CurveAlign Users Manual

# Contents

[Contents 1](#_Toc358110538)

[1 Introduction 1](#_Toc358110539)

[2 QuickStart Instructions 2](#_Toc358110540)

[3 Output files 3](#_Toc358110541)

[4 Output images 3](#_Toc358110542)

[4.1 Overlay image 3](#_Toc358110543)

[4.2 Map Image 4](#_Toc358110544)

[4.2.1 How processed map files are created: 5](#_Toc358110545)

[4.3 Reconstructed Image 6](#_Toc358110546)

[5 Boundary Analysis 7](#_Toc358110547)

[6 Stacks 8](#_Toc358110548)

[6.1 Stacks with Boundaries 9](#_Toc358110549)

[6.2 Stacks without Boundaries 9](#_Toc358110550)

[7 Batch mode 9](#_Toc358110551)

[7.1 Batch mode with CSV boundaries 11](#_Toc358110552)

[7.2 Batch mode without boundaries 11](#_Toc358110553)

[7.3 Batch mode with Tiff boundaries 11](#_Toc358110554)

[7.4 Batch mode with CT-FIRE input 11](#_Toc358110555)

[8 Status Label 12](#_Toc358110556)

# Introduction

The purpose of CurveAlign is to compute features that describe collagen interactions with epithelial cells. It was developed in order to search for stromal changes that are correlated with disease in images of collagen and epithelial cells. We have recently used CurveAlign for the feature calculations performed the paper Bredfeldt, et. al. Journal of Pathology Informatics, 2014. Collagen images may be furnished by any appropriate imaging technique, however we have focused here on SHG images of collagen. Epithelial cell information is input into CurveAlign as an 8-bit mask file, that must be pre-registered with the collagen image, where white pixels correspond to epithelial cell regions and black pixels correspond to anything else in the image. These mask files can be generated by any appropriate means, such as manual ROI annotation in FIJI or using segmentation tools in MATLAB or FIJI. The output features can then be used to potentially classify images or fibers using machine learning techniques. Machine learning is not incorporated into CurveAlign, but is available through many open source tools such as Weka and R. The GUI in CurveAlign is modular, so that the main user interface is in a separate window from the outputs. This allows for the user to resize the output windows to their preferred size. The main user interface window is shown below.

Figure . Main user interface for CurveAlign.

# QuickStart Instructions

1. Select the fiber processing mode.
2. Select the boundary processing mode.
3. Click “Get Images” and browse to an image. Images may be single or stacks. If you would like to process more than one image in a batch, just select more than one image in the dialog using the technique appropriate for your operating system. In batch mode, images are not displayed.
4. If a manual boundary is required (not available in batch mode), hold down the Alt key and click points along the desired path on the image window. Release the Alt key when finished.
5. Enter the keep threshold level and select the desired outputs using the checkboxes on the bottom of the main window.
6. Click the run button.
7. The output data will automatically be placed in a folder called CA\_Out in the directory with the original images.
8. If you have drawn a boundary, you will be prompted to give a file name and location for the boundary points to be saved. These will be saved as a \*.csv file. This file can be used again in the future by selecting CSV boundary as the boundary processing mode.
9. The current function being performed by CurveAlign is listed in the Status label at the bottom of the main window.

# Relationship with CT-FIRE

CT-FIRE is another software tool developed for processing images of fibers. CT-FIRE traces fibers and outputs a database of fiber objects. This output database can be used as an input to CurveAlign (the subject of this manual). CurveAlign's main purpose is to compare fibers with boundaries and fibers to each other to measure density and alignment. When CT-FIRE inputs are used, the fibers are compared to region boundaries, either drawn manually, or created by any segmentation tool.

CT-FIRE and CurveAlign may someday be combined, but we thought it was useful to have them separate and easy enough to just connect them by reading the CT-FIRE outputs into CurveAlign when needed.

# Fiber processing modes

There are four fiber processing modes.

