V 0.1   Sept. 2, 2022
University of California, Irvine

Mulveyj@uci.edu
Patterson Lab

ImageJ DSSIM User Guide

Katen Iyer and Justin Mulvey

This guide will explain how to get started using the DSSIM (structural dissimilarity index measure) ImageJ plugin. Installation instructions are on the last page.

[This version of the guide is a work in progress, and [] will be used to make notes where we expect improvements in the next iteration. However, we think the plugin and guide are very usable at this stage, and we are open to any feedback]. [**Note we are currently optimizing performance. We recommend a maximum dataset size of 200 MB for this iteration.** Large files may take excessive RAM and several hours to complete in this iteration]

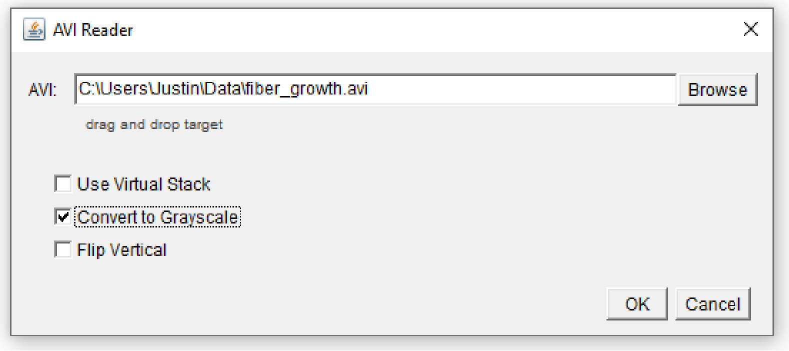
DSSIM Introduction

While structural changes in microscopy videos are often visually apparent, numerically quantifying the rate of change, temporal scale, and spatial scale of dynamic behavior remains challenging. Here, we describe structural dissimilarity index measure (DSSIM) analysis as a general metric to quantify dynamic behavior in microscopy videos. DSSIM, also called SSIM, is an established mathematical measure of the perceived difference between two images, which compares local variation in the mean, variance, and cross-correlation to produce a new dissimilarity image. This metric was originally developed to quantify signal degradation between the source and destination in cable television, before it was adopted by the machine learning community as a metric for evaluating 2D and 3D model performance against a ground truth. Our lab discovered it was also a great metric for highlighting and quantifying structural dynamics in microscopy videos. In the document that follows, we give step-by-step instructions to apply and tune this analysis for microscopy videos. In addition to this document, the SSIM Wikipedia is a good place to learn about the mathematical foundation of SSIM/DSSIM analysis.

Importing data

The DSSIM plugin can only be run on an ImageJ 'image stack'. The easiest way to import data is to either drag and drop all the frame files/movie file onto the ImageJ windows, or by going to File→Import. It may also be necessary to convert individually opened frames to a stack with Image→Stacks→Images to Stack. The Bioformats plugin may be required to import certain types of microscopy files.

The analysis will not work for RGB images, so they must be converted to a single channel. This can be done upon importing a video file by checking **Convert to grayscale**, or by going to Image→Type and selecting 8-bit, 16-bit, or 32-bit.



