Landes Research Group
Super-resolution kinetics program

Developed equally by Jixin Chen, Lydia Kisley, Joey Tauzin, and Bo Shuang in October 2013

Purpose:

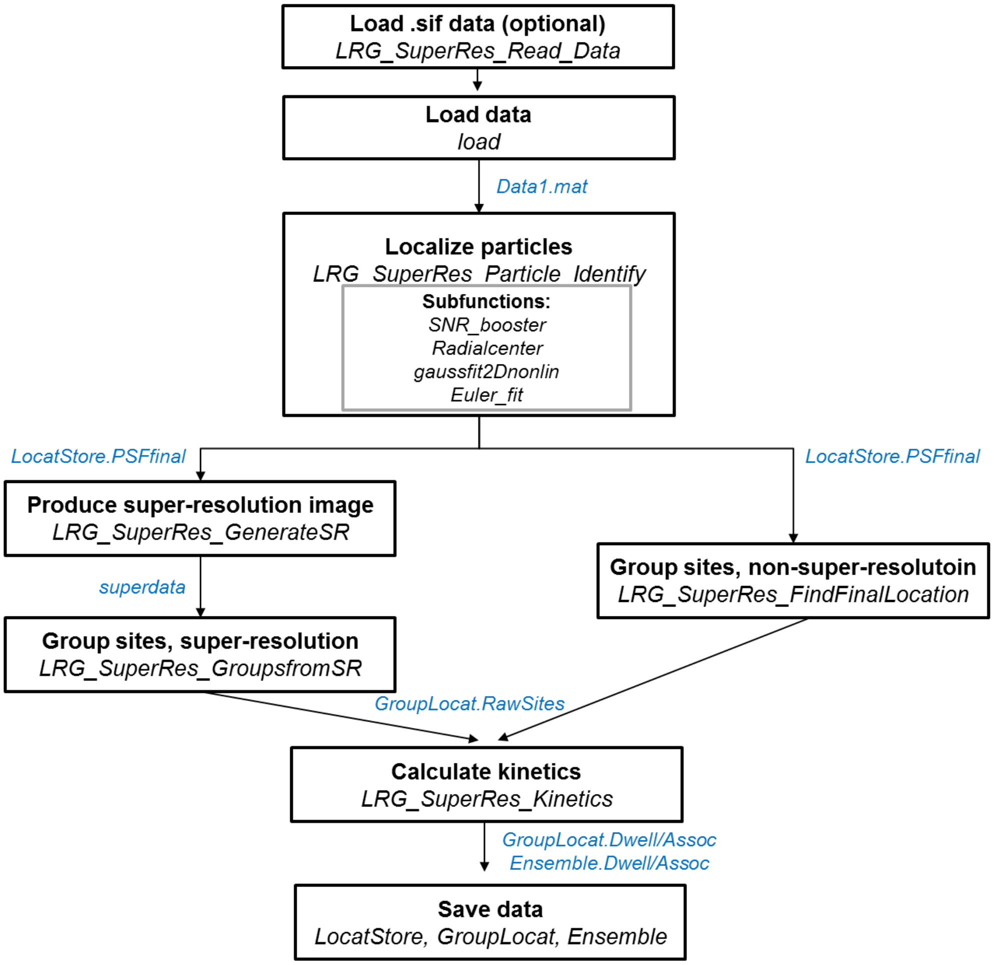
Super-resolution imaging and single molecule kinetics have the powerful potential to understand a variety of interfacial interactions. This program is a user-friendly method to do so. It is broken into three parts: 1) identify where particles are interacting at the interface, 2) group those interactions to individual sites, either by diffraction-limited or super-resolved locations, and 3) extract the kinetics from the individual sites. The results include spatial information of where molecules are interacting and how long they are there at resolutions down to ~30 nm.

History:

This program has been developed using previous naming conventions used in the SMS_VIS particle tracking and SMC molecule identification programs, such as the 'e.' structure and PSFData. Earlier versions of the super-resolution program written by Jixin Chen are in files named 'SuperPosition-Date' with functions named 'JC_function.m' and were used for the data analysis in Chen 2013 and Kisley 201X. While the naming conventions for molecule identification follows similar names to SMS_VIS/SMC, the method in this program varies drastically and is detailed in Shuang 2013.

Hierarchy:

The main function LRG_SuperRes_Main defines all of the input variables and calls LRG_SuperRes_Run that runs the program according to the following hierarchy (important output/input listed in blue):



Input variables:
The main function LRG_SuperRes_Main is where the user defines all of the input variables, as described below. Default values are for wide field imaging at 30 ms integration times, 2.5x magnification.

