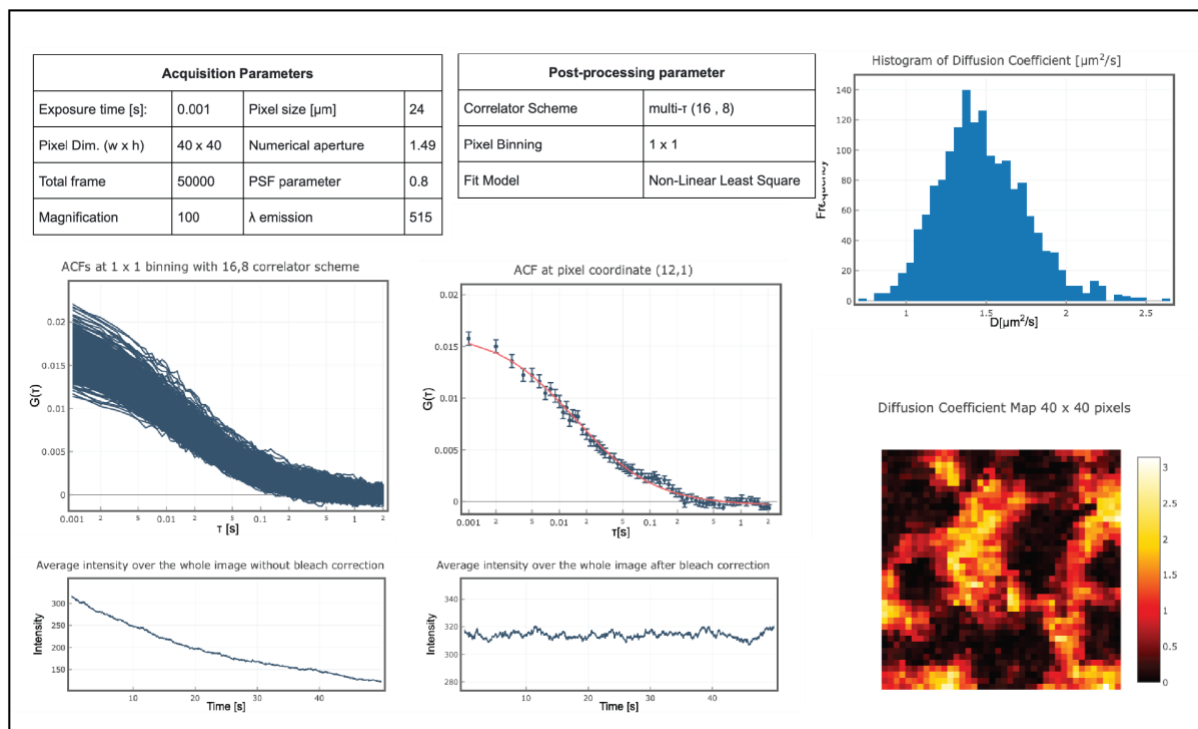# A guide to
Imaging **FCS**
in APEER

14 July 2020

This user manuals consist of a set of data analysis tool – in the form of (APEER) modules – for camera based Fluorescence Correlation Spectroscopy (FCS). Each of the modules and its subsequent workflows is fully based on Imaging FCS 1.52, which is an ImageJ plugin for the purposes of calculating, viewing, and analysing spatio-temporal correlation function from 16-bit GRAY TIFF STACK files. It calculates correlations for each pixel or group of pixels in the image along the time axis (i.e., the z-axis in the stack).

Table of Contents

1. Introduction (what does Imaging FCS in APEER DO?)

This user manual is based on Imaging FCS 1.52 that allows calculating spatio-temporal auto- and cross-correlations of an image stack (16/32-bit TIFF). It can perform the following tasks:

What does Imaging FCS in APEER do?

1) Calculation of temporal autocorrelation function (ACF) either at each pixel (or square/ rectangular areas produced by $n \times n$ or $n \times m$ binning) of an image or region of interest (ROI).

2) Dual-Colour Fluorescence Cross-Correlation Spectroscopy (DC-FCCS) if the image is split on the camera according to wavelength.

3) To perform polynomial bleach corrections of the data.

4) To analyse the correlations by fitting with appropriate fit function [1]. The modules can fit auto-correlations, spatial cross-correlations, and dual-colour cross-correlations. Models currently include only two dimensional one diffusive component. We provide simple non-linear least squares or general least squares algorithm.

5) To calculate Model probabilities of fit functions using Bayesian Model Selection.

6) Creation of histogram and map of Diffusion Coefficient in a html document to be from a browser.

7) To create spreadsheet containing post-processed data.

[1] *This document does not explain the theory of correlation analysis as it is only concerned with the use of the Imaging FCS plugin. For details on the theory please see the references.*

2. Modules

2.1. Autocorrelation

Input: 16-bit or 32-bit images in any of standard file format: TIFF, OME-TIFF, TIF

Output: Post-processed data in html and spreadsheet

Correlations are performed on stacks of multiple images acquired at different time points. Each image is made up from a certain number of pixels, and each pixels is associated with an integer value. It allows an entire stacks of frames to be stored as a single "multi-plane" file. Most commercial software allows saving in TIFF format. Conversion from other proprietary images (e.g., czi) can be achieved using "OME-TIFF Converter" module between the input image and the start of the first modules in any of your workflow.

When to use:

Analysing Imaging FCS measurements by mapping concentration and diffusion coefficient in planar structures such as supported lipid bilayers or plasma membranes of cells, and live organism generated either by the exponentially decaying evanescent wave in a Total Internal Reflection Fluorescence (TIRF) microscope or a thin light sheet generated by a Selective Plain Illumination Microscope (SPIM).
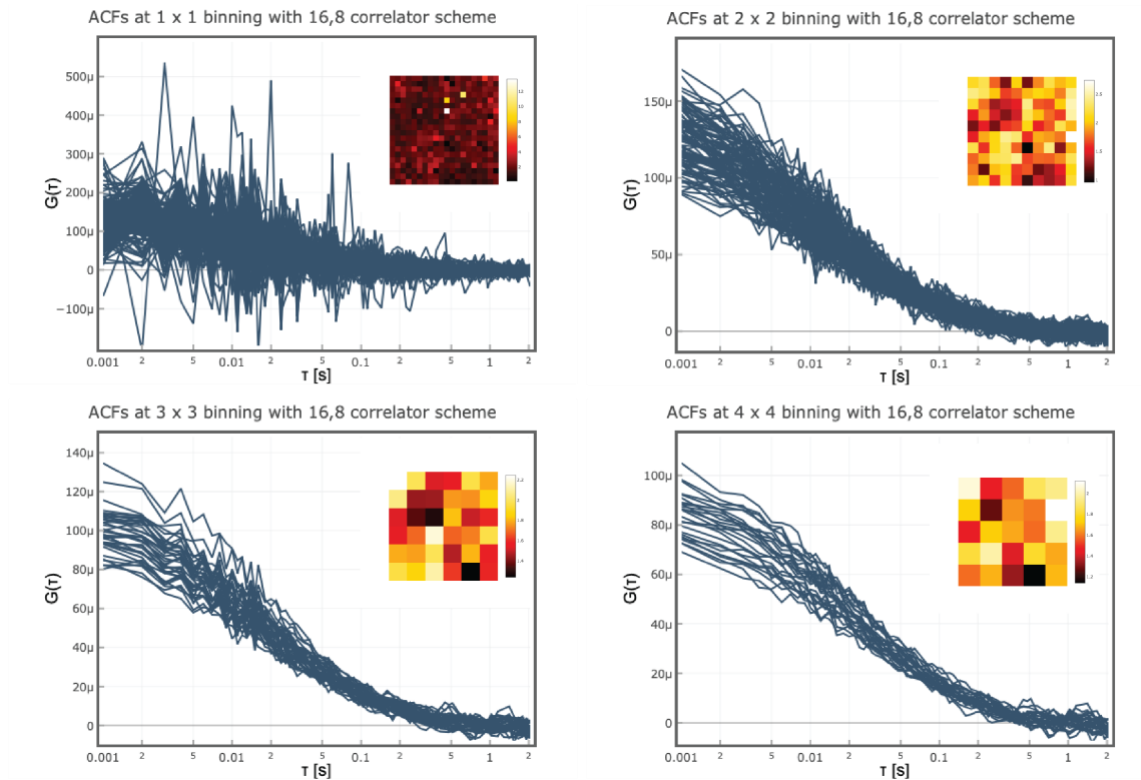
How to use:

It is originally intended to be used after the "ROI selector" module if bleach correction is not desired or after the "Polynomial bleach correction" module if otherwise.