AVI: C:\Users\Justin\Data\fiber_growth.avi

□ Use Virtual Stack
☑ Convert to Grayscale
□ Flip Vertical

Preprocessing

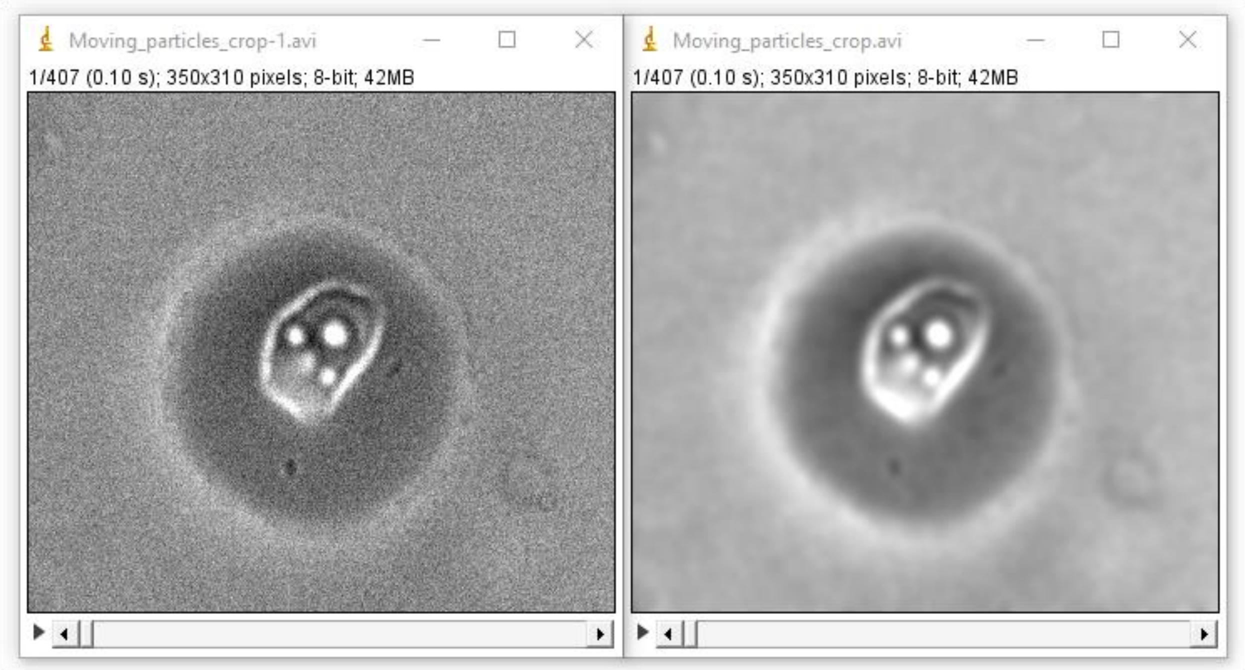
DSSIM analysis is sensitive to noise which is frequently found in microscopy videos. Thus, noise must be minimized for an effective analysis. Three common methods of removing noise are z-binning (temporal binning, frame averaging), x-y binning (spatial binning, pixel binning), and Gaussian-Blur denoising (low-pass FFT filter, local Gaussian-weighted average). All of these operations can be done on an ImageJ 'image stack'**. The preprocessing will affect the outcome of DSSIM analysis so it must be carefully considered and iterated upon.** In general, it is often best to start with excessive data binning to capture the largest dynamics in the dataset before moving backward to improve spatial and temporal resolution.

1. z binning: z binning is an effective way to remove noise by averaging information across multiple frames in a video, which also reduces dataset size for analysis. However, z binning will reduce temporal resolution, and can make dynamics appear blurry. . Z binning can be accessed with Image→Adjust→Size by adjusting the **Depth.** We recommend z binning as much as possible while still ensuring the dynamics of interest are not lost. In low framerate videos, z binning is not recommended

2. x-y binning: x-y binning is an effective way to remove noise while also reducing the size of the dataset for faster analysis. However, x-y binning will reduce the spatial resolution of the analysis. x-y binning can be accessed with Image→Adjust→Size. We recommend a maximum resolution of 1024 x 1024, although it is often beneficial to start smaller (or with a crop). We recommend users check the box for **Constrain Aspect Ratio** and **Average When Downsizing,** and set interpolation to bicubic. It is

also recommended to scale with integers (2048 → 1024).

3. **Gaussian-Blur:** One method of removing Gaussian noise is with a Gaussian-blur, which can be accessed with Process→Filters→Convolve→Gaussian Blur. Increasing the **Sigma** value will improve noise reduction but degrade feature sharpness. We recommend starting with a **Sigma** value 1/200 of the native resolution of the data (**Sigma** = 5 for a 1k x 1k image). Optimal **Sigma** will vary substantially depending on the type of microscopy and signal-to-noise ratio of the image.



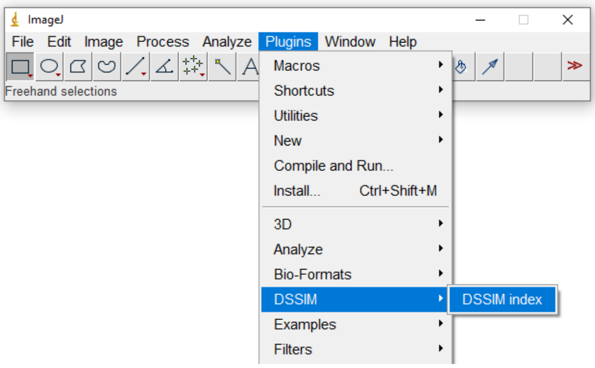
1/407 (0.10 s); 350x310 pixels; 8-bit; 42MB

