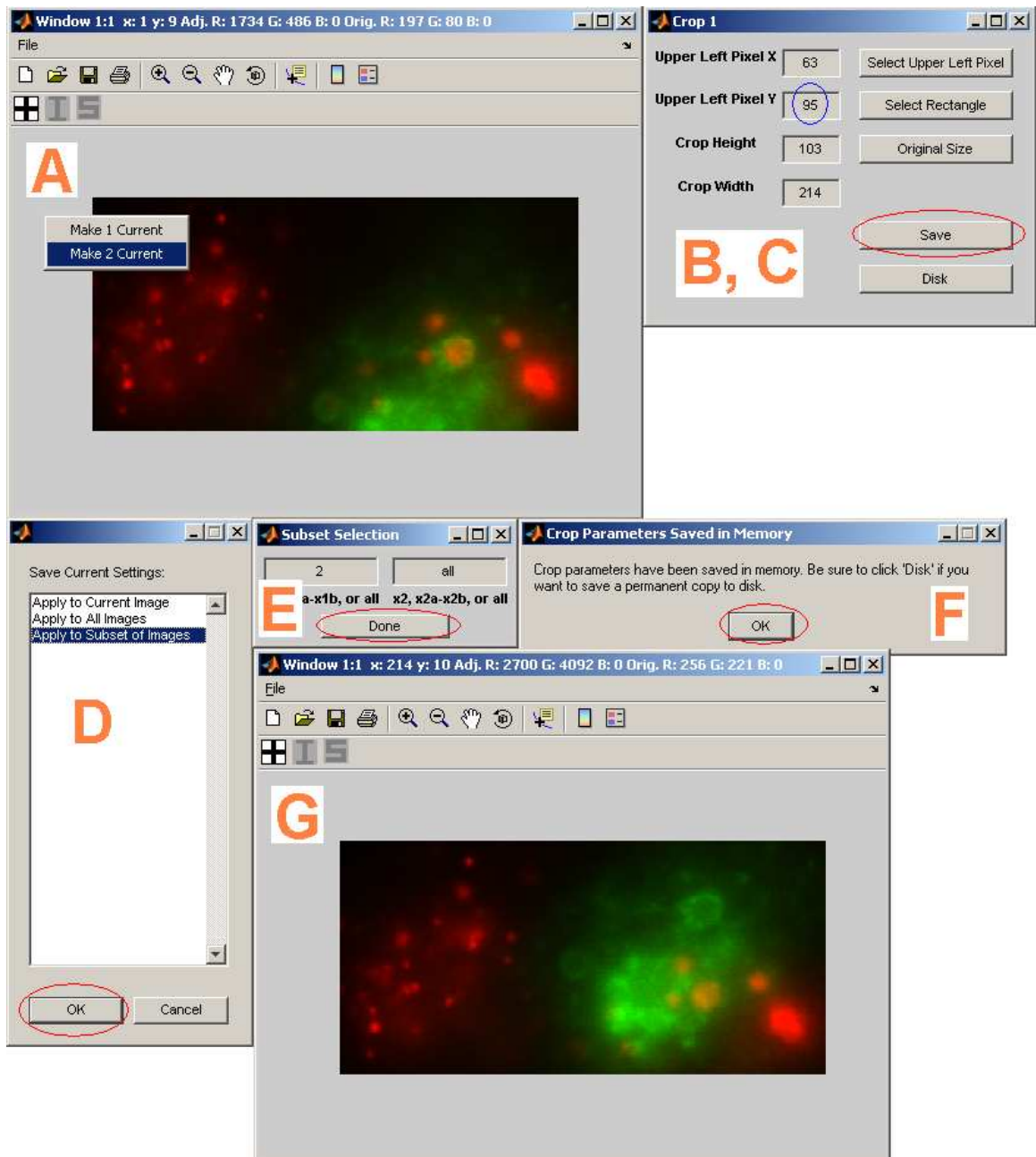
Figure 3.5.11: Cropping while in overlay viewing mode

Step 19: *Save crop settings to disk by overwriting existing CropSet object* Let's save the new crop settings to disk. Click the **Disk** button in the crop tool figure. Because we had saved the previous crop settings under the file name 'small', a figure now pops up asking whether we want to overwrite 'small' with the new settings (**Save**), or save the new settings in a new CropSet object (**Save As**). Click **Save** to overwrite 'small' with the new crop settings. Click **OK** in the confirmation box.

• If the **Save As** button is clicked, MIATool will ask for a file name to save the new CropSet

object under. This new CropSet object will then become the new current CropSet object, and subsequent overwrite operations will be applied to it until another 'save as' introduces a new current CropSet object.

- The 'Save/Save As' mechanism works in the exact same way with the intensity adjustment tool and DisplaySet objects.
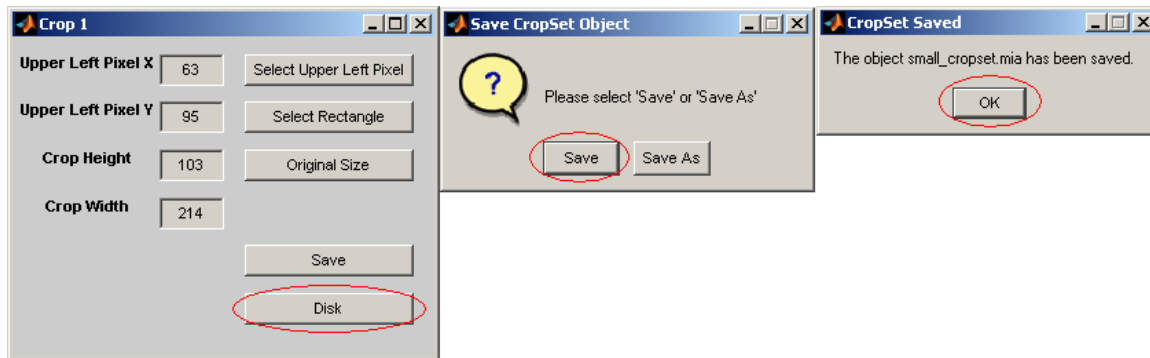
Figure 3.5.12: Overwriting existing CropSet object on disk with new crop settings

Step 20: _Close MIATool viewer and main menu_ We are done with this portion of the tutorial. Close the MIATool viewer by clicking the close (×) button in the upper right-hand corner of the figure containing the viewer sliders. You can close the main menu figure as well by clicking its close (×) button.

3.6   Writing processed images to disk

So far in this tutorial we have performed pixel intensity adjustments and cropping, and have been able to see the processed images in the MIATool viewer. We have also learned to save the intensity and crop settings in the form of DisplaySet and CropSet objects so that we can use the same settings for viewing at a later time. We have, however, not yet seen how we can generate actual processed images as output. In this section, we demonstrate how to use MIATool to write processed images to disk in the tiff format.

Step 1: _Open MIATool main menu and load MIATool directory_ Make sure that you have the MIATool main menu figure opened and the MIATool directory 'C:\miatooldemo1' loaded.

- Refer to Section 3.1 for help on opening the main menu figure, and Section 3.3 for help on loading the MIATool directory.

Step 2: _Initiate opening of MIATool viewer_ Select **MIATool → Open DisplayTool**.

Step 3: _Choose ImageSet object to create processed images from_ Select the ImageSet object 'original_imageset.mia'. Click **OK**.

45

Step 4: *Choose previously saved CropSet object* Select 'CropSet' from the 'Object Type' drop-down menu. Highlight 'small_cropset.mia' in the 'Available Objects' list box and click the **Add** button to add it to the 'Selected Objects' list box. (See Figure 3.6.1, left panel.)

Step 5: *Choose previously saved DisplaySet object* Select 'DisplaySet' from the 'Object Type' drop-down menu. Highlight 'autoadjust_displayset.mia' in the 'Available Objects' list box and click the **Add** button to add it to the 'Selected Objects' list box. Click **Done**. (See Figure 3.6.1, middle panel.)

Step 6: *Decline LabelSet and SegmentationSet object selection* In the next figure that pops up, simply click **Done** since we won't be doing any image labeling or segmentation in this part of the tutorial. (See Figure 3.6.1, right panel.)
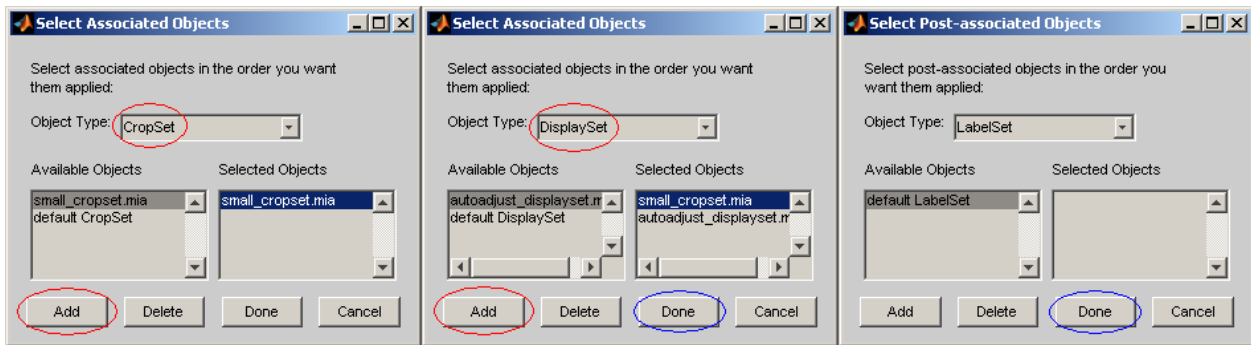
Figure 3.6.1: Opening image viewer with existing CropSet and existing DisplaySet

Step 7: *Scroll viewer sliders and view images* Scroll through the images and see for yourself that they are intensity-adjusted and cropped exactly as before. This is because we are viewing with the DisplaySet object we saved previously in Section 3.4 and the CropSet object we saved previously in Section 3.5.

Step 8: *Write processed images to disk* To write the intensity-adjusted and cropped images exactly as you see them in the viewer to disk in tiff format, perform the following. (The procedure is similar to the import process of Section 3.2.)

A. Click the **Apply** button in the figure containing the viewer sliders.

B. In the file browser, select the root of the C drive as the location to create the output MIATool directory under which the processed images will reside. Click **OK**.

C. Specify **miatooldemo2** as the name of the root folder of the output MIATool directory. Click **OK**.

D. Suppose we want the first output series of images to be written as the frames of a multi-frame tiff file. Select **multi-frame tif** from the **Series Type** drop-down menu. Enter **ic** (for **i**ntensity-adjusted and **c**ropped) in the optional **File Name Prefix** text box. Click **OK**.

E. Suppose we want the second output series of images to be written as individual tiff files. Select **n.tif** from the **Series Type** drop-down menu. Leave the optional **File Name Prefix** text box empty. Click **OK**.

F. We are asked to give the file name for the default ImageSet object that will be created for the processed images. Leave the text box empty to accept the default file name 'imageset1'. Click **OK**.

G. A confirmation for the creation of the output MIATool directory appears. Click **OK**.

- As always you can specify a different location to create the MIATool directory and a different name for the root folder of the MIATool directory. However, the remainder of this tutorial will assume this MIATool directory to be located at 'C:\miatooldemo2'.

- The **Apply** button will write to disk images processed using the intensity and crop settings last saved to memory using the **Save** buttons of the intensity adjustment and crop tools. It does not matter whether the settings have been saved to disk.
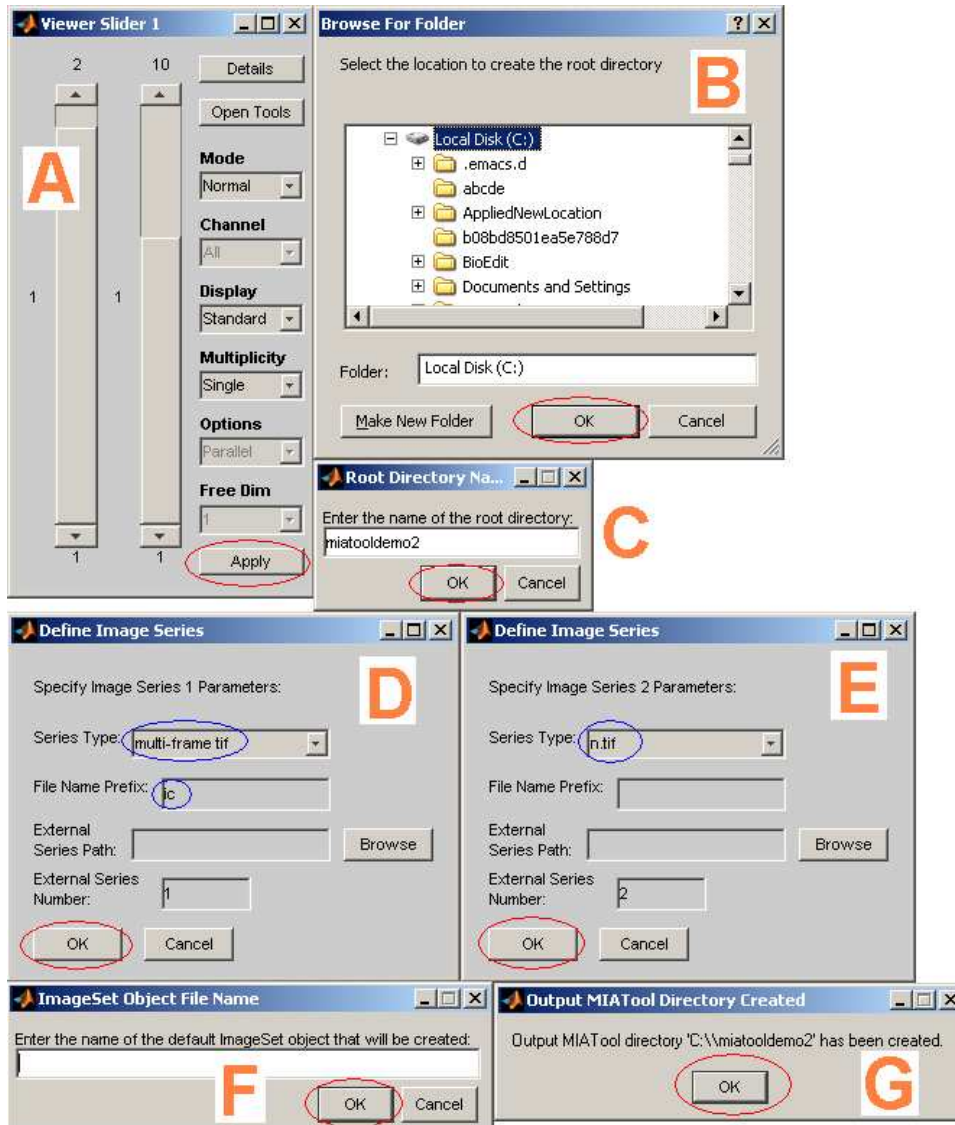
Figure 3.6.2: Creating intensity-adjusted and cropped tiff images on disk

Step 9: *Close MIATool viewer* Close the MIATool viewer by clicking the close (✕) button in the upper right-hand corner of the figure containing the viewer sliders.

Step 10: *Look at MIATool directory containing processed images* Use Windows Explorer to verify what has been created on disk.

A. Verify that the MIATool directory 'C:\miatooldemo2' has been created, and that it contains the subfolders 'ImageFiles' and 'MIAToolObjects' as well as the file 'miatooldir_miatooldirectory.mia', which needs to be present in all MIATool directories.

B. Verify that the 'ImageFiles' subfolder contains the two subfolders 'Series1' and 'Series2'. Inside the 'Series1' subfolder you should find the multi-frame tiff file 'ic_m1.tif'. The prefix 'ic' is what we specified for output series 1, and the 'm1' stands for '**m**ulti-frame tiff, series

48

1'. Inside the 'Series2' subfolder, you should find 10 individual tiff images with file names 's2_1.tif', 's2_2.tif', ..., 's2_10.tif'. The 's2' stands for 'single tiff, series **2**', and there is no prefix because we did not specify one for output series 2.

C. Verify that the 'MIAToolObjects' subfolder contains the subfolder 'ImageSet1', which in turn contains the default ImageSet object 'imageset1_imageset.mia'. The file name 'imageset1' was automatically assigned by MIATool because we did not provide one.
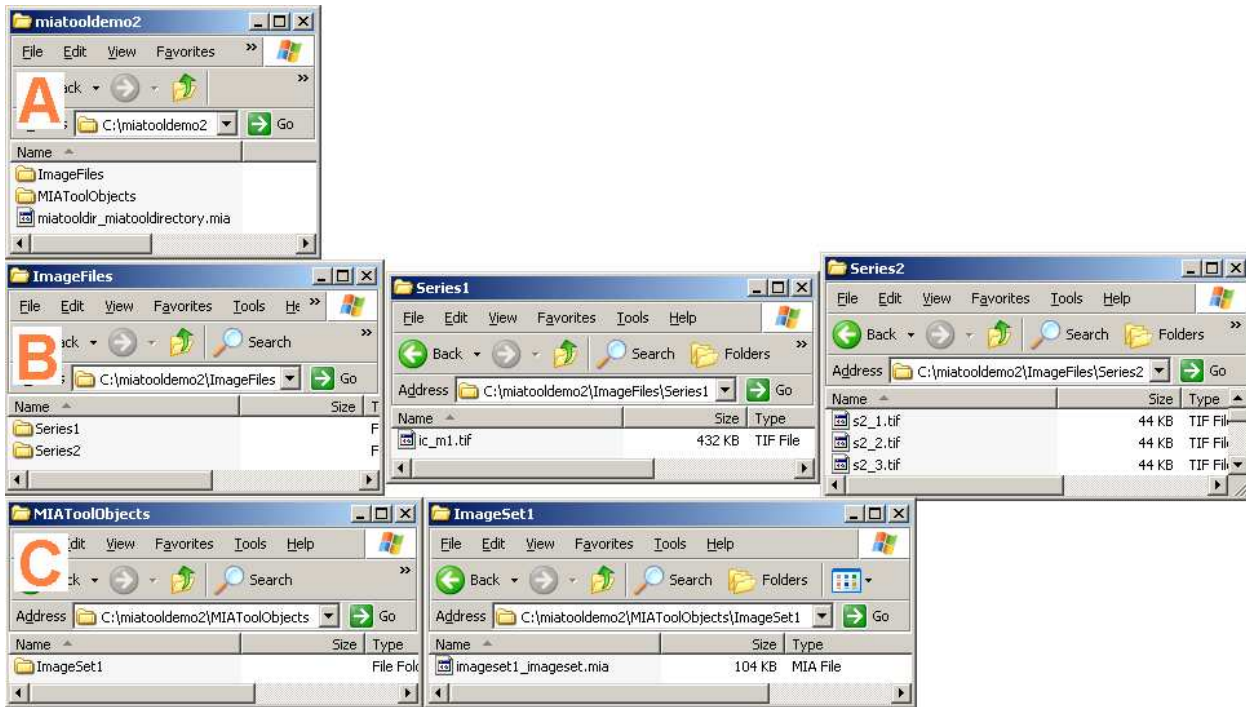
Figure 3.6.3: MIATool directory containing processed images

Step 11: *Verify processed images by viewing them using MIATool viewer* Verify that the images in the output MIATool directory are processed (i.e., intensity-adjusted and cropped) images by doing the following.

A. Select **MIATool** → **Load MIATool Directory** from the MIATool main menu figure. Using the directory browser that pops up, select the MIATool directory 'C:\miatooldemo2'. Click **OK**. When loading is finished, the message 'Loaded: C:\miatooldemo2' will be displayed in the main menu figure.

B. Select **MIATool** → **Open DisplayTool** from the MIATool main menu figure. In the list box that pops up, select the ImageSet object 'imageset1_imageset.mia' and click **OK**.

C. Because we are not looking to do any intensity adjustments or cropping, we do not need to select any CropSet or DisplaySet objects. Simply click **Done**.