The Autocorrelation (ACF) module is configured with the following user input:

Input Image: Choose single stack from the output of Bleach correction module or directly from camera

Pixel binning: The module provides an option to bin the data which is a process in which the intensities of adjacent pixels are added-up. The number of pixels in X and Y direction to be binned (bin) is determined by the user.

The following shows how pixel binning improves the SNR at the expense of worse spatial resolution. It is recommended to choose the bin size small enough without compromising the quality of the ACF and therefore fit values and high enough without losing too much spatial resolution.

ACFs at 1 x 1 binning with 16,8 correlator scheme

ACFs at 2 x 2 binning with 16,8 correlator scheme

ACFs at 3 x 3 binning with 16,8 correlator scheme

ACFs at 4 x 4 binning with 16,8 correlator scheme

Exposure time: Input the period in seconds between the start of each scan.

Correlator P: The number of correlation channels for the first group of bin times. The bin time is equal to the Exposure time/ acquisition time. This can be set to an integer multiple of 4 ranging from 8 to 64. All following bin times will have half the number of channels, i.e. for 8 for P = 16

Correlator Q: This is the number of groups of bin times (including the first). The bin time within a group is constant but it will always twice as large as the bin time of the preceding group.

Example: Using a Correlator P of 16 and Correlator Q of 6 with a Frame time of 1 ms results in the following time points for the correlation:

Group 1: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16

Group 2: 18, 20, 22, 24, 26, 28, 30, 32

Group 3: 36, 40, 44, 48, 52, 56, 60, 64

Group 4: 72, 80 …

Group 5: 144, 160 …

Group 6: 288, 320 … 512

The module also calculates the 0 lagtime (τ) channel. It is an indication of the shot noise, but actually it is not used in the data fitting. The most common configuration for a typical Diffusion Coefficient (*D*) = $1 - 2\ \mu m^2/s$ is of (16, 8) but could be extended to any of (*p, q*) correlator structure. The choice is an arbitrary one depending of Diffusion Coefficient one's expect. For instance, (16, 8) covers the same maximum lagtime (τ) as (8, 9), (32, 7), (64, 6) option.
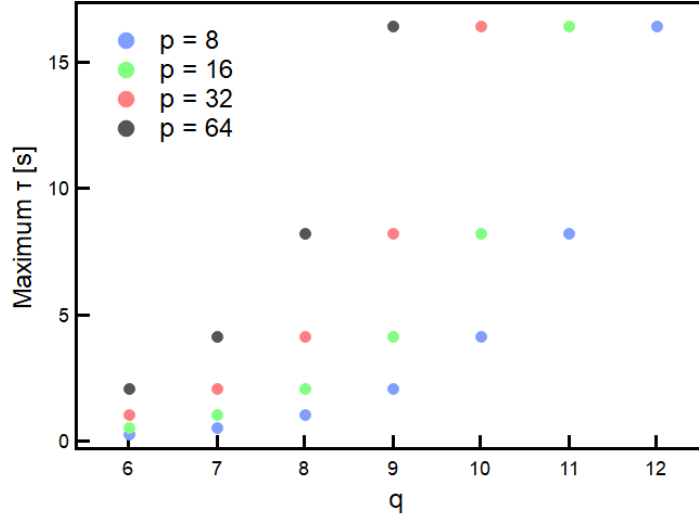
It is also worth noting, the minimum number of frames (*Fr$_{min}$*) needed for a (*p, q*) correlator is

$$Fr_{min} = p - 1 + \frac{p}{2} \sum_{i=1}^{q-1} 2^i$$

The bare minimum frame needed to perform correlation is tabulated in *Table 1*. To carry out correlations, it is advised to perform the correlation with higher number of frames in order to increase the precision of the calculated correlation. The minimum number of frames needed to have at least 100 points in the last channel is tabulated in *Table 2*.

Table 1: Minimum frame (bare minimum) for a given (p, q) architecture

p	q								
	6	7	8	9	10	11	12	13	14
16	544	1088	2176	4352	8704	17408	34816	69632	139264
32	1056	2112	4224	8448	16896	33792	67584	135168	270336
64	2080	4160	8320	16640	33280	66560	133120	266240	532480

Table 2: Minimum frame needed to yield at least 100 points in the last channel given a (p, q) architecture

p	q								
	6	7	8	9	10	11	12	13	14
16	3712	7424	14848	29696	59392	118784	237568	465136	950272
32	4224	8448	16896	33792	67584	135168	270336	540672	1081344
64	5248	10496	20992	41984	83968	167936	335872	671744	1343488

Fitting option: We tested the weighted NLS and GLS fits on simulations. Both fits need a minimum number of frames to converge to the simulated diffusion coefficient. This is about 50,000 frames for the NLS fit and somewhat longer for the GLS fit. However, for the same number of frames the GLS fit will have an about 20% - 30% smaller standard deviation for the diffusion coefficient. Bayesian Model Selection uses either NLS or GLS fits and fits multiple models. So it will take correspondingly longer and probably should be used only in the final evaluation. In future version we will include an alternative model selection methods.

For an accurate prediction of D the module requires input of lateral dimension of PSF. The PSF of the optical system has to be determined experimentally to obtain correct values of D and N. The one-dimensional PSF can be approximated by a Gaussian function centred at x_0 with standard deviation σ.

$$PSF_{(x,x_0)} = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-x_0)^2}{2\sigma^2}}$$

Where standard deviation (σ):

$$\sigma = \frac{\sigma_0 \lambda_{em}}{NA}$$

For simplicity the modules assume σ_0 = 0.8 with NA, magnification and pixel size as indicated by user. A PSF calibration of a typical TIRF system with 100x magnification, 1.49 NA, and $24\ \mu m$ pixel size yield $\sigma_0 \approx 0.8$. This value, which we will call σ, characterizes the $1/e^2$ radius of the PSF in the lateral direction (along the light sheet). For calibration of one's TIRM and light sheet microscope PSF, please proceed to Imaging FCS 1.52 program – a plugin runs under ImageJ.

Camera pixel size*: actual camera pixel size in micro meter

Magnification*: Magnification of the optical system. This is required to calculate the pixel size in object space (=pixel size/magnification).

NA*: Numerical aperture of the microscope objective, used to calculate the PSF.

Emission wavelength*: : Wavelength of emission, λ_{em} , for the fluorophore used. This value is used to calculate the PSF.

Lower threshold : Any pixels (first frame) with intensity lower than the threshold value will be excluded from analysis.

Upper threshold : Any pixels (first frame) with intensity higher than the threshold value will be excluded from analysis.

*Note the following parameters are only necessary if fitting option is selected

2.2. Polynomial bleach correction

*Input: 16-bit images in any of standard file format: TIFF, OME-TIFF, TIF*

Output: 32-bit images in TIFF format

The analysis of ACF assumes fluorescence emission in an equilibrium state, that is the intensity counts average do not change over time scale of the experiment. However, any physical or chemical processes which takes the system away from equilibrium such as photobleaching is difficult to avoid. Hence, successful application of FCS relies on the ability to correct for intensity trace before performing autocorrelation if photobleaching is substantial.

If bleaching is present, the data can be corrected by one of four methods. For simplicity we only include a polynomial bleach correction. Since bleach correction is a memory intensive process and limits in the allocated memory in docker at run time. The correction method works by sampling 1000 equally spaced intensity trace and fit each of sampled traced at by a polynomial fit.

