



A User's guide to DSCoMAN

DSCoMAN version 1.0

1 Introduction

Although imaging of perfusion using T_2 or T_2^* -weighted MR images (called *dynamic susceptibility MR imaging*) has been performed for many years, the methods used to analyze these images are, as a rule, far from widely accepted. The use of this type of imaging, while promising, is arguably far from validated for any particular clinical application.

One barrier to the validation of these techniques is that there is a lack of widely available reference methods to analyze these images. In the absence of comparable acquisition and analysis methods, it is difficult to directly compare quantitative results obtained in one laboratory with that of another. The function of a reference technique is not necessarily to provide the ideal analysis scheme, but to provide a reasonable method that can be easily applied for the purposes of comparison.

As a first step towards providing a reference technique, particularly for the application of dynamic susceptibility MR imaging to brain tumors, we have developed the DSCoMAN software package.

DSCoMAN stands for **D**ynamic **S**usceptibility **C**ontrast **M**R **A**nalysis. This software implements a method of dynamic susceptibility contrast (DSC) MR analysis that takes into account that the blood-brain barrier is disrupted in patients with high grade gliomas. When there is a significant blood-brain barrier breakdown, then the relative cerebral blood volume (rCBV) obtained by trapezoidal integration of concentration time curves underestimates true cerebral blood volume due to contrast agent leakage[1]. The DSCoMAN software implements the method described by Boxerman et al [1]. The cerebral blood volume maps corrected using this method were recently shown to more accurately correlate with tumor grade in patients with glioma than uncorrected methods. The DSCoMAN software generates the corrected relative Cerebral Blood Volume maps from DSC perfusion images. It can also generate maps of K₂ (a measurement of contrast agent leakage). In addition, DSCoMAN can be used to create Time to Peak (TTP) maps, as well as estimates of relative Mean Transit Time (rMTT) and the relative Cerebral Blood Flow (rCBF).

2 What is needed to run DSCoMAN?

2.1 Hardware requirement

DSCoMAN is designed to run on any hardware running an operating system that supports ImageJ, a public domain open source Java-based image processing program authored and maintained by Wayne Rasband at the National Institute of Mental Health. The operating systems supported by ImageJ include Mac, Windows and Linux platforms.

2.2 Software requirement

DSCoMAN has been implemented as plugins to ImageJ. ImageJ (rsb.info.nih.gov/ij/) must be installed before the DSCoMAN plugins. We recommend installing the version 1.36 or a later version of ImageJ bundled with Java 1.5. This can be obtained from <http://rsb.info.nih.gov/ij/download.html>.

2.3 MR Imaging data

DSCoMAN software uses images acquired from dynamic susceptibility MR imaging as input. Most frequently, we expect that these images will be available as “DICOM images,” although any 8-bit or 16-bit image format that can be opened in ImageJ could be used (for example, uncompressed tif, jpg or Analyze image). One advantage of using DICOM images is that DSCoMAN will take default values of parameters such as slice locations, echo time etc. from the DICOM header.

This software considers the opened dynamic susceptibility MR imaging set to be a “4D” stack of images in (xy)zt order composed of images repeated at the same number of image locations (n_l) for a certain number of time points (n_t). In fact, the stack of images in the image set should be composed of (image width) * (image height) * n_l * n_t pixels. It is important that some of the early time points be obtained before the contrast-agent is administered to establish the baseline signal intensity. The data obtained earliest should begin the stack, with subsets proceeding in order of increasing time so that the data obtained last ends the stack.

HINT: If your stack is in (xy)tz order, Michael Abramoff has provided the handy Hypervolume Shuffler (rsb.info.nih.gov/ij/plugins/hypervolume-shuffler.html) that can place these into (xy)zt order.

For the purposes of making stacks, it certainly helps to have all of the relevant images in a single directory.

3 Installation procedure

3.1 Install ImageJ

If you have not already installed ImageJ, download and install this free program by following the instructions at rsb.info.nih.gov/ij/download.html. We recommend installing the version 1.36 or a later version of ImageJ bundled with Java 1.5. The DCSoMAN software requires Java 1.5.