D. We won't be doing any labeling or manual segmentation, so simply click **Done**.

49

E. Scroll through the 20 images and see for yourself that they are intensity-adjusted and cropped.
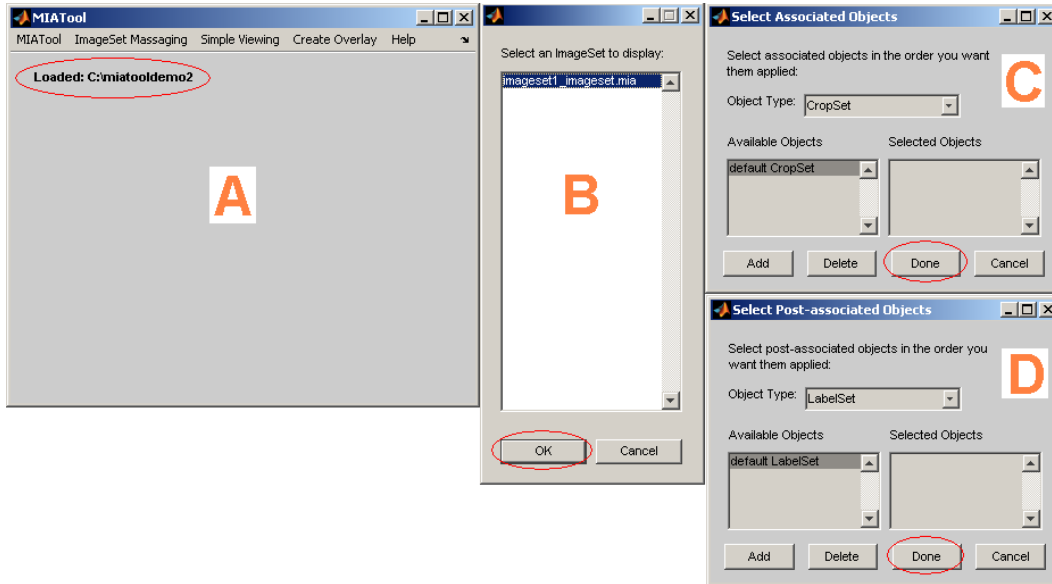
Figure 3.6.4: Opening MIATool viewer to view processed images

Step 12: *Close MIATool viewer and main menu* Close the MIATool viewer by clicking the close (×) button in the upper right-hand corner of the figure containing the viewer sliders. You can close the main menu figure as well by clicking its close (×) button.

3.7 Writing overlay images to disk

The 'Apply' button introduced in the previous section does not support the generation of overlay images on disk. Instead, MIATool provides a separate menu option for writing overlay images to disk. In this section, we demonstrate this menu option using the intensity-adjusted and cropped images created in the previous section. A subsection follows that uses the created overlay images to demonstrate the viewing and processing of RGB images in MIATool.

Step 1: *Open MIATool main menu and load MIATool directory* Make sure that you have the MIATool main menu figure opened and the MIATool directory 'C:\miatooldemo2' loaded. The MIATool directory 'C:\miatooldemo2' contains the intensity-adjusted and cropped images that we want to create overlays with.

- Refer to Section 3.1 for help on opening the main menu figure, and Section 3.3 for help on loading the MIATool directory.

Step 2: *Initiate writing of overlay images to disk* Select **Create Overlay** → **Create Overlay Images** from the MIATool main menu figure.

50

- The writing of overlay images to disk is independent of the MIATool viewer. All that's required is that the MIATool directory containing the images to overlay is loaded.

Step 3: *Choose ImageSet object with which to create overlay images* Select the default ImageSet object 'imageset1_imageset.mia' and click **OK**. The overlay images will be created from the two series of intensity-adjusted and cropped images referenced by this ImageSet object.
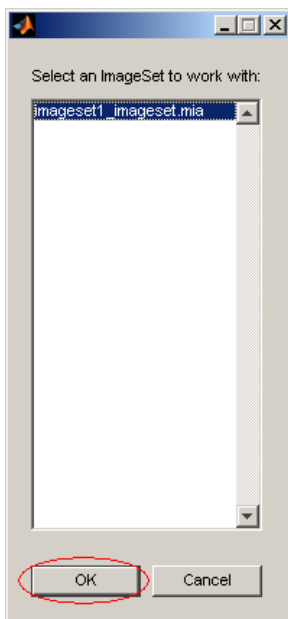
Figure 3.7.1: Selecting ImageSet object from which to generate overlays

Step 4: *Specify different overlay combinations of input image series* We can now overlay the two input series of intensity-adjusted and cropped images in many different combinations. Let's create five different overlays (and hence five output image series) using the two input image series.

A. For each output image series (i.e., each overlay combination) we can specify up to three different colors. For the first output image series, all we want to see is all 10 images of the first input series in blue. To do so, disable the red and green channels by clicking once each of the buttons Red and Green. When disabled, you will see their corresponding input boxes grayed out. Select the value **1** (for input series 1) from the **Series Number** drop-down menu of the blue channel. Since we want all 10 images of input series 1, the **Start Frame**, **End Frame**, and **Increment** text boxes of the blue channel should have the values **1**, **10**, and **1**, respectively. An increment value of 1 means to take every image starting with the start frame number and ending with the end frame number. Click **OK**.

B. For the second output image series, we want to see the odd-numbered images of the second input series in yellow. To do so, disable the blue channel by clicking the button Blue once. Select the value **2** (for input series 2) from the **Series Number** drop-down menus of the red and green channels. Since we want only the odd-numbered images of input series 2, the **Start Frame**, **End Frame**, and **Increment** text boxes of the red channel should have the values

51

1, **10**, and **2**, respectively. The same is true for the text boxes of the green channel. An increment value of 2 means to take every second image starting with the start frame number and ending with the end frame number. Click **OK**.

C. For the third output image series, we want to see the even-numbered images of the first input series in green and the even-numbered images of the second input series in red. To do so, disable the blue channel by clicking the button Blue once. For the red channel, select the value **2** (for input series 2) from the **Series Number** drop-down menu, and enter the values **2**, **10**, and **2** in the **Start Frame**, **End Frame**, and **Increment** text boxes, respectively. For the green channel, select the value **1** (for input series 1) from the **Series Number** drop-down menu, and enter the values **2**, **10**, and **2** in the **Start Frame**, **End Frame**, and **Increment** text boxes, respectively. Click **OK**.

D. For the fourth output image series, we want to see the odd-numbered images of the first input series in yellow and the even-numbered images of the second input series in blue. For the red channel, select the value **1** (for input series 1) from the **Series Number** drop-down menu, and enter the values **1**, **10**, and **2** in the **Start Frame**, **End Frame**, and **Increment** text boxes, respectively. For the green channel, select the value **1** (for input series 1) from the **Series Number** drop-down menu, and enter the values **1**, **10**, and **2** in the **Start Frame**, **End Frame**, and **Increment** text boxes, respectively. For the blue channel, select the value **2** (for input series 2) from the **Series Number** drop-down menu, and enter the values **2**, **10**, and **2** in the **Start Frame**, **End Frame**, and **Increment** text boxes, respectively. Click **OK**.

E. For the fifth output image series, we want to see the odd-numbered images of the first input series in turquoise and the even-numbered images of the first input series in red. For the red channel, select the value **1** (for input series 1) from the **Series Number** drop-down menu, and enter the values **2**, **10**, and **2** in the **Start Frame**, **End Frame**, and **Increment** text boxes, respectively. For the green channel, select the value **1** (for input series 1) from the **Series Number** drop-down menu, and enter the values **1**, **10**, and **2** in the **Start Frame**, **End Frame**, and **Increment** text boxes, respectively. For the blue channel, select the value **1** (for input series 1) from the **Series Number** drop-down menu, and enter the values **1**, **10**, and **2** in the **Start Frame**, **End Frame**, and **Increment** text boxes, respectively. Click **OK**.

F. We have specified the five different overlays we want. In the next input figure, click **Stop** to end the overlay specification process.

- In the overlay specification of an output image series, clicking the buttons Red , Green , and Blue will alternately enable and disable the respective color channels.

Figure 3.7.2: Specifying output series of overlay images

Step 5: *Specify output MIATool directory and output image series format* Perform the following to complete the creation of overlay images on disk.

A. In the file browser, select the root of the C drive as the location to create the output MIATool directory under which the overlay images will reside. Click **OK**.

B. Specify **miatooldemo3** as the name of the root folder of the output MIATool directory. Click **OK**.

C. Suppose we want the first output series of images to be written as the frames of a multi-frame tiff file. Select **multi-frame tif** from the **Series Type** drop-down menu. Enter **b1** (for **b**lue, input series **1**) in the optional **File Name Prefix** text box. Click **OK**.

D. Suppose we want the second output series of images to be written as individual tiff files. Select **n.tif** from the **Series Type** drop-down menu. Enter **y2o** (for **y**ellow, input series **2**, **o**dd frame numbers) in the optional **File Name Prefix** text box. Click **OK**.

E. For the third output series, select **multi-frame tif** from the **Series Type** drop-down menu and enter **r2eg1e** (for **r**ed, input series **2**, **e**ven frame numbers; **g**reen, input series **1**, **e**ven frame numbers) in the optional **File Name Prefix** text box. Click **OK**.

F. For the fourth output series, select **n.tif** from the **Series Type** drop-down menu and enter **y1ob2e** (for **y**ellow, input series **1**, **o**dd frame numbers; **b**lue, input series **2**, **e**ven frame numbers) in the optional **File Name Prefix** text box. Click **OK**.

G. For the fifth output series, select **multi-frame tif** from the **Series Type** drop-down menu and enter **r1et1o** (for **r**ed, input series **1**, **e**ven frame numbers; **t**urquoise, input series **1**, **o**dd frame numbers) in the optional **File Name Prefix** text box. Click **OK**.

H. We are asked to give the file name for the default ImageSet object that will be created for the overlay images. Leave the text box empty to accept the default file name 'imageset1'. Click **OK**.

I. A confirmation for the creation of the output MIATool directory appears. Click **OK**.

- As always you can specify a different location to create the MIATool directory and a different name for the root folder of the MIATool directory. However, the remainder of this tutorial will assume this MIATool directory to be located at 'C:\miatooldemo3'.
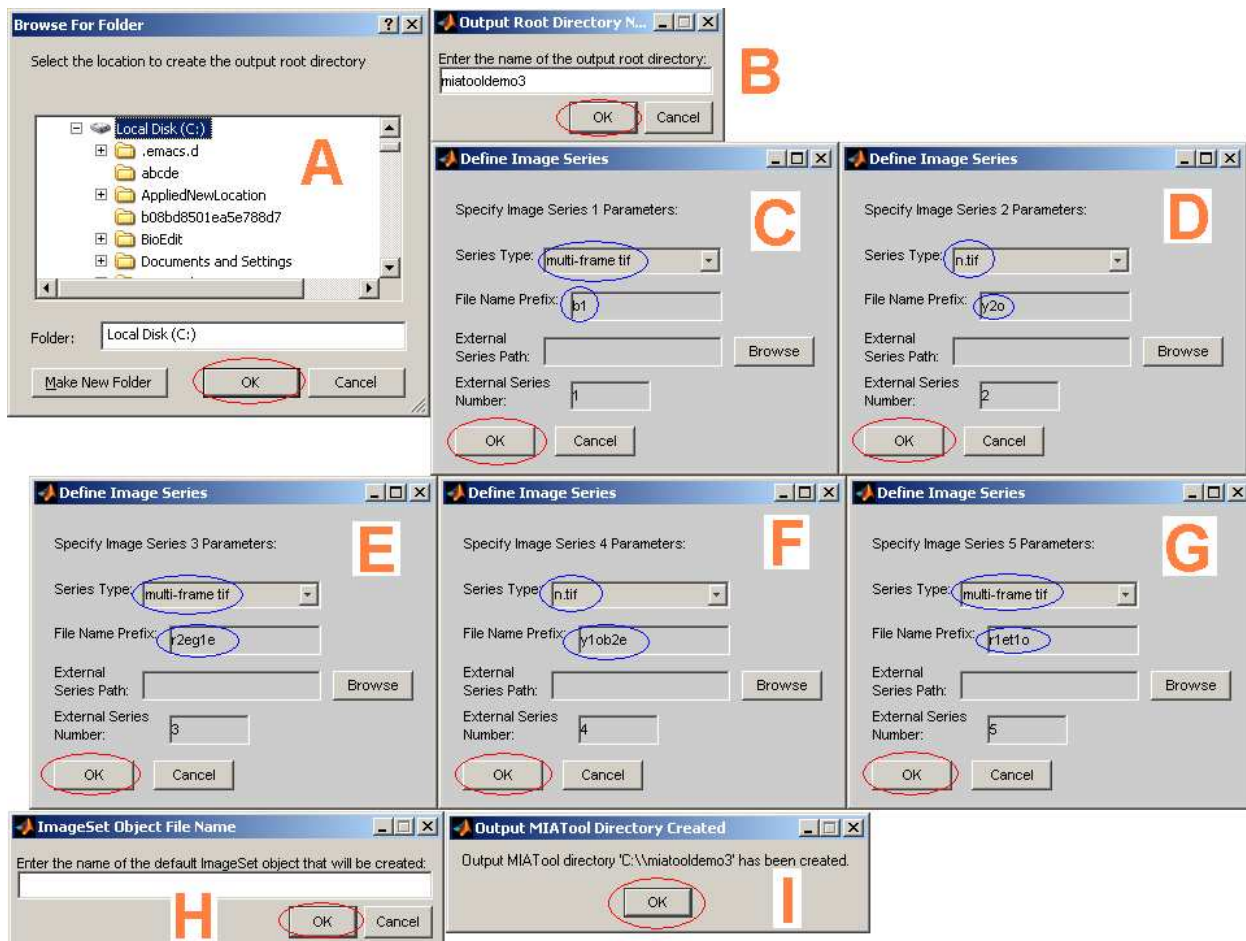
Figure 3.7.3: Specifying MIATool directory for overlay images

Step 6: *Look at MIATool directory containing overlay images* Use Windows Explorer to verify what has been created on disk. You should see that the MIATool directory 'C:\miatooldemo3'

has been created, and that it contains the subfolders 'ImageFiles' and 'MIAToolObjects' as well as the file 'miatooldir_miatooldirectory.mia'. The 'ImageFiles' subfolder should contain five series subfolders, and each series subfolder should contain either a multi-frame tiff file or single tiff images with file name prefixes as specified in the previous step. The 'MIAToolObjects' subfolder should contain the subfolder 'ImageSet1', which in turn should contain the default ImageSet object 'imageset1_imageset.mia'. Details concerning MIATool's file naming conventions can be found in Sections 3.2 and 3.6 and will not be repeated here. See Figure 3.7.4 for screen shots of the various folders and files that should have been created.
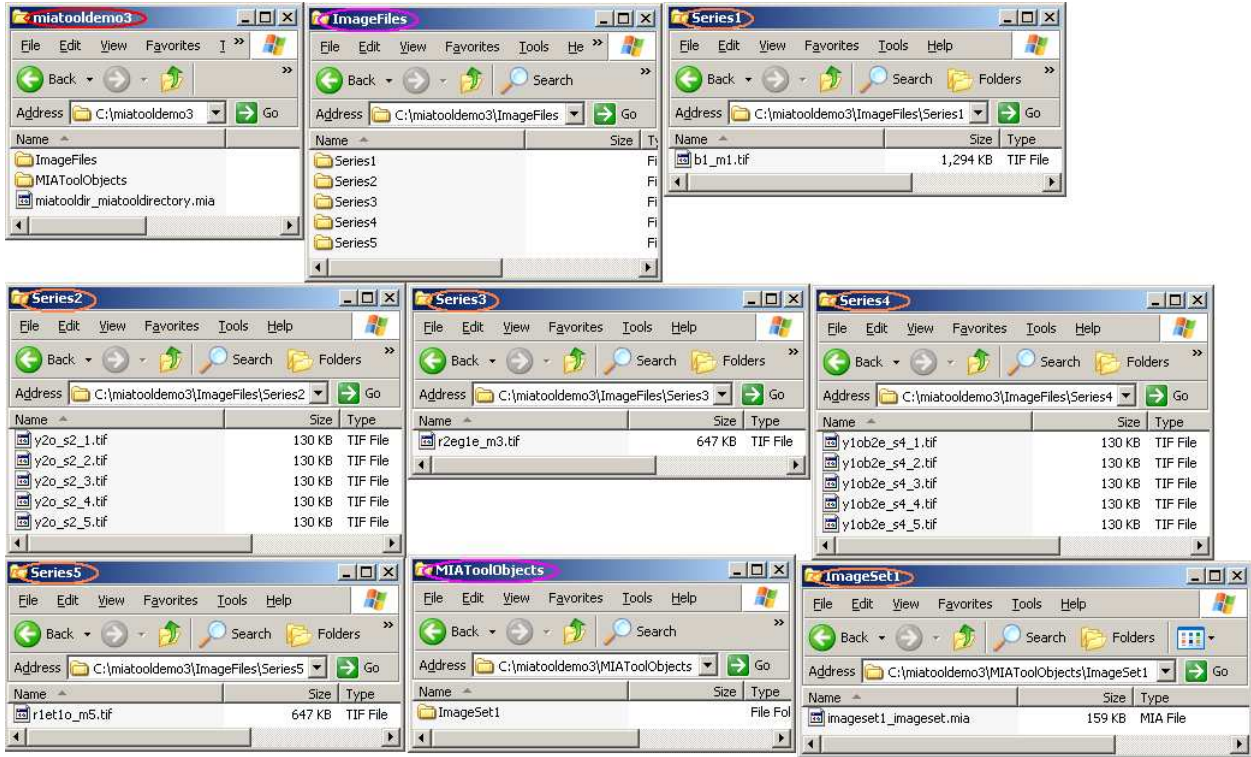
Figure 3.7.4: MIATool directory containing overlay images

3.7.1   Viewing and processing RGB images

MIATool supports the viewing and processing of RGB images. In this section we will view the overlay images we have created and provide some important information for working with RGB images.

Step 7: *Load MIATool directory containing overlay images* Load the MIATool directory 'C:\miatooldemo3', which contains the overlay images we have created. Once loaded, you should see the message 'Loaded: C:\miatooldemo3' displayed in the MIATool main menu figure.

- Refer to Section 3.3 for help on loading the MIATool directory.

55

Step 8: *Initiate opening of MIATool viewer* Select **MIATool → Open DisplayTool** from the MIATool main menu figure.

Step 9: *Choose ImageSet object to view and process* Select the ImageSet object 'imageset1_imageset.mia'. Click **OK**.

Step 10: *Choose a default CropSet object* Select 'CropSet' from the 'Object Type' drop-down menu. Highlight 'default CropSet' in the 'Available Objects' list box and click the **Add** button to add it to the 'Selected Objects' list box. (See Figure 3.7.5, left panel.)

Step 11: *Choose a default DisplaySet object* Select 'DisplaySet' from the 'Object Type' drop-down menu. Highlight 'default DisplaySet' in the 'Available Objects' list box and click the **Add** button to add it to the 'Selected Objects' list box. Click **Done**. (See Figure 3.7.5, middle panel.)

Step 12: *Decline LabelSet and SegmentationSet object selection* In the next figure that pops up, simply click **Done** since we won't be doing any image labeling or segmentation in this part of the tutorial. (See Figure 3.7.5, right panel.)
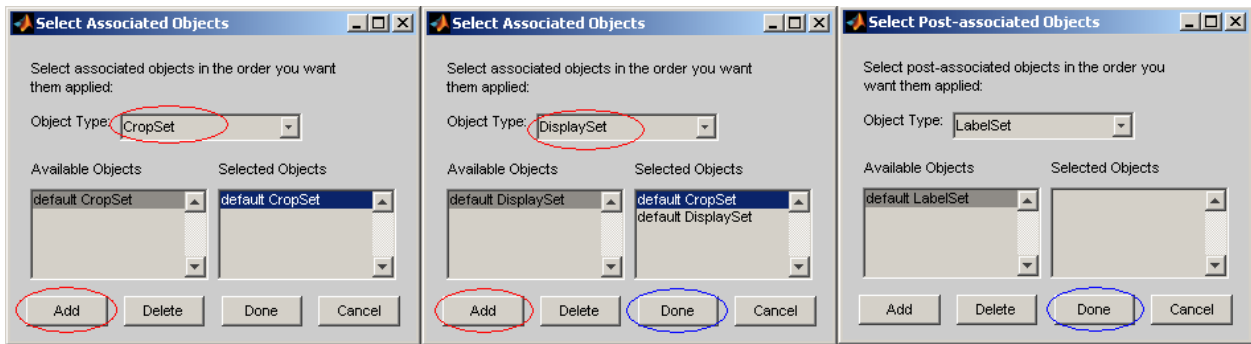
Figure 3.7.5: Opening image viewer with default CropSet and default DisplaySet

Step 13: *Scroll viewer slider and verify overlay images* Scroll the first viewer slider and quickly verify that the five different overlays are as we had specified them. See Figure 3.7.6 for screen shots of what you should see. Also notice that the maximum value of the second viewer slider is 10 only for the first image series. For the second, third, fourth, and fifth series, the maximum value of the second viewer slider is 5. This is correct because the first series was the only one where we chose to generate 10 overlay images during the overlay specification. For all other series we chose only the odd and/or even numbered frames, and hence they only have 5 overlay images each.