1. Curvelet Transform: The curvelet transform is performed on the image(s) and each curvelet corresponds to an observation in the feature list. Each curvelet has a unique angle and position. In this case, fiber length and curvature are not available, since curvelets are independent from one another.
2. CT-FIRE Segments: The output from CT-FIRE is used as the input, providing information about the fibers in the image. Each fiber segment in the CT-FIRE output file corresponds to an observation in the feature set. Each segment has a unique angle and position. In this case, each segment is given a fiber length and curvature value that corresponds to the length and curvature of the entire fiber this segment belongs to.
3. CT-FIRE Fibers: The output from CT-FIRE is used as the input, providing information about the fibers in the image. Each fiber center point in the CT-FIRE output file corresponds to an observation in the feature set. Each center point has a unique angle corresponding to the average angle of the fiber. The position is the midpoint between the fiber end points. In this case, each center point is given a fiber length and curvature value that corresponds to the length and curvature of the entire fiber this center point belongs to.
4. CT-FIRE Fiber Endpoints: The output from CT-FIRE is used as the input, providing information about the fibers in the image. Each fiber endpoint in the CT-FIRE output file corresponds to an observation in the feature set. Each endpoint has a unique angle and position. The angle corresponds to the angle of the entire fiber. In this case, each endpoint is given a fiber length and curvature value that corresponds to the length and curvature of the entire fiber this segment belongs to.

When CT-FIRE outputs are used as inputs to CurveAlign, the CT-FIRE output files must be named according to a strict naming convention. For example, if the image is named the following:

1B-a1.tif

Then the CT-FIRE output file must be named the following:

ctFIREout\_1B-a1.mat

At the time of writing this manual, the above naming convention has been adopted by the CT-FIRE tool as well, so the outputs from CT-FIRE may be directly used as inputs to CurveAlign.

# Boundary processing modes

There are four boundary processing modes.

1. No boundaries: Fibers will be compared to eachother, but not to a boundary. For example, features about fiber alignment will still be computed and output in the feature list.
2. Manual boundaries: The user is required to manually draw boundaries with the Alt-key + mouse clicking technique. This option is not allowed if multiple input images are selected to be processed in a batch. If this option is selected and multiple files are to be processed, then the "No Boundaries" option is automatically selected.
3. CSV boundaries: Previously stored boundary files can be used that contain the x-y positions of points along the boundary.
   1. CSV files must adhere to the following naming convention. If the image is named: TACS-3a.jpg, then the CSV file must be named: boundary for TACS-3a.jpg.csv Batch processing is allowed in this mode.
4. Tiff boundaries: Mask files are used to indicate where region boundaries are located. The mask files should be 8-bit files where the inside of ROIs are given a value of 255 and outside a value of 0. This allows for features identifying whether a fiber is inside an ROI or not. These files must be in pixel for pixel registry with the image files and therefore have the same dimensions as the image files.
   1. Tiff boundary files must adhere to the following naming convention. If the image is named: 1B-a5.tif, then the tiff boundary file must be named: mask for 1B-a5.tif.tif.

When boundaries are used as part of the analysis, CurveAlign computes up to 2 relative angles per observation. The first is the nearest point angle and the second is the intersection point angle. The nearest point angle is the relative angle between the observation point and the tangent angle of the region boundary at the point nearest to the observation point. The intersection point angle is the relative angle between the observation point and the intersection point of the fiber, interpolated out a user defined distance, and a boundary.

# Input files

CurveAlign requires up to three types of input files.

Image files: These files contain images of fibers. These can be produced by any method available, however we have particularly focused on SHG images of collagen fibers.

CT-FIRE files: The output of CT-FIRE having been run on the above mentioned "Image files."

Boundary files: The boundary files can be CSV files or Tiff files. See the section about Boundary Modes above to learn about naming conventions for these file types.

# Output files

There are several files that are generated by the CurveAlign software for each image processed. These files and their descriptions are listed in the table below:

|  |  |
| --- | --- |
| **Filename** | **Description** |
| \*\_hist.csv | List of bin values and numbers of observations in each bin. Bin values correspond to relative angle in if boundaries are used, and absolute angle if no boundary is used. |

# Output images

A few output images are saved in .tiff format in the selected output directory. These images are explained below.

## Overlay image

This image shows the position and angle of each curvelet that is being used in the analysis (green) and each curvelet that is ignored (red) as an overlay on the original image. If a boundary is selected, the boundary is also included in the overlay.

TACS-3a_overlay.tiff

Figure . Overlay image showing the curvelets that are within the specified distance from the boundary in green, the curvelets outside the specified boundary in red, and the boundary in yellow.

## Map Image

The map image is intended to help the user identify the spatial distribution of curvelet angles within the image.