3.2 Adjust Memory

In order to handle large sets of images, maximize the RAM available to ImageJ. Use the *Edit/Options/Memory* command to make more than the default 128MB available to ImageJ. It often helps to increase the memory allocation option to approximately 2/3 of available RAM (e.g. `-mx170m` on a 256MB machine). Larger increases may lead to memory thrashing. Another way to make more memory available to ImageJ is by running from the command line and using the `-Xmx` option. Note that

ImageJ is limited to 64MB when you run it by double clicking on ij.jar.

3.3 Install the DSCoMAN plugins

Follow these steps:

1. Make a directory called DSCoMAN in your ImageJ plugins folder. Unzip the DSCoMAN software and save it in the DSCoMAN directory in your plugins folder. The zip file will contain a jar file called DSCoMAN_1.0.jar containing the DSCoMAN classes. The zip file will also contain class files for the ROITimeCourse and the HyperVolume plugins used by the DSCoMAN software.

Note: The ROITimeCourse and the HyperVolume class files are also a part of the TOPPCAT package. If you already have TOPPCAT in your ImageJ/plugins folder, then you do not need to put these class files into your DSCoMAN folder. (ImageJ will give a Duplicate Command warning, if you have multiple classes with the same name in your plugins folder).

2. This version of DSCoMAN depends on other utility classes that can be obtained by clicking the link for “Download the Utilities” on the DSCoMAN webpage. Unzip the file called DBLAB_1.0.zip and save it into the main ImageJ plugins folder i.e. ImageJ_Home\plugins where Image_Home is the home directory of your ImageJ installation. This will save the DBLAB.jar into the main plugins folder.

Note: If you had a previous version of DBLAB.jar in your ImageJ/plugins folder, overwrite it with the latest DBLAB.jar available on the DSCoMAN webpage.

3. Now, restart ImageJ. You should see three different items under DSCoMAN in the plugins menu: Boxerman Weisskoff, DSCoMAN GUI and Raw Perfusion. The DSCoMAN GUI is a user interface for calling other plugins. The Boxerman Weisskoff plugin generates the corrected rCBV map using the methods described in [1]. The Raw Perfusion plugin generates maps of parameters such as Time To Peak (TTP), relative Mean Transit Time (rMTT) and relative Cerebral Blood Flow (rCBF).

4 DSCoMAN Operation

DSCoMAN can be used either in manual mode or in macro mode. To get familiar with the DSCoMAN software and to perform small scale analyses, we suggest using manual mode. Macro mode is useful when you need to perform a number of repetitive analyses.

4.1 Manual mode

The easiest way to use manual mode is to click on the ImageJ plugins menu and then click on the DSCoMAN GUI plugin. This will show an interface as seen in Figure 1.

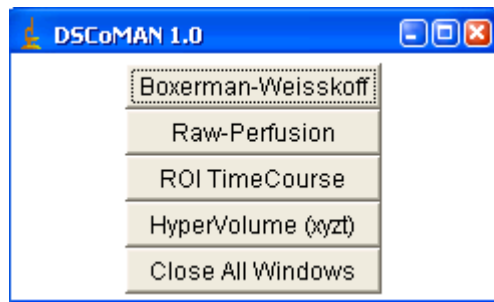


Figure 1: DSCoMAN 1.0 GUI

Using ImageJ, open the dynamic susceptibility contrast MR images that you want to analyze as a stack of images. You can then use the menu items on the DSCoMAN GUI shown in Figure 1 to perform different functions as described in this section.

4.1.1 Boxerman-Weisskoff

The Boxerman-Weisskoff plugin generates the corrected rCBV map using the methods described in [1].

To perform Boxerman-Weisskoff analysis on a set of dynamic susceptibility contrast MR images:

1. Open the dynamic susceptibility contrast MR images as a stack in ImageJ (for example, use File ... Import ... Image Sequence).
2. Place a region of interest in the background of the image and then click on Boxerman-Weisskoff in the menu in Figure 1. The background ROI is required for computing the whole brain mask. Boxerman-Weisskoff analysis is only performed on whole brain voxels defined in this mask.