Figure 3.7.6: First image of each of the five overlay series

Step 14: *View single channel of RGB image as a grayscale image* You can choose to view only a particular color channel of an RGB image using the **Channel** drop-down menu in the figure containing the viewer sliders. Choose **Red**, **Green**, or **Blue** from the **Channel** drop-down menu to view the respective color channel as a grayscale image. Figure 3.7.7 shows the viewer configured to display the blue channel of the first series of overlays. (If you view the red or green channel of the first series of overlays, you will see grayscale images of zero intensity because the first series of overlays are strictly blue images.)

- The **Display** drop-down menu is enabled only when you are viewing a single color channel. Since an individual color channel is displayed as a grayscale image, the mesh and contour display modes can be used to view it in addition to the standard display mode. In contrast, when all three color channels are viewed at the same time (i.e., when **All** is selected from the **Channel** drop-down menu), the displayed image is RGB and the only available display mode is the standard mode.
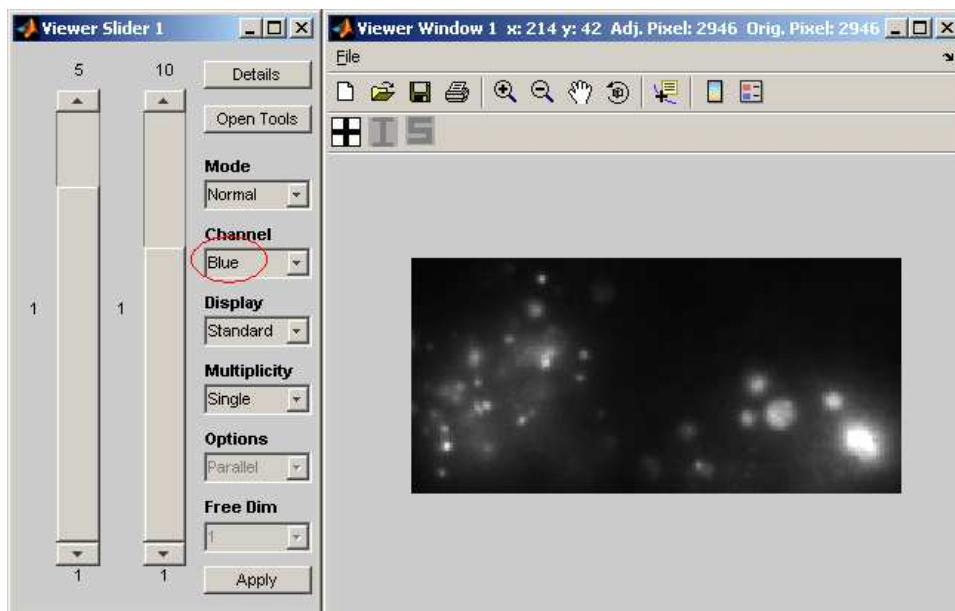
Figure 3.7.7: Viewing the blue channel of the first series of overlays

Step 15: *View RGB images in parallel viewing mode* You can view multiple RGB images at the same time, but the only valid viewing mode you can use is the parallel viewing mode. Set the **Channel** drop-down menu back to **All**, and select **Multiple** from the **Multiplicity** drop-down menu. The **Options** drop-down menu should default to **Parallel**, which is what we want. The **Free Dim** drop-down menu should default to **1**, which is also what we want. You should now have five image windows. Drag them apart and scroll the second viewer slider to view the five series of overlays simultaneously. Note, however, that you will get an error if the second viewer slider goes beyond the value 5. This is because the second, third, fourth, and fifth series of overlays only have five images each!

- When viewing RGB images in multiple viewing mode, you will run into errors if you select

Step 16: *Activate crosshair and view pixel intensity profiles of RGB images* Set the **Multiplicity** drop-down menu back to **Single**. Activate the crosshair and the pixel intensity profiles (see Section 3.4.1 if you need help). Scroll the viewer sliders and watch the pixel intensity profiles change as the current image changes. Note that when the **Channel** drop-down menu is set to **All**, the pixel intensity profiles displayed are those of the red channel. To view the pixel intensity profiles of the green or blue channel, you need to set the **Channel** drop-down menu to **Green** or **Blue**, respectively. Take, for example, any image of the third series and watch the intensity profiles change as you select **All**, **Red**, **Green**, **Blue** from the **Channel** drop-down menu in turn.

- When the **Channel** drop-down menu is set to **All**, the pixel intensity profiles displayed will be those of the red channel. To view the intensity profiles of another channel, set the **Channel** drop-down menu to the desired channel.

Step 17: *Deactivate crosshair and pixel intensity profiles* Deactivate the crosshair and the pixel intensity profiles by clicking the crosshair toolbar button. Set the **Channel** drop-down menu to **All**.

Step 18: *Adjust pixel intensities of RGB images* MIATool's pixel intensity adjustment tool supports independent intensity adjustment of the three channels of an RGB image. First open the intensity adjustment tool by clicking the **Open Tools** button in the figure containing the viewer sliders. In the list box that appears, highlight **2: displayset** and click **OK**. Now perform the following:

A. We will use the fourth image of the fourth series for the demonstration. Set both viewer sliders to the value **4**.

B. Notice that the **Channel** drop-down menu of the intensity adjustment tool is set to **Red** by default, meaning that the settings currently shown in the intensity adjustment tool applies to the red channel of the RGB image. Select **Multiply** from the **Method** drop-down menu of the intensity adjustment tool and enter the value **2** in the **Multiply** text box. In the image window you should see an increase in the intensity of the red channel.

C. Select **Green** from the **Channel** drop-down menu of the intensity adjustment tool. Enter the value **3** in the **Multiply** text box. In the image window you should see an increase in the intensity of the green channel.

D. Select **Blue** from the **Channel** drop-down menu of the intensity adjustment tool. Enter the value **4** in the **Multiply** text box. In the image window you should see an increase in the intensity of the blue channel.

E. Just as with grayscale images, to keep the intensity settings we need to save them to memory. Let's apply the settings to all images. Click the **Save** button in the intensity adjustment tool and in the list box that comes up, select **Apply to All Images** and click **OK**. In the confirmation box that appears, click **OK**. Now scroll the viewer sliders and see how the images look.
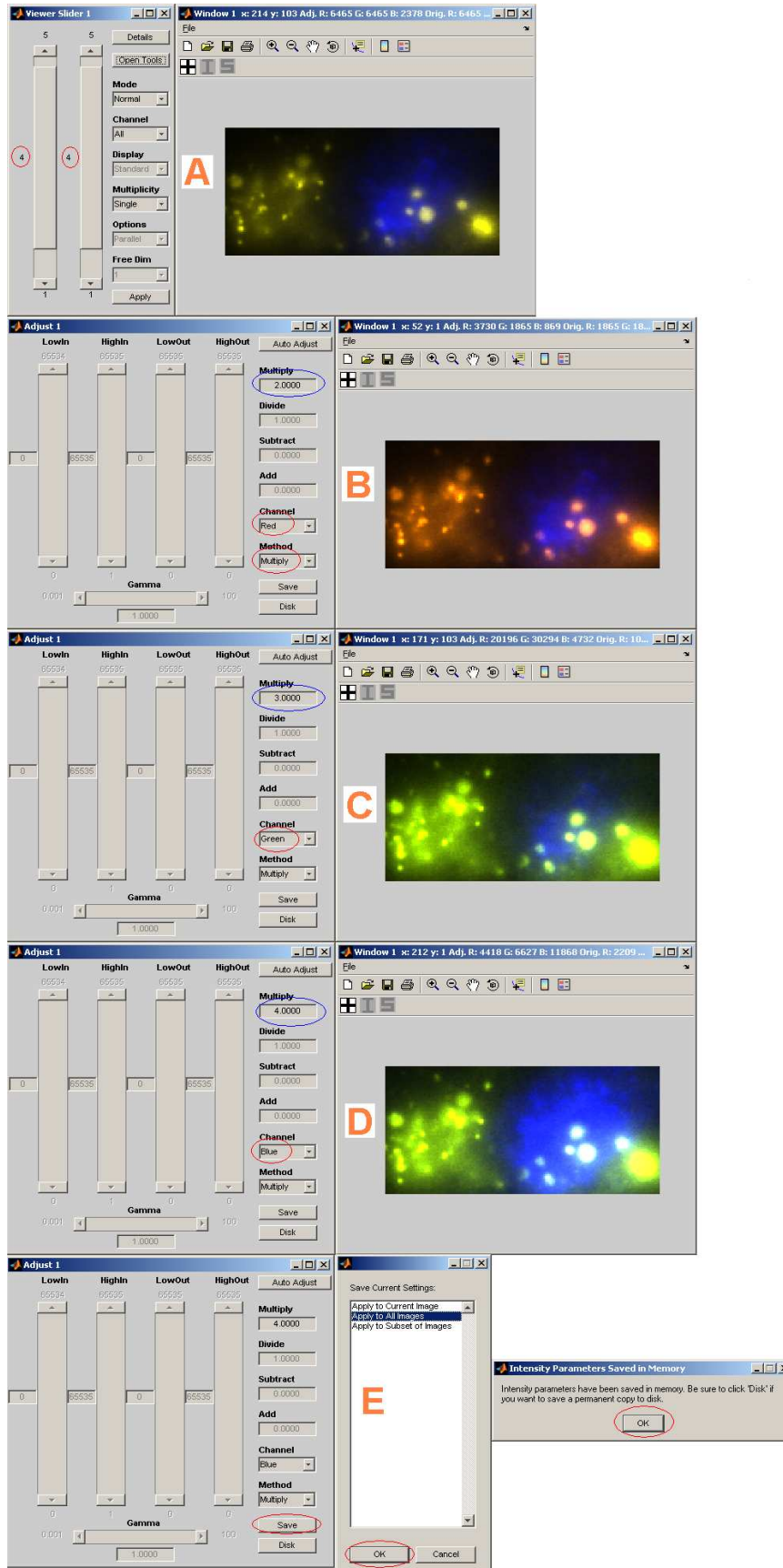
Figure 3.7.8: Pixel intensity adjustment of an RGB image

Step 19: *Crop RGB images* MIATool's crop tool can be used with RGB images in exactly the same way as with grayscale images (see Section 3.5). Open the crop tool by clicking the **Open Tools** button in the figure containing the viewer sliders. In the list box that appears, highlight **1: cropset** and click **OK**. In the crop tool figure, enter the following in the given order: **Crop Height = 53**, **Crop Width = 120**, **Upper Left Pixel X = 58**, **Upper Left Pixel Y = 28**. Click the **Save** button in the crop tool and in the list box that comes up, select **Apply to All Images** and click **OK**. In the confirmation box that appears, click **OK**. Now scroll the viewer sliders and see all the images cropped using the same settings.

Step 20: *Close MIATool viewer and main menu* We are finished with this part of the tutorial. Close the MIATool viewer by clicking the close (×) button in the upper right-hand corner of the figure containing the viewer sliders. You can close the main menu figure as well by clicking its close (×) button.

3.8   Manual image segmentation

MIATool provides a tool for the manual segmentation of images. The segmentation tool works only with grayscale images. To visually distinguish segmented portions of an image from the rest of the image, the MIATool viewer displays the grayscale image in the green channel of an RGB image and the segmented portions of the grayscale image in the red and blue channels of the same RGB image. Segmented portions displayed in the red channel will appear in yellow, and those displayed in the blue channel will appear in turquoise. This section will demonstrate the manual segmentation tool using the intensity-adjusted and cropped images we created in Section 3.6.

Step 1: *Open MIATool main menu and load MIATool directory* Make sure that you have the MIATool main menu figure opened and the MIATool directory 'C:\miatooldemo2' loaded. The MIATool directory 'C:\miatooldemo2' contains the intensity-adjusted and cropped images that we want to manually segment.

- Refer to Section 3.1 for help on opening the main menu figure, and Section 3.3 for help on loading the MIATool directory.

Step 2: *Initiate opening of MIATool viewer* Select **MIATool → Open DisplayTool**.

Step 3: *Choose ImageSet object to view and segment* Select the ImageSet object 'imageset1_imageset.mia'. Click **OK**.

Step 4: *Decline CropSet and DisplaySet object selection* In the next figure that pops up, simply click **Done** since we won't be doing any cropping or pixel intensity adjustment in this part of the tutorial. (See Figure 3.8.1, left panel.)

Step 5: *Select first default SegmentationSet object* In the next figure that appears, select 'SegmentationSet' from the 'Object Type' drop-down menu. Highlight 'default SegmentationSet' in the 'Available Objects' list box and click the **Add** button to add it to the 'Selected Objects' list box. (See Figure 3.8.1, middle panel.) This SegmentationSet object will allow us to open one instance of the manual segmentation tool and create segmented parts that appear in yellow.

- To be able to segment the images you are viewing, you must select a SegmentationSet object at this step.

- A 'SegmentationSet' object is a container of the segmented parts of the images referred to by an ImageSet object. In our present case where there are a total of 20 images, the SegmentationSet object will be automatically configured to store the segmented parts for 20 images.

- A **default** SegmentationSet object is one that does not contain any segmented parts.

Step 6: *Select second default SegmentationSet object* Highlight 'default SegmentationSet' in the 'Available Objects' list box again and click the **Add** button to add it to the 'Selected Objects' list box. (See Figure 3.8.1, right panel.) This second SegmentationSet object will allow us to open a second instance of the manual segmentation tool and create segmented parts that appear in turquoise. Click **Done**.

- You can choose **up to 2** SegmentationSet objects during the opening of the MIATool viewer. The first chosen SegmentationSet object will have its segmented parts displayed in yellow, while the second chosen SegmentationSet object will have its segmented parts displayed in turquoise.
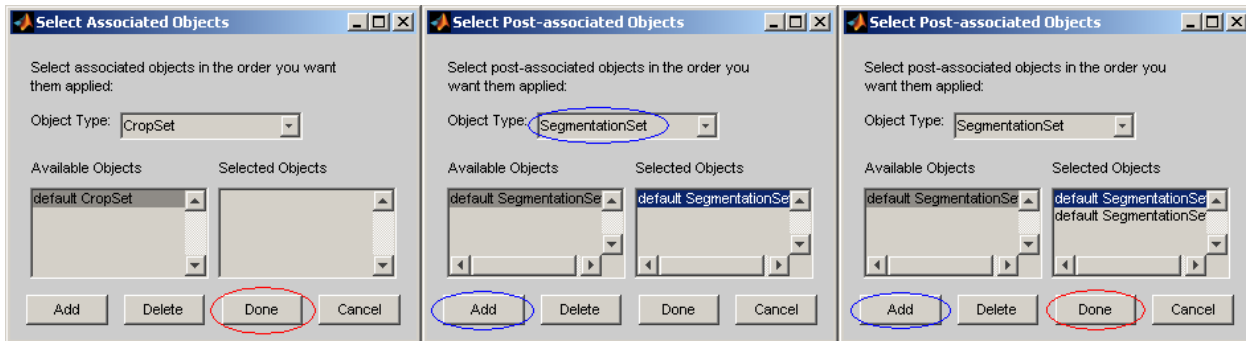
Figure 3.8.1: Opening image viewer with two default SegmentationSet objects

Step 7: *Notice grayscale images displayed in green* When the viewer opens, notice that even though we are working with grayscale images, they are displayed in the green channel of an RGB image with zero intensity red and blue channels.

Step 8: *Open two instances of manual segmentation tool* Open two instances of the manual segmentation tool, one for each SegmentationSet object we selected during the opening of the viewer, by clicking the **Open Tools** button in the figure containing the viewer sliders. In the list box that comes up, click the **Select all** button to select both SegmentationSet objects, then click **OK**. Two segmentation tool figures will open, one on top of the other. Drag them apart with your mouse. They should look identical except for the name in the title bar. One figure should be named 'Segmentation 1' and the other 'Segmentation 2'. The former will produce segmented parts that appear in yellow, while the latter will produce segmented parts that appear in turquoise.
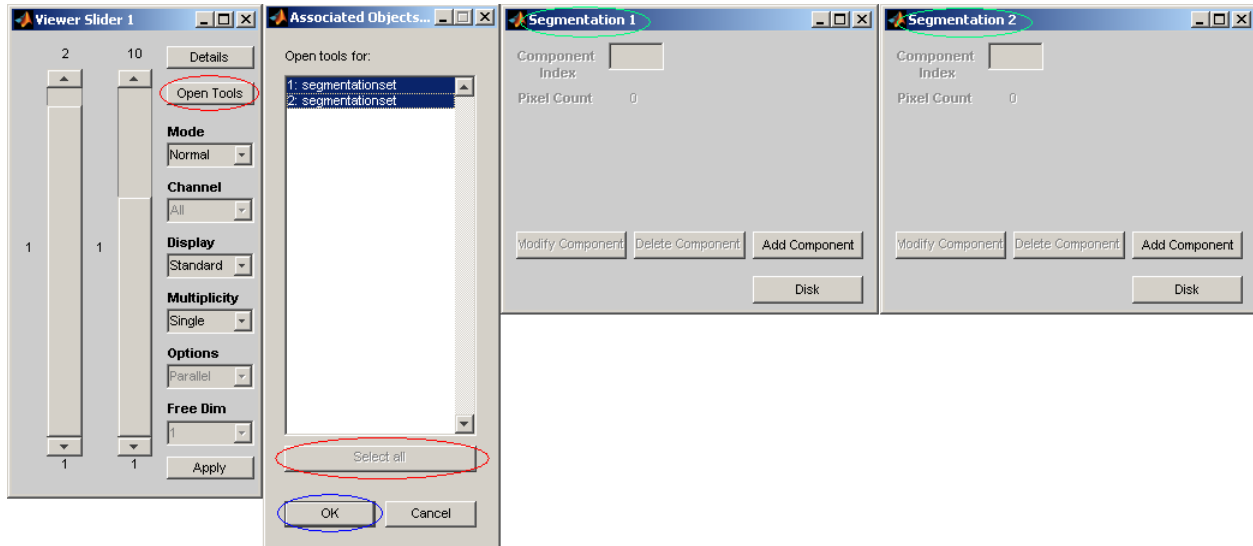
Figure 3.8.2: Opening two instances of the manual segmentation tool

Step 9: *Add components to image* Let's start by adding some segmented parts, which we will refer to as 'components', to image 1 of series 1. Make sure that both viewer sliders are set to the value 1. Perform the following:

A. ***Add yellow square component*** In the segmentation tool figure **Segmentation 1**, click the **Add Component** button. In the figure that comes up, note that the **Region Type** drop-down menu defaults to **Square**, which allows the creation of a square component. Enter the value **11** in the **Square Length** text box, which requires an odd positive integer. This specifies an 11-pixel-by-11-pixel square component. Click the **Click Center** button to interactively select the center pixel of the square component. Click **once** inside the image, approximately at the center of the round object encircled in the image window of Figure 3.8.3, panel A. A star will appear to mark the position. Press <**return**>. You will now see the **Center X** and **Center Y** text boxes populated with the x and y coordinates of the position you clicked. You can repeat the selection of the center pixel as many times as you want, and each time the previous position coordinates will be overwritten. You can also change the values in the text boxes directly. Click **Done**. You should now find that the round object has been marked by a yellow square component. The segmentation tool figure **Segmentation 1** should now show a **1** in the **Component Index** text box, indicating that the square component we just added has been assigned the unique identifier 1 and is taken to be the current component on which the **Modify Component** and **Delete Component** buttons will act. It should also indicate a 121-pixel (11x11 = 121) component in the **Pixel Count** field.