When to use:

There are two ways to probe for the presence of photobleaching. First, is the slow decrease of the intensity over time. Second, strong deviation of G_∞ from 0. In general parameter G_∞ will deviate from 0 by less than 1% in solution measurements.
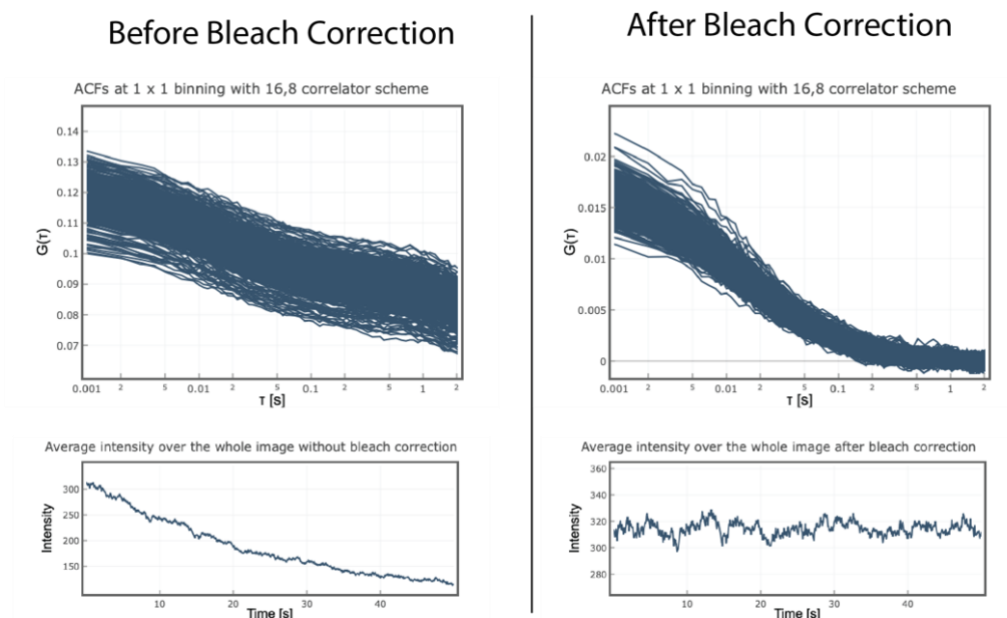
How to use:

It is originally intended to be used after the "ROI selector" module and before the "Autocorrelation" module.

The bleach correction module is configured with the following user input:

Polynomial degree: In our experience a degree in the range of 4 to 8 is sufficient for bleach correction. The polynomial degree should not be selected too high as otherwise the correction can influence also the fast fluctuations which are investigated and the resulting diffusion coefficient will be biased.

2.3. Region of Interest selector

Input: 16-bit images in any of standard file format: TIFF, OME-TIFF, TIF

Output: cropped 16-bit images in TIFF format

Due to default memory limit allocated per module and to ensure a successful autocorrelation workflows, this module crops an image stack along X and Y axis to a maximum of 40 pixels in each direction. The Region of Interest (ROI), sometimes referred to as Area of Interest (AOI) is configured with the following features (the first pixel has a coordinate $(1, 1)$):
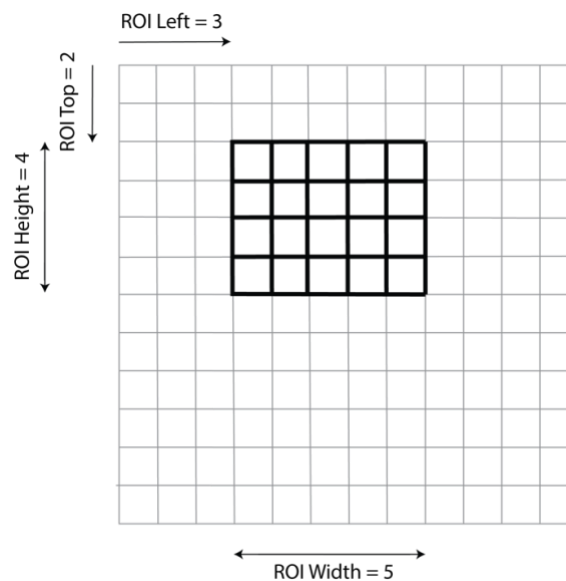
ROI Left: Configures the left hand coordinate of cropped image stack relative to the original area of interest.

ROI Top: Configures the top hand coordinate of cropped image stack relative to the original area of interest.

ROI Width: Configures the width of cropped image stack.

ROI Height: Configures the height of cropped image stack.

When to use:

It is required as a the first step of all FCS workflow.

How to use:

It is originally intended being used before "Polynomial bleach correction" module if bleach correction is desired or "Autocorrelation" module if bleach correction is not required.

2.4. Z binning

Input: 16-bit images in any of standard file format: TIFF, OME-TIFF, TIF

Output: 16-bit images TIFF format

Z binning creates a new substack with a new size smaller by a factor of *Z Shrink factor*. While the Z shrink factor can take on any value, if *group size* does not divide evenly into the stack size, the last group of pixels will be excluded.

For example, 50,000 slices stack and a *Shrink Factor* of 2 output a 16-bit 25,000 multi-plane TIFF. If the Shrink Factor is 3, new substack will contain 16,666 planes with the last 2 planes unused.
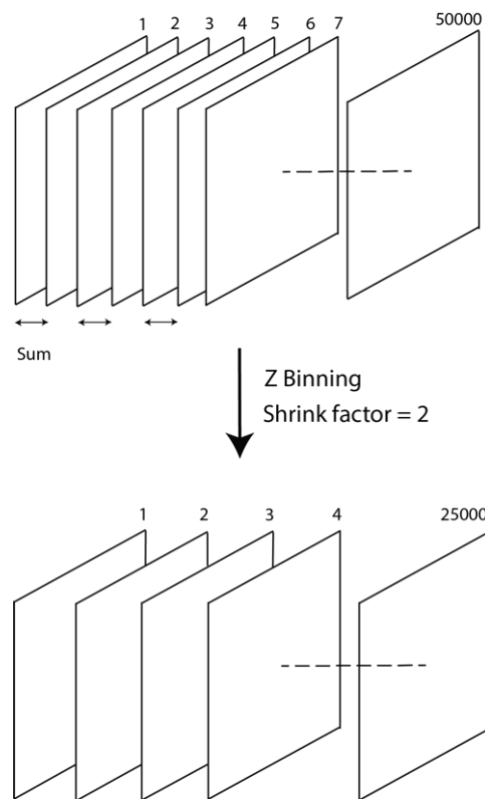
When to use:

It is an optional module to improve the standard deviation of the measurement at the expense of worse temporal resolution. As a rule of thumb, one has to measure at about 10 times faster than the fastest process one wishes to observe. Typically a 1 ms acquisition time per frame is sufficient for observing diffusion dynamics at a range of $D = 1 - 2 \ \mu m^2/s$ and 2 ms frame time suffice for observing slower dynamics, i.e. $D = 0.1 \ \mu m^2/s$.

How to use:

It is originally intended to be used after the "OME-TIFF Converter" module and before the "Polynomial bleach correction" module or "Autocorrelation" module

If one chooses to include the "Z binning" module, it is imperative to provide the correct acquisition time per frame in the subsequent "Autocorrelation" module which is equivalent to the original frame time x *Shrink Factor*. Failure to do so will influence the time axis of resulting ACFs, therefore, biasing the fit results.

2.5. Background Subtraction

Input: 16 bit image stacks from any of standard file format: TIFF, OME-TIFF, TIF.

Output: Background corrected 16-bit images TIFF format.

EMCCDs often shows pattern even on images taken with the shutter closed. Background subtraction module not only allow user to remove pattern noise from low EM gains images. The feature is useful if one wants to avoid fitting background pixels. This speeds up the calculation and reduce fitting problem as background pixels have no correlations and the fit will return meaningless value.
This module allows correcting the data with a background image or subtracting a constant value across all pixels (if background image is not available).

When to use:

Background correction is required in a very low intensity measurements e.g. N&B analysis where one can sometime see the camera noise or FCS with sCMOS camera.

How to use:

Obtain background file:

- "background" file is an image stack recorded with the shutter closed at the same exact location with the same parameters and same number of frames and dimension as the actual data file.

What if there is no background file available?

- The correction is not crucial in FCS as signal-to-noise ratio is typically quite large and thus the patterned noise in negligible.
- Background subtraction will not influence the parameter Diffusion coefficient (D) in the fits.

Background subtraction is configured with the following features, there are 2 modes available depending if background file is available :

Input sample image stacks: Image stacks to be background corrected.

