SOFI Software Manual

May 29, 2022

Contents

1 Introduction

The super-resolution optical fluctuation (SOFI) imaging technique is based on higher-order statistical analysis of the time-series of fluctuating fluorophores. The spatiotemporal cross-cumulant analysis leads to a super-resolution point-spread function raised to the power of the cumulant order n, i.e., a direct resolution improvement of $\sqrt{n}$ for the reconstructed SOFI image in all directions. With subsequent postprocessing either by Fourier reweighing or deconvolution, the resolution increases with growing cumulant order n (named "SOFI order"). As an example, 4th order SOFI achieves up to 4-fold resolution enhancement.

In addition to increasing the resolution, cross-cumulant analysis also increases the pixel sampling. This does not only work in 2D imaging, but also allows finer sampling and resolution increase in the z-direction if e.g., data is acquired in two or more axially separated planes simultaneously (here referred to as biplane and 3D SOFI with more than 2 image planes). The same principle allows the generation of additional color channels when using spectral cross-cumulation between two or more simultaneously acquired color channels (here referred to as multicolor SOFI). Using SOFI, one can perform different quantitative analysis. Combining 3 orders of SOFI analysis allows the estimation of molecular parameters such as the on-time ratio, the molecular brightness, and density. Only a few hundred or thousand frames are needed for cumulant calculation. After collecting the frames, they need to be analyzed and post-processed. The software described in this manual allows to generate super-resolved images for 2D imaging and molecular parameter estimation, biplane, 3D multiplane SOFI and for SOFI based multicolor imaging.

2 Definitions

SOFI super-resolution optical fluctuation imaging

Raw data refers to camera images, saved e.g. as tiff files or in a proprietary format. The pixel gray values are Analog Digital Units (ADU) and include an offset.

Metadata is data contained in the raw data that describes the acquisition settings. Additional camera metadata might be required to read in files stored in binary format.

ROI regions of interest: 1. Unless specified, the software divides the camera data into multiple planes and uses the maximal region for each plane. It is possible to crop the maximum common ROI after co-registration of all planes. 2. Users can specify an ROI in the config file. Only coordinates in this ROI are then used for further analysis.

Subsequence the input image sequence is often divided into smaller sequences referred to as subsequences or batches, that are analyzed separately and subsequently averaged.

SNR Signal-to-noise ratio

FRC Fourier Ring Correlation

PSF point spread function

3 Installation

3.1 Quick Start

The SOFI package is developed in MATLAB with the core cumulant calculation provided for GPU (CUDA) and for CPU (C++ code implemented via Mexx files).

The code was tested in MATLAB R2016b and MATLAB R2021A, under Windows 7, Windows 10 and Mac and requires the Image Processing and Parallel Computing toolboxes. No additional software installation is required.

1. Clone the repository

```
        git clone https://github.com/kgrussmayer/sofipackage.git
```

2. Download test data here and move them into ./data folder into the sofipackage repository. There should be:

   - ./sofi/sofi2d
   - ./sofi/sofi3d
   - ./sofi/sofi3mc
   - ./sofi/sofibiplane

3. Open sofipackage directory in MATLAB. Please make sure that it is set as ”current folder” and it appears in the MATLAB path.
4. Launch tests ./tests/test_expected_results.m All tests should pass, results and figures will be saved in automatically created results folder. The analysis of all the test data should take approx. 3min on a standard desktop computer (with at least 16GB RAM).
5. Run your own experiments. Example experiments for all 4 SOFI modalities are in ./experiments this includes SOFI2D, SOFI3D, SOFI multicolor and SOFI biplane. For a new experiment with SOFI3D, create a new copy of experiment_sofi3d.m into a new folder for example ./experiment/-sofi3d/20210630. Edit configuration in the new experiment (change input files, output path etc.). Run the experiment while always keeping sofipackage as the ”current folder” in MATLAB.

GPU Version

The cumulant calculation can be accelerated by GPU. It requires that the computer has a CUDA-enabled NVIDIA GPU (http://ch.mathworks.com/discovery/MATLAB-gpu.html).

If you have the MATLAB Parallel Processing Toolbox and a NVIDIA graphics card, write the following command on the ”Command Window”:

```
gpuDevice
```

It should display:

```
ans =
        CUDADevice with properties:
```

with a list of properties. Note the 'ComputeCapibility' of your graphics card.

Compilation for your GPU card:
Step 1: note the 'ComputeCapibility' of your graphics card which is for example '2.0' in the case of NVIDIA Geforce GTX 480. The compute capability of a graphics card can also be found here:
https://developer.nvidia.com/cuda-gpus
Step 2: Set the MATLAB current folder to `multicolor_sofi\funcs\private`. The tab ”Current folder” should display files including 'gpu.cu', 'gpu.ptx', 'nvcc.m' and 'nvccbat.bat'.
Step 3: Execute the following command on the ”Command Window”:

```
nvcc -arch=sm_20 -ptx gpu.cu
```

This command launches the NVIDIA compiler to recompile gpu.cu. Make sure that the -arch option is set to the compute capability of your CUDA-capable graphics card. In the example displayed above, it is 2.0 (the compute capibility of NVIDIA Geforce GTX 480).

CPU Version

We recommend using the GPU version of the SOFI algorithm (see above). The Mexx files included in the current software package are compiled for Win32bit systems and we provide the original files that the user can compile for their computer system.

3.2 Example Workflow

After creating a new copy of one of the SOFI analysis folders (SOFI2D/SOFI3D/Multicolor SOFI/Biplane SOFI), the configuration needs to be edited depending on the particular experiment. The following example workflow (for SOFI3D) outlines what should be configured before running the software.