General Parameters
`e.codedir` – string containing the path where the program is located
`e.runsuperres(`true or false) false = find kinetics using diffraction limited data. true = find kinetics using superlocalized data
`e.startframe` – The first frame to analyze
`e.stopframe` – The last frame to analyze
`e.xmin` - ROI x minimum value (default=1)
`e.xmax` - ROI x maximum value (default=512)
`e.ymin` - ROI y minimum value (default=1)
`e.ymax` - ROI y maximum value (default=512)
`e.pixelSize` - size of 1 pixel in nm (default=64)

Data Parameters
`e.path` – string of the path where data is located
`e.filename` – string of the name of the file to read in if reading in a single file. Do not include extension.
`e.loadsif` – (true or false) true = load data from a .sif file (if this is the first analysis for instance) false = data will be loaded from a .mat file

Particle Identification Parameters
`e.SNR_booster –` (true or false) Boost the signal to noise ratio.
`e.local_thd` – (true or false) true=use local background, false=use global background
`e.wide2` – in pixels; local maximum cutoff distance (real # between 1 and 5. default = 3)
`e.Gauss_width` – in pixels; width of the PSF Gaussian. (real # between 1 and 3. default = 2)
`e.fitting` - Fitting method to use. 'rc' = radial symmetry, 'gs' = gaussian fitting, 'el' = Euler fitting
`e.test` – (true or false) Generate a figure showing identified particle locations in the first frame. Careful, this will be done for each frame.

Diffraction Limited Event Grouping Parameters
`e.sameLocation –` (in pixels) Distance threshold for grouping particles in diffraction limited data (real # default = 2)

SuperResolution Parameters
`e.nzoom` – Super resolution zoom value (for example. If e.nzoom=20, 1 pixel = 20 pixels in new super resolution image; default = 20)
`e.sigmarad` – (in nm) sigma radius used to generate super resolution data. 1 event plotted as a 2D Gaussian w/ amplitude =1 event and standard deviation of e.sigmarad. (default = 25 nm). Reduce this value if the resolution is expected to be higher for the sample. Typical values can be as small as <4 nm representing <10 nm resolution depending on the photon counts of a single particle/molecule, signal noise ratio, and sample drifting.

Super Resolution Event Grouping Parameters
`e.FinalLocatSigma` – the standard deviation of the single binding site represented by multiple binding events, or the location precision of each molecule represented by the centers in multiple frames. This value is used to generate a Gaussian standard 2D peak with this standard deviation. Then the spot on the super-resolution image is picked out and compared with the standard peak using a cross correlation function. (Default = 0.5 pixel).
`e.nevent` – the minimum peak intensity of the spot on the super-resolution image to be selected for further analysis. Any spot that has maximum intensity less than this value will not be analyzed. Thus the value distinguish the specific and nonspecific interactions. (Default = 5).

`e.SRCorrFactor` – the minimum cross correlation factor between a spot and the standard Gaussian peak. This controls the shape of the spot. Only locations with cross correlation factors of their super-resolved spots larger than this value will be analyzed. The other locations will be excluded. (Default = 0.6).
`e.FinalLocatThresh` – times of e.FinalLocatSigma is the distance to search for a location chosen with the right intensity and shape. This means that for each centroid find in each frame, the closest group location will be find and the distance will be compared to 3 times e.FinalLocatSigma. If the distance is smaller, than this centroid will be grouped in the location and marked as specific event. Otherwise the event will be marked as nonspecific that will not be further analyzed. (Default = 3).

Parameters for kinetics analysis
`e.dataSpace –` (in ms) Dead time of the detector (default = 32)
`e.dataTime –` (in ms) Integration time (default = 30)

Description of each function:
LRG_particle_identify (written by Bo Shuang)

A thorough description of identifying particles is presented in Shuang, et al. 2013. Each frame is read into the function. First, if desired, the signal to noise ratio is increased in each frame by convolution between the frame and a 3 × 3 matrix of ones.

Then, either the global or local background level is calculated. For the local background, each 50×50 pixel region is selected and the local background and local noise level are calculated, while the global is based on the entire frame. From these values, the corresponding intensity threshold is equal to the background plus 3 times standard deviation of local noise.

From the threshold, local intensity maximums are identified. Pixels larger than their neighbors (pixels within e.wide2 distance) and both the center and the neighbors must larger than the local threshold. LocatStore.PSFDataID records all the pixel positions of local maximums.

From the local intensity maximums, super-resolution position refinement is performed using the method indicated by e.fitting (‘rc’ radial symmetry (see Parthasarathy 2012); ‘gs’ gaussian fitting; ‘el’ Euler fitting (see Shuang, in preparation)). The locations and parameters are recorded in LocatStore.PSFData.