**Input background image stacks:** Background image stacks if available.

Background number*: A constant value to be subtracted across all pixels

*Note the following parameter is only important if one has no associated dark images available.

1. Mode 1 → If background file is available, correct every pixels for its noise individually
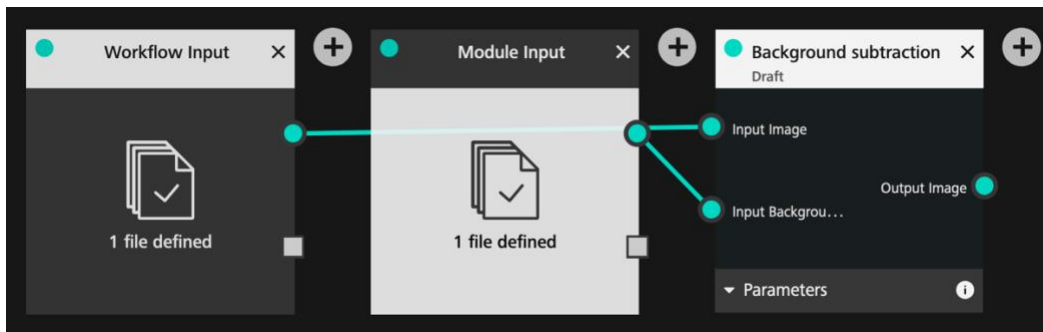
2. Mode 2 → If background file is not available, subtract a constant value across all pixels.
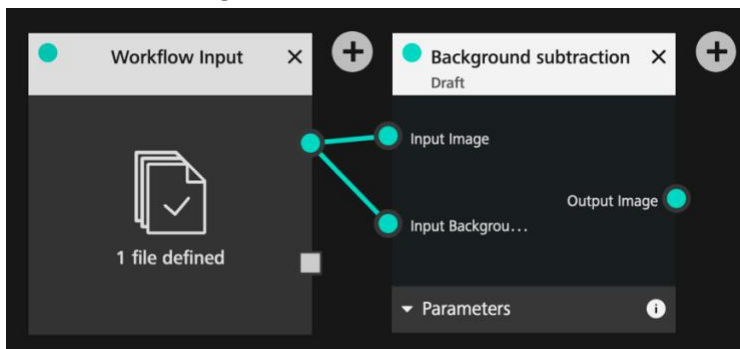
Since APEER will not allow module to run without all input connected, connect "Input Background" to the same source. This will allow user to manually determine a constant background OR let the module automatically determine the smallest pixel value in the "sample stack" to be subtracted across all pixels.

To automatically determine smallest pixel value (default option if background file is not available):

Enter any integer value > 100,000

2.6. Image Splitter

Input: 16 image stacks in any of standard file format: TIFF, OME-TIFF, TIF

Output: Equally divided image stacks. Both of the same dimension.

When to use:

This module is originally intended to be used along with DC-FCCS module. A typical DC-FCCS experiment requires image to be split according to wavelength onto to parts of the camera chip so that images can be displayed side by side on a single camera chip. This module divides an image into two equal halves which are to be cross-correlated.
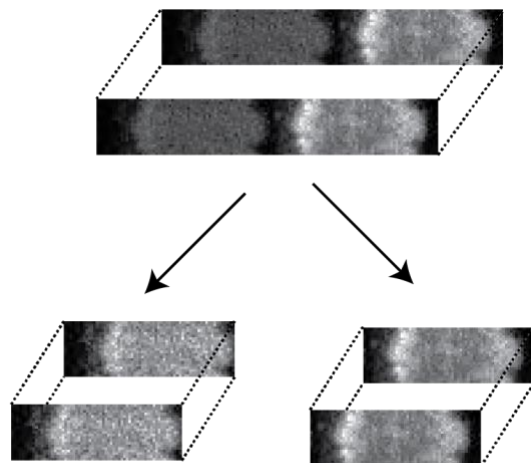
How to use:

The Image Splitter module is configured with the following user input:

Split Left – Right or Split Top - Bottom: Split Images into two equal dimensions
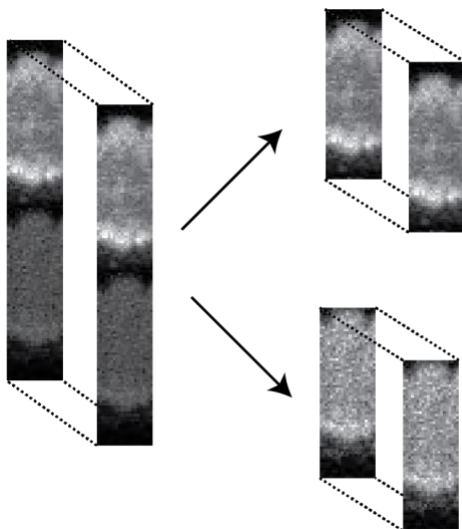
Split Top – Bottom   Split Left – Right

2.7. DC-FCCS

Input: 16/ 32 bit image stacks from green and red channel in any of standard file format: TIFF, OME-TIFF, TIF

*Output: Post-processed data in html and spreadsheet*

Upon the end of the workflow, html file will provide the average intensity trace from each channel, cross-correlation functions, individual autocorrelation function in each channel, and the fit results. If one wishes to depict individual cross-correlations (CCF) , green ACF, red ACF or overlays of CCF and ACFs at each pixels one could make use of the toggle button and slider.
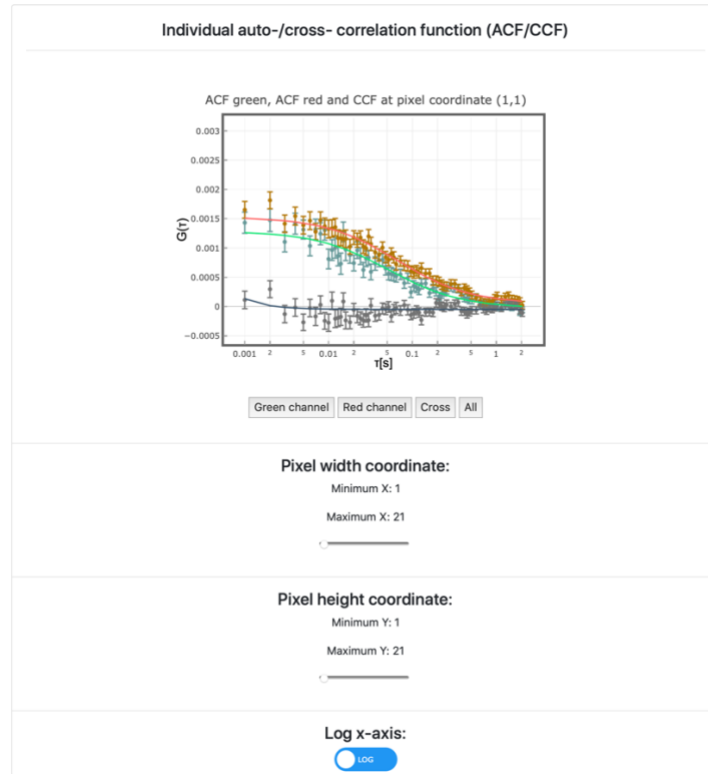