1. Define the input path, the calibraton file path, and the datatype of your images. You can specify a specific image file to be processed, or leave the ”imageName” setting blank to process all the files within the input folder.
2. For the calibration settings, you can use the default settings or tune them to try to get better output images.
3. In the cumulant calculation section, you should specify settings such as:
   - the SOFI orders you would like to be processed (higher orders take longer processing times)
   - length of the subsequence used in cumulant calculation, which can be tuned to get better results
   - number of the frame you would like to start processing from (leave blank to process from the start)
   - number of the frame you would like to end processing at (leave blank to process till the end)
   - step size of the sliding window moving over subsequences, which can be tuned to get better results
   - number of planes in the setup
   - file extension
   - channel weights to correct differences in intensity within the different channels
4. For the post-processing settings, you can use the default settings or tune them (e.g. modify the deconvolution parameters, such as number of iterations, whether to do linearization, filtering and averaging or not, which PSF model to use)
5. In the output settings section, you can modify parameters like the number of bits of the output images, whether to save figures or not (and in which format). You should specify the output folder where the output images will be saved. After this you can finally run the software.

debugging section

4 SOFI Theory

SOFI applies high order statistics to exploit the temporal blinking sequence of fluorescent emitters. More precisely, SOFI is based on calculating spatio-temporal cross-cumulants to obtain a super-resolved, background-free and noise-reduced image using a conventional widefield microscope. As stated by Dertinger et al. (PNAS, 2009), the fluctuating emitters should fulfill the following conditions:

1. The markers should switch between at least two optically distinguishable states, e.g. a dark and a bright state.
2. Each emitter switches between the states repeatedly and independently in a stochastic manner.
3. The point-spread image of each emitter has to extend over several camera pixels.

As shown in Figure 1a, images of stochastically blinking emitters are recorded such that the point-spread function (PSF) is spread over several camera pixels. By acquiring a stack of images over time, one obtains a time trace for each pixel (Figure 1b). The time trace contains the sum of the contributions of every fluorophore whose PSF reaches the pixel. Assuming N independently fluctuating emitters, the detected intensity can be described as

$$I(\mathbf{r},t) = \sum_{k=1}^{N} \epsilon_k U\left(\mathbf{r}-\mathbf{r}_k\right) s_k(t) + b(\mathbf{r}) + n(\mathbf{r},t) \tag{1}$$

where ϵ_k is the molecular brightness, $U\left(r - r_k\right)$ is the PSF of the optical system, $s_k(t)$ denotes a switching function (normalized fluctuation sequence, $s_k(t) \in \{0,1\}$), b(r) is a constant background, and n(r, t) represents an additive noise. The sample is assumed to be stationary during the image acquisition.

Random blinking of an emitter is spatio-temporally correlated with itself and uncorrelated with neighboring emitters. For each pixel, a measure of correlation in a form of an nth order cumulant is calculated for a better discrimination of emitters inside the PSF volume. Using cross-cumulants, values for virtual pixels in between the physical pixel grid can be also calculated in order to obtain a finer sampling of the image. Figure 1c shows a 2nd order cross-cumulant with various time lags calculated for each pixel time trace (Figure 1b). Note that the 2nd order cross-cumulant is mathematically equivalent to the 2nd cross-correlation. Cumulants are appropriate for a generalization of SOFI to higher orders, because unlike higher order correlations, nth order cumulants does not contain lower order cross-terms which would hamper the resolution enhancement.

Generally, spatio-temporal cumulants can be calculated with various time lags. For reducing the computational complexity and ensuring the maximum of the signal, zero time lag is used. Virtual pixels can be calculated in between the physical pixels acquired by the camera using cross-cumulants and followed by a flattening operation (i.e. assigning proper weights to virtual pixels). By applying the nth order cumulant to Eq. 1, we obtain

$$\kappa_n\{I(\mathbf{r},t)\}(\tau) = \kappa_n \left\{ \sum_{k=1}^{M} \epsilon_k U\left(\mathbf{r}-\mathbf{r}_k\right) s_k(t) + b(\mathbf{r}) + n(\mathbf{r},t) \right\}(\tau) \tag{2}$$

Using the fundamental properties of cumulants like additivity and semi-invariance, the nth order cumulant with zero time lag can be written as

$$\kappa_n\{I(\mathbf{r},t)\} = \sum_{k=1}^{N} \epsilon_k^n U^n\left(\mathbf{r}-\mathbf{r}_k\right) \kappa_n\left\{s_k(t)\right\} + \kappa_n\{b(\mathbf{r})\} + \kappa_n\{n(\mathbf{r},t)\} \tag{3}$$

For (n ⩾ 2), under the assumption of uncorrelated noise and stationary background, the terms b(r) and n(r, t) will cancel out. For an nth order cumulant, the PSF is raised to the nth power. In consequence, the spatial resolution is improved by a factor of $\sqrt{n}$. Therefore, increasing the cumulant order yields an image with an enhanced spatial resolution. Since a multiplication in the spatial domain corresponds to a convolution in the frequency domain, the cut-off frequency of Un (r) is in principle n-times higher than that of U (r). Consequently,by applying deconvolution and a subsequent rescaling, the nth order cumulant image exhibits up to an n-fold resolution improvement.

SOFI usually assumes STORM like blinking model of the fluorophores i.e. the fluorophores reversibly switch between a bright and a dark state. In Deschout and Lukeš et al., it was shown that SOFI can be applied also to PALM data. In the PALM photo-physical model, the emitter activation is assumed as non-reversible, however, since the emitter is activated, it exhibits several quick blinking events prior to be finally bleached. On a shorter time scale (within one subsequence of input dataset), the emitter fluctuates. If the emitter fluctuates between two different states (an on-state S_{on} and a dark state S_{off}), we can define the on-time ratio as

$$\rho = \frac{\tau_{\text{on}}}{\tau_{\text{on}} + \tau_{\text{off}}} \tag{4}$$

where τ_{on} and τ_{off} are the characteristic lifetimes of S_{on} and S_{off} states. The nth order cumulant $\kappa_n\{s_k(t)\}$ is in this model described by a Bernoulli distribution with probability ρ_{on} and approximated by an nth order polynomial function of the on-time ratio (further referred to as a cumulant function)

$$f_n(\rho_{\text{on}}) = \rho_{\text{on}}(1 - \rho_{\text{on}}) \frac{\partial f_{n-1}}{\partial \rho_{\text{on}}} \tag{5}$$

Under these conditions, the nth order spatio-temporal cross-cumulant can be approximated as:

$$\kappa_n\{I(\mathbf{r}, \mathbf{t})\} \approx \epsilon^n f_n(\rho_{\text{on}}) \sum_{k=1}^{N} U^n(\mathbf{r} - \mathbf{r}_k) \tag{6}$$

As a characteristic of SOFI, any stationary background is strongly suppressed, uncorrelated noise is reduced and signals of several blinking emitters can be discriminated beyond the diffraction limit.

This section is adapted from the thesis of Tomas Lukes, please check it or the papers by Dertinger et al. (PNAS 2009) and Geissbuehler et al. (Optical Nanoscopy, 2012) for more information.