Finally, further filtering is performed based on the second momentum of the particle compared with the second momentum of the same fitting region with Gaussian noise. Only if the second momentum of the particle is smaller than the 90 % of the second momentum of Gaussian noise, we will consider this particle as a real particle and record its positions. These locations are recorded in LocatStore.PSFfilteredraw.

The final locations are then stored in LocatStore.PSFfinal. For a check of each frame, an image will be produced if e.test==1.

------------------INPUT------------------
im: a single frame
e.SNR_enhance = true || false, true if want to use SNR_booster function to increase the SNR ratio before analysis
e.local_thd = true || false, true if want to use local background and threshold
e.wide2 = 1, 1.5, 2, 2.4, 2.9, 3, ... 5, cut off distance to define a local maximum,
e.Gauss_width = 1 to 3, Gaussian width of the PSF. default is 2.
e.fitting = 'rc' || 'gs' || 'el', fitting funtion to be used, radial symmetry:'rc', Gauss fitting: 'gs', Euler fitting: 'el'. default is 'rc'
e.test = true || false, true if using test mode to generate a figure after particle identification. default is false.
------------------OUTPUT------------------
LocatStore:

.PSFDataID (n×2 matrix recording each particle’s pixel position)

.PSFData (n×6 matrix recording each particle’s refined position (y&x), second momentum (y&x), intensity, local background)

.PSFfilteredraw (the same as PSFData but with noise filtered out based on second momentum (the corresponding raw is set to 0s))

.PSFfinal (the same as PSFData but with noise filtered out based on second momentum (the corresponding raw is removed))

LRG_SuperRes_FindFinalLocation (written by Joey Tauzin)

This function first makes a list of the coordinates of all identified events and then groups them based on a distance threshold specified in e.sameLocation. This function is for use with diffraction limited data.

------------------INPUT------------------

LocatStore - The structure containing the identified event information. Specifically the field PSFfinal is used by this function

e - The structure containing the user defined parameters. Specifically this function uses e.sameLocation

------------------OUTPUT------------------

GroupLocat(1:n) - Is a structure where each element contains the fields RawSites and Centroid.

n is the number of identified unique sites. GroupLocat(1:n).RawSites has all of the events grouped to that site and is a m by 4 matrix where m is the number of events in the group and each row contains the event info formatted as [y x frame# I].

GroupLocat(1:n).Centroid contains the final locations calculated by averaging the x and y positions of all events grouped to a unique location. It is a 1 by 4 matrixx and has the format [y x stdy stdx]

LRG_SuperRes_GenerateSR (written by Jixin Chen)

Briefly, on a virtual picture with each pixel 1/e.nzoom size of original image, each event in a frame will be replaced by a Gaussian 2D spot with standard deviation e.sigmarad in both x and y axis and peak maximum 1. Then all the events in all the frames will be summed up to give a final super-resolution image.

------------------INPUT------------------

LocatStore: contain all the location information

e.nzoom: times of amplify of the image size. Suggesting value 10 or 20.

e.sigmarad: unit (nm). The size of the super-resolution Gaussian image of each event in standard deviation. FWHM about 2.36*sigma.

------------------OUTPUT------------------

superdata: a matrix with size length*e.nzoom and width*e.nzoom

LRG_SuperRes_GroupsfromSR (written by Jixin Chen)

Get the grouped final locations from the super-resolution image. Set a threshold sigma for the dispersion of events at a set location. Generate a Gaussian standard with this sigma and compare the standard to the super-resolution (SR) image. Set a correlation factor threshhold e.SRCorrFactor for the shape and a number of event threshold e.nevent.

After selection of the final locations by cross correlation, group the events with the following rule: screen the events in a single frame and find the nearest final location; if the distance is less then e.FinalLocatThresh*the sigma, group it, otherwise mark the events as nonspecific binding events. Screen events to find the closest final location. If the distance is <e.FinalLocatThresh*e.FinalLocatSigma, put the event in the group of the final location. Otherwise, mark the events as nonspecific binding by adding a column in LocatStore.

Nonspecific binding events will not show up in the grouped events based on the user-defined input e.nevent.

------------------INPUT------------------

LocatStore

superdata

e. – global variable, uses e.nzoom (default 20), e.FinalLocatSigma (default 0.5),e.nevent (default 5), e.SRCorrFactor (default 0.6) e.FinalLocatThresh (default 3)

PSFDataReduced is used. modify this as needed.

------------------OUTPUT------------------

GroupLocat(1 : number of sites), LocatStore structure GroupLocat: .Centroid(y,x, sigmay, sigmax) sigma being the dispersion of the group members

.RawSites(y,x,frame, intensity;) sort column by frame number

LocatStore column 7 is set to a flag for specific (1) and nonspecific binding (0).

LRG_SuperRes_Kinetics (written by Lydia Kisley)

Based on the individual grouped locations, the kinetics are extracted. Going through