Beside fitting the three individual correlation functions namely: ACFs from green channel, ACFs from red channel and CCFs an additional quantity called cross-correlation amount (*q*) is defied according to [13] as:

$$q = \frac{\max{(N_{green}, N_{red})}}{N_{cross}}$$

And provides a semi-quantitative measure of the fraction of co-diffusing green and red particles. Theoretically, *q* is expected to assume values between 0 (no cross-correlation) and 1 (complete cross-correlation). However, due to imperfect fitting of noisy curves, values > 1 or < 0 can be occasionally encountered. Values > 1 are usually caused by errors in determination of N by fitting very noisy correlation curves, especially in situation where ACF amplitudes are very low. Since such values usually suffer from very large errors, they are excluded from further analysis by setting *q* in such pixels to 0. Values of *q* < 0. Can result from errors in fitting CCFs with amplitudes close to 0; *q* can be set to 0 in such pixels by applying thresholds.
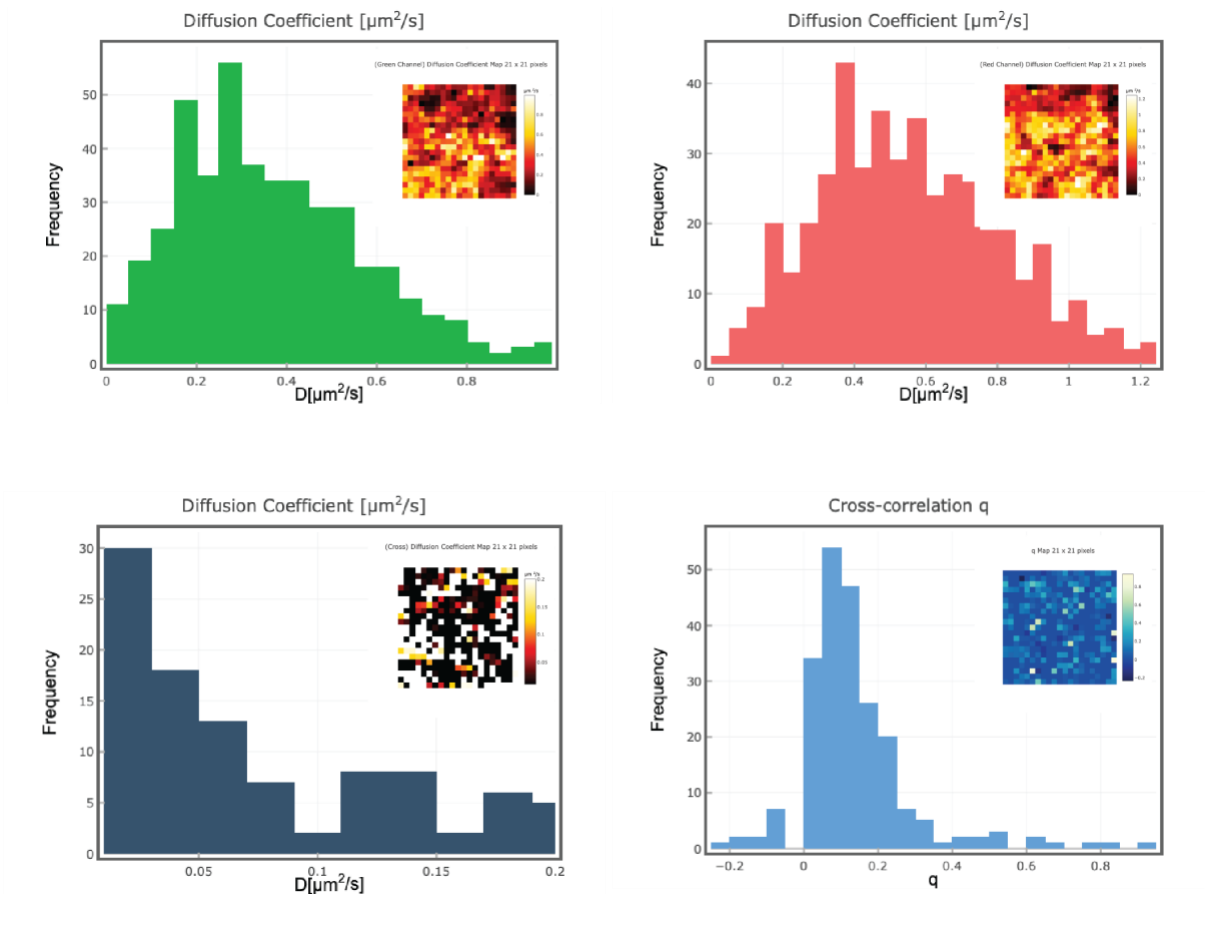
Thresholding functionality can be used to filter out outliers in the parameters maps. By setting the lower and upper limit of fit parameter D or cross-correlation amount (q), the histogram and parameters maps will be updated to better visualize the distribution of parameters. Nevertheless, the original parameter values are kept and saved in the output files (spreadsheet).
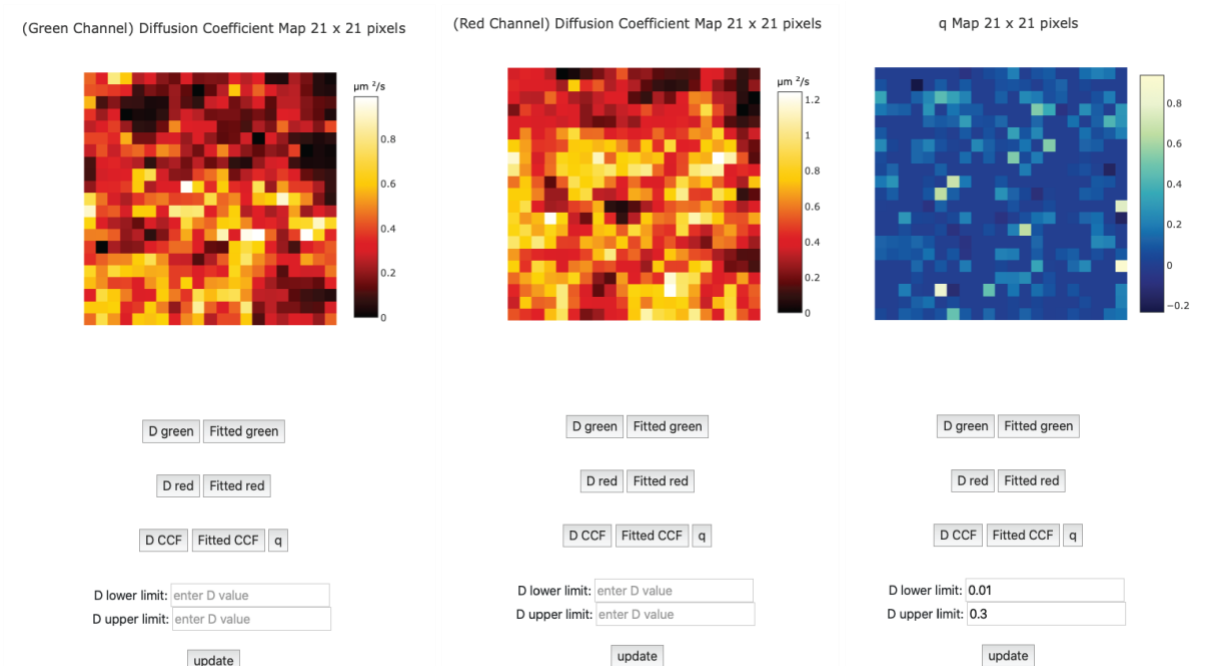
When to use:

DC-FCCS is a tool to measure molecular interactions. It is based on the observation of fluorescence at two distinct wavelength ranges. By cross-correlation the different signals one can determine whether the different molecules/particles interact. Analysing DC-FCCS measurements can be achieved either by the exponentially decaying evanescent wave in a Total Internal Reflection Fluorescence (TIRF) microscope or a thin light sheet generated by a Selective Plain Illumination Microscope (SPIM).

How to use:

DC-FCCS measurements assume that the image was split according to wavelength onto two parts of the image sensor. Since the module correlate image stacks from two distinct wavelength most commonly green and red channel, the module is intended to be used after the "Image Splitter" module followed by "ROI selector" module and the "Polynomial bleach correction" module.

The Dual Colour – Fluorescence Cross Correlation Spectroscopy (FCCS) module is configured with the following user input:

Input Image green: Image stacks from green channel to be analysed.

Input Image red: Same region of Image stacks from red channel to be analysed.

Pixel binning: The module provides an option to bin the data which is a process in which the intensities of adjacent pixels are added-up. The number of pixels in X and Y direction to be binned (bin) is determined by the user.

Exposure time: Input the period in seconds between the start of each scan.

Correlator P: The number of correlation channels for the first group of bin times. The bin time is equal to the Exposure time/ acquisition time. This can be set to an integer multiple of 4 ranging from 8 to 64. All following bin times will have half the number of channels, i.e. for 8 for P = 16

Correlator Q: This is the number of groups of bin times (including the first). The bin time within a group is constant but it will always twice as large as the bin time of the preceding group.