5 *Overview of SOFI Processing*

The SOFI technique is based on spatiotemporal cross-cumulant analysis and leads to a super-resolved image by processing hundreds or thousands of frames. There are several flavors of SOFI, e.g. 2D, 3D and multicolor SOFI, which will be outlined in the following sections.

5.1 *2D SOFI Workflow*

The nonlinear response to molecular brightness (ϵ^n in Eq. 6) represents a limitation in practice for 3rd and higher orders where most of the structural details are hidden in the background due to a few brightest spots. Geissbuehler et al. (Optical Nanoscopy, 2012) proposed a reformulation of the original SOFI concept called balanced SOFI (bSOFI). The workflow for 2D SOFI processing is based on the bSOFI algorithm. The different steps are depicted in Figure 2.

The first step in balancing the cumulant is a deconvolution in order to correct the non-linear response to brightness without compromising the resolution improvement. Assuming a perfect deconvolution applied to the nth order cumulant in Eq. 6, the deconvolved cumulant image can be expressed as

$$\hat{\kappa}_n\{I(\mathbf{r},\mathbf{t})\} \approx \epsilon^n f_n\left(\rho_{\text{on}}\right)\sum_{k=1}^{N}\delta\left(\mathbf{r}-\mathbf{r}_k\right) \tag{7}$$

In the next step, a linearization of the brightness response is performed by taking the nth root. The result is then reconvolved with the n-times size-reduced PSF to limit the final resolution to a physically reasonable value. Figure 3b shows 2D SOFI images up to the 4th cumulant order after flattening and linearization. Cumulants are proportional to the nth order polynomial of the on-time ratio $f_n(\rho_{on})$. Close to the roots of the polynomial, the SNR of the cumulant drops. These roots in the interval $\rho_{on} = [0,1]$ are at different positions for different cumulant orders. Once the cumulants are linearized by the aforementioned procedure, it is possible to combine linearized cumulant of order n with a linearized cumulant of order n-1 using an on-time ratiomap $\rho_{on}(r)$ to restore the badly defined areas with low SNR.

For more information, check the thesis of Tomas Lukes or the paper by Geissbuehler et al. (Optical Nanoscopy, 2012).

5.2 Cross-cumulant Calculation

Cumulants are a statistical measure related to moments. Because cumulants are additive, the cumulant of a sum of independently fluctuating fluorophores corresponds to the sum of the cumulant of each individual fluorophore. This leads to a point-spread function raised to the power of the cumulant order n and therefore a resolution improvement of $\sqrt{n}$, respectively almost n with subsequent Fourier filtering. (Geissbuehler et al., Optical Nanoscopy 2012).

The process of calculating cross-cumulants is depicted in Figures 4 and 5. The cross-cumulant $\kappa_{x,n}$ of order n for a pixel group can be calculated as a sum over all partitions. Each addend consists of a prefactor depending on the number of parts in the particular partition P times a partition product, which evaluates the parts p as a temporally averaged product.

5.3 SOFI Workflow

The general processing steps for SOFI, depicted in Figure 6, are as follows:

1. Raw cross-cumulation
2. Flatten cumulants
3. Linearization
4. Balance cumulants

As a pre-processing step in 2D SOFI you may apply drift correction. As a post-processing step you may perform molecular parameter estimation. For multi-plane (3D) SOFI, there are extra steps of plane weight correction and transformation of planes to have properly aligned pairs of consecutive planes and to align the whole 3D stack (outlined in Figure 6).

5.4 3D Cross-cumulant Combinations

Figure 7 shows an example of different 3D cross-cumulant combinations.

5.5 Multicolor SOFI

To be inserted once the codes have been merged. See Spectral cross-cumulants for multicolor super-resolved SOFI imaging https://doi.org/10.1038/s41467-020-16841-1

maybe this can be filled by Kristin?

5.6 SOFI Processing Options Explained

All the processing options are set in the config file and are briefly explained in the following sections.

Input and Calibration Settings

These settings let you control which data files will be processed; typically we provide the path to the folder that contains all the measurements and specify the file type by providing the file extension. The software will select all corresponding data in the folder. It is possible to specify a region of interest that should be analyzed. The settings for calibration are also set in this section.

Tomas can fill this with the assumed input settings.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% INPUT FOLDERS AND FILENAMES %% CALIBRATION FILE %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
settings.io.imagePath = 'YourImagePathHere';
settings.io.imageName = []; % if empty, all the files in the path will be processed
settings.io.pnc = 'YourCalibrationImagePathHere'; % calibration
% input file format - the same for all image inputs
% '.tif'; '.tiff' ; '.bin' ; '.dat'
settings.io.fext = '.dat';

select = []; % '*_ND04_582_75_50ms*' if select is empty, the program will read automaticly all the measurement names in the path
                % if select is a *string* enclosed by wildcarts, only folders with names containing
                % the select will be selected

%%%%%%%%%%%%%%%%%%%%%%%%
% CALIBRATION SETTINGS %
%%%%%%%%%%%%%%%%%%%%%%%%
settings.cal.logsize = 2.1; % size of the Laplacian of Gaussian filter
settings.cal.alol = 15;     % lower limit of the segments area
settings.cal.aupl = 100;    % upper limit of the segments area
settings.cal.maxshiftx = 200; % maximum coregistration shift in pixels

settings.cal.figs = 1;      % export calibration figures {0,1}
settings.cal.roix = [];     % roi used in calibration procedure
settings.cal.roiy = [];     % roi used in calibration procedure
settings.cal.px_tol = 15;       % tolerance for tentative correspondences
settings.cal.bgth = -20;    % threshold, for backgroudn suppression
settings.cal.order = 2;     % cumulant order used for registration
settings.cal.saveCal = 1;
settings.cal.correctForPSFshape = 0; % apply an extra factor to the calibration to take into account the shape of the psf
```

Cumulant Calculation

These settings offer control on which cumulant orders are calculated, which frames are evaluated (start and stop), and the size of a subsequence used in cumulant calculation. ”subseqlength” is the subsequence size, i.e. the data is divided into chunks of subseqlength frames, processed and all subsequences are averaged. This is a popular way to suppress the influence of photobleaching on cumulant analysis (anything introducing correlations will result in SOFI signal, but not necessarily contribute to enhancing the resolution, see Smoothness correction for better SOFI imaging https://www.nature.com/articles/s41598-021-87164-4). Subseqlength should be significantly smaller (5 times) smaller than the photobleaching halftime, but needs to allow sufficient sampling of blinking statistics. The second part of this section is SOFI-flavor-dependent and allows the user to specify parameters such as the number of planes in the setup and the channel weights which are used to correct differences in intensity within the different channels.