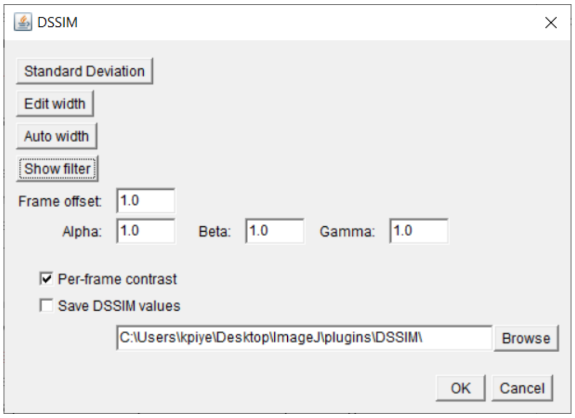
1/407 (0.10 s); 350x310 pixels; 8-bit; 42MB

The above figure shows a raw frame containing a high level of noise (left) compared to a processed frame that has been denoised with a Gaussian Blur (right). There are several other powerful tools for preprocessing such as: band-pass filters, background removal algorithms, and AI-based denoising approaches (noise2void for ImageJ).

DSSIM Analysis

With the desired 'image stack' already open, click Plugins→DSSIM→DSSIM Index to launch the plugin.





Frame offset: 1.0
Alpha: 1.0 Beta: 1.0 Gamma: 1.0
☑ Per-frame contrast
☐ Save DSSIM values
C:\Users\kpiye\Desktop\ImageJ\plugins\DSSIM\

The SSIM Wikipedia is a good place to learn about the mathematical foundation of SSIM/DSSIM analysis. There are three main components of DSSIM analysis: **Filter Settings (Standard Deviation, Edit Width, Auto Width)**, **Frame Offset**, and **Exponent Values (α, β, γ)**. The goal of the DSSIM analysis is to detect dynamic behavior in microscopy datasets. In short, the **Filter Setting** affects the spatial scale of dynamics detected, the **Frame Offset** affects the temporal scale of dynamics detects, and the **Exponents** control the nature of dynamics detected.

1. **Filter Settings:** [We are working on reducing this to a single numeric input so it will be much simpler]. The **Filter Settings** section will allow you to edit the parameters of the local neighborhood compared between each image in the video/image stack. The default value is a Gaussian weighted neighborhood with a first standard deviation of 1.5 pixels. For higher resolution datasets, we have found it is often better to go to 5 or 10 pixel first standard deviations. To change this value, click **Standard Deviation**, then enter the value in the prompt. Next, click **Auto Width**. You can directly edit the filter width by clicking **Edit Width**, the value must be an odd number. Note higher standard deviations will capture larger, less localized dynamics but also reduce the effects of abrupt changes. Additionally, higher standard deviations will increase computational time. **Show filter** will display a 10x enlarged version of the Gaussian filter which the analysis will use. You are able to see the alterations made in the filter settings window.
2. **Frame Offset:** DSSIM Analysis creates a dissimilarity image by comparing two frames. A **Frame Offset** of 1 will calculate DSSIM(n,n+1) (3 compared to 4, 4 compared to 5) and a **Frame Offset** of 3 will calculate DSSIM(n,n+3) (3 compared to 6, 4 compared to 7).

Adjusting this is very useful for detecting fast dynamics (smaller offsets) versus slow dynamics (larger offsets). Increasing frame offset will not increase computational time. Note the DSSIM stack will have fewer frames equal to the number of the frame offset. (value must be and integer greater than 0)

3. **Exponents(α,β,γ):** DSSIM compares local variation in the mean, variance, and cross-correlation (luminance, contrast, and structure) which are multiplied together to produce a final DSSIM value for each neighborhood (see SSIM Wikipedia for more details). Alpha, beta, and gamma correspond to the exponents of these values, which can be scaled to control their influence on the result. For example, if local variance is not useful for capturing dynamics, the beta exponent can be set to 0 which will remove the contribution. We recommend leaving all exponents set to 1. It is possible to view the individual contributions of each channel by setting only one exponent to 1.

$$\mathrm{SSIM}(x,y) = l(x,y)^{\alpha} \cdot c(x,y)^{\beta} \cdot s(x,y)^{\gamma}$$