The raw map (\_rawmap.tiff) file codes the angle of the curvelet into a grey scale value. The pixel in the image where the center of the curvelet is located is given a value between 0 and 255 that corresponds to 0 to 90 degrees when a boundary is selected and 0 to 180 degrees when a boundary is not selected. This file may be further processed in matlab or imagej according to the users preferences.

The processed map file (\_procmap.tiff) is a processed version of the raw map file overlayed on the original image. The output is intended to show regions of aligned structures that are perpendicular to the boundary, in the case of a boundary selection, or regions of generally aligned structures, in the case where no boundary is selected.

### How processed map files are created:

When a boundary is selected, the raw map file translates 0 to 90 degrees into 0 to 255 in gray scale. The center location of each curvelet is given a gray level corresponding to its angle with respect to the boundary. Then a square max filter is applied with a size of 12 pixels on a side, followed by a Gaussian disc filter with a sigma of 4 pixels. The color scale is then set to 0-20 degrees = black, 20-45 degrees = green, 45-60 degrees = yellow, and 60-90 degrees = red. The map image is overlayed on the original image with transparency set to 0.5. See Figure 3.

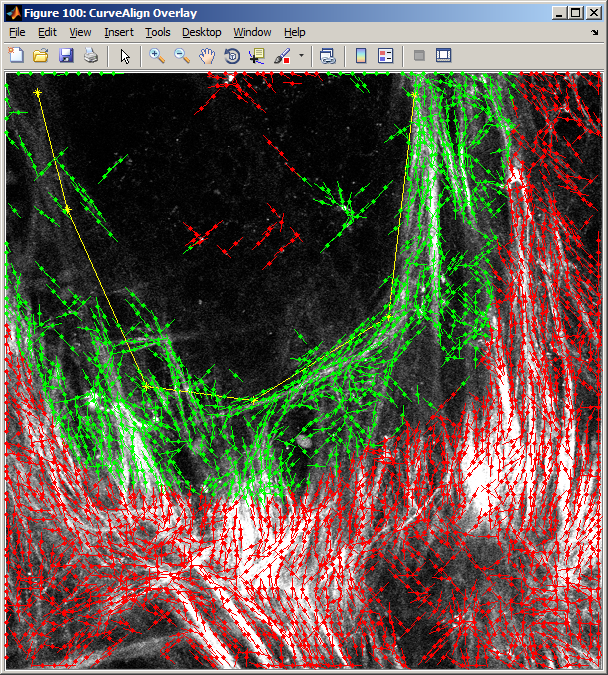
TACS-3a_procmap.tiff

Figure . Over lay (left) and map (right) images when a boundary is selected.

When a boundary is not selected, the raw map file translates 0 to 180 degrees into 0 to 255 in gray scale. The center location of each curvelet is given a gray level corresponding to its absolute angle in the image. Then a square, standard-deviation filter is applied with a size of 64 pixels on a side. The result is then subtracted from 127.5 (half of 255), followed by a 12 pixel max filter, followed by a Gaussian disc filter with sigma of 4 pixels. The color scale is then set to 0-80 black, 80-90 green, 90-100 yellow and 100-127.5 red. In this case, red indicates areas of highly aligned structures, while black and green indicate areas of more randomly aligned structures. The map image is overlayed on the original image with transparency set to 0.5. See Figure 4.