Note that the Boxerman-Weisskoff plugin requires the information about the image times at each time point in order to perform the analysis. If this information is available in the image header, then on clicking on Boxerman-Weisskoff in the menu in Figure 1, it will automatically read the image times and display a file called ImageTimeTable.txt containing the time for each time point. However, if this information is not available in the image header, then the user must supply the ImageTimeTable.txt file containing the time for each time point (one time per line).

3. When you click on the Boxerman-Weisskoff menu in Figure 1, the Boxerman-Weisskoff control panel shown in Figure 2 should appear. This control panel allows the user to provide various input parameters. It also allows the user to select the maps that should be displayed. Section 4.1.1.1 describes the different images that can be generated using Boxerman-Weisskoff. Section 4.1.1.2 describes the numeric input parameters in Figure 2.
4. Click on the OK button to activate the plugin.

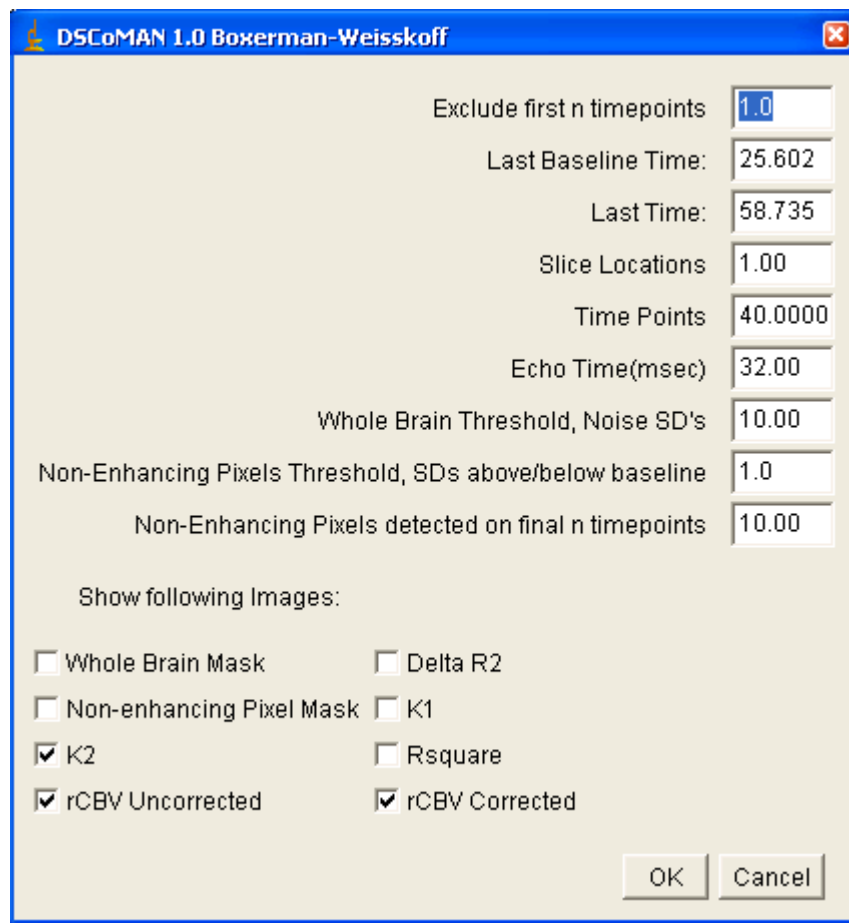


Figure 2: DSCoMAN 1.0 Boxerman-Weisskoff Control Panel

4.1.1.1 Map selection

In Figure 2, you can use the checkboxes to choose which images/maps should be generated. This section will describe the images that can be generated by the Boxerman-Weisskoff plugin. This will also help in understanding some of the numeric parameters in the control panel in Figure 2.

Whole Brain Mask: This is the first step before starting any actual analysis. We compute a whole brain mask for each slice in the dataset so that only locations with an average baseline signal intensity more than “x” times the standard deviation of background noise are included. Here, x specified by the user using the input parameter **Whole Brain Threshold, Noise SD’s**). Background noise standard deviation is calculated using the user defined background ROI. Further analysis is done only on voxels that are present in the whole brain mask.