```
%%%%%%%%%%%%%%%%%%%%%%%%
% CUMULANT CALCULATION %
%%%%%%%%%%%%%%%%%%%%%%%%
settings.sys.orders = 1:4; % sofi orders to be calculated (cumulant orders)
settings.sys.subseqlength = 400; % length of a subsequence used for cumulant calculation
settings.sys.start = 500; %[]; start from frame start; if empty, it starts from the first frame
settings.sys.end = []; % end at frame end; if empty, it reads until the last frame

%--- SOFI 3D ---%
settings.sys.subseqstep = 400; % size of a sliding window, if subseqstep = subseqlength => no overlap, if subseqstep < subseqlength => overlap of subsequences
settings.sys.nplanes = 8; % number of planes in the setup
settings.sys.nss = []; % first n subsequences to be processed, if empty = process all
settings.sys.fext = '.dat'; %settings.io.fext; % '.tif'; '.tiff' ; '.bin' ; '.dat'
% channel weights []
%settings.sys.ch_weights = [0.93, 0.88, 0.80, 1.00; 0.84, 0.81, 0.85, 0.99];
settings.sys.ch_weights = [2, 4, 6, 8; 7, 5, 3, 1];
% for more options see funcs/sofi/reorderData.m
% orange [166.6, 151.2, 139.2, 154.5;141.75, 147.2, 148.1, 181]
% red established Dec. 2016 average of 6 scans [0.88, 0.93, 0.73, 1; 0.88, 0.82, 0.80, 0.94]
% orange established Dec. 2016 average of 4 scans [0.93, 0.88, 0.80, 1.00; 0.84, 0.81, 0.85, 0.99]
% the channel weights for the prism setup should be [2, 4, 6, 8; 7, 5, 3, 1]
```

Output Settings

These settings let you control where the SOFI results will be stored; typically we use the path to the folder that contains all the measurements and append the most important processing options to the folder name. In this section you can set parameters such as the number of bits of the output images, whether to save figures or not (and in which format).

Tomas can edit this after it is implemented. After code rearrangement, there should be a more uniform naming convention for the results. For 2D processing we provide e.g. bleaching curves and molecular parameter estimate files in a seperate folder, SOFI results are directly saved and have respective file endings: _CX raw cumulants with X cumulant order, _SOFIX for standard processing or _SOFI_linX for adaptive linearization.

```
%%%%%%%%%%%%%%%%%%%
% OUTPUT SETTINGS %
%%%%%%%%%%%%%%%%%%%
settings.io.bits = 16;          % number of bits of the output tif file {8,16}
settings.io.figs = 1;           % create figures yes/no {1,0}
settings.io.figshow = 0;        % show figures yes/no {1,0}
settings.io.figsave = 1;        % save figures (which figures - please explain difference in scaling) yes/no {1,0}
settings.io.figformat = 'png';  % figure will be saved in the specified format {'fig','png','pdf'}
settings.io.matsave = 0;        % save results in matfile (can take a lot of memory on the drive)
settings.io.seq = 0; % first n 3D super-resolved images that will be saved in a sequence
settings.io.nowID = datestr(now,'yyyymmdd_HHMM');

% Output path where results will be saved
outputpath = [settings.io.imagePath];

% test outputpath
settings.io.outputpath = 'YourOutputPathHere';
```

Post-processing Settings

It is optional to deconvolve, linearize (two outputs for 2D data: standard sqrt or adaptive linearization) and reconvolve the raw cumulants. The standard deconvolution method is Lucy-Richardson deconvolution with a Gaussian psf. You can define the psf FWHM in pixels and specify the number of iterations. We usually use 5-10 iterations but this needs to be fine-tuned for each dataset and checked for artifacts and overdeconvolution. It is also possible to use an Airy-function as a psf model. Since Lucy-Richardson has a positivity constraint it is not always the optimal deconvolution algorithm. For 2D SOFI, an algorithm based on augmented lagrangian deconvolution can be used as an extra deconvolution and is better for low signal-to-noise data. This has not been published in detail, but is described in the thesis of Tomas Lukes.

```
% fwhm in [x, y, z] in pixels, [3 3 2] is default
settings.dec.fwhm = [3 3 2]; % sofi3d

% Richardson Lucy deconvolution Matlab implementation as standard option
settings.dec.iter = 10; %20; % number of iterations for Richardson-Lucy deconvolution, 5 is default
settings.dec.lin = 1; % turn the linearization step on/off (on = 1, off = 0)
settings.dec.medfilt = 1; % apply median filtering (on = 1, off = 0)
settings.dec.avgn = 5; % apply averaging before deconvolution (1 = no averaging, 2 = average two consecutive SR images etc.)
settings.dec.orders = settings.sys.orders; % orders to be deconvolved
settings.dec.nplanes = settings.sys.nplanes;
settings.dec.reconvolve = 0;
settings.dec.psfmodel = 'gaussian'; % 'gaussian', 'airy'
```

The following sections are part of the 2D SOFI config file:

I/O Processing Settings

We provide different options for bleaching correction and drift correction. Bleaching correction intends to fit the average fluorescence intensity over time using an exponential function and corrects the data to suppress the unwanted correlations from photobleaching. Obviously, drift will result in a loss of correlation. In our experience, drift correction based on cross-correlation between the different SOFI subsequences is often precise enough. If available, drift correction based on fiducial markers is more precise. You can supply e.g. such a file created in ThunderSTORM ImageJ plugin or from `LBEN_PALM` MATLAB codes. The latter options have not been used for a while. In this section, it is possible to concatenate consecutive files with the same core file name and a different numbering into one file.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%
% I/O PROCESSING SETTINGS %
%%%%%%%%%%%%%%%%%%%%%%%%%%%

%--- 2D SOFI ---%
% Bleaching correction settings
settings.io.blcor = 1;              % bleaching correction off/on {0,1}
settings.blcor.type = 'monoexp';    % {monoexp, iir}
settings.blcor.MaxCorrSamp = 5000;
settings.io.dcor = 1;% turn on or off drift correction, if on
% - specify path to drift correction file if drift correction method is other than SOFI
settings.dcor.type = 'SOFI'; % drift correction method {'TS','LBEN_PALM','SOFI'}
% We assume the drift correction file to be : [settings.io.imageFile,settings.dcor.tag]
settings.dcor.tag = '_drift';%'driftcor'; % additional tag for drif corr. file {_drift_corr, drift,driftcor}
% TS: drift correction based on ThunderSTORM .json file (either from fiducial markers or cross-correlation)
% LBEN_PALM: molecule localisations from PALM setup
% SOFI: first cross-correlation between 100 frames subsequences and first 100 frames, performed for mean of stack
% then followed by cross-correlation between first and other SOFI2 subsequences
settings.io.ro = 0; % reorder Nikon data
%----------------%

settings.io.concatOn = 0;   % concatenate consecutive images no/yes {0,1}
settings.io.concatSave = 0; % save concatenate images in one file no/yes {0,1}
```