Example: Using a Correlator P of 16 and Correlator Q of 6 with a Frame time of 1 ms results in the following time points for the correlation:

Group 1: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16

Group 2: 18, 20, 22, 24, 26, 28, 30, 32

Group 3: 36, 40, 44, 48, 52, 56, 60, 64

Group 4: 72, 80 …

Group 5: 144, 160 …

Group 6: 288, 320 … 512

Fitting option: We tested the weighted NLS and GLS fits on simulations. Both fits need a minimum number of frames to converge to the simulated diffusion coefficient. This is about 50,000 frames for the NLS fit and somewhat longer for the GLS fit. However, for the same number of frames the GLS fit will have an about 20% - 30% smaller standard deviation for the diffusion coefficient. Bayesian Model Selection uses either NLS or GLS fits and fits multiple models. So it will take correspondingly longer and probably should be used only in the final evaluation. In future version we will include an alternative model selection methods.

For an accurate prediction of D the module requires input of lateral dimension of green and red PSF . The PSFs of the optical system has to be determined experimentally to obtain correct values of D and N. The one-dimensional PSF can be approximated by a Gaussian function centred at x_0 with standard deviation σ.

$$PSF_{(x,x_0)} = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-x_0)^2}{2\sigma^2}}$$

Where standard deviation (σ):

$$\sigma = \frac{\sigma_0 \lambda_{em}}{NA}$$

For simplicity the modules assume σ_0 = 0.8 for both green and red PSF with NA, magnification and pixel size as indicated by user. A PSF calibration of a typical TIRF system with 100x magnification, 1.49 NA, and $24\ \mu m$ pixel size yield $\sigma_0 \approx$ 0.8 . This value, which we will call σ, characterizes the $1/e^2$ radius of the PSF in the lateral direction (along the light sheet). For calibration of one's TIRM and light sheet microscope PSF, please proceed to Imaging FCS 1.52 program – a plugin runs under ImageJ.

Camera pixel size*: actual camera pixel size in micro meter

Magnification*: Magnification of the optical system. This is required to calculate the pixel size in object space (=pixel size/magnification).

NA*: Numerical aperture of the microscope objective, used to calculate the PSF.

Emission wavelength green*: : Wavelength of emission, $\lambda_{em\ green}$, for the green fluorophore used. This value is used to calculate the green PSF.

Emission wavelength red*: : Wavelength of emission, $\lambda_{em\ red}$, for the red fluorophore used. This value is used to calculate the red PSF.

Lower threshold : Intensity threshold is based on the intensity counts of green image stacks. Any pixels (first frame of green image stacks) with intensity lower than the threshold value will be excluded from analysis.
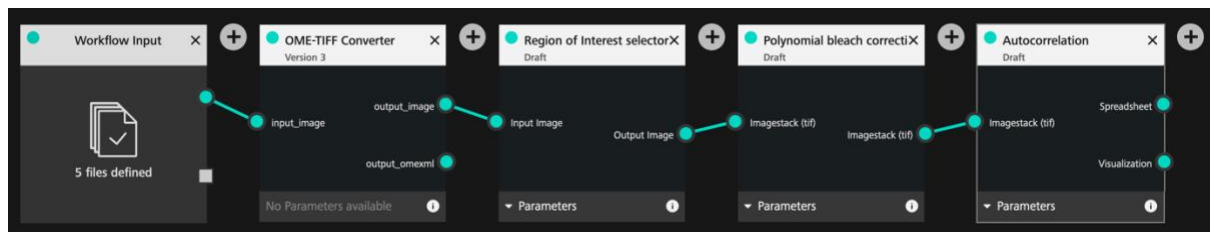
Upper threshold : Intensity threshold is based on the intensity counts of green image stacks. Any pixels (first frame of green image stacks) with intensity higher than the threshold value will be excluded from analysis.
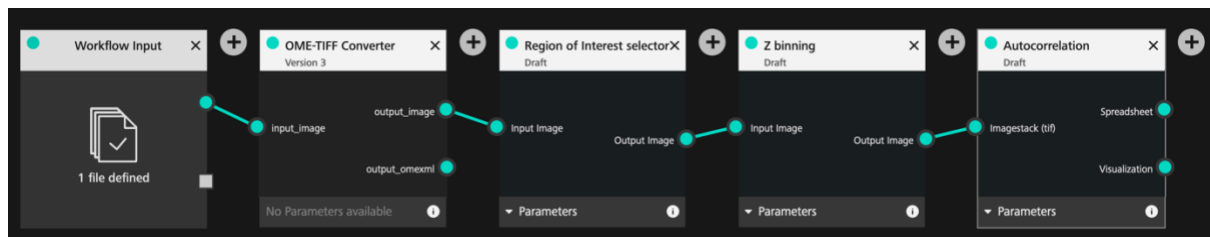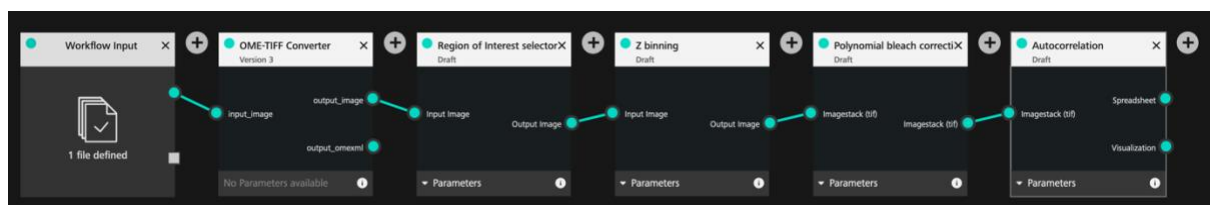
3. Workflow

Autocorrelation
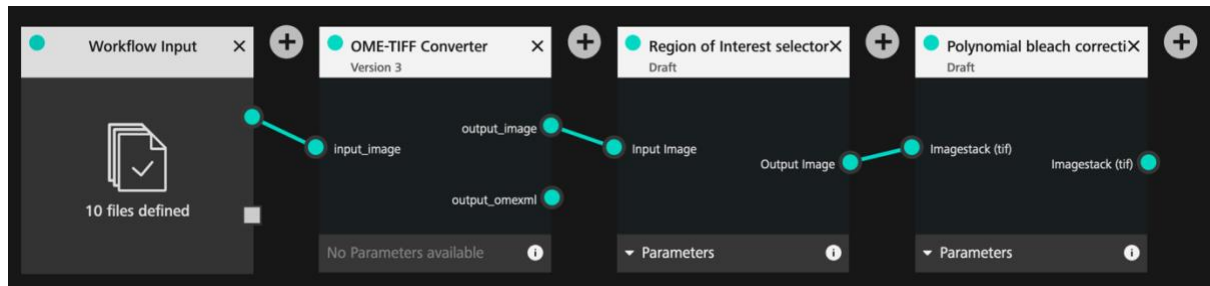
Autocorrelation + Bleach correction

Autocorrelation + Frame binning

Autocorrelation + Bleach correction + Frame binning

4. Annex 1: Fitting Functions

Note that the full calculations for the fitting functions have been provided in the literature (10, 13, 19-24) and can also be obtained from interactive CDF files on our website. Therefore, we do not provide the full derivations here. In particular, we left out all calculations of the effective volumes and just included their results in the final functions.

A note on the SPIM model: At the moment, the given fitting function does not take into account pixel cross-talk. It will fit ACFs but the recovered D is slightly biased (up to 30%). But it will give incorrect and distance dependent D values for spatial CCFs. A new model will be provided in the next upgrade.

The autocorrelation function *G(τ)* is typically expressed as

$$G(\tau) = g(\tau) + G_{\infty}$$

where *g(τ)* is the temporal part of the autocorrelation function and G_{∞} is the value of convergence for $\tau \to \infty$. The value of G_{∞} is 1 or 0 depending on the definition of the autocorrelation function (see our website here; note that you need the CDF plugin from Wolfram, which is freely downloadable, to see this interactive pages). It is typically left as a fit parameter as the condition $\tau \to \infty$ cannot be fulfilled. The value of G_{∞} differs from its ideal value usually very little, within a range of a few %.