Non-enhancing Pixel Mask: This mask is computed for each slice in the dataset and consists of those voxels that did not demonstrate signal intensity enhancement (averaged over the final “x” time points), greater than “y” standard deviation above that pixel’s average baseline intensity. Here, x and y are user specified (x specified by the input parameter **Non-Enhancing Pixels detected on final n timepoints**, and y is specified using the input parameter **Non-Enhancing Pixels Threshold, SDs above/below_baseline**).

Delta R2: This represents the relaxivity-time curve, a parameter related to the concentration of gadolinium in the voxel. It is computed as $-(1/TE) \ln (S(t)/S_0)$, where TE is the echo time, S(t) is the

dynamic signal intensity and S_0 is the average baseline signal intensity)

K1: This is the map of the parameter K1 as defined in equation 1 in [1]

K2: This is the map of the parameter K2 as defined in equation 1 in [1]. The K2 term reflects the effects of contrast agent leakage.

Rsquare: This is the map of R^2 for the linear fit of the model in equation 1 in [1].

rCBV Uncorrected: This is the relative Cerebral Blood volume map that does not take into account contrast agent leakage.

rCBV Corrected: This is the relative Cerebral Blood volume map corrected for contrast agent leakage. It is defined in equation 2 in [1].

4.1.1.2 Input parameters

This section describes the numeric parameters in the input dialog show in Figure 2. If values are available, Boxerman-Weisskoff software reads default values of these parameters from the image header.

Exclude first n timepoints: In the DSC perfusion image, the first few time points may have very high peaks. These points can be excluded from the analysis by specifying the number of initial time points to exclude.

Last Baseline Time: This is the last time (in seconds) before the gadolinium contrast appears in the venous region of interest. See section 4.1.2 for help in calculating the last baseline time. This value is used to decide the number of time points to use to calculate the average baseline signal intensity. The Time to Peak (TTP) is calculated as the time taken for the curve to peak from the last baseline time.

Last Time: This is the last time (in seconds) to be included in the analysis.

Slice Locations: The total number of slice locations in the input image.

Time Points: The total number of time points in the input image.

Echo Time (msec): The echo time (in msec) of the dynamic susceptibility contrast-enhanced MR sequence.

Whole Brain Threshold, Noise SD's: This is used to define the whole brain mask. If you specify a value "x" for this parameter, then the whole brain mask consists of voxels with average baseline signal intensity exceeding "x" SD's of the background noise.

Non-Enhancing Pixels Threshold, SDs above/below_baseline: This is used to define the Non-enhancing pixel mask. If you specify a value "x" for this parameter, then the Non-enhancing pixel mask consists of all the pixels that did not demonstrate signal intensity enhancement greater than "x" SD's above that pixel's average baseline signal intensity.

Non-Enhancing Pixels detected on final n timepoints: This parameter specifies the number of tail time points to use while computing the signal intensity enhancement for generation of the Non-enhancing pixel mask. For example, if the parameter for **Non-Enhancing Pixels detected on final n timepoints** has value “x” and the parameter for **Non-Enhancing Pixels Threshold, SDs above/below_baseline** has value “y”, then the Non-enhancing pixel mask consists of all the pixels that did not demonstrate signal intensity enhancement (signal intensity averaged over the final “x” time points) greater than “y” SD’s above that pixel’s average baseline signal intensity.

4.1.2 Raw-Perfusion

When you click on the Raw-Perfusion button in Figure 1, the Raw-Perfusion plugin is invoked. This plugin can be used for generating maps of parameters such as Time To Peak (TTP), relative Mean Transit Time (rMTT) and relative Cerebral Blood Flow (rCBF).

To perform the Raw-Perfusion analysis on dynamic susceptibility contrast MR images:

1. Open the dynamic susceptibility contrast MR images as a stack in ImageJ (for example, use File ... Import ... Image Sequence).
2. Place a region of interest in the background of the image and then click on Raw-Perfusion in the menu in Figure 1. The background region of interest is required for computing the whole brain mask. The Raw-Perfusion plugin does analysis on the whole brain voxels defined in this mask.

Note that the Raw-Perfusion plugin requires the information about the image times at each time point in order to perform the analysis. If this information is available in the image header, then on clicking on Raw-Perfusion in the menu in Figure 1, it will automatically read the image times and display a file called ImageTimeTable.txt containing the time for each time point. However, if this information is not available in the image header, then the user must supply the ImageTimeTable.txt file containing the time for each time point (one time per line).