Molecular Parameter Estimation

Using three different SOFI orders and/or estimation of the on-time the molecular density, the on-time ratio of the fluorophore blinking kinetics and the molecular brightness is possible. For more informaton, check Mapping molecular statistics with balanced super-resolution optical fluctuation imaging (bSOFI) 10.1186/2192-2853-1-4 or Complementarity of PALM and SOFI for super-resolution live-cell imaging of focal adhesions https://doi.org/10.1038/ncomms13693.

```
% Molecular parameters
settings.molpar.run = 1; % turn on or off
settings.molpar.thresh = 0.05; % bacground suppression threshold
```

On-time Estimation

Using a second-order cumulant (same as cross-correlation) analysis with varying lag-time, the blinking can be analyzed and under certain conditions the on-time ratio can be easily determined using an exponential fit - this is basically FCS. For more information, check Mapping molecular statistics with balanced super-resolution optical fluctuation imaging (bSOFI) 10.1186/2192-2853-1-4.

```
% Estimate Ton
settings.ton.run = 1; % turn on or off
settings.ton.subseqlength = 500;
settings.ton.numtau = 20;
```

FRC Resolution Estimation

SOFI specific implementation of Fourier-Ring correlation to estimate the image resolution, see Complementarity of PALM and SOFI for super-resolution live-cell imaging of focal adhesions https://doi.org/10.1038/ncomms13693.

```
% FRC calculation
settings.frc.run = 0; % turn on or off
settings.frc.orders = 1:3;
settings.frc.bcgsub = 1;
settings.frc.pixelsize = settings.sys.pxy; % projected pixel size (in xy) [nm]
```

Jacknife SNR Estimation

This enables estimation of the signal-to-noise ratio using Jacknife resampling and is SOFI specific. The algorithm is computationally expensive. More information can be found in Complementarity of PALM and SOFI for super-resolution live-cell imaging of focal adhesions https://doi.org/10.1038/ncomms13693 and Model-free uncertainty estimation in stochastical optical fluctuation imaging (SOFI) leads to a doubled temporal resolution doi:10.1364/BOE.7.000467.

```
% Jacknife SNR estimation
settings.sys.jk = 0; % turn on/off the Jacknife SNR estimation
settings.jk.orders = 1:2;
settings.sys.block = 1;
```