If the amplitude *G(0)* of the autocorrelation function *G(τ)* is inversely proportional to the number of particles in the observation volume, then

$$G(0) = \frac{1}{N} + G_{\infty}$$

For a full derivation see (10, 23). Therefore, we can write

$$G(\tau) = \frac{1}{N} \frac{g(\tau)}{g(0)} + G_{\infty}$$

Below we will give the functions of $g(\tau)$ from which $g(0)$ and $G(\tau)$ can be derived. Note that this definition is used in all model function in this plugin. However, it makes strictly only sense for the temporal auto- and cross-correlations without a spatial shift. For spatial cross-correlations N is more difficult to interpret as it depends on the particles that pass from one area to the next and the other particles that contribute to the fluorescence but not to the correlations.

The model for diffusion and flow for one particular component is given by

$$
g_i(\tau) = \left(\sqrt{\frac{4D_i\tau + \sigma_{xy}^2}{\pi}} \left(e^{-\frac{(a+r_x-\tau v_x)^2}{4D_i\tau+\sigma_{xy}^2}} + e^{-\frac{(a-r_x+\tau v_x)^2}{4D_i\tau+\sigma_{xy}^2}} - 2e^{-\frac{(r_x-\tau v_x)^2}{4D_i\tau+\sigma_{xy}^2}} \right) + (a+r_x-\tau v_x) Erf\left(\frac{(a+r_x-\tau v_x)}{\sqrt{4D_i\tau+\sigma_{xy}^2}} \right) \right.
$$
$$
\left. + (a-r_x+\tau v_x) Erf\left(\frac{(a-r_x+\tau v_x)}{\sqrt{4D_i\tau+\sigma_{xy}^2}} \right) - 2(r_x-\tau v_x) Erf\left(\frac{(r_x-\tau v_x)}{\sqrt{4D_i\tau+\sigma_{xy}^2}} \right) \right)
$$
$$
\times \left(\sqrt{\frac{4D_i\tau + \sigma_{xy}^2}{\pi}} \left(e^{-\frac{(a+r_y-\tau v_y)^2}{4D_i\tau+\sigma_{xy}^2}} + e^{-\frac{(a-r_y+\tau v_y)^2}{4D_i\tau+\sigma_{xy}^2}} - 2e^{-\frac{(r_y-\tau v_y)^2}{4D_i\tau+\sigma_{xy}^2}} \right) + (a+r_y-\tau v_y) Erf\left(\frac{(a+r_y-\tau v_y)}{\sqrt{4D_i\tau+\sigma_{xy}^2}} \right) \right.
$$
$$
\left. + (a-r_y+\tau v_y) Erf\left(\frac{(a-r_y+\tau v_y)}{\sqrt{4D_i\tau+\sigma_{xy}^2}} \right) - 2(r_y-\tau v_y) Erf\left(\frac{(r_y-\tau v_y)}{\sqrt{4D_i\tau+\sigma_{xy}^2}} \right) \right)
$$
$$
\times \left(1 + \frac{4D\tau}{\sigma_z^2} \right)^{-\frac{1}{2}}
$$

Here the first two terms describe the diffusion and flow in the *xy* directions along the pixel, and the last term describes the diffusion (no flow is incorporated at the moment for the SPIM case) in the *z* direction, orthogonal to the pixels and across the light sheet.

The ***observation volumes*** in imaging FCS are given by

$$A_{eff}(\tau) = \frac{a^2}{\left(\frac{\sigma_0}{a\sqrt{\pi}} \left(e^{-\frac{a^2}{\sigma_0^2}} - 1 \right) + Erf\left(\frac{a}{\sigma_0} \right) \right)^2}$$

for ITIR-FCS, and by

$$V_{eff} = \frac{\sqrt{\pi}a^2\sigma_z}{\left(\frac{\sigma_{xy}}{a\sqrt{\pi}} \left(e^{-\frac{a^2}{\sigma_{xy}^2}} - 1 \right) + Erf\left(\frac{a}{\sigma_{xy}} \right) \right)^2}$$

for SPIM-FCS.

For DC-FCCS we provide here the simplest cross-correlation function. It assumes one diffusion coefficient and light sheets with different size. But both light sheets are perfectly centred:

$$G_{SPIM}(\tau) = \frac{1}{N} \frac{\left(\frac{\sqrt{4D\tau + \frac{\sigma_{01}^2}{2} + \frac{\sigma_{02}^2}{2}}}{a\sqrt{\pi}} \left(e^{-\frac{a^2}{4D\tau + \frac{\sigma_{01}^2}{2} + \frac{\sigma_{02}^2}{2}}} - 1 \right) + Erf\left(\frac{a}{\sqrt{4D\tau + \frac{\sigma_{01}^2}{2} + \frac{\sigma_{02}^2}{2}}} \right) \right)^2 \times \left(1 + \frac{8D\tau}{\sigma_{z,01}^2 + \sigma_{z,02}^2} \right)^{-\frac{1}{2}}}{\left(\frac{\sqrt{\frac{\sigma_{01}^2}{2} + \frac{\sigma_{02}^2}{2}}}{a\sqrt{\pi}} \left(e^{-\frac{a^2}{\frac{\sigma_{01}^2}{2} + \frac{\sigma_{02}^2}{2}}} - 1 \right) + Erf\left(\frac{a}{\frac{\sigma_{01}^2}{2} + \frac{\sigma_{02}^2}{2}} \right) \right)^2} + G_{\infty}$$

In the plugin we allow a possible shift in the light sheet direction (r_z) but assume that the pixels for the cross-correlations are perfectly aligned. This leads to the following function (13).

$$G_{SPIM}(\tau)=\frac{1}{N}\frac{\left(\frac{\sqrt{4D\tau+\frac{\sigma_{01}^2}{2}+\frac{\sigma_{02}^2}{2}}}{a\sqrt{\pi}}\left(e^{-\frac{a^2}{4D\tau+\frac{\sigma_{01}^2}{2}+\frac{\sigma_{02}^2}{2}}}-1\right)+Erf\left(\frac{a}{\sqrt{4D\tau+\frac{\sigma_{01}^2}{2}+\frac{\sigma_{02}^2}{2}}}\right)\right)^2\times\left(1+\frac{8D\tau}{\sigma_{z,01}^2+\sigma_{z,02}^2}\right)^{-\frac{1}{2}}e^{-\frac{2r_z^2}{8D\tau+\sigma_{z,01}^2+\sigma_{z,02}^2}}}{\left(\frac{\sqrt{\frac{\sigma_{01}^2}{2}+\frac{\sigma_{02}^2}{2}}}{a\sqrt{\pi}}\left(e^{-\frac{a^2}{\frac{\sigma_{01}^2}{2}+\frac{\sigma_{02}^2}{2}}}-1\right)+Erf\left(\frac{a}{\frac{\sigma_{01}^2}{2}+\frac{\sigma_{02}^2}{2}}\right)\right)^2 e^{-\frac{2r_z^2}{\sigma_{z,01}^2+\sigma_{z,02}^2}}}+G_\infty$$

The Java code for Imaging FCS contains the functions for the most general two component cases and their derivatives for all parameters for data fitting.

Non-square, rectangular binning

In Fig. A2-1 we show the general definition for the derivation of the function. However, we implemented at the moment rectangular areas of the same dimensions and orientations only. This means that (k-l) = (m-n) and (q-p) = (s-r).
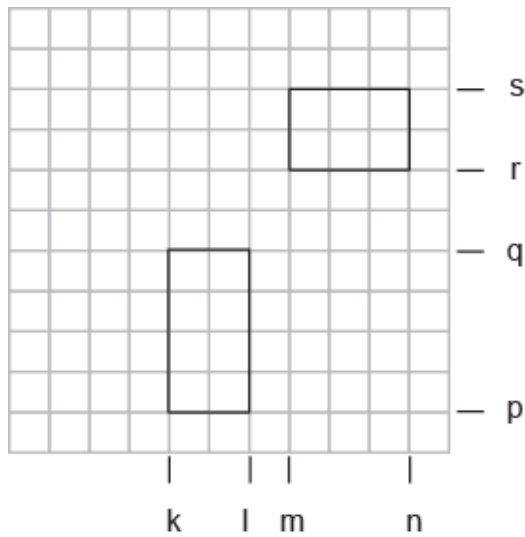

Fig. A2-1: Schematic representation of the regions on a CCD chip which were used for the derivation of the generalized cross-correlation functions of ITIR-FCCS.