3. When you click on the Raw Perfusion menu in Figure 1, the Raw Perfusion control panel shown in Figure 3 should appear. This control panel allows the user to provide various input parameters. It also allows the user to select the maps that should be displayed. Section 4.1.2.1 describes the different images that can be generated using the Raw Perfusion plugin. Section 4.1.2.2 describes the numeric input parameters in Figure 3.

DSCOMAN Version 1.0

Exclude first n timepoints: 1.0

Last Baseline Time: 25.603

Last Time: 58.735

Slice Locations: 16.00

Time Points: 40.0000

Echo Time(msec): 32.00

Whole Brain Threshold, Noise SD's: 10.00

delta R2 curve to use: Raw delta R2

Parameters for Last baseline Time Computation:

Last Baseline Time Computation Method: Method 3: Use threshold percent of the area under the delta R2 curve

Threshold number of noise standard deviations: 3.0

Threshold % of the area under the delta R2 curve: 5.0

Show following Images:

Whole Brain Mask Delta R2

Corrected Delta R2 rMTT

rCBF TTP

OK Cancel

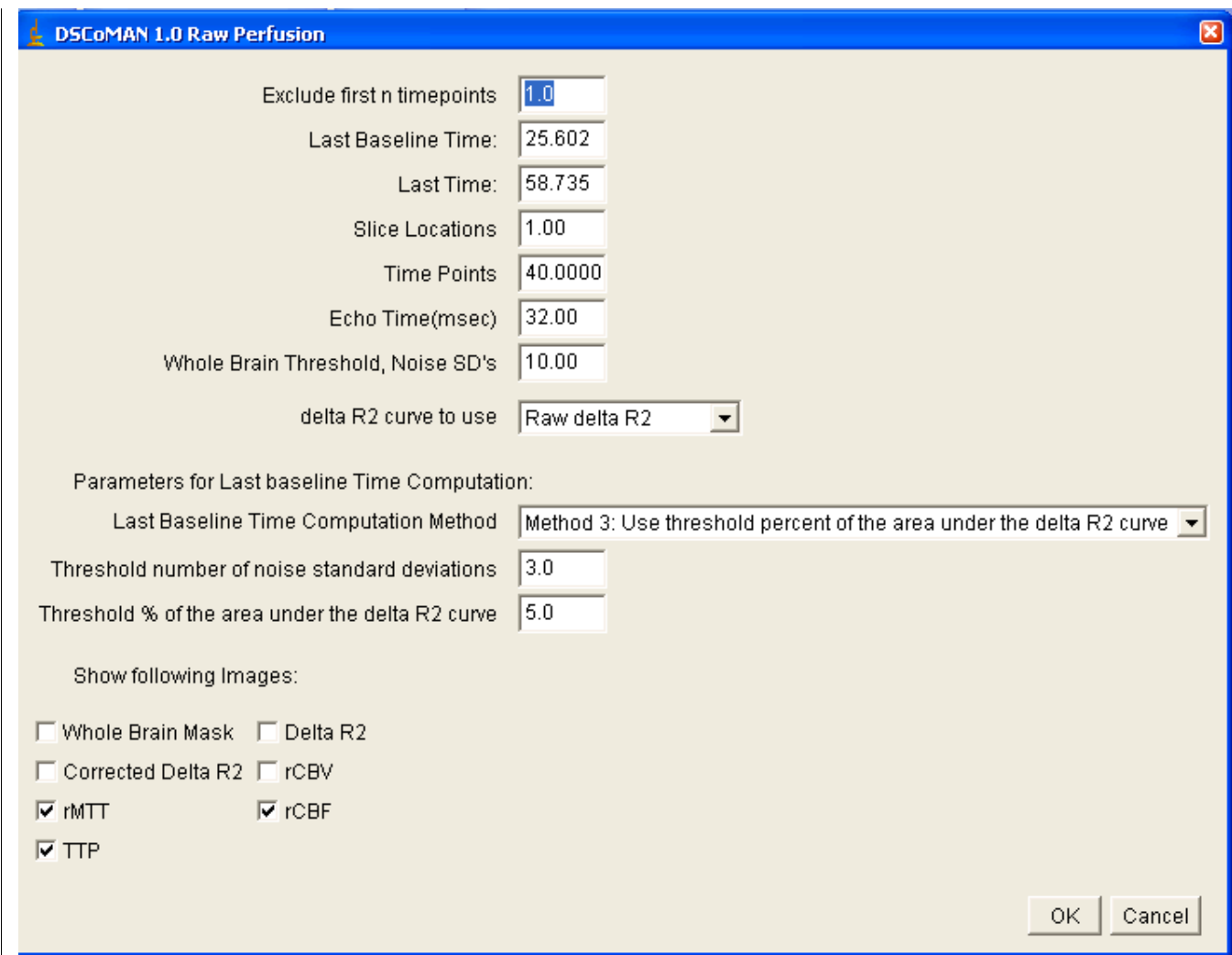


Figure 3: DSCoMAN 1.0 Raw Perfusion Control Panel

4.1.2.1 Map Selection

In Figure 3, you can use the checkboxes to choose which images/maps should be generated. This section will describe the images that can be generated by the Raw-Perfusion plugin. This will also help in understanding some of the numeric parameters in the control panel in Figure 3.

Whole Brain Mask: This is the first step before starting any actual analysis. We compute a whole brain mask for each slice in the dataset so that only locations with an average baseline signal intensity more than “x” times the standard deviation of background noise are included. Here, x specified by the user using the input parameter **Whole Brain Threshold, Noise SD’s**). Background noise standard deviation is calculated using the user defined background ROI. Further analysis is done only on voxels that are present in the whole brain mask.

Delta R2: This represents the relaxivity-time curve, a parameter related to the concentration of gadolinium in the voxel. It is computed as $-(1/TE) \ln (S(t)/S_0)$, where TE is the echo time, S(t) is the dynamic signal intensity and S₀ is the average baseline signal intensity)

Corrected Delta R2: This is the delta R2 curve corrected for contrast agent leakage according to the equation A11 in the Appendix of [1]. This computation involves fitting a curve to the data to determine

the parameters K1 and K2 as described in [1].