B. ***Add yellow component using square threshold method*** In the segmentation tool figure **Segmentation 1**, click the **Add Component** button. In the figure that comes up, select **Square Threshold** from the **Region Type** drop-down menu, which allows the creation of an irregularly-shaped component from a square component based on pixel intensity values. Enter the value **13** in the **Square Length** text box. This specifies a 13-pixel-by-13-pixel

64

square component to start with. Instead of interactively selecting the center of the square with the **Click Center** button, enter directly the value **59** in the **Center X** text box and the value **44** in the **Center Y** text box. Now enter the value **25000** in the **Threshold** text box. This tells the segmentation tool to form the final component by taking pixels within the specified square that have an **adjusted** intensity value equal to or greater than 25000. Click **Done**. You should now find that the round object encircled in the image window of Figure 3.8.3, panel B, has been marked by a irregularly-shaped yellow component. The segmentation tool figure **Segmentation 1** should now show a **2** in the **Component Index** text box, indicating that the irregularly-shaped component we just added has been assigned the unique identifier 2 and is taken to be the new current component on which the **Modify Component** and **Delete Component** buttons will act. The segmentation tool figure should also indicate in the **Pixel Count** field that the newly-added component is composed of 83 pixels.

C. ***Add turquoise polygonal component*** In the segmentation tool figure **Segmentation 2**, click the **Add Component** button. In the figure that comes up, select **Polygon** from the **Region Type** drop-down menu, which allows the creation of a polygonal component. Click the **Click Polygon** button to interactively select the vertices of the polygonal component. Click a few times around the periphery of the big round object in the lower right-hand corner of the image so as to approximate its shape with a polygon. This round object is encircled in the image window of Figure 3.8.3, panel C. A dashed line will extend to each position you click and help you to visualize the polygon. After you finish clicking, press <**return**>. Though nothing will be shown, pressing the return key tells the segmentation tool to record the chosen polygon. You can repeat the selection of the polygon vertices as many times as you want, and each time the previous vertices will be overwritten. Click **Done**. You should now find that the round object has been marked by a turquoise polygonal component. The segmentation tool figure **Segmentation 2** should now show a **1** in the **Component Index** text box, indicating that the polygonal component we just added has been assigned the unique identifier 1 and is taken to be the current component on which the **Modify Component** and **Delete Component** buttons will act. Keep in mind that this second instance of the segmentation tool works with a different SegmentationSet object than the first instance, so the identifier 1 here does not conflict with the identifier 1 assigned by the first instance of the segmentation tool to the yellow square component. The segmentation tool figure **Segmentation 2** should also display the number of pixels comprising the component in the **Pixel Count** field.

D. ***Add turquoise component using polygon threshold method*** In the segmentation tool figure **Segmentation 2**, click the **Add Component** button. In the figure that comes up, select **Polygon Threshold** from the **Region Type** drop-down menu, which allows the creation of a component from a polygonal component based on pixel intensity values. Click the **Click Polygon** button to interactively select the vertices of the polygonal component. Click three times to make a triangle that encloses the cluster of small objects on the left-hand side of the image. This triangle is shown in the image window of Figure 3.8.3, panel D. A dashed line will extend to each position you click to help you visualize the triangle. After you finish clicking, press <**return**>. Though nothing will be shown, pressing the return key tells the segmentation tool to record the chosen triangle. Now enter the value **20000** in the **Threshold** text box. This tells the segmentation tool to form the final component by taking pixels within the specified triangle that have an **adjusted** intensity value equal to or greater

than 20000. Click **Done**. You should now see the cluster of small bright objects marked by a turquoise component. **Note that a component can consist of disjoint pixels!** The segmentation tool figure **Segmentation 2** should now show a **2** in the **Component Index** text box, indicating that the component we just added has been assigned the unique identifier 2 and is taken to be the new current component on which the **Modify Component** and **Delete Component** buttons will act. The segmentation tool figure **Segmentation 2** should also display the number of pixels comprising the component in the **Pixel Count** field.

- In general, to add components to a particular image, first scroll to that image to make it the current image, then add components to it as demonstrated in this step.
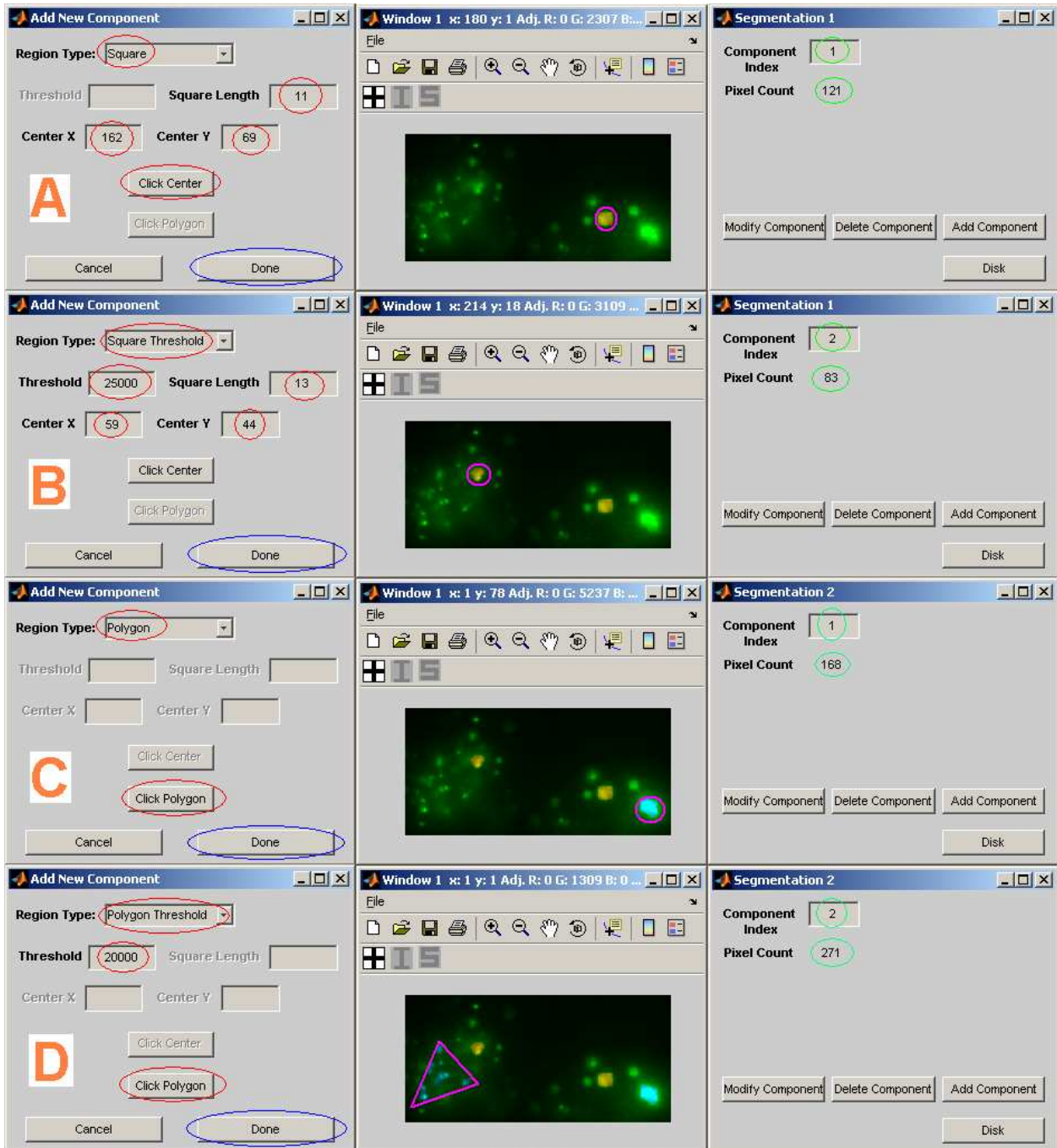
Figure 3.8.3: Four ways to add a component using the manual segmentation tool

Step 10: _Change current component_ For each image that has at least one component associated with it, one of the components is taken to be the current component on which the **Modify Component** and **Delete Component** buttons of the segmentation tool figure will act. This current component is indicated by the identifier in the **Component Index** text box of the segmentation tool figure. The number of pixels that comprise the current component is also shown in

the **Pixel Count** field of the segmentation tool figure. You can change the current component in two different ways.

A. ***Click on a component to make it current*** You can make a component the current component simply by clicking on one of its pixels. Click on the two yellow components in turn and watch the **Component Index** text box and the **Pixel Count** field of the segmentation tool figure **Segmentation 1** update to reflect the last-clicked component. If you don't see any change, make sure that you are clicking on a yellow pixel! Do the same with the two turquoise components and observe the same behavior in the segmentation tool figure **Segmentation 2**.

B. ***Specify component identifier to make it current*** You can make a component the current component by directly entering its integer identifier in the **Component Index** text box of the segmentation tool figure. Take either segmentation tool figure and change the value in its **Component Index** text box back and forth between 1 and 2. With each change you should find the value of the **Pixel Count** field updated to reflect the specified current component.

Step 11: *Added components are automatically saved to memory* Scroll the viewer sliders to change the current image to something other than image 1 of series 1, then change the current image back to image 1 of series 1. You will find that the components we have added are still there. This is because unlike the intensity adjustment tool and the crop tool, added components are automatically saved to memory. There is no 'Save' button that needs to be clicked.

Step 12: *Modify a component* Make image 1 of series 1 the current image if it isn't already. Click on the big turquoise object in the lower right-hand corner of the image (circled object in image window of Figure 3.8.3, panel C) to make it the current component. Verify that the segmentation tool figure **Segmentation 2** shows a **Component Index** of 1. Now perform the following to see how one can modify the current component.

A. Click the **Modify Component** button in the segmentation tool figure **Segmentation 2**.

B. In the figure that comes up, you will find a list box that enumerates the x and y coordinates of all the pixels that comprise the current component (i.e., the big turquoise object in the lower right-hand corner of the image). We can modify the component by removing pixels from it and/or by adding pixels to it. Here we will demonstrate only the delete operation.

C. Select approximately the first half of the coordinates list by doing the following. Highlight the first coordinate in the list with your mouse. Scroll the list box slider down to approximately the middle of the list. While holding down the **Shift** key, click on any one of the coordinates visible in the list box. This selects all coordinates from the top of the list to where you just clicked. Click the **Delete** button to remove from the component the pixels corresponding to the selected coordinates.

D. In the image window you should find that the turquoise component has lost approximately half of its original pixels.

E. Click **Done** to accept the modification we've performed. The modification will automatically be saved to memory.

F. Verify in the segmentation tool figure **Segmentation 2** that the value of the **Pixel Count** field has been updated to reflect a significantly smaller number of pixels that now comprise the component.

- In the figure that contains the list of pixel coordinates, the **Cancel** button may be used to discard any changes you've made as long as you have not clicked the **Done** button. Clicking the **Done** button will commit the changes you've made.

- You can delete and add pixels to the current component as many times as you want before clicking the **Done** button. The coordinates that are present after all the deletions and additions are what the segmentation tool will accept as the modified component when you click the **Done** button.

- You can add pixels to the current component using the **Add** button. When clicked, a 'Coordinates Selection' input figure will come up. The input figure will have two text boxes, one for entering the x coordinate(s) and one for entering the y coordinate(s) of the pixel(s) you want to add to the component. In addition to a valid single coordinate value, either text box will also accept a valid coordinate range (e.g., 4-10 for x=4 to x=10) so that multiple pixels can be specified and added in one add operation. Either text box will also accept **all** as input, which specifies all valid x coordinates or all valid y coordinates. This input figure is very much analogous to the 'Subset Selection' figure detailed in Step 13 of Section 3.4. The acceptable input formats are identical, but just keep in mind that here the numbers you enter refer to x and y pixel coordinates of the current image.
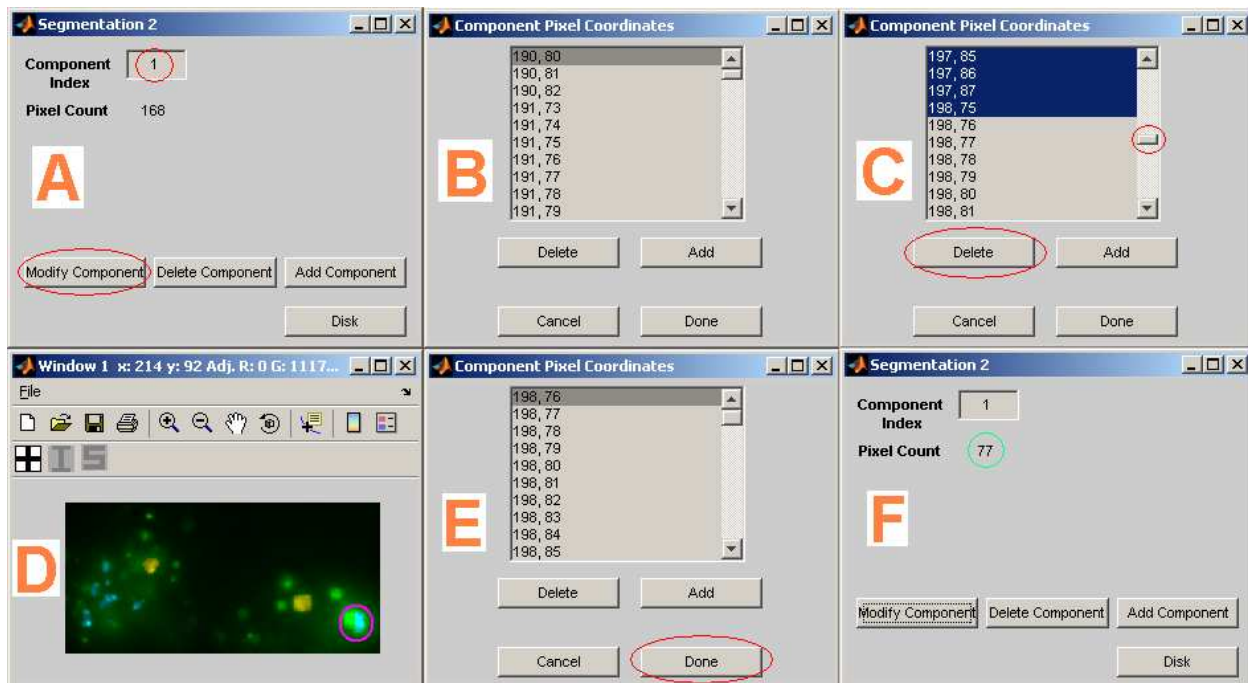

Figure 3.8.4: Removing pixels from the current component

Step 13: *Delete a component* To see how one can delete a component, first click on the yellow object in the left-hand half of the image (circled object in image window of Figure 3.8.3, panel B) to make it the current component. Verify that the segmentation tool figure **Segmentation 1** shows a **Component Index** of 2. Now to delete this component, simply click the **Delete Component** button in the segmentation tool figure **Segmentation 1**. You should find that the yellow component has been removed from the displayed image (i.e., has reverted back to green), and that the **Component Index** text box has been updated with the value **1**, indicating that the current component is now the first yellow component we added, which is also the only yellow component left. As in the case of addition and modification, the deletion of components is automatically saved to memory.

- Once you click the **Delete Component** button, the current component is permanently deleted. Unless it has previously been saved to a SegmentationSet object on disk, the only way to recover it is to use **Add Component**, possibly followed by **Modify Component**, to recreate it. That of course assumes that you know the exact pixels that comprised the deleted component.
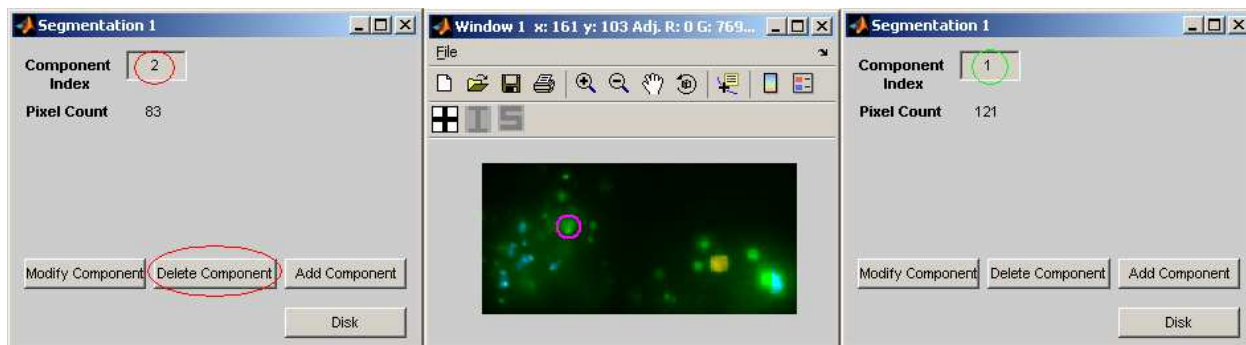
Figure 3.8.5: Deleting the current component

Step 14: *Save components to disk in SegmentationSet objects* Though they are automatically saved to memory, the components created using the manual segmentation tool will be lost once you close the MIATool viewer. To have them available for later viewing and use, they need to be saved in the form of a SegmentationSet object to a MIATool directory on disk. To save the remaining yellow component we currently see in the viewer to disk, click the **Disk** button of segmentation tool figure **Segmentation 1**. When asked for the file name of the SegmentationSet object, enter **yellow** to help us remember that this SegmentationSet object is associated with yellow components. Click **OK** after you enter the file name, and again click **OK** to close the confirmation box. Similarly, to disk the turquoise components we currently see in the viewer, click the **Disk** button of segmentation tool figure **Segmentation 2**. When asked for the file name of the SegmentationSet object, enter **turquoise**. Click **OK** after you enter the file name, and again click **OK** to close the confirmation box. Figure 3.8.6 gives screen shots of the 'segmentationset objects' subfolder and its contents which you should now find under the MIATool directory 'C:\miatooldemo2'.
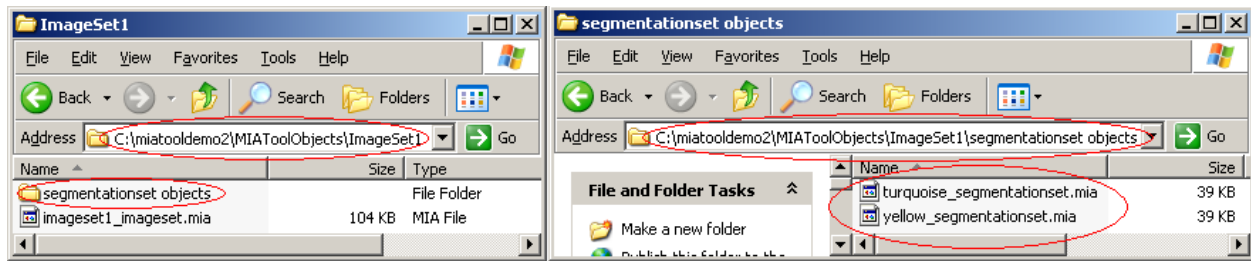
Figure 3.8.6: SegmentationSet objects stored in MIATool directory on disk

Step 15: *Write segmented images to disk* You can write the segmented images you see in the viewer to disk as RGB images. To do so we will again use the **Apply** button in the viewer sliders figure as we did in Section 3.6.

A. Click the **Apply** button in the figure containing the viewer sliders.

B. In the file browser, select the root of the C drive as the location to create the output MIATool directory under which the RGB images will reside. Click **OK**.

C. Specify **miatooldemo4** as the name of the root folder of the output MIATool directory. Click **OK**.

D. Select **multi-frame tif** from the **Series Type** drop-down menu to save the first series of RGB images as a multi-frame tiff file. Leave the optional **File Name Prefix** text box empty. Click **OK**.