The function was originally derived in ref [1]:

$$g_{general}(\tau) = \frac{1}{\left(4 <C> (k-l)(m-n)(p-q)(r-s)\right)} \times$$

$$\begin{bmatrix} \dfrac{2\sqrt{D\tau+\sigma^2}}{\sqrt{\pi}} \left(e^{-\frac{(l-m+v_x\tau)^2}{4(D\tau+\sigma^2)}} + e^{-\frac{(k-n+v_x\tau)^2}{4(D\tau+\sigma^2)}} - e^{-\frac{(l-n+v_x\tau)^2}{4(D\tau+\sigma^2)}} - e^{-\frac{(k-m+v_x\tau)^2}{4(D\tau+\sigma^2)}} \right) \\[2em] + (l-m+v_x\tau)\, erf\left(\dfrac{l-m+v_x\tau}{2\sqrt{D\tau+\sigma^2}} \right) + (k-n+v_x\tau)\, erf\left(\dfrac{k-n+v_x\tau}{2\sqrt{D\tau+\sigma^2}} \right) \\[2em] - (l-n+v_x\tau)\, erf\left(\dfrac{l-n+v_x\tau}{2\sqrt{D\tau+\sigma^2}} \right) - (k-m+v_x\tau)\, erf\left(\dfrac{k-m+v_x\tau}{2\sqrt{D\tau+\sigma^2}} \right) \end{bmatrix} \times$$

$$\begin{bmatrix} \dfrac{2\sqrt{D\tau+\sigma^2}}{\sqrt{\pi}} \left(e^{-\frac{(q-r+v_y\tau)^2}{4(D\tau+\sigma^2)}} + e^{-\frac{(p-s+v_y\tau)^2}{4(D\tau+\sigma^2)}} - e^{-\frac{(q-s+v_y\tau)^2}{4(D\tau+\sigma^2)}} - e^{-\frac{(p-r+v_y\tau)^2}{4(D\tau+\sigma^2)}} \right) \\[2em] + (q-r+v_y\tau)\, erf\left(\dfrac{q-r+v_y\tau}{2\sqrt{D\tau+\sigma^2}} \right) + (p-s+v_y\tau)\, erf\left(\dfrac{p-s+v_y\tau}{2\sqrt{D\tau+\sigma^2}} \right) \\[2em] - (q-s+v_y\tau)\, erf\left(\dfrac{q-s+v_y\tau}{2\sqrt{D\tau+\sigma^2}} \right) - (p-r+v_y\tau)\, erf\left(\dfrac{p-r+v_y\tau}{2\sqrt{D\tau+\sigma^2}} \right) \end{bmatrix}$$

5. References

1. Sankaran, J., M. Manna, L. Guo, R. Kraut, and T. Wohland. 2009. Diffusion, Transport, and Cell Membrane Organization Investigated by Imaging Fluorescence Cross-Correlation Spectroscopy. Biophysj. 97: 2630–2639.

2. He, J., S.-M. Guo, and M. Bathe. 2012. Bayesian Approach to the Analysis of Fluorescence Correlation Spectroscopy Data I: Theory. Anal Chem. 84: 3871–3879.

3. Guo, S.-M., J. He, N. Monnier, G. Sun, T. Wohland, and M. Bathe. 2012. Bayesian Approach to the Analysis of Fluorescence Correlation Spectroscopy Data II: Application to Simulated and In Vitro Data. Anal Chem. 84: 3880–3888.

4. Guo, S.-M., N. Bag, A. Mishra, T. Wohland, and M. Bathe. 2014. Bayesian Total Internal Reflection Fluorescence Correlation Spectroscopy Reveals hIAPP-Induced Plasma Membrane Domain Organization in Live Cells. Biophysj. 106: 190–200.

5. Bag, N., J. Sankaran, A. Paul, R.S. Kraut, and T. Wohland. 2012. Calibration and limits of camera-based fluorescence correlation spectroscopy: a supported lipid bilayer study. ChemPhysChem. 13: 2784–2794.

6. Wawrezinieck, L., H. Rigneault, D. Marguet, and P.-F. Lenne. 2005. Fluorescence correlation spectroscopy diffusion laws to probe the submicron cell membrane organization. Biophysj. 89: 4029–4042.

7. Brinkmeier, M., K. Dörre, J. Stephan, and M. Eigen. 1999. Two-beam cross-correlation: a method to characterize transport phenomena in micrometer-sized structures. Anal Chem. 71: 609–616.

8. Przybylski, A., B.X.R. Thiel, J. Keller-Findeisen, B. Stock, and M. Bates. 2017. Gpufit: An open-source toolkit for GPU-accelerated curve fitting. Sci. Rep. : 1–9.

9. Wohland, T., R. Rigler, and H. Vogel. 2001. The standard deviation in fluorescence correlation spectroscopy. Biophys J. 80: 2987–2999.

10. Sankaran, J., N. Bag, R.S. Kraut, and T. Wohland. 2013. Accuracy and Precision in Camera-Based Fluorescence Correlation Spectroscopy Measurements. Anal Chem. 85: 3948–3954.

11. Macháň, R., Y.H. Foo, and T. Wohland. 2016. On the Equivalence of FCS and FRAP: Simultaneous Lipid Membrane Measurements. Biophysj. 111: 152–161.

12. Freedman, D., and P. Diaconis. 1981. On the Histogram as a Density Estimator: L2 Theory. Probab. Theory and Relat. Fields. : 453–476.

13. Krieger, J.W., A.P. Singh, C.S. Garbe, T. Wohland, and J. Langowski. 2014. Dual-Color Fluorescence Cross-Correlation Spectroscopy on a Single Plane Illumination Microscope (SPIM-FCCS). Opt Express. 22: 2358.

14.   Bag, N., X.W. Ng, J. Sankaran, and T. Wohland. 2016. Spatiotemporal mapping of diffusion dynamics and organization in plasma membranes. Methods Appl. Fluoresc. : 1–15.

15.   Shusterman, R., S. Alon, T. Gavrinyov, and O. Krichevsky. 2004. Monomer Dynamics in Double- and Single-Stranded DNA Polymers. Phys. Rev. Lett. 92: 048303.

16.   Unruh, J.R., and E. Gratton. 2008. Analysis of Molecular Concentration and Brightness from Fluorescence Fluctuation Data with an Electron Multiplied CCD Camera. Biophysj. 95: 5385–5398.

17.   Ries, J., S. Chiantia, and P. Schwille. 2009. Accurate Determination of Membrane Dynamics with Line-Scan FCS. Biophysj. 96: 1999–2008.

18.   Hirsch, M., R.J. Wareham, M.L. Martin-Fernandez, M.P. Hobson, and D.J. Rolfe. 2013. A stochastic model for electron multiplication charge-coupled devices--from theory to practice. PLoS ONE. 8: e53671.

19.   Guo, L., J.Y. Har, J. Sankaran, Y. Hong, B. Kannan, and T. Wohland. 2008. Molecular Diffusion Measurement in Lipid Bilayers over Wide Concentration Ranges: A Comparative Study. ChemPhysChem. 9: 721–728.

20.   Ries, J., E.P. Petrov, and P. Schwille. 2008. Total Internal Reflection Fluorescence Correlation Spectroscopy: Effects of Lateral Diffusion and Surface-Generated Fluorescence. Biophys J. 95: 390–399.

21.   Wohland, T., X. Shi, J. Sankaran, and E. Stelzer. 2010. Single plane illumination fluorescence correlation spectroscopy (SPIM-FCS) probes inhomogeneous three-dimensional environments. Opt Express. 18: 10627–10641.

22.   Sankaran, J., X. Shi, L.Y. Ho, E.H.K. Stelzer, and T. Wohland. 2010. ImFCS: a software for imaging FCS data analysis and visualization. Opt Express. 18: 25468–25481.

23.   Singh, A.P., J.W. Krieger, J. Buchholz, E. Charbon, J. Langowski, and T. Wohland. 2013. The performance of 2D array detectors for light sheet based fluorescence correlation spectroscopy. Opt Express. 21: 8652.

24.   Thompson, N.L. 1991. Fluorescence correlation spectroscopy. In: Lakowicz JR, editor. Topics in Fluorescence Spectroscopy, Vol. 1: Techniques. New York: Plenum Press. pp. 337–378.