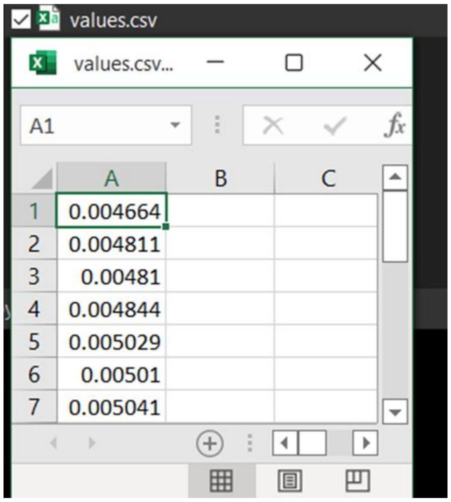
$$l(x,y) = \frac{2\mu_x\mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1}$$

$$c(x,y) = \frac{2\sigma_x\sigma_y + c_2}{\sigma_x^2 + \sigma_y^2 + c_2}$$

$$s(x,y) = \frac{\sigma_{xy} + c_3}{\sigma_x\sigma_y + c_3}$$

Per-frame contrast removes outliers on a per-frame basis in output DSSIM Stack. This is good for datasets undergoing a dramatic change at the beginning/end of the data. However, it makes it impossible to directly compare frames in the DSSIM Stack based on intensity values. Unchecking this box will use the same contrast mapping for every frame in the DSSIM Stack, making it possible to compare frames directly. You can also change the global contrast boundaries of the DSSIM Stack with Image→Adjust→Brightness/Contrast, then setting the min and max contrast boundary or hit the **Auto Contrast** button. We recommend starting with the per-frame contrast box unchecked, but then checking the box if no obvious dynamics are observed due to outlier frames in the DSSIM Stack.

Save DSSIM Values: The average DSSIM values for each frame of the DSSIM image can be saved by checking this box. This is usefully for quantifying average dynamics between frames in the dataset. It will be saved as a .csv file to the folder path selected with **Browse**.



values.csv

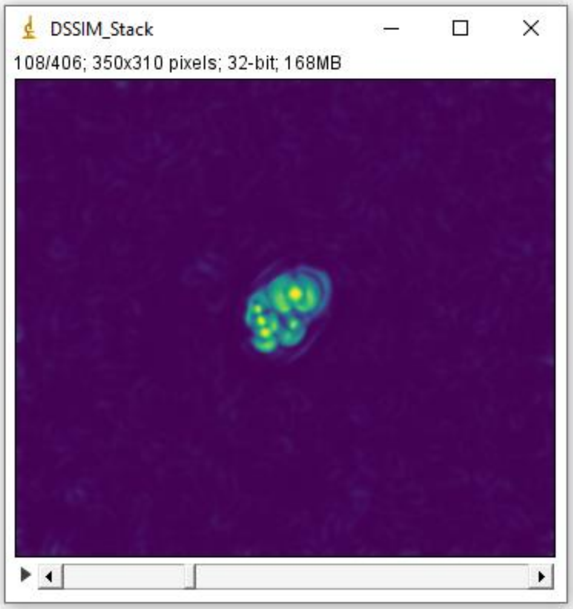
values.csv...

A1

	A	B	C
1	0.004664		
2	0.004811		
3	0.00481		
4	0.004844		
5	0.005029		
6	0.00501		
7	0.005041		

After entering the settings click OK to run the analysis. Depending on the size of the dataset and computer, this takes anywhere from a few seconds to several hours. [We are currently working on multicore optimization and a progress bar]. The DSSIM Stack will appear once the plugin calculations have finished.

Once the DSSIM Stack has been created, the LUT can be changed to add color. This can be done with Image→Color→Edit LUT, then finding the 'lut' folder in the ImageJ install folder. We recommend using viridis.lut or another linear colormap (see The misuse of color in science communication). [we are working on automatically finding this file].



DSSIM_Stack

108/406; 350x310 pixels; 32-bit; 168MB

Example DSSIM Stack with viridis LUT. Bright pixels represent regions of high dynamics.