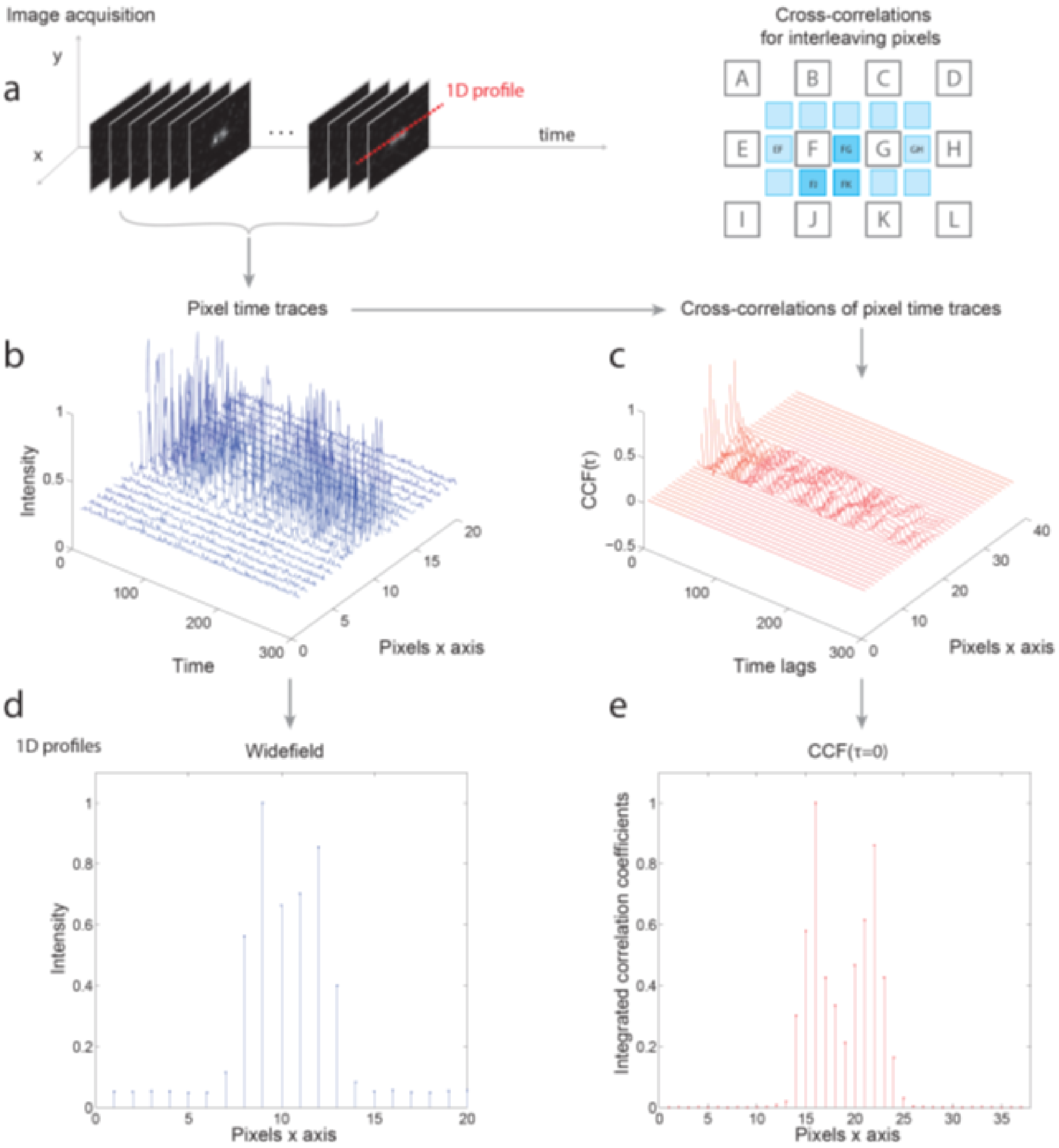


Figure 1: The SOFI principle in a one dimensional example. (a) An image sequence of two blinking emitters. (b) Intensity time traces of a line profile shown in (a). (c) 2nd order cross-cumulants calculated from the intensity time traces (b) for all time lags. Using crosscumulants, the interleaving pixels are also calculated. Note that the 2nd order cross-cumulant is mathematically equivalent to the 2nd cross-correlation. (d) The temporal average of (b) i.e.the widefield image. (e) The 2nd order cross-cumulants for □=0.

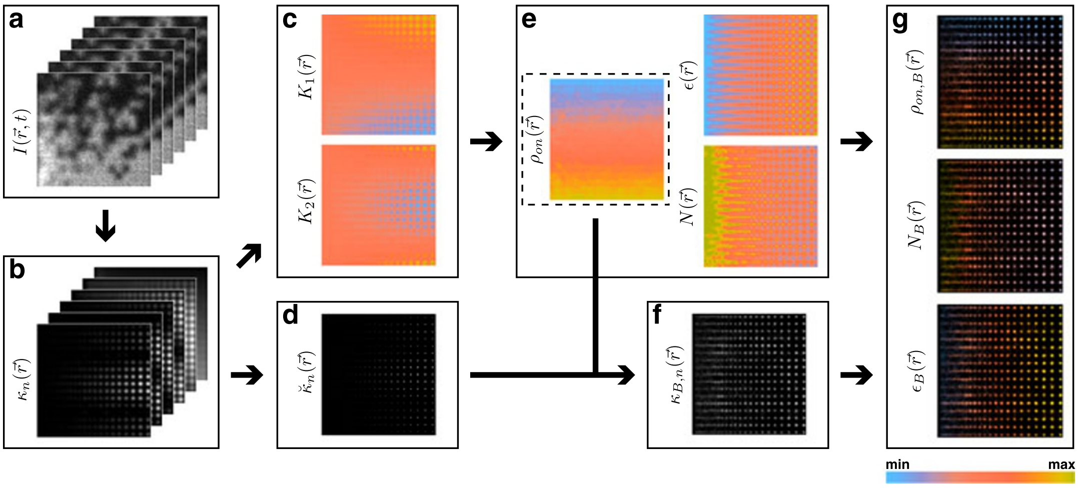


Figure 2: Flowchart to illustrate the different steps of the bSOFI algorithm for 2D SOFI analysis, from Mapping molecular statistics with balanced super-resolution optical fluctuation imaging (bSOFI) 10.1186/2192-2853-1-4. (a) Raw data. (b) Cross-cumulant computation up to order n without time lags. (c) Cumulant ratios K1 and K2 . (d) Deconvolved cumulant of order n. (e) Solution for the spatial distribution of the molecular brightness ϵ, on-time ratio ρ_{on} and density N. (f) Balanced cumulant of order n. (g) Color-coded molecular parameter maps overlaid with a balanced cumulant as a transparency mask.

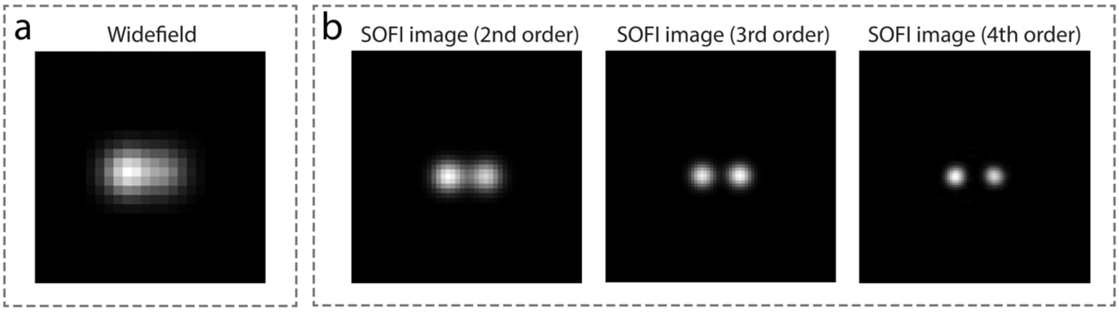
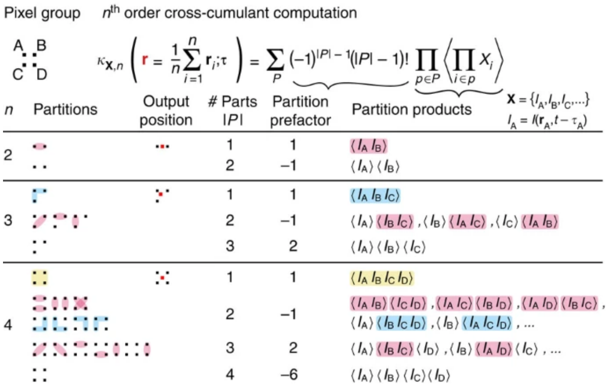


Figure 3: SOFI example. (a) Widefield image (temporal average of the input image sequence shown in Fig. 1a. (b) Linearized 2D SOFI images up to the 4th order cumulant.