rCBV: This is the relative Cerebral Volume map. It is computed from the trapezoidal integration of the deltaR2 map from the first time after the last baseline time to the last time specified by the user. That is, this integration is done over the part of the deltaR2 curve after the arrival of contrast. The deltaR2 map used depends on the choice made in the input parameter –**delta R2 curve to use**.

rMTT: This is the relative Mean Transit Time map. The relative Mean Transit Time is defined in equation A1 in the Appendix.

rCBF: This is the relative Cerebral Blood Flow map. The Cerebral Blood Flow is defined as the ratio of the relative Cerebral Blood Volume to the relative Mean Transit Time

TTP: This is the Time to Peak map. The Time to Peak (TTP) is calculated as the time taken for the delta R2 curve to peak from the **Last Baseline Time**. The **Last Baseline Time** is a user specified parameter and is explained in section 4.1.1.2.

4.1.2.2 Input parameters

This section describes the numeric parameters in the input dialog show in Figure 3. If values are available, Raw Perfusion software reads default values of these parameters from the image header.

Exclude first n timepoints: In the DSC perfusion image, the first few time points may have very high peaks. These points can be excluded from the analysis by specifying the number of initial time points to exclude.

Last Baseline Time: This is the last time (in seconds) before the gadolinium contrast appears in the venous region of interest. See section 4.1.2 for help in calculating the last baseline time. This value is used to decide the number of time points to use to calculate the average baseline signal intensity.

Last Time: This is the last time (in seconds) to be included in the analysis.

Slice Locations: The total number of slice locations in the input image.

Time Points: The total number of time points in the input image.

Echo Time (msec): The echo time (in msec) of the dynamic susceptibility contrast-enhanced MR sequence.

Whole Brain Threshold, Noise SD's: This is used to define the whole brain mask. If you specify a value “x” for this parameter, then the whole brain mask consists of voxels with average baseline signal intensity exceeding “x” SD's of the background noise.

delta R2 curve to use: This is a popup menu to choose the delta R2 curve that should be used for the TTP, rMTT and rCBF rCBF and rCBV calculations. You can choose the raw delta R2 curve or the corrected delta R2 curve from a Boxerman-Weisskoff analysis.