E. Select **n.tif** from the **Series Type** drop-down menu to save the second series of RGB images as individual tiff files. Leave the optional **File Name Prefix** text box empty. Click **OK**.

F. Leave the text box empty to accept the default file name 'imageset1' for the default ImageSet object that will be created for the RGB images. Click **OK**.

G. Click **OK** in the confirmation box for the creation of the output MIATool directory.

- As always you can specify a different location to create the MIATool directory and a different name for the root folder of the MIATool directory. However, the remainder of this tutorial will assume this MIATool directory to be located at 'C:\miatooldemo4'.

- The **Apply** button will write to disk the segmented images as they are currently seen in the viewer. It does not matter whether the components have been saved to disk.

Step 16: *Close MIATool viewer* Close the MIATool viewer by clicking the close (✕) button in the upper right-hand corner of the figure containing the viewer sliders.

Step 17: *Verify segmented images by viewing them using MIATool viewer* Let's verify the RGB images we've just created by viewing them with the MIATool viewer. First load the

71

MIATool directory 'C:\miatooldemo4 (refer to Section 3.3 if you need help). Once loaded, select **MIATool → Open DisplayTool** from the MIATool main menu figure. In the list box that comes up, select the ImageSet object 'imageset1_imageset.mia' and click **OK**. In the next figure that pops up, simply click **Done** since we won't require any cropping or intensity adjustment. Similarly, simply click **Done** in the subsequent figure since we won't be doing any image labeling or segmentation. When the viewer opens, you should see RGB images that look identical to the segmented images we were looking at previously (i.e., all images are strictly green except image 1 of series 1, which has some yellow and turquoise).

- Make image 1 of series 1 the current image. Use the **Channel** drop-down menu in the figure with the viewer sliders to view the red, green, and blue channels individually. In the red channel you should find only the parts that are yellow in the RGB image. In the green channel you should see the entire image. In the blue channel, you should see only the parts that are turquoise in the RGB image.

Step 18: *Close MIATool viewer and main menu* We are done with this part of the tutorial. Close the MIATool viewer by clicking the close (×) button in the upper right-hand corner of the figure containing the viewer sliders. You can close the main menu figure as well by clicking its close (×) button.

3.9   Image labeling

MIATool provides a tool for the labeling of images. The label tool works with both grayscale and RGB images, and supports labeling with lines and text. This section will demonstrate the label tool using the RGB images we created at the end of Section 3.8.

Step 1: *Open MIATool main menu and load MIATool directory* Make sure that you have the MIATool main menu figure opened and the MIATool directory 'C:\miatooldemo4' loaded. The MIATool directory 'C:\miatooldemo4' contains the RGB images that we want to label.

- Refer to Section 3.1 for help on opening the main menu figure, and Section 3.3 for help on loading the MIATool directory.

Step 2: *Initiate opening of MIATool viewer* Select **MIATool → Open DisplayTool**.

Step 3: *Choose ImageSet object to view and label* Select the ImageSet object 'imageset1_imageset.mia'. Click **OK**.

Step 4: *Decline CropSet and DisplaySet object selection* In the next figure that pops up, simply click **Done** since we won't be doing any cropping or pixel intensity adjustment in this part of the tutorial. (See Figure 3.9.1, left panel.)

Step 5: *Select a default LabelSet object* In the next figure that appears, select 'LabelSet' from the 'Object Type' drop-down menu. Highlight 'default LabelSet' in the 'Available Objects' list

box and click the **Add** button to add it to the 'Selected Objects' list box. (See Figure 3.9.1, right panel.) This LabelSet object will allow us to place labels on the RGB images. Click **Done**.

- To be able to label the images you are viewing, you must select a LabelSet object at this step.

- A 'LabelSet' object is a container of the labels placed on the images referred to by an ImageSet object. In our present case where there are a total of 20 images, the LabelSet object will be automatically configured to store labels for 20 images.

- A **default** LabelSet object is one that does not contain any labels.

- You can open the viewer with multiple LabelSet objects, each of which will have an independent label tool associated with it. You can use multiple LabelSet objects to store different types of labels for the same images. For example, you can use one LabelSet object to store only line labels and another to store only text labels. For another example, you can store labels of different colors, sizes, fonts, etc., in different LabelSet objects.
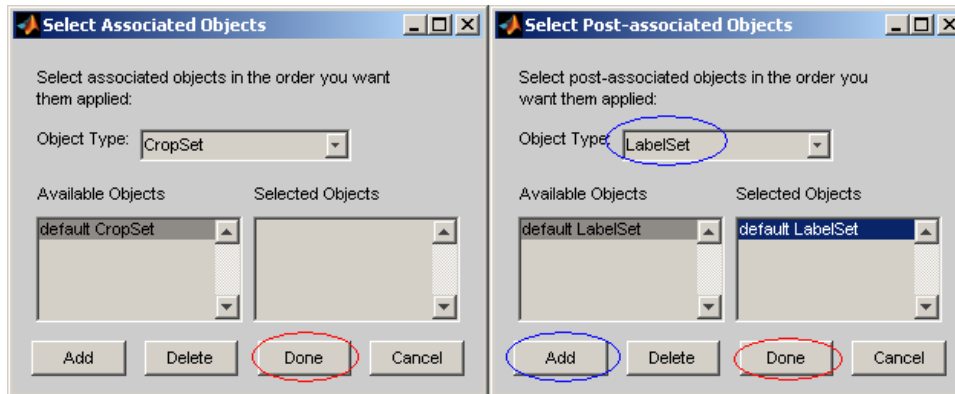

Figure 3.9.1: Opening image viewer with a default LabelSet object

Step 6: *Open label tool* After the viewer comes up, open the label tool by clicking the **Open Tools** button in the figure containing the viewer sliders. In the list box that comes up, select **1: labelset** and click **OK**. The label tool figure should open.
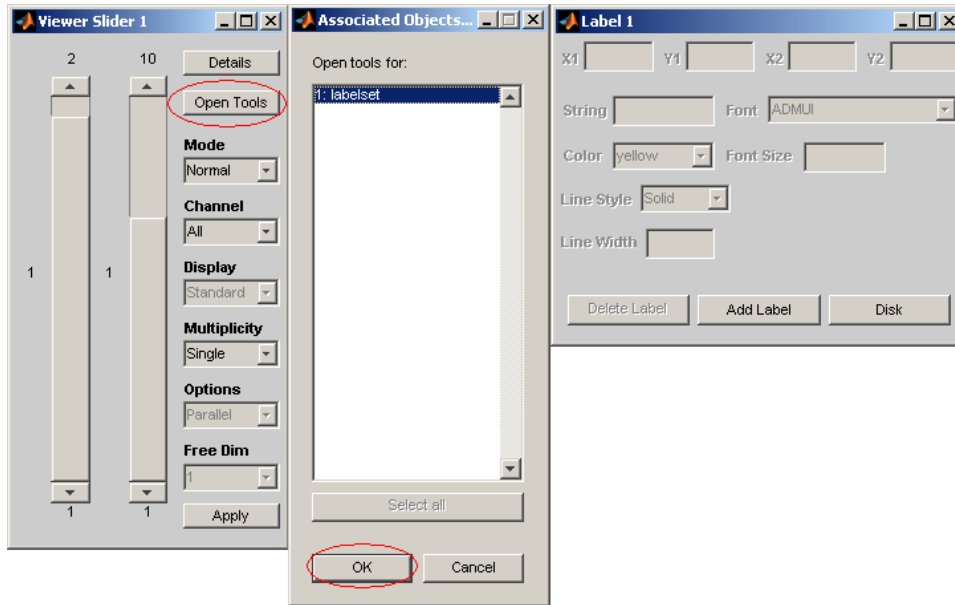
Figure 3.9.2: Opening the label tool

Step 7: ***Add labels to image*** Let's start by adding one each of the two types of labels supported by the label tool - a text label and a line label - to image 1 of series 1. Make sure that both viewer sliders are set to the value 1. Perform the following:

A. ***Add a text label*** In the label tool figure, click the **Add Label** button. In the figure that comes up, note that the **Label Type** drop-down menu defaults to **Text**, which allows the creation of a text label. To specify the position of the label, click the **Click X1 Y1** button. When you move your mouse over the image window, a crosshair will be present to help you select a location for the label. Click anywhere inside the image where you would like to place the text label. You will now see the **X1** and **Y1** text boxes in the label tool figure populated with the x and y coordinates of the position you clicked. You can repeat the interactive selection of the position as many times as you want, and each time the previous position coordinates will be overwritten. You can also change the values in the **X1** and **Y1** text boxes directly. If you know the precise coordinates at which to place the label, you can enter them in the text boxes without having to do the interactive selection at all. Now specify the text for the label by entering it in the **String** text box. For the demonstration we will enter the text **MIATool**, but you can enter anything you like. Click **Done**. You should now find in the image window the text label displayed at the position you chose. The text will be in red by default, and it will be enclosed in a blue box to indicate that it is the current label. The current label is the one on which the **Delete Label** button in the label tool figure will act, and whose attributes like color and font size are displayed and modifiable in the various text boxes and drop-down menus in the label tool figure. All the enabled controls in the label tool figure should now show the values of the various attributes of the text label you just added (e.g., color is red, font size is 10, etc.).

B. ***Add a line label*** In the label tool figure, click the **Add Label** button. In the figure that comes up, select **Line** from the **Label Type** drop-down menu, which allows the creation of

74

a line label. To specify the position of one end of the line, click the **Click X1 Y1** button. Click anywhere inside the image where you would like one end of the line label to be. You will now see the **X1** and **Y1** text boxes in the label tool figure populated with the x and y coordinates of the position you clicked. Now specify the position of the other end of the line by clicking the **Click X2 Y2** button and then clicking anywhere inside the image where you would like the other end of the line label to be. You will now see the **X2** and **Y2** text boxes in the label tool figure populated with the x and y coordinates of the position you clicked. You can repeat the interactive selection of either position as many times as you want, and each time the previous position coordinates will be overwritten. You can also change the values in the four text boxes directly. If you know the precise coordinates of either or both ends of the line label, you can enter them in the text boxes without having to do the interactive selection at all. Click **Done**. You should now find in the image window the line label displayed as you have specified. The line will be red by default, and it will appear to be selected as it's ends will be marked by small squares. The selected state indicates that it is the new current label on which the **Delete Label** button in the label tool figure will act, and whose attributes like color and line width are displayed and modifiable in the various text boxes and drop-down menus in the label tool figure. Notice that the text label has lost its enclosing blue box since it is no longer the current label. All the enabled controls in the label tool figure should now show the values of the various attributes of the line label you just added (e.g., color is red, line width is 1, etc.).

- In general, to add labels to a particular image, first scroll to that image to make it the current image, then add labels to it as demonstrated in this step.
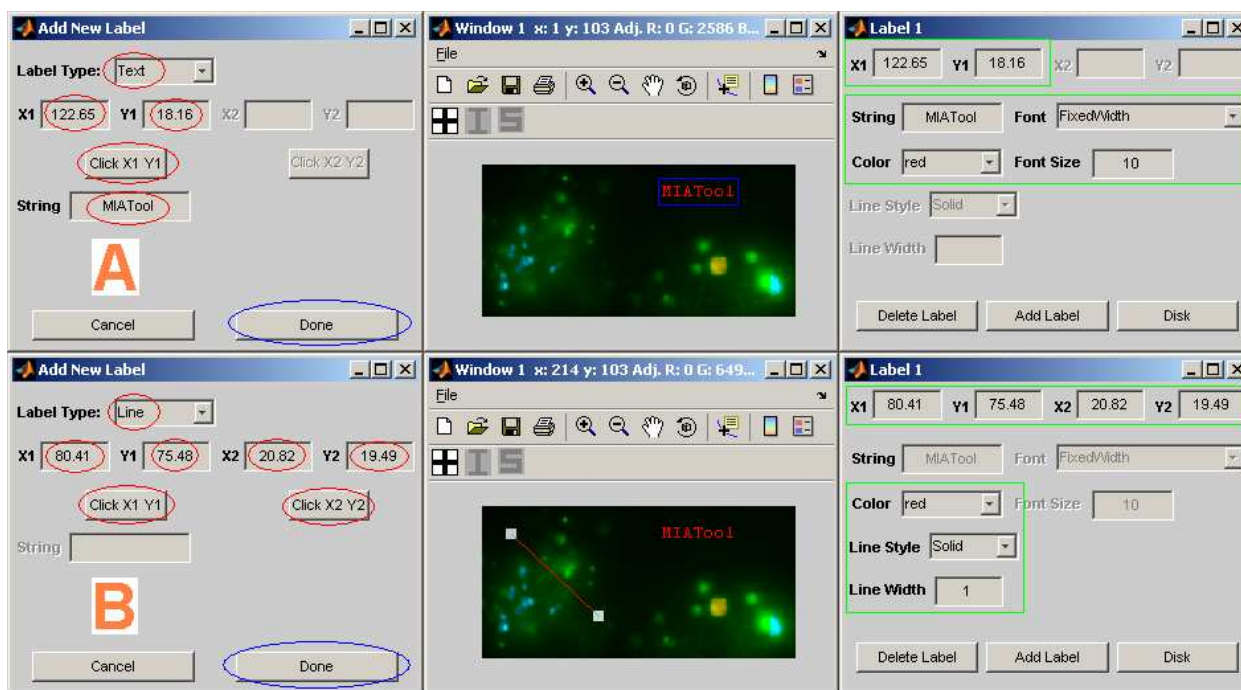

Figure 3.9.3: Adding a text label and a line label using the label tool

Step 8: *Change current label* For each image that has at least one label associated with it, one of the labels is taken to be the current label on which the **Delete Label** button in the label tool figure will act, and whose attributes are displayed and modifiable in the various text boxes and drop-down menus in the label tool figure. This current label is indicated by an enclosing blue box if it happens to be a text label, or by square markers at both of its end points if it happens to be a line label. You can change the current label by simply clicking on the label that you want to make the current label. Click on the text label and line label in turn and watch the various attribute text boxes and drop-down menus of the label tool figure update to reflect the last-clicked label.

Step 9: *Added labels are automatically saved to memory* Scroll the viewer sliders to change the current image to something other than image 1 of series 1, then change the current image back to image 1 of series 1. You will find that the labels we have added are still there. Like components added using the manual segmentation tool, added labels are automatically saved to memory. There is no 'Save' button that needs to be clicked.

Step 10: *Modify a label* Make image 1 of series 1 the current image if it isn't already. Click on the text label to make it the current label. Now perform the following to see how one can modify the current label.

A. Select **white** from the **Color** drop-down menu in the label tool figure. Verify in the image window that the text of the label is now displayed in white.

B. Select **Comic Sans MS** from the **Font** drop-down menu in the label tool figure. Verify in the image window that the font of the text has changed.

C. Change the value in the **Font Size** text box to **15**. Verify in the image window that the text has become larger.

D. Change the text in the **String** text box to something different. Here we will change it to **viewer**, but you can change it to anything you like. Verify in the image window that the text has changed accordingly.

E. Change the value in the **X1** text box to something different. Here we will change it from 122.65 to 60. You can of course change it to anything you like. Verify in the image window that the position of the text has shifted horizontally. You can try changing the **Y1** text box as well to shift the text label vertically.

F. Click on the line label to make it the current label.

G. Select **cyan** from the **Color** drop-down menu in the label tool figure. Verify in the image window that the line is now displayed in cyan.

H. Change the value in the **Line Width** text box to **3**. Verify in the image window that the line has become thicker.

I. Select **Dashed** from the **Line Style** drop-down menu in the label tool figure. Verify in the image window that the line has changed from a solid line to a dashed line.

J. Change the value in the **X2** text box to be the same as that in the **X1** text box. Verify in the image window that the line is now a vertical line.

K. Change the value in the **Y1** or **Y2** text box to make the vertical line longer or shorter. Here we will change the value of the **Y2** text box to make the line shorter. Verify in the image window that the line has changed accordingly.

L. Scroll the viewer sliders to change the current image to something other than image 1 of series 1, then change the current image back to image 1 of series 1. You will find that the changes we made to the labels are still there. This is because changes made to the attributes of a label are automatically saved to memory.
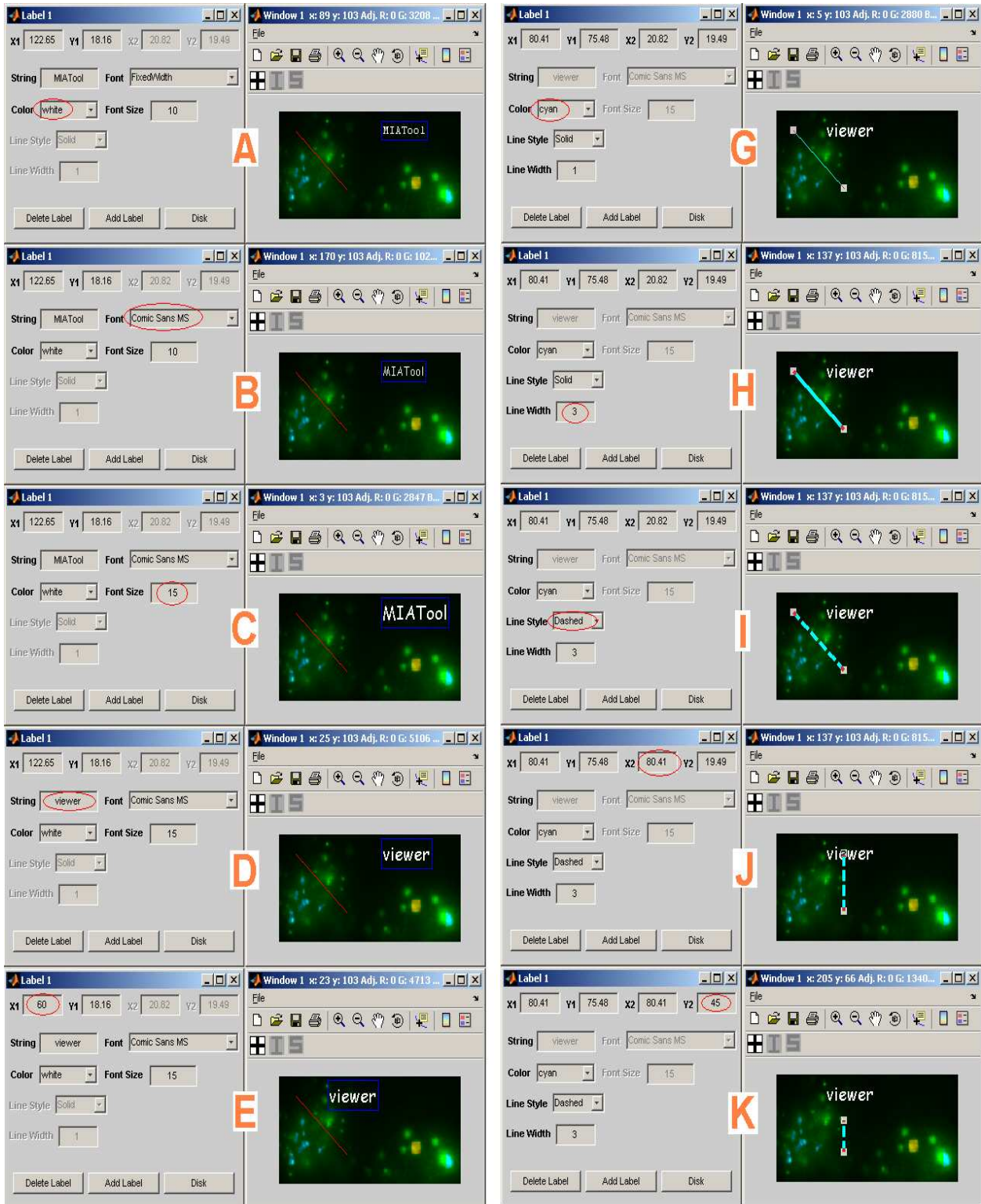
Figure 3.9.4: Modifying the attributes of text and line labels

Step 11: *Delete a label* To see how one can delete a label, first click on the line object to make it the current label. To delete this line, simply click the **Delete Label** button in the label tool figure. You should find in the image window that the line has been removed, and that the various text boxes and drop-down menus in the label tool figure have been updated to reflect the attributes of the text label, indicating that the current label is now the text label, which happens to also be the only label left. As in the case of addition and modification, the deletion of labels is automatically saved to memory.