Pixel group n^{th} order cross-cumulant computation

A B
C D

$$\kappa_{\mathbf{X},n}\left(\mathbf{r}=\frac{1}{n}\sum_{i=1}^{n}\mathbf{r}_i;\tau\right)=\sum_P\underbrace{(-1)^{|P|-1}(|P|-1)!}\underbrace{\prod_{p\in P}\left\langle\prod_{i\in p}X_i\right\rangle}$$

$\mathbf{X}=\{I_A,I_B,I_C,...\}$
$I_A=I(\mathbf{r}_A,t-\tau_A)$

n	Partitions	Output position	# Parts $\|P\|$	Partition prefactor	Partition products
2			1	1	$\langle I_A I_B\rangle$
			2	−1	$\langle I_A\rangle\langle I_B\rangle$
3			1	1	$\langle I_A I_B I_C\rangle$
			2	−1	$\langle I_A\rangle\langle I_B I_C\rangle$, $\langle I_B\rangle\langle I_A I_C\rangle$, $\langle I_C\rangle\langle I_A I_B\rangle$
			3	2	$\langle I_A\rangle\langle I_B\rangle\langle I_C\rangle$
4			1	1	$\langle I_A I_B I_C I_D\rangle$
			2	−1	$\langle I_A I_B\rangle\langle I_C I_D\rangle$, $\langle I_A I_C\rangle\langle I_B I_D\rangle$, $\langle I_A I_D\rangle\langle I_B I_C\rangle$, $\langle I_A\rangle\langle I_B I_C I_D\rangle$, $\langle I_B\rangle\langle I_A I_C I_D\rangle$, ...
			3	2	$\langle I_A\rangle\langle I_B I_C\rangle\langle I_D\rangle$, $\langle I_B\rangle\langle I_A I_D\rangle\langle I_C\rangle$, ...
			4	−6	$\langle I_A\rangle\langle I_B\rangle\langle I_C\rangle\langle I_D\rangle$

Figure 4: Cross-cumulant calculation, from Live-cell multiplane three-dimensional super-resolution optical fluctuation imaging 10.1038/ncomms6830

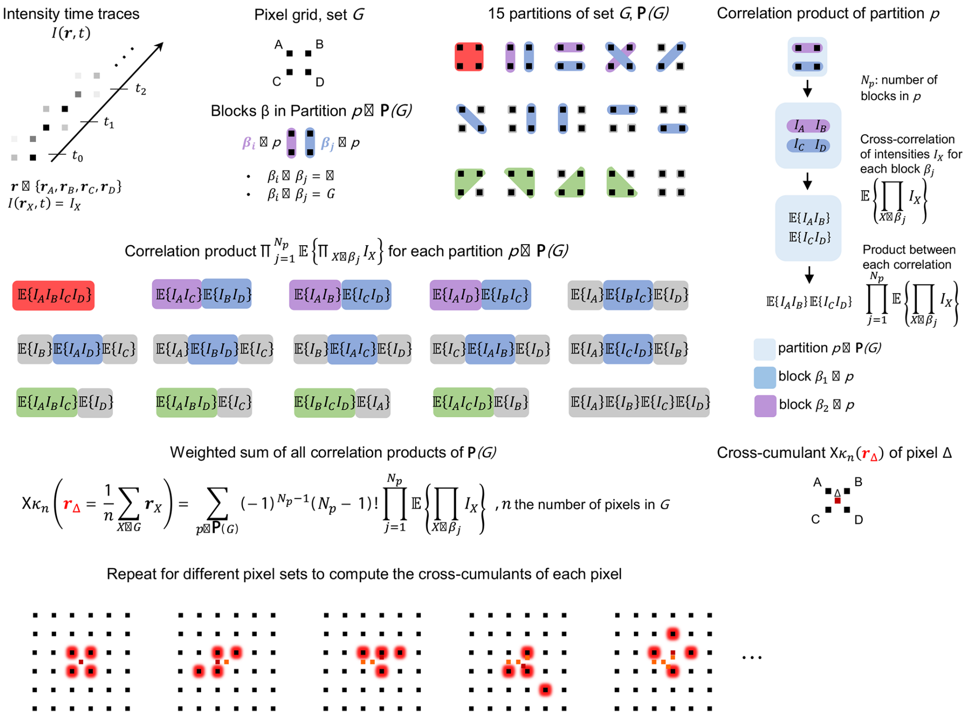

Intensity time traces
$I(\boldsymbol{r},t)$
t_2
t_1
t_0
$\boldsymbol{r}$ ⊠ $\{\boldsymbol{r}_A,\boldsymbol{r}_B,\boldsymbol{r}_C,\boldsymbol{r}_D\}$
$I(\boldsymbol{r}_X,t) = I_X$
Pixel grid, set G
A
B
C
D
Blocks β in Partition p⊠ P(G)
β_i ⊠ p
β_j ⊠ p
β_i ⊠ β_j = ⊠
β_i ⊠ β_j = G
15 partitions of set G, P(G)
Correlation product of partition p
N_p: number of blocks in p
I_A I_B
I_C I_D
Cross-correlation of intensities I_X for each block β_j
$\mathbb{E}\left\{\prod_{X⊠\beta_j} I_X\right\}$
$\mathbb{E}\{I_AI_B\}$
$\mathbb{E}\{I_CI_D\}$
Product between each correlation
$\mathbb{E}\{I_AI_B\}\mathbb{E}\{I_CI_D\}$
$\prod_{j=1}^{N_p}\mathbb{E}\left\{\prod_{X⊠\beta_j} I_X\right\}$
Correlation product $\Pi_{j=1}^{N_p}\mathbb{E}\left\{\Pi_{X⊠\beta_j} I_X\right\}$ for each partition p⊠ P(G)
$\mathbb{E}\{I_AI_BI_CI_D\}$
$\mathbb{E}\{I_AI_C\}\mathbb{E}\{I_BI_D\}$
$\mathbb{E}\{I_AI_B\}\mathbb{E}\{I_CI_D\}$
$\mathbb{E}\{I_AI_D\}\mathbb{E}\{I_BI_C\}$
$\mathbb{E}\{I_A\}\mathbb{E}\{I_BI_C\}\mathbb{E}\{I_D\}$
$\mathbb{E}\{I_B\}\mathbb{E}\{I_AI_D\}\mathbb{E}\{I_C\}$
$\mathbb{E}\{I_A\}\mathbb{E}\{I_BI_D\}\mathbb{E}\{I_C\}$
$\mathbb{E}\{I_B\}\mathbb{E}\{I_AI_C\}\mathbb{E}\{I_D\}$
$\mathbb{E}\{I_C\}\mathbb{E}\{I_AI_B\}\mathbb{E}\{I_D\}$
$\mathbb{E}\{I_A\}\mathbb{E}\{I_CI_D\}\mathbb{E}\{I_B\}$
$\mathbb{E}\{I_AI_BI_C\}\mathbb{E}\{I_D\}$
$\mathbb{E}\{I_AI_BI_D\}\mathbb{E}\{I_C\}$
$\mathbb{E}\{I_BI_CI_D\}\mathbb{E}\{I_A\}$
$\mathbb{E}\{I_AI_CI_D\}\mathbb{E}\{I_B\}$
$\mathbb{E}\{I_A\}\mathbb{E}\{I_B\}\mathbb{E}\{I_C\}\mathbb{E}\{I_D\}$
partition p⊠ P(G)
block β_1 ⊠ p
block β_2 ⊠ p
Weighted sum of all correlation products of P(G)
$X\kappa_n\left(\boldsymbol{r}_\Delta = \frac{1}{n}\sum_{X⊠G}\boldsymbol{r}_X\right) = \sum_{p⊠\mathbf{P}(G)}(-1)^{N_p-1}(N_p-1)!\prod_{j=1}^{N_p}\mathbb{E}\left\{\prod_{X⊠\beta_j} I_X\right\}$, n the number of pixels in G
Cross-cumulant $X\kappa_n(\boldsymbol{r}_\Delta)$ of pixel Δ
A
B
Δ
C
D
Repeat for different pixel sets to compute the cross-cumulants of each pixel
…