Last Baseline Time Computation Method: This is a pop-up menu to choose the method to use for computing the last baseline time. The possibilities are:

1. **Method 1: Use user specified Last Baseline Time** – This method uses the Last Baseline Time specified by the user in Figure 3. It uses the same value of the last baseline time for each pixel.
2. **Method 2: Use threshold SD below baseline intensity** – This method uses a threshold SD to compute the last baseline time individually for each pixel. In this case, the last baseline time is identified when the signal intensity goes below the specified SD's (specified by input parameter - **Threshold number of noise standard deviations**) of the baseline signal intensity. Note that this computation is done on a pixel-to-pixel basis.
3. **Method 3: Use threshold percent of the area under the delta R2 curve** – This method uses the area under the delta R2 curve to compute the last baseline time for each pixel. For a given pixel, it computes the area under the delta R2 curve at each time point. It identifies the last baseline time as the point at which the area under the curve at a particular time exceeds a certain user-specified threshold (specified by input parameter – **Threshold % of the area under the delta R2 curve**) of the total area.

Threshold number of noise standard deviation: This parameter specifies the number of noise standard deviations to be used for computation of the last baseline time using Method 2 described above. Say, the value of this parameter is x. Then, the last baseline time for a signal intensity curve is the time at which the signal intensity goes 'x' SD's below the average baseline signal intensity.

Threshold % of the area under the delta R2 curve: This parameter specifies a threshold to use for the area under the delta R2 curve for the computation of the last baseline time using Method 3 described above.

4.1.2.3 Limitations

The limitations associated with mean transit time and cerebral blood flow measurements generated with the Raw Perfusion plugin should be carefully considered. This plugin uses first moment of the uncorrected delta R2 vs. time curve to calculate mean transit time. This approximation, as has been pointed out in several publications, provides at best a rough estimate of relative mean transit time. Since the mean transit times generated by this technique are used to compute cerebral blood flow, similar precautions apply to the blood flow maps. The cerebral blood volumes calculated should also be considered relative, not absolute measurements.

Improved quantitation of mean transit time and flow could be obtained by fitting the underlying curves to a gamma variate equations and/or deconvolution of the tissues curves using an arterial input function, but these techniques are not offered in this version of the DSCoMAN software.

4.1.3 ROI TimeCourse

When you click on the ROI TimeCourse button in Figure 1, the ROI TimeCourse plugin is invoked. This plugin can be used to see the variation in signal intensity over time.

The ROI TimeCourse plugin is part of the TOPPCAT software package. The download instructions are available at: http://dblab.duhs.duke.edu/modules/dblabs_topcat/index.php?id=1

For a region of interest on an image, ROI Time Course creates a plot window as seen in Figure 4 showing the variation in the mean signal intensity within the location specified by the region of interest

over all time points. This is very useful for calculating the proper last baseline time point before the arrival of the contrast bolus. In the plot, the x-axis is designated in seconds and the y-axis is signal intensity.

Note: placing a cursor over the plot window will help indicate the time in seconds corresponding to the location of the cursor. Selecting the “List” button generates a list the coordinates of the points on the curve.