- Once you click the **Delete Label** button, the current label is permanently deleted. Unless it has previously been saved to a LabelSet object on disk, the only way to recover it is to use **Add Label**, possibly followed by modification of some attributes, to recreate it. That of course assumes that you know exactly the attributes of the deleted label.
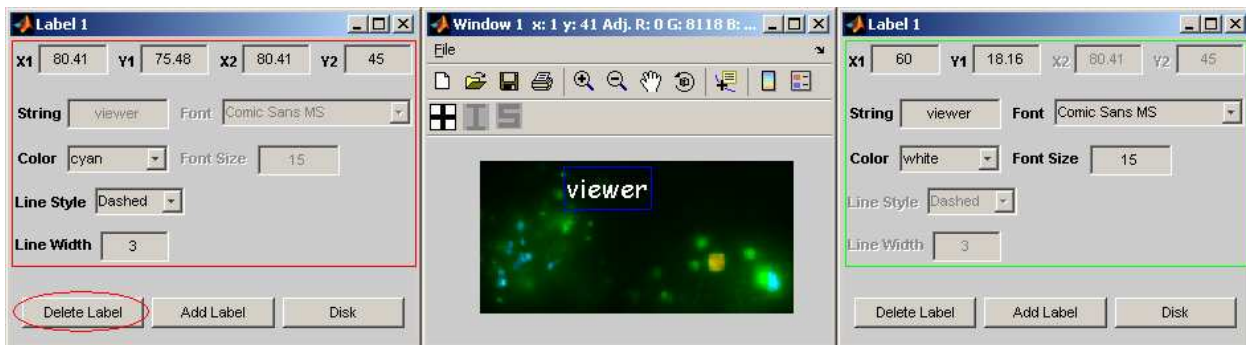
Figure 3.9.5: Deleting the current label

Step 12: *Save labels to disk in a LabelSet object* Though they are automatically saved to memory, the labels created using the label tool will be lost once you close the MIATool viewer. To have them available for later viewing and use, they need to be saved in the form of a LabelSet object to a MIATool directory on disk. To save the remaining text label we currently see in the viewer to disk, click the **Disk** button of the label tool figure. When asked for the file name of the LabelSet object, simply leave the text box empty and click **OK** to accept the default file name of 'labelset1'. Click **OK** to close the confirmation box. Figure 3.9.6 gives screen shots of the 'labelset objects' subfolder and its contents which you should now find under the MIATool directory 'C:\miatooldemo4'.
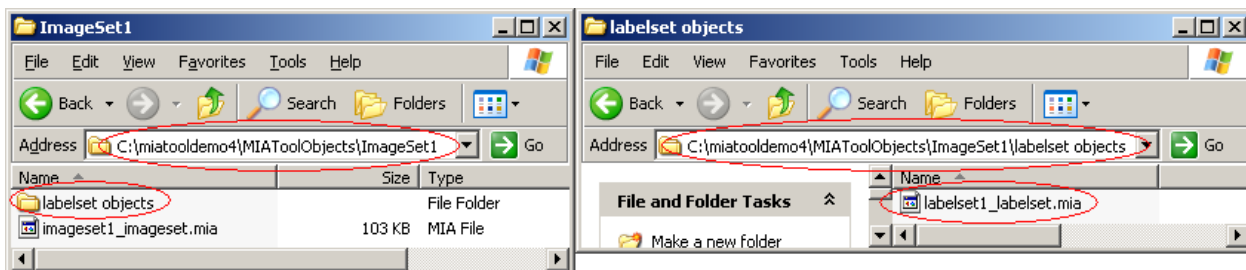
Figure 3.9.6: LabelSet object stored in MIATool directory on disk

79

Step 13: *Creation of images with labels not currently supported* While you can write intensity-adjusted, cropped, and segmented images you see in the viewer to disk using the **Apply** button in the viewer sliders figure, MIATool does **not** currently support the creation of images containing the labels that you see in the viewer. The **Apply** button will work when you have labels displayed, but the images written to disk will **not** contain the labels.

Step 14: *Close MIATool viewer* Close the MIATool viewer by clicking the close (×) button in the upper right-hand corner of the figure containing the viewer sliders.

Step 15: *Verify disked labels by viewing them using MIATool viewer* Let's verify that the disked LabelSet object contains the correct information by reopening the MIATool viewer with that LabelSet object. Select **MIATool → Open DisplayTool** from the MIATool main menu figure. In the list box that comes up, select the ImageSet object 'imageset1_imageset.mia' and click **OK**. In the next figure that pops up, simply click **Done** since we won't require any cropping or intensity adjustment. In the subsequent figure, select **LabelSet** from the **Object Type** drop-down menu. Highlight **labelset1_labelset.mia** in the **Available Objects** list box and click the **Add** button to add it to the **Selected Objects** list box. Click **Done**. When the viewer opens, you should see the remaining text label in image 1 of series 1, and no labels in any other image. That is, we have recovered things exactly as they were when we closed the viewer in the previous step.

- Open the label tool and practice adding, modifying, and deleting some more labels. Try adding labels to different images. When you are done, save the labels using the **Disk** button in the label tool figure. You can choose to overwrite the existing LabelSet object on disk, or you can choose to save things to a different LabelSet object.

Step 16: *Close MIATool viewer and main menu* We are done with this part of the tutorial. Close the MIATool viewer by clicking the close (×) button in the upper right-hand corner of the figure containing the viewer sliders. You can close the main menu figure as well by clicking its close (×) button.

3.10 ImageSet massaging

So far in this tutorial we have dealt with default ImageSet objects which contain an arrangement of image pointers that mirrors how the images are physically stored on disk. Specifically, default ImageSet objects contain exactly one pointer per image in the MIATool directory, and the pointers are arranged in a 2D array where each row corresponds to the images belonging to a 'Series' subfolder on disk. For more details concerning default ImageSet objects, please refer to Step 6 of Section 3.2.

In this part of the tutorial, we demonstrate the 'massaging' of the array of image pointers in an ImageSet object to produce another ImageSet object that has a different arrangement of image pointers. It is important to be able to have different arrangements of pointers. For example, for some particular purpose we may only be interested in viewing and processing a certain subset of the images on disk. In such a case, instead of using the default ImageSet object, it is sufficient and perhaps easier to work with an ImageSet object that has pointers to only the images of interest. For other applications, it may be necessary to view and process a given image in different ways.

For such applications, it makes sense to work with an ImageSet object that contains more than one pointer to each image. In what follows, we will use the intensity-adjusted and cropped images we created in Section 3.6 to demonstrate the two massaging methods currently supported by MIATool - the **subset** operation and the **replicate** operation.

Step 1: ***Open MIATool main menu and load MIATool directory*** Make sure that you have the MIATool main menu figure opened and the MIATool directory 'C:\miatooldemo2' loaded. The MIATool directory 'C:\miatooldemo2' contains the intensity-adjusted and cropped images that we want to work with.

- Refer to Section 3.1 for help on opening the main menu figure, and Section 3.3 for help on loading the MIATool directory.

3.10.1   Taking a subset of image pointers

In this section, we demonstrate the use of the subset operation to create an ImageSet object that contains pointers only to the images we are interested in. The example will also demonstrate how the subset operation can be used to logically split a physical series of images on disk into multiple **logical series**. Please keep in mind that while what we demonstrate here with a small data set composed of only 20 images may seem insignificant, in real life the subset operation can be a very useful tool when a data set consists of thousands or tens of thousands of images and you are only interested in working with a small subset of it.

The subset operation requires as input an ImageSet object with a 2D array of image pointers and always outputs an ImageSet object with a 2D array of image pointers. A default ImageSet object is therefore always a valid input to the subset operation. Likewise, the output ImageSet object of a subset operation is always a valid input to another subset operation.

Step 2: ***Initiate subset operation*** In the MIATool main menu figure, select **ImageSet Massaging → Subset**.

- The use of the subset operation is independent of the MIATool viewer. All that's required is that the MIATool directory containing the ImageSet object to which you want to apply the subset operation is loaded.

Step 3: ***Choose ImageSet object to apply subset operation to*** Select the ImageSet object 'imageset1_imageset.mia'. Click **OK**. This is the default ImageSet object which contains two rows of ten image pointers each. The first row of pointers references the intensity-adjusted and cropped images of series 1 on disk, and the second row of pointers references those of series 2 on disk.

Step 4: ***Specify output logical image series*** In the next figure that comes up, perform the following.

A. Suppose that of the ten images belonging to series 2 on disk we are interested only in images

2 through 7, and suppose that we want these six images of interest to be the first **logical series** in the ImageSet object we are creating. Select **Series 2** from the **Series Number** drop-down menu and enter the value **2** in the **Start** text box and the value **7** in the **End** text box. Click the **Add** button and you will see '2 2 7' (for series=2, start frame=2, end frame=7) added to the **New Series List** list box.

B. Suppose that we are interested in all ten images belonging to series 1 on disk, but that we want to work with them as two different **logical series** because even though they all belong to the same series on disk, images 1 through 4 are actually unrelated to images 5 through 10. Suppose we want images 5 through 10 to be the second **logical series** in our output ImageSet object. Select **Series 1** from the **Series Number** drop-down menu and enter the value **5** in the **Start** text box and the value **10** in the **End** text box. Click the **Add** button and you will see '1 5 10' (for series=1, start frame=5, end frame=10) added to the **New Series List** list box.

C. Images 1 through 4 of series 1 on disk will be the third **logical series** in our output ImageSet object. Select **Series 1** from the **Series Number** drop-down menu and enter the value **1** in the **Start** text box and the value **4** in the **End** text box. Click the **Add** button and you will see '1 1 4' (for series=1, start frame=1, end frame=4) added to the **New Series List** list box. We have finished specifying the logical series of our output ImageSet object. Click **OK**.

D. In the input box that comes up, enter **subset** for the file name of our output ImageSet object to help us remember what it is. Click **OK**.

E. Click **OK** in the confirmation box that comes up.

- When specifying output logical series, if you make a mistake and want to remove an item added to the **New Series List** list box, simply select that item in the list box and click the **Delete** button.

- You can actually use the subset operation to create an ImageSet object with multiple pointers to the same physical image. For example, when specifying the output logical series, you can select the same input series more than once and specify the same frame numbers from that input series more than once. Multiple pointers to the same physical image will allow you to assign and store different settings (e.g., pixel intensity settings, crop settings, etc.) for that image.
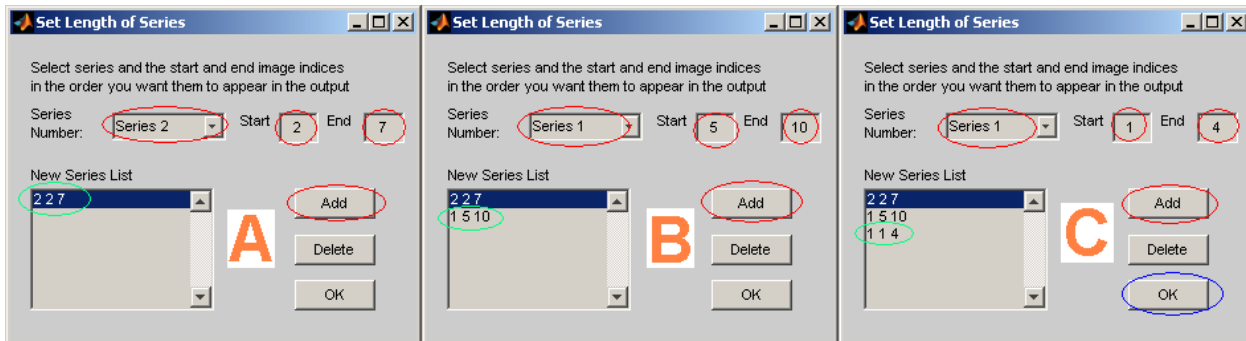

Figure 3.10.1: Specifying logical series of output ImageSet object of the subset operation

Step 5: *Verify output ImageSet object of subset operation has been saved to disk* Verify that a new ImageSet object has been saved under the MIATool directory 'C:\miatooldemo2'. You should find a new 'ImageSet2' subfolder under the 'MIAToolObjects' subfolder, and inside the 'ImageSet2' subfolder you should find the file 'subset_imageset.mia', the ImageSet object we have just created. **Note that we have not changed anything else in the MIATool directory. In particular, look in the 'ImageFiles' subfolder and verify that the two series of images are exactly as they were before. We have not changed the way the images are arranged on disk and we have not created any additional series subfolders or images. We have only created a new arrangement of pointers that reference these images on disk.**

Figure 3.10.2: Output ImageSet object of subset operation saved in MIATool directory on disk

Step 6: *Open MIATool viewer to view output ImageSet object of subset operation* Let's verify the output ImageSet object of the subset operation by viewing the images it points to. Select **MIATool → Open DisplayTool** from the MIATool main menu figure. In the list box that comes up, select the ImageSet object 'subset_imageset.mia' and click **OK**. We can intensity-adjust, crop, segment, and label the images referenced by this ImageSet object in exactly the same way as we've shown in this tutorial for default ImageSet objects. For our current purpose, let's load just a default DisplaySet object. In the next figure that pops up, select **DisplaySet** from the **Object Type** drop-down menu. Highlight **default DisplaySet** in the **Available Objects** list box and click the **Add** button to add it to the **Selected Objects** list box. Click **Done**. In the subsequent figure, simply click **Done** as we do not want a SegmentationSet or a LabelSet object.

- The default DisplaySet, CropSet, SegmentationSet, and LabelSet objects, if selected during the opening of the viewer, will all be configured automatically to match the arrangement of pointers in the ImageSet object they are being loaded with. In other words, they will be able to keep track and store settings independently for each pointer in the ImageSet object. The default DisplaySet object selected in this step, for example, will be able to keep track and store intensity settings for the 15 images referenced by the ImageSet object 'subset_imageset.mia'.
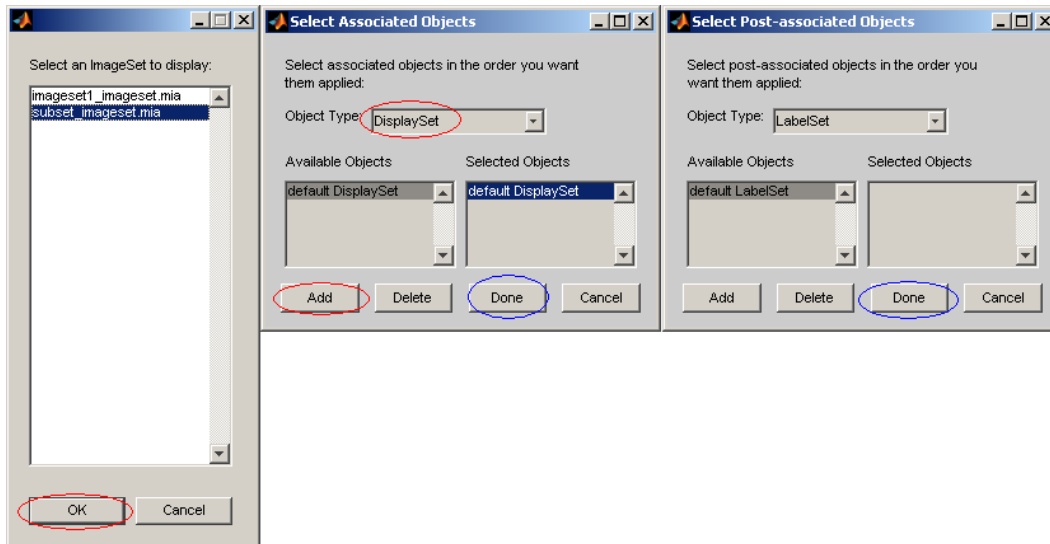
Figure 3.10.3: Opening MIATool viewer with output ImageSet object of subset operation

Step 7: *Verify correspondence between logical and physical image series and frame numbers* When the viewer opens, you will find two viewer sliders. The value of the first viewer slider corresponds to the **logical** series index. It has a maximum value of 3 for the three logical series we created. The value of the second viewer slider corresponds to the **logical** frame number within a logical series. For example, logical series 1 has six images with logical frame numbers 1 through 6. However, they correspond to frame numbers 2 through 7 of physical series 2 on disk. Note that logical series 2 also has six images (frame numbers 5 through 10 of physical series 1), but that logical series 3 has four images (frame numbers 1 through 4 of physical series 1). (See Figure 3.10.4.) Use the **Details** button in the viewer sliders figure to verify the correspondence between the logical series and frame numbers and the physical series and frame numbers. For example, for logical frame number 1 of logical series 1, the details figure should indicate the image file 's2_2.tif', which is frame 2 of series 2 on disk. Note that the 'Series' field in the details figure reports the logical series index, but the 'Index within series' field reports the physical frame number.

- For help on the **Details** button, refer to Steps 6 and 7 of Section 3.4.

- Open the pixel intensity adjustment tool and make some intensity adjustments on the images. After you are done, save the intensity settings to disk as a DisplaySet object. Then use Windows Explorer to verify that the DisplaySet object has been saved to a newly created 'displayset objects' subfolder under the 'ImageSet2' subfolder of the MIATool directory 'C:\miatooldemo2'.

- **Default** ImageSet objects are special in that they have a direct correspondence between the logical and physical series indices and frame numbers. That is, logical series 1 is the same as physical series 1, and logical frame number 1 is the same as physical frame number 1. Because of this direct correspondence, we have not made a distinction between logical and physical series and frame numbers until now. The fact is, however, that **when working with ImageSet objects, the 'series' that we speak of is always the logical series, which may or may not correspond to a physical series.**
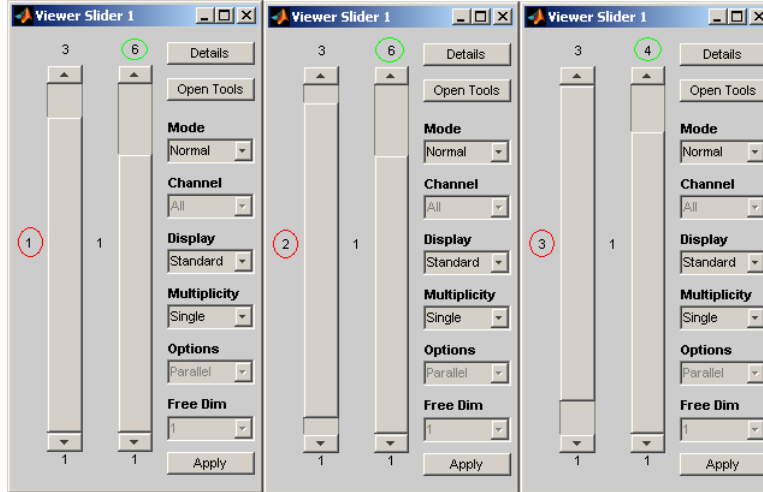
84

Figure 3.10.4: ImageSet object with three logical series

Step 8: *Close MIATool viewer* Close the MIATool viewer by clicking the close (×) button in the upper right-hand corner of the figure containing the viewer sliders.

3.10.2 Replicating image pointers

In this section, we demonstrate the use of the replicate operation to create an ImageSet object that contains multiple copies of all the image pointers in an input ImageSet object. The replicate operation takes as input the number of copies of the pointers to make, and creates an arrangement of pointers consisting of a stack of the specified number of identical copies of the array of pointers in the input ImageSet object. Consider for example a 2x10 array of image pointers, which is what we have used extensively in this tutorial. Suppose we specify to the replicate operation to make 3 copies of the 2x10 pointer array. Then the output pointer array is simply a stack of 3 copies of the 2x10 pointer array - a 2x10x3 array - where all 3 copies have the same orientation. That is, the pointers at positions (x,y,1), (x,y,2), and (x,y,3), x=1,2, y=1,2,...,10, of the 3D array are copies of the same pointer.

In general, **the replicate operation can accept any ImageSet object as input, and its output will always be an ImageSet object containing an image pointer array that has one more dimension than the image pointer array of the input ImageSet object.** Here we will demonstrate the replicate operation and give a practical example of its usefulness.

Step 9: *Initiate replicate operation* In the MIATool main menu figure, select **ImageSet Massaging → Replicate**.