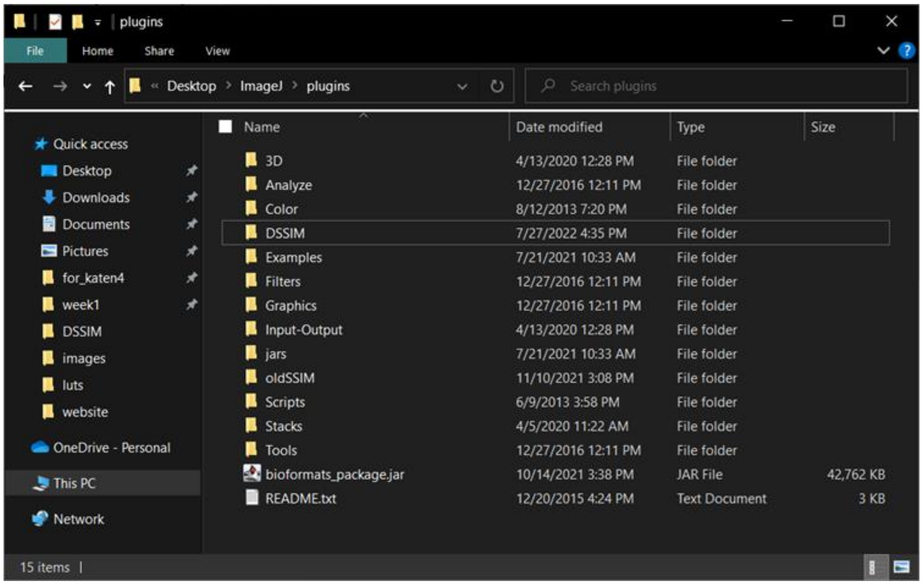
Data Interpretation and Advanced Analysis

[section to be added]

- Higher DSSIM value means more structural dynamics.
- Regional analysis methods
- Quantitative analysis

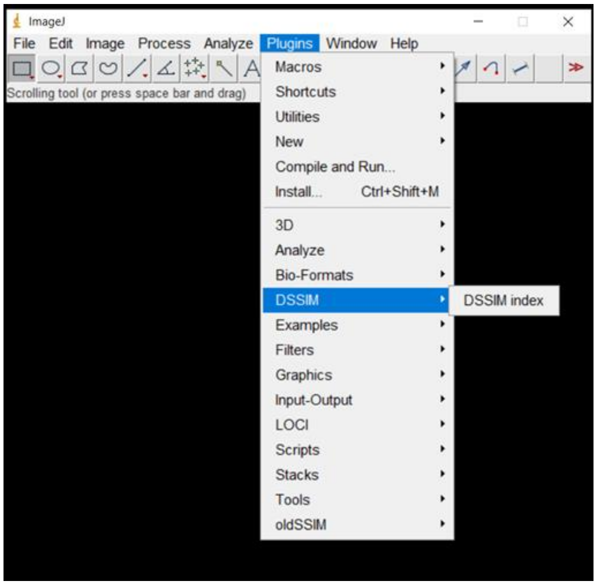
Installation:

Place the DSSIM folder in the ImageJ plugin folder.



Name	Date modified	Type	Size
3D	4/13/2020 12:28 PM	File folder	
Analyze	12/27/2016 12:11 PM	File folder	
Color	8/12/2013 7:20 PM	File folder	
DSSIM	7/27/2022 4:35 PM	File folder	
Examples	7/21/2021 10:33 AM	File folder	
Filters	12/27/2016 12:11 PM	File folder	
Graphics	12/27/2016 12:11 PM	File folder	
Input-Output	4/13/2020 12:28 PM	File folder	
jars	7/21/2021 10:33 AM	File folder	
oldSSIM	11/10/2021 3:08 PM	File folder	
Scripts	6/9/2013 3:58 PM	File folder	
Stacks	4/5/2020 11:22 AM	File folder	
Tools	12/27/2016 12:11 PM	File folder	
bioformats_package.jar	10/14/2021 3:38 PM	JAR File	42,762 KB
README.txt	12/20/2015 4:24 PM	Text Document	3 KB

When you launch ImageJ you should be able to find DSSIM under the plugins dropdown.


ImageJ
File
Edit
Image
Process
Analyze
Plugins
Window
Help
Scrolling tool (or press space bar and drag)
Macros
Shortcuts
Utilities
New
Compile and Run...
Install... Ctrl+Shift+M
3D
Analyze
Bio-Formats
DSSIM
DSSIM index
Examples
Filters
Graphics
Input-Output
LOCI
Scripts
Stacks
Tools
oldSSIM