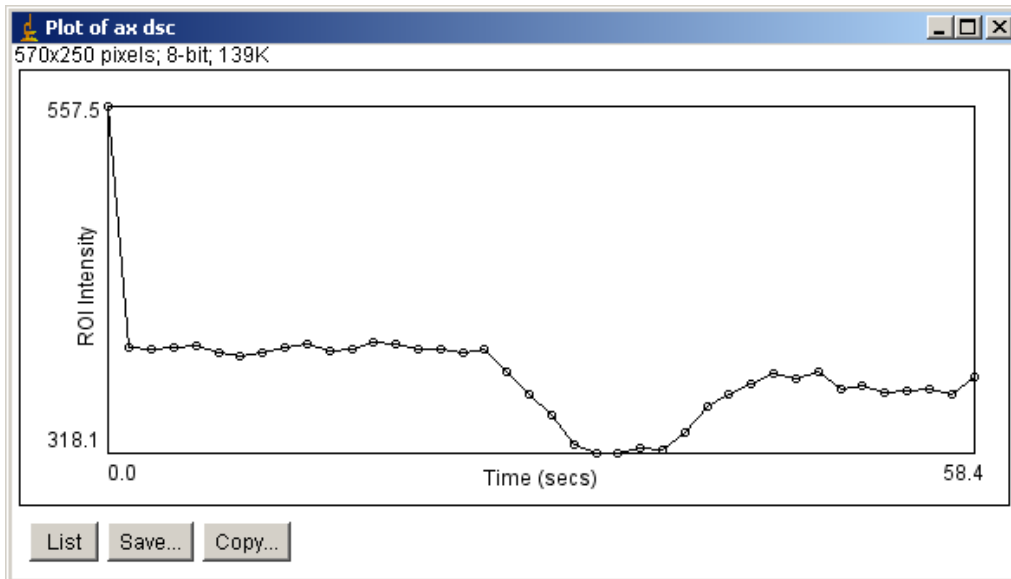


Fig. 4. Sample output from the ROI time course plugin.

4.1.4 HyperVolume

When you click on the HyperVolume button in Figure 1, the HyperVolume plugin is invoked. This plugin can be used to display a four dimensional (xyzt) dynamic stack with two slider bars for easy manipulation in both the time and z dimensions.

The HyperVolume plugin is part of the TOPPCAT software package. The download instructions are available at: http://dblab.duhs.duke.edu/modules/dblabs_topcat/index.php?id=1

HyperVolume makes use of the most excellent HyperVolume Browser plugin from Patrick Pirrotte and Jerome Mutterer (rsb.info.nih.gov/ij/plugins/hypervolume-browser.html).

4.2 Macro mode

The DSCoMAN software can be operated using ImageJ’s macro language. This is very useful when you need to perform a number of repetitive analyses. To experiment with macro language, use the ImageJ command Plugins...Macros...Record, and/or refer to the ImageJ website.

One disadvantage to using the macro mode is that you must explicitly supply all parameters within the macro text. Specifically, the parameters supplied in the macro text override default parameters. It is possible, however, to modify the macro text to supply these parameters in an automated way by using our Query Dicom Header plugin (rsb.info.nih.gov/ij/plugins/query-header.html) within the macro to make the macro text file aware of the Dicom parameters used in the stacks.

Here is a sample ImageJ macro code to run the Boxerman-Weisskoff plugin. It provides the required input parameters and specifies that the following maps be generated: K1, K2, Rsquare, rCBV UnCorrected and rCBV Corrected. This is specified by the boolean parameters k1, k2, rsquare, rcbv_uncorrected and rcbv_corrected in the input parameter list.

```
run("Boxerman Weisskoff", "exclude_first_n_timepoints=1 last_baseline_time=25.603
last_time=58.735 slice_locations=16 time_points=40.0000 echo_time(msec)=32
whole_brain_threshold_noise_sd's=10 non-enhancing_pixels_threshold_sds_above/below_baseline=1
non-enhancing_pixels_detected_on_final_n_timepoints=10 k1 k2 rsquare rcbv_uncorrected
rcbv_corrected");
```

Appendix

The rMTT (for relative mean transit time) can be estimated as the first moment of the delta R2 curve:

$$A1) \quad rMTT = \frac{\int_0^t t \cdot \Delta R2(t)}{\int_0^t \Delta R2(t)}$$

This integration is done from the first time after the last baseline time to the last time specified by the user. That is, this integration is done over the part of the curve after the arrival of contrast. The initial baseline portion of the curve is ignored.

References

1. Boxerman JL, Schmainda KM, Weisskoff RM. **Relative Cerebral Blood Volume Maps Corrected for Contrast Agent Extravasation Significantly Correlate with Glioma Tumor Grade, Whereas Uncorrected Maps do not.** AJNR Am J Neuroradiol 2006;27:859-67
2. Weisskoff RM, Boxerman JL, Sorensen AG, et al. **Simultaneous blood volume and permeability mapping using a single Gd-based contrast injection.** Proceedings of the Society of Magnetic Resonance, Second Annual Meeting; 1994 Aug 6–12; San Francisco, Calif.; Berkeley: Society of Magnetic Resonance; 1994:279