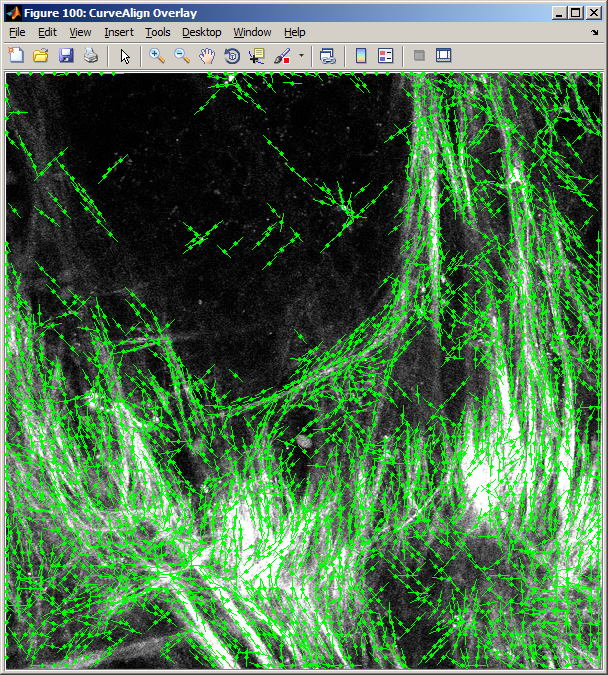
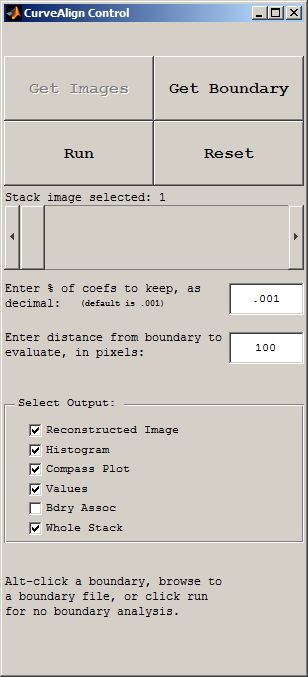
TACS-3a_procmap.tiff

Figure . Overlay and map images when no boundary is selected.

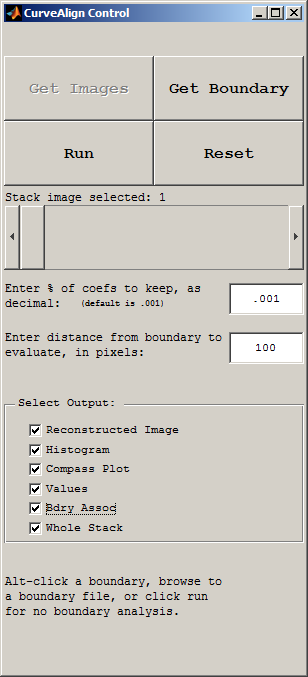
# Boundary Analysis

The user is able to analyze the curvelets that fall only within a certain distance from a boundary. To enter this distance, first a boundary must be drawn or opened from file. Then the distance in pixels should be entered into the edit box.



Boundaries are created by alt-clicking on the original image that is opened in the CurveAlign GUI. When alt is released, the boundary is ended and no additional points may be added to the boundary. Boundary files can also be opened by browsing to a saved boundary file using the Get Boundary button.

The position on the boundary that is being compared to each curvelet may be visualized as well. This allows the user to see where the angle comparisons are being made. Blue lines are drawn on the overlay output image from each curvelet to the point on the boundary that the curvelet is associated with.



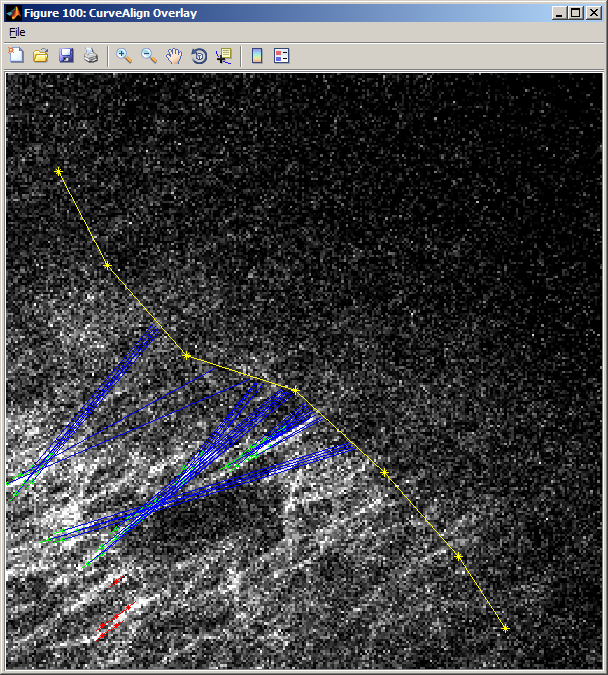
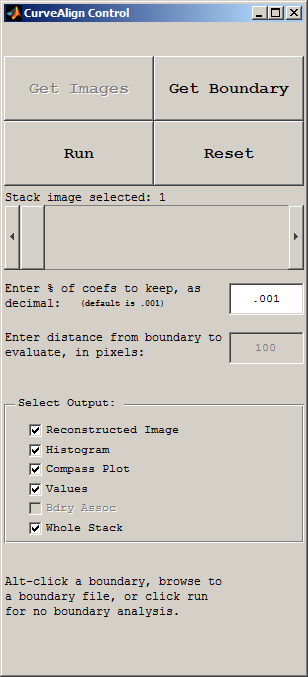