- The use of the replicate operation is independent of the MIATool viewer. All that's required is that the MIATool directory containing the ImageSet object whose pointer array you want to replicate is loaded.

Step 10: *Choose ImageSet object to apply replicate operation to* Select the ImageSet object

85

'imageset1_imageset.mia'. Click **OK**. This is the default ImageSet object which contains two rows of ten image pointers each (i.e., a 2x10 pointer array). The first row of pointers references the intensity-adjusted and cropped images of series 1 on disk, and the second row of pointers references those of series 2 on disk.

Step 11: *Specify replication factor* In the input box that comes up, enter the value **3** to indicate that we want to make 3 copies of the image pointer array in the default ImageSet object. Click **OK**. (See Figure 3.10.5, left panel.)

Step 12: *Provide file name for output ImageSet object of replicate operation* In the input box that comes up next, enter **replicate** for the file name of our output ImageSet object to help us remember what it is. Click **OK**. (See Figure 3.10.5, middle panel.)

Step 13: *Replicate operation completed* In the confirmation box, click **OK**. (See Figure 3.10.5, right panel.)
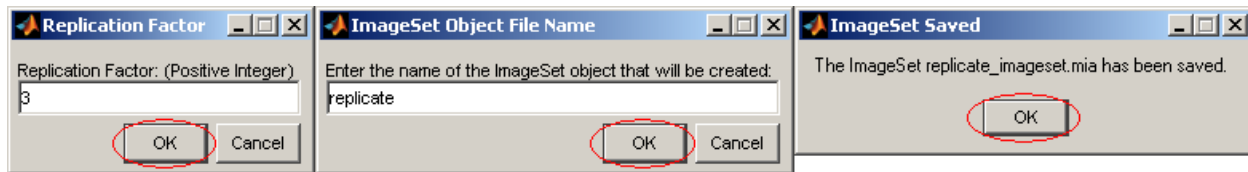
Figure 3.10.5: Specifying replication factor and file name of output ImageSet object

Step 14: *Verify output ImageSet object of replicate operation has been saved to disk* Verify that a new ImageSet object has been saved under the MIATool directory 'C:\miatooldemo2'. You should find a new 'ImageSet3' subfolder under the 'MIAToolObjects' subfolder, and inside the 'ImageSet3' subfolder you should find the file 'replicate_imageset.mia', the ImageSet object we have just created. **Note that as in the case of the subset operation, we have not changed anything else in the MIATool directory. We have not increased the number of physical image series on disk, nor have we created additional images on disk. We have only created a new arrangement of pointers where there are three pointers referencing each of the 20 images on disk.**

Figure 3.10.6: Output ImageSet object of replicate operation saved in MIATool directory on disk

Step 15: *Open MIATool viewer to view output ImageSet object of replicate operation*

86

Let's verify the output ImageSet object of the replicate operation by viewing the images it points to. Select **MIATool** → **Open DisplayTool** from the MIATool main menu figure. In t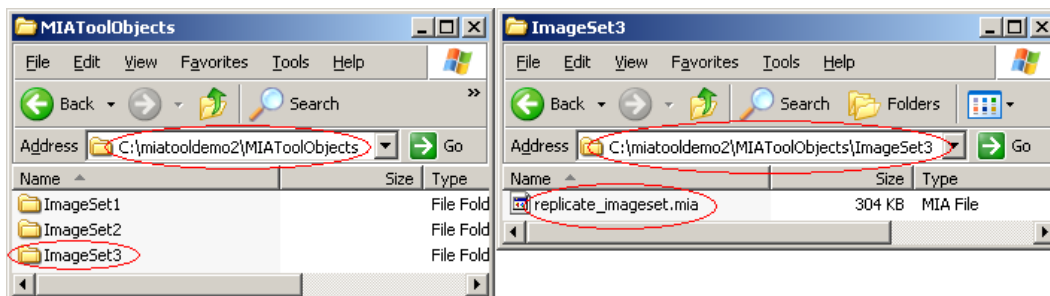he list box that comes up, select the ImageSet object 'replicate_imageset.mia' and click **OK**. We can intensity-adjust, crop, segment, and label the images referenced by this ImageSet object in exactly the same way as we've shown in this tutorial for default ImageSet objects. For our current purpose, let's load just a default DisplaySet object. In the next figure that pops up, select **DisplaySet** from the **Object Type** drop-down menu. Highlight **default DisplaySet** in the **Available Objects** list box and click the **Add** button to add it to the **Selected Objects** list box. Click **Done**. In the subsequent figure, simply click **Done** as we do not want a SegmentationSet or a LabelSet object.
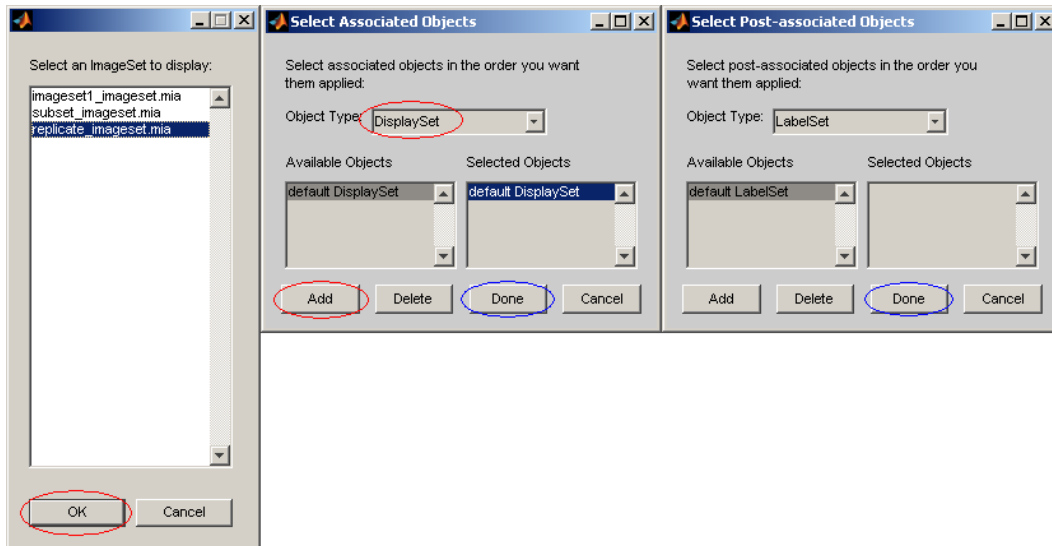

Figure 3.10.7: Opening MIATool viewer with output ImageSet object of replicate operation

Step 16: *Verify copies of same pointer reference same physical image* When the viewer opens, you will find three viewer sliders that will allow us to traverse a 2x10x3 pointer array (see Figure 3.10.8. Recall that our input was the default ImageSet object which has a 2x10 pointer array. When we specified a replication factor of 3, MIATool made 3 copies of the 2x10 pointer array and stacked them to form a 2x10x3 pointer array. Hence we have the 3 sliders for navigating through a 3D array. The first and second viewer sliders correspond as in the case of the default ImageSet object to the series number and frame number of the image on disk, respectively. Together the values of the first and second sliders specify a unique image on disk. For a given physical image specified by the first and second sliders, the value of the third viewer slider, which has a maximum value of 3, indicates which of the 3 copies of the pointer to that physical image we're currently working with. Perform the following to verify that these copies of the same pointer refer to the same physical image:

A. Set the first and second viewer sliders to any values you want, then scroll the third viewer slider and visually verify that the image displayed in the image window is the same regardless of the value of the third slider. This is because when you scroll the third slider, you're going through three copies of the same pointer, and hence the displayed image is the same image.

B. With the first and second viewer sliders set to any values you want, again scroll the third

87

viewer slider. As you change the value of the third slider, use the **Details** button in the viewer sliders figure to verify that you're looking at the same physical image. For example, if you set the first slider to 1 and the second slider to 3, then for all values of the third slider you should see that the physical image is frame 3 of the multi-frame tiff file 'ic_m1.tif'.

- For help on the **Details** button, refer to Steps 6 and 7 of Section 3.4.

- The ImageSet object 'replicate_imageset.mia' actually contains six **logical** image series, three of which correspond to physical series 1 and the other three correspond to physical series 2. The reason for this is because the input default ImageSet object had 2 logical series (Recall that for a default ImageSet object, there is a direct correspondence between logical and physical series. Therefore, 2 physical series means 2 logical series.), and since we specified a replication factor of 3, the replicate operation made 3 copies of the 2 logical series to give 6 logical series in the output ImageSet object. You can use the **Details** button in the viewer sliders figure to look at the logical series an image pointer belongs to. The logical series index is given in the **Series** field of the details figure. In general, given an input ImageSet object with N logical series and a replication factor of M, the output ImageSet object of the replicate operation will have $N \times M$ logical series.

- The **Apply** button in the viewer sliders figure actually converts logical series to physical series on disk. Since the ImageSet object 'replicate_imageset.mia' has six logical series, using the **Apply** button now will cause six series subfolders of ten images each to be written to a new MIATool directory on disk. Furthermore, the default ImageSet object created will have a 6x10 image pointer array.
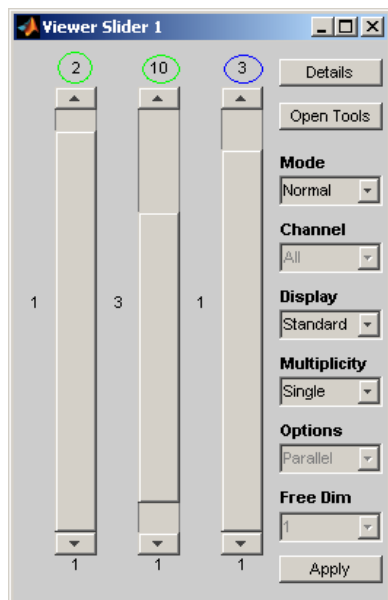

Figure 3.10.8: ImageSet object with a 3D image pointer array

Step 17: *Use copies of same pointer to view overlays of same images with different intensity settings* Let us now see a practical application using the ImageSet object 'replicate_imageset.mia'. Suppose we want to view overlays of the correspondingly numbered images in

physical series 1 and 2, at three different pixel intensity settings. We can do this with the default ImageSet object, but because there we will only have one pointer per physical image, we will have to save the three different intensity settings in three different DisplaySet objects. Furthermore, with one instance of the viewer, at any given time we will only be able to view the overlays with one of the intensity settings. In contrast, because the ImageSet object 'replicate_imageset.mia' has three pointers per physical image, we can save the three different intensity settings in a single DisplaySet object. Moreover, with one instance of the viewer, we will be able to quickly go back and forth between the three different intensity settings. Perform the following:

A. Open the pixel intensity adjustment tool by clicking the **Open Tools** button in the viewer sliders figure. In the list box that comes up, select **1: displayset** and click **OK**.

B. In the intensity adjustment tool figure, make sure the **Method** drop-down menu is set to **Imadjust** as we will be using the 'imadjust' method to adjust the pixel intensities.

C. For our first intensity setting, enter **40000** in the text box associated with (to the left of) the **HighIn** slider in the intensity adjustment tool figure. You should immediately see a change in the displayed image. Click the **Save** button and in the list box that appears, select **Apply to Subset of Images** and click **OK**. In the Subset Selection figure that comes up next, you should see three text boxes since we are dealing with a 3D pointer array. The leftmost box corresponds to the first dimension/slider, the middle box corresponds to the second dimension/slider, and the rightmost box corresponds to the third dimension/slider. Enter **all** in the leftmost text box, **all** in the middle text box, and **1** in the rightmost text box. These values specify that we want to apply the current intensity setting to the first copy of pointers to all 20 physical images. Click **Done**. Click **OK** in the confirmation box.

D. For our second intensity setting, enter **20000** in the text box associated with (to the left of) the **HighIn** slider in the intensity adjustment tool figure. You should immediately see a change in the displayed image. Click the **Save** button and in the list box that appears, select **Apply to Subset of Images** and click **OK**. In the Subset Selection figure that comes up next, enter **all** in the leftmost text box, **all** in the middle text box, and **2** in the rightmost text box. These values specify that we want to apply the current intensity setting to the second copy of pointers to all 20 physical images. Click **Done**. Click **OK** in the confirmation box.

E. For our third intensity setting, enter **10000** in the text box associated with (to the left of) the **HighIn** slider in the intensity adjustment tool figure. You should immediately see a change in the displayed image. Click the **Save** button and in the list box that appears, select **Apply to Subset of Images** and click **OK**. In the Subset Selection figure that comes up next, enter **all** in the leftmost text box, **all** in the middle text box, and **3** in the rightmost text box. These values specify that we want to apply the current intensity setting to the third copy of pointers to all 20 physical images. Click **Done**. Click **OK** in the confirmation box.

F. If you like, use the **Disk** button in the intensity adjustment tool figure to save the intensity settings we've created to disk as a DisplaySet object. The disked DisplaySet object will reside in a newly created 'displayset objects' subfolder under the 'ImageSet3' subfolder of the MIATool directory 'C:\miatooldemo2'.

G. You can close the intensity adjustment tool figure if you want by clicking its close (×) button.

H. With the first 2 viewer sliders set to any values you like, scroll the third viewer slider to see the same physical image displayed with different intensity settings.

I. Select **Multiple** from the **Multiplicity** drop-down menu in the viewer sliders figure. Since the viewer defaults to the parallel display mode and the first dimension as the 'free dimension', this will display correspondingly numbered images in physical series 1 and 2 simultaneously, but in separate windows. Position the two image windows side by side and scroll the third viewer slider to see the pair of images with different intensity settings.

J. To view overlays of correspondingly numbered images in physical series 1 and 2, select **Overlay** from the **Options** drop-down menu in the viewer sliders figure. Again, scroll the third viewer slider to see overlays of the same pair of physical images with different intensity settings. Figure 3.10.9 shows the three different overlays for the first pair of physical images.

- While in overlay mode, select **3** from the **Free Dim** drop-down menu in the viewer sliders figure. Scroll the first and second viewer sliders to view overlays of the three differently intensity-adjusted versions of each physical image.

- With the **Free Dim** drop-down menu in the viewer sliders figure set to **3**, select **Parallel** from the **Options** drop-down menu. Scroll the first and second viewer sliders to view simultaneously the three differently intensity-adjusted versions of each physical image in three separate windows.
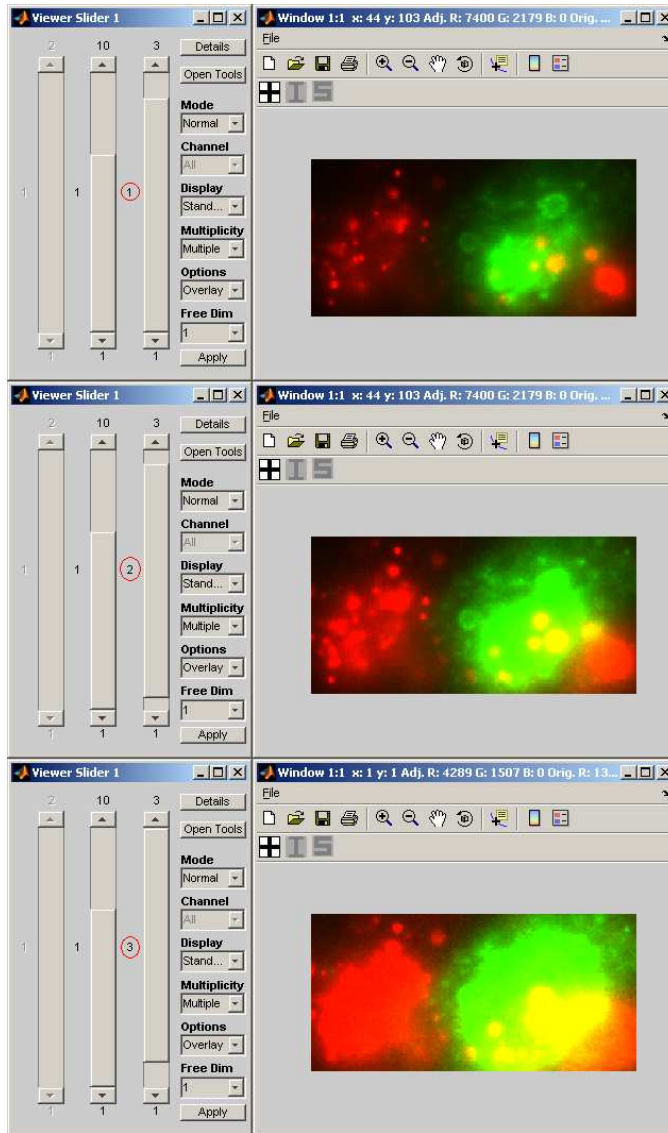
Figure 3.10.9: Overlays of a pair of images with three different intensity settings

Step 18: ***Close MIATool viewer and main menu*** We are done with this part of the tutorial. Close the MIATool viewer by clicking the close ($\boxed{\times}$) button in the upper right-hand corner of the figure containing the viewer sliders. You can close the main menu figure as well by clicking its close ($\boxed{\times}$) button.

3.11 Multiple open viewers

It is perhaps clear by now that a single instance of the MIATool viewer can support the viewing of only one ImageSet object. So far in this tutorial, we have had only a single instance of the viewer open at any given time. MIATool , however, does support having multiple viewers open at the same time, meaning that different ImageSet objects can be viewed and processed concurrently in separate viewers. There is one limitation - these ImageSet objects must all reside in the MIATool

91

directory that is currently loaded in the MIATool main menu figure.

In this section, we demonstrate two simultaneously open viewers using the MIATool directory 'C:\miatooldemo2', which by this point in the tutorial should have three ImageSet objects we can work with.

Step 1: *Open MIATool main menu and load MIATool directory* Make sure that you have the MIATool main menu figure opened and the MIATool directory 'C:\miatooldemo2' loaded.

- Refer to Section 3.1 for help on opening the main menu figure, and Section 3.3 for help on loading the MIATool directory.

Step 2: *Open one instance of MIATool viewer with output ImageSet object of subset operation* Let's open one instance of the viewer with the ImageSet object generated with the subset operation in Section 3.10.1. Select **MIATool → Open DisplayTool** from the MIATool main menu figure. In the list box that comes up, select the ImageSet object 'subset_imageset.mia' and click **OK**. In the next figure that pops up, select **DisplaySet** from the **Object Type** drop-down menu. Highlight **default DisplaySet** or any other DisplaySet object you may have created in the **Available Objects** list box. Click the **Add** button to add it to the **Selected Objects** list box. Click **Done**. In the subsequent figure, optionally use the **Object Type** drop-down menu and the **Add** button to choose a default LabelSet object or a default SegmentationSet object or both. Click **Done**. The viewer should now be opened.

Step 3: *Open second instance of MIATool viewer with output ImageSet object of replicate operation* Without first closing the viewer we've just opened, let's open a second instance of the viewer with the ImageSet object generated with the replicate operation in Section 3.10.2. Select **MIATool → Open DisplayTool** from the MIATool main menu figure. In the list box that comes up, select the ImageSet object 'replicate_imageset.mia' and click **OK**. In the next figure that pops up, select **DisplaySet** from the **Object Type** drop-down menu. Highlight **default DisplaySet** or any other DisplaySet object you may have created in the **Available Objects** list box. Click the **Add** button to add it to the **Selected Objects** list box. Click **Done**. In the subsequent figure, optionally use the **Object Type** drop-down menu and the **Add** button to choose a default LabelSet object or a default SegmentationSet object or both. Click **Done**. A second instance of the viewer should now be opened.

Step 4: *Verify independence of the two open viewers* Scroll the viewer sliders of both viewers and verify that they control only the image displayed in their respective image window. In other words, verify that the two viewers work independently of each other.

Step 5: *Verify independence of each viewer's associated intensity adjustment tool* Use the **Open Tools** button of each viewer to open its pixel intensity adjustment tool. (Choose **1: displayset** in the list box that comes up and click **OK**.) Play with the intensity settings of the current image in each viewer using the viewer's associated intensity adjustment tool. Verify that each intensity adjustment tool controls only the intensity settings of the current image in its associated viewer.