Figure 5: Cross-cumulant calculation, from SOFI simulation tool: A software package for simulating and testing super-resolution optical fluctuation imaging 10.1371/journal.pone.0161602

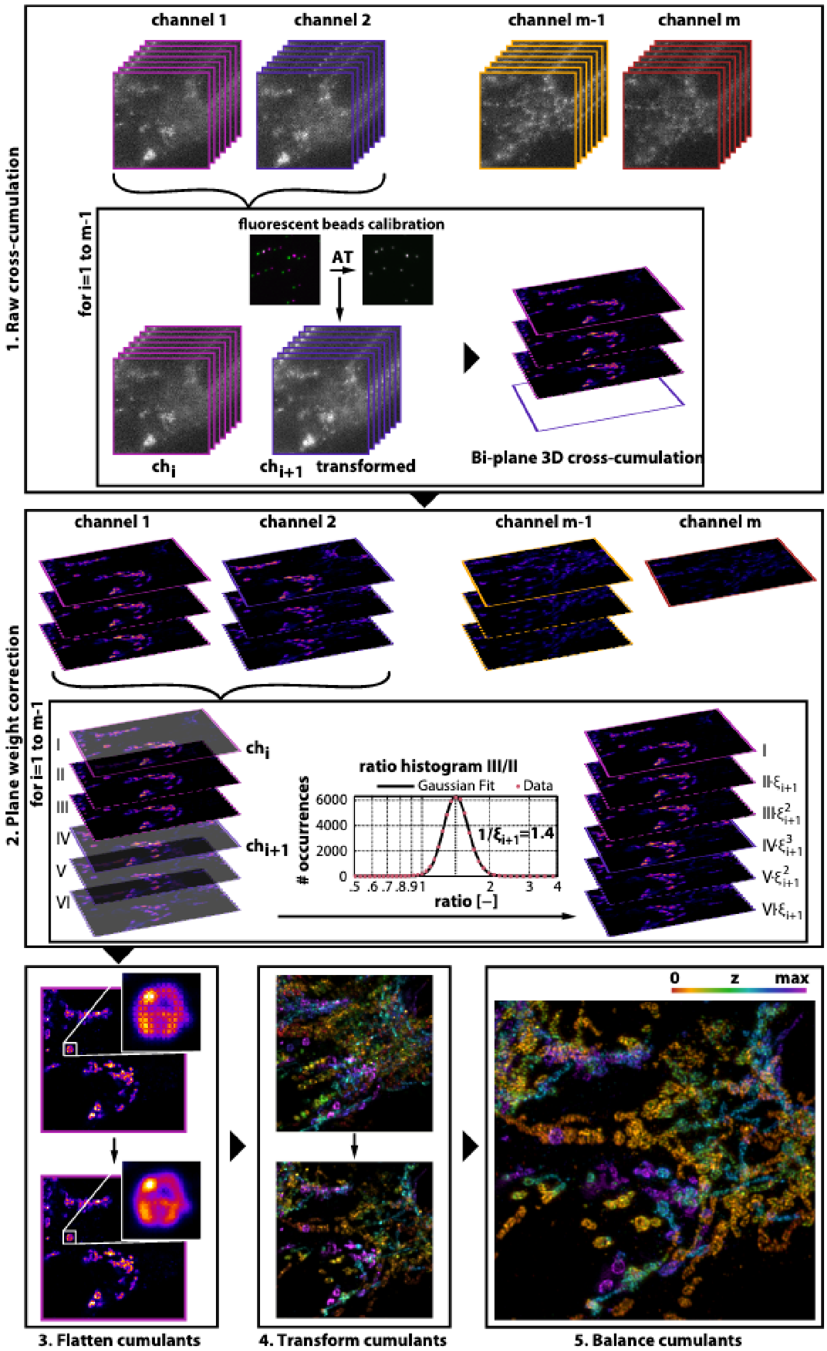


Figure 6: Multi-plane workflow, from: Live-cell multiplane three-dimensional super-resolution optical fluctuation imaging 10.1038/ncomms6830

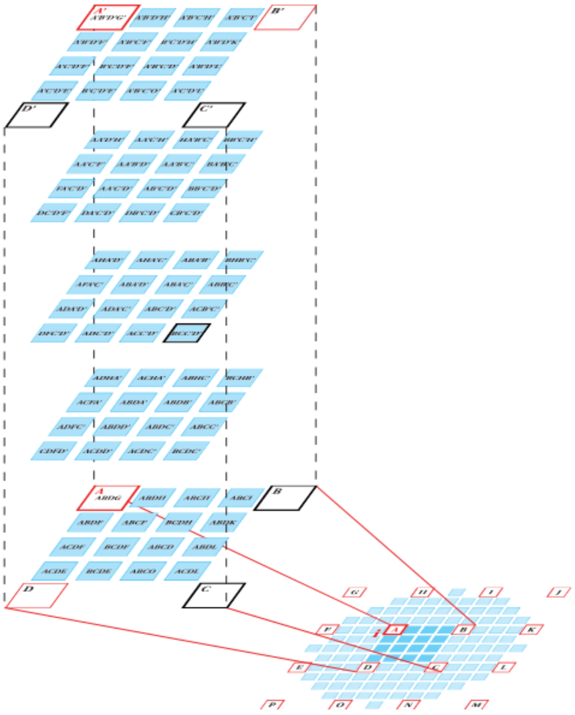


Figure 7: 3D cross-cumulant combinations, from Live-cell multiplane three-dimensional super-resolution optical fluctuation imaging 10.1038/ncomms6830