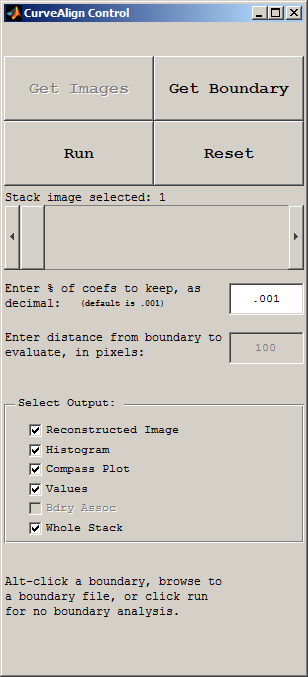
Figure . An example of an overlay images where the associations between curvelets and the boundary are indicated by blue lines.

# Stacks

CurveAlignV2.0 allows for the user to select stacks of images. When a stack is opened, the slider bar is enabled, allowing the user to choose which image in the stack should be displayed.



There is a checkbox at the bottom of the main window that allows for the user to select if the whole stack should be processed, or only the current image should be processed.



When a stack is being analyzed, then the output files that are images will also be in stack format. To open these stacks in ImageJ, the LOCI bioformats importer must be used. For some reason, these tiff stacks cannot be drag and dropped into ImageJ, this will be investigated further in the next release of CurveAlign.

The other output files, for example the histogram file and the curvelet angle spreadsheet, are produced such that each image in the stack creates a new output file. For example a stack with 4 images will generate 4 histogram files named stack\_1\_hist.csv, stack\_2\_hist.csv, stack\_3\_hist.csv, etc.

## Stacks with Boundaries

If a stack is being processed, then only a single boundary can be used for the entire stack. Future versions of this tool will allow for a different boundary in each image plane in a stack.

## Stacks without Boundaries

When no boundary file is selected, then each image in the stack is analyzed as described in the section above about image analysis without a boundary.

# Batch mode

Batch mode is used to process multiple images and potentially associated boundaries. To use batch mode, click the Get Images button and select more than one image in the file selection dialog box. Each image will be processed according to the modes currently selected.

CurveAlign will first search for boundary files in the chosen directory. If there are boundary files, then CurveAlign will process all images that are found to be associated with the located boundary files. Images may be a mixture of both individual images and stacks. If the image is a stack, then the entire stack will be processed.

The batch mode output files will be stored in a directory named CA\_Out and will include all of the outputs available from the CurveAlign software.

## Batch mode with CSV boundaries

If boundaries are drawn with the CurveAlign program, they should be saved in the same directory as the images. The boundary files should be named like the following:

Image file name = TACS-3a.jpg

Boundary file name = boundary for TACS-3a.jpg.csv

## Batch mode without boundaries

To process a directory of images without boundaries, then just place the images in a directory by themselves (without any boundary files), then run CurveAlign with select all of the images in that directory. In this case, the distance from the boundary and boundary association edit boxes will be ignored.

## Batch mode with Tiff boundaries

In batch mode, boundaries may also be imported as tiff files. Tiff boundary files must be 8bit binary mask images where the inside of a region has a value of 255 and everything else must be 0. Boundary tiff files must have the same number of pixels (length and width) as the original image and may be produced by hand or by segmentation in ImageJ/FIJI. Tiff boundary files should be named according to the following convention:

Image file name = 1B-a5.tif

Boundary file name = mask for 1B-a5.tif.tif

## Batch mode with CT-FIRE input

To compare the results of the CT-FIRE software to boundaries (either TIFF or csv), then choose one of the CT-FIRE fiber processing modes on the main GUI. The following naming convention should be used:

Image file name = 1B-a1.tif

CT-FIRE results file = ctFIREout\_1B-a1.mat

# Status Label

To allow the user to keep track of what is happening in the program, there is a status label on the bottom of the main window. This label gives hints about what the user should do next or shows the current task that the program is working on.