- If you chose a default Labelset object or a default Segmentationset object or both, open the label tool and/or manual segmentation tool associated with each viewer. Add labels and components and verify that the tools apply only to the current image in their associated viewer.

- Use the **Disk** button in each intensity adjustment tool to save to disk the intensity settings for the images displayed in its associated viewer. Verify, using Windows Explorer, that for the first instance of the viewer, the disked DisplaySet object resides under the 'ImageSet2' subfolder (where the ImageSet object created from the subset operation is located) of the MIATool directory 'C:\miatooldemo2'. Verify that for the second instance of the viewer, the disked DisplaySet object resides under the 'ImageSet3' subfolder (where the ImageSet object created from the replicate operation is located).

- For a given viewer, the intensity adjustment tool, the crop tool, the segmentation tool, and the label tool can be open and used at the same time. There is, however, one word of caution. If you want to do both cropping **and** either segmentation or labeling or both, it is recommended that you decide on the crop settings first before you do any segmentation or labeling. Once you have the crop settings you want and have started to create components and/or labels, do not change the crop settings again. Furthermore, the next time you open the viewer with the LabelSet object and/or SegmentationSet object previously created under certain crop settings, make sure that you also select the CropSet object that will give you those crop settings. If these recommendations are not followed, then you will not see the labels and/or components at the correct locations in the image.

- Open a third instance of the viewer with the default ImageSet object 'imageset1_imageset.mia'. Load whatever objects you like and verify that you can work with it independently of the other two viewers.

Step 6: *Close MIATool viewers and main menu* We are done with this part of the tutorial. Close each MIATool viewer by clicking the close (×) button in the upper right-hand corner of its viewer sliders figure. You can close the main menu figure as well by clicking its close (×) button.

3.12   Quick viewing of images without the MIATool directory

So far in this tutorial we have demonstrated the standard usage of MIATool where a MIATool directory is first created/loaded before a viewer can be opened to view the images. Though it is not considered to be standard usage and comes with many limitations, MIATool supports a quick way to view images without having to go through a MIATool directory. This alternative method of image viewing is referred to as **simple viewing** because it is intended for viewing images only and offers very limited processing capabilities. It is recommended that all processing is done through the standard usage of MIATool .

Instead of specifying a MIATool directory and an ImageSet object, simple viewing only requires that you specify the folders containing the images you want to view. However, the images in a given folder must conform to one of the following formats:

- **n.tif** - Images are **individual**, **16-bit**, **uncompressed tiff** files. If there are N images, then

93

they must be named '1.tif', '2.tif', ..., 'N.tif'.

- **multi-frame tif** - Images are frames of a **single**, **16-bit**, **uncompressed**, **multi-frame tiff** file. The multi-frame tiff file can have any file name of the form **'anyname.tif'**.

- **MIATool n.tif** - Individual tiff files in a 'Series' subfolder of a MIATool directory.

- **MIATool multi-frame tif** - Frames of a multi-frame tiff file in a 'Series' subfolder of a MIATool directory.

Simple viewing is limited in the following ways:

- You will be given the option to choose a default CropSet object and a default DisplaySet object during the opening of the viewer, which means that you will be able to crop the images and make intensity adjustments to the images during viewing. However, the **Disk** buttons in the crop tool and the intensity adjustment tool will not work properly, and you will **not** be able to save crop settings and intensity settings to disk for later use. Once the viewer is closed, all settings will be lost.

- The manual segmentation tool and the label tool will not be available.

- You will not be able to do ImageSet massaging.

- You will not be able to create overlay images on disk from the images you choose for simple viewing.

- The **Apply** button in the viewer sliders figure will not work properly. You will not be able to write any images, processed or unprocessed, to disk.

- Only images conforming to one of the four formats enumerated above can be viewed with simple viewing.

We now give a demonstration of the viewing of three series of images using simple viewing.

Step 1: *Open MIATool main menu* Make sure that you have the MIATool main menu figure opened.

- Refer to Section 3.1 for help on opening the main menu figure.

Step 2: *Initiate simple viewing* Select **Simple Viewing** → **Simple Viewing** from the MIATool main menu figure.

Step 3: *Specify image series to view using simple viewing* Perform the following to specify three series of images to view using simple viewing:

A. The first series of images we will choose is the grayscale multi-frame tiff file in the sample data subfolder under your MIATool installation directory. In the figure that appears, click

the **Browse** button and use the directory browser to select the folder '\V1.1\SampleData \grayscale_multiframe_tif' under your MIATool installation directory. Click **OK**. The 'Series Path' text box will now contain the full path to the selected folder. Because the images in the selected folder are in the form of a single multi-frame tiff file, make sure the option **multi-frame tif** is selected in the 'Series Type' drop-down menu. Click **OK**.

B. The second series of images we will choose is the grayscale multi-frame tiff file in the **Series1** subfolder of the MIATool directory 'C:\miatooldemo2'. In the figure that comes up, click the **Browse** button and use the directory browser to select the folder 'C:\miatooldemo2\ImageFiles\Series1'. Click **OK**. The 'Series Path' text box will now contain the full path to the selected folder. Because this is a multi-frame tiff file in a MIATool directory, make sure the option **MIATool multi-frame tif** is selected in the 'Series Type' drop-down menu. Click **OK**.

C. The third series of images we will choose is the series of five individual RGB tiff images in the **Series4** subfolder of the MIATool directory 'C:\miatooldemo3'. In the figure that comes up, click the **Browse** button and use the directory browser to select the folder 'C:\miatooldemo3\ImageFiles\Series4'. Click **OK**. The 'Series Path' text box will now contain the full path to the selected folder. Because this is a series of individual tiff files in a MIATool directory, make sure the option **MIATool n.tif** is selected in the 'Series Type' drop-down menu. Click **OK**.

D. We have now specified the three series of images we want to view. In the next figure that appears, click **Cancel**.

E. The next figure gives us the option to open the viewer with a default CropSet object and/or a default DisplaySet object. Choose a default CropSet object by highlighting **default CropSet** in the **Available Objects** list box and clicking the **Add** button to add it to the **Selected Objects** list box.

F. Choose a default DisplaySet object by highlighting **default DisplaySet** in the **Available Objects** list box and clicking the **Add** button to add it to the **Selected Objects** list box. Click **Done**.
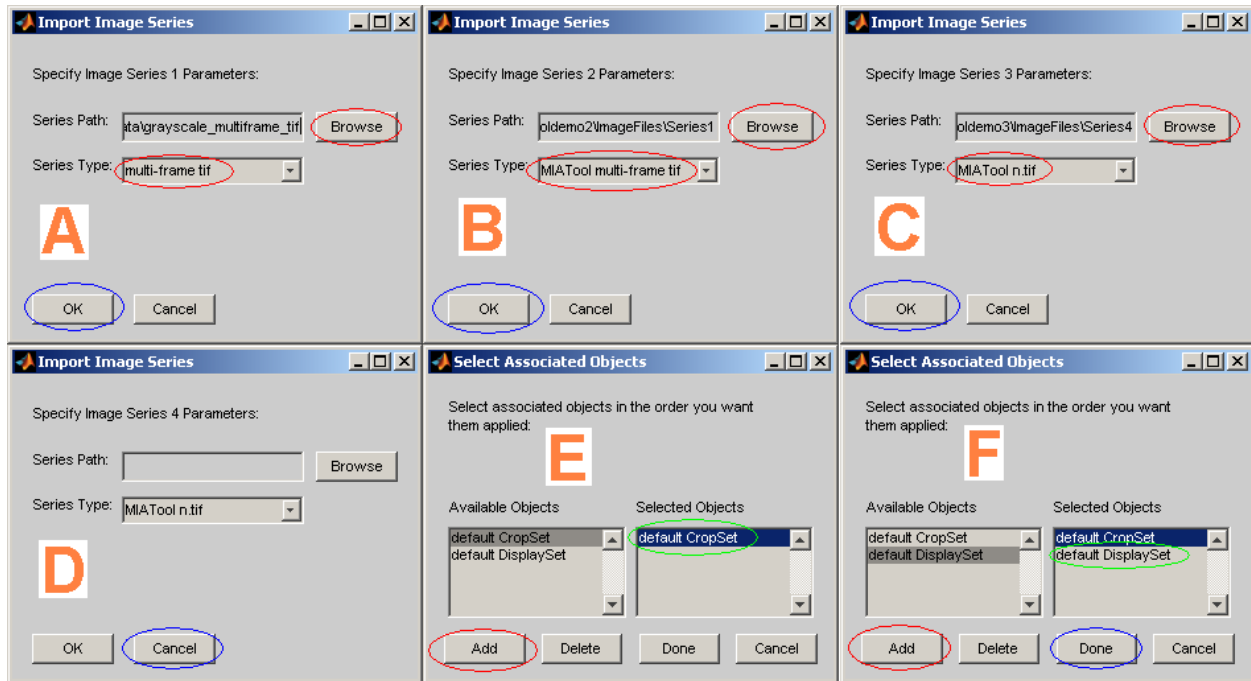
Figure 3.12.1: Specifying series of images to view using simple viewing

Step 4: *View and verify selected series of images* In the viewer that opens, you will find two viewer sliders. The first slider has a maximum value of 3 and its value indicates the index of the image series you are looking at. The three image series we chose are numbered in the order we specified them during the opening of the viewer. The value of the second slider corresponds to the image/frame index within a series. Depending on the series you are looking at, the second slider has different maximum values. Do the following:

A. Scroll the viewer sliders and visually verify that the first series consists of ten unprocessed grayscale images where nothing can be seen, that the second series consists of ten grayscale intensity-adjusted and cropped images, and that the third series consists of five RGB intensity-adjusted and cropped overlay images. See Figure 3.12.2.

B. Use the **Details** button in the viewer sliders figure to verify the physical locations of these images on disk.

C. Use the **Open Tools** button in the viewer sliders figure to open the intensity adjustment tool (select **2: displayset** from the list box and click **OK**). Notice as you scroll the viewer sliders that various controls in both the viewer sliders figure and the intensity adjustment tool figure are enabled or disabled depending on whether the current image is grayscale or RGB. These controls are enclosed in blue boxes in Figure 3.12.2.

D. Use the intensity adjustment tool to play with the intensity settings of the images. However, be sure that when using the **Save** button, keep in mind that settings for a grayscale (RGB) image can only be applied to other grayscale (RGB) images. The software will not work

96

correctly if you try to apply settings for a grayscale image to RGB images, or vice versa. Also recall that the **Disk** button will **not** work properly!

- Simple viewing always opens with two viewer sliders. The value of the first slider corresponds to the series index (the image series are numbered in the order you specified them during the opening of the viewer), and the value of the second slider corresponds to the image/frame index within a series.

- With simple viewing you are actually working with an ImageSet object. Unlike the standard usage of MIATool , however, this ImageSet object is internal and you will not be able to save a copy of it to disk.

- As shown in this example, the MIATool viewer can work with ImageSet objects that reference both grayscale and RGB images. This is true for both simple viewing and the standard usage of MIATool. Keep in mind, however, that when working with RGB images, the various restrictions detailed in Section 3.7.1 will apply.

- Open the crop tool and play with the crop settings of the images. Recall that cropping works the same for grayscale and RGB images. However, since we have images of different sizes, keep in mind that if you want to apply a crop setting to images of different sizes, the setting must be valid for **all** the image sizes involved.
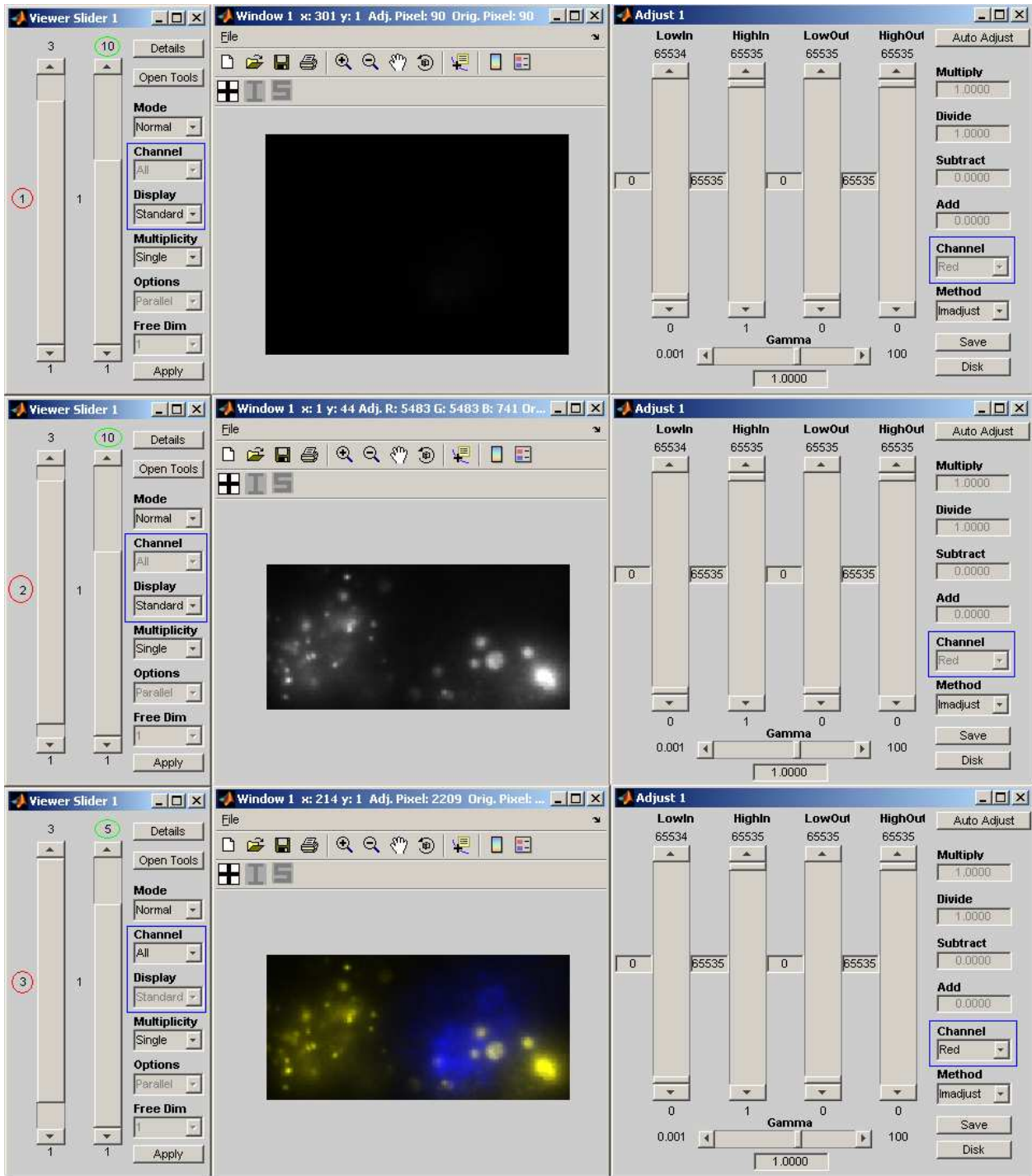
Figure 3.12.2: Three series of images viewed using simple viewing

Step 5: *Simple viewing creates nothing on disk* Note that simple viewing does not move or make copies of the source images on disk. The sources image are loaded for display from their original locations on disk. In fact, simple viewing does not create anything on disk.

Step 6: *Close MIATool viewer and main menu* We are done with this part of the tutorial. Close the MIATool viewer by clicking its close (✕) button in the upper right-hand corner of its viewer sliders figure. You can close the main menu figure as well by clicking its close (✕) button.

3.13   Deleting objects from MIATool directories

MIATool does not currently support the deletion of images or objects from MIATool directories. All deletions must be done directly through Windows Explorer. Unless you want to discard an entire MIATool directory, in which case you can simply delete the entire directory structure using Windows Explorer, it is not recommended that you delete anything from a MIATool directory. In particular, to guarantee that MIATool will be able to continue to work with a MIATool directory, you should not delete any images or ImageSet objects and the folders that contain them. You should also take care not to delete the object 'miatooldir_miatooldirectory.mia'.

Though not recommended, you can, however, use Windows Explorer to delete CropSet objects, DisplaySet objects, SegmenationSet objects, and LabelSet objects from a MIATool directory. If you want to, you can also delete an entire 'displayset objects', 'cropset objects', 'segmentationset objects', or 'labelset objects' subfolder. You might choose to delete an object when you have mistakenly saved an unwanted object to disk, or perhaps when you've saved too many objects to disk and you want to remove any unnecessary ones to save disk space. Whatever the reason you have for deletion, please do it with care as MIATool does not provide any means of recovering deleted objects.

After you've deleted objects or entire subfolders containing objects of the four types mentioned, you must **refresh** the MIATool directory so that MIATool will continue to be able to work with it. The procedure for object deletion from a MIATool directory is as follows:

A. Use Windows Explorer to delete particular CropSet objects, DisplaySet objects, SegmenationSet objects, and LabelSet objects from a MIATool directory. You can also delete an entire 'displayset objects', 'cropset objects', 'segmentationset objects', or 'labelset objects' subfolder.

B. Open the MIATool main menu figure.

C. Select **MIATool → Refresh MIATool Directory** from the MIATool main menu figure.

D. Use the directory browser that comes up to select the root folder of the MIATool directory you have deleted objects from. For example, if you have deleted objects from the MIATool directory 'C:\miatooldemo2', then use the browser to select the folder 'C:\miatooldemo2'.

E. The refresh is complete when you see a confirmation box stating that the refresh is complete. Click the **OK** button in the confirmation box to finish the process.

After refreshing a MIATool directory, it is likely that when you open the viewer with one of its ImageSet objects, the selection boxes listing the remaining DisplaySet, CropSet, SegmentationSet, and LabelSet objects for you to choose from will list the objects in an order different from before you deleted objects and refreshed the MIATool directory. This will be true for all ImageSet objects

in the refreshed MIATool directory, even the ones you did not delete any associated objects of. This is a normal side effect of refreshing a MIATool directory that does not affect the normal functioning of MIATool .

3.14   Miscellaneous

Here we give brief descriptions of a number of things the previous sections of the tutorial did not touch on.

- You can move an entire MIATool directory to a different location on disk using Windows Explorer. The first time you load a MIATool directory from a new location, you may notice a longer loading time. If there are many objects in the MIATool directory, then the loading time can be significantly longer than normal. This is a normal behavior on the part of MIATool and should only happen the first time you load a MIATool directory from a new location. Subsequent loading from the new location should have the normal loading time.

- You have probably noticed in the MIATool viewer's image window the toolbar above the crosshair, the intensity profile, and the intensity profile scaling toolbar buttons. This toolbar is not specific to MIATool , but rather is common to all figures in MATLAB. Of the available buttons in it, the **Zoom In** (magnifying glass with plus sign), **Zoom Out** (magnifying glass with minus sign), and **Pan** (hand) buttons may be especially useful to the MIATool user. You can use them to zoom in, zoom out, and pan the current image. The settings that you choose for the current image using these buttons will be applied to the other images when you scroll the viewer sliders. To deactivate any of these buttons, simply click on it again.

- You have probably noticed the **Fast** option available in the **Mode** drop-down menu of the viewer sliders figure. This option is intended to allow you to view the images at a faster rate as you scroll through them. It is, however, not currently well-supported and not recommended as part of the normal usage of MIATool . It is nevertheless functional, and you can try it if you like. It does, however, impose the following limitations:

    - You cannot use the crop tool, the intensity adjustment tool, the label tool, or the segmentation tool while you are in the **Fast** viewing mode. All the controls in these tools will be disabled.
    - Even if you have the crosshair activated, you will not be able to see and use the crosshair while your are in the **Fast** viewing mode.
    - Intensity profiles will not update as you scroll through the images in **Fast** viewing mode.
    - Even if you have a SegmentationSet object and/or a LabelSet object loaded, you will not be able to see the components and labels that are visible in the **Normal** viewing mode.

References

[Chao et al., 2007] Chao, J., Long, P., Ward, E. S., and Ober, R. J. (2007). Design and application of the Microscopy Image Analysis Tool. *Engineering in Medicine and Biology Workshop, 2007 IEEE Dallas*, pages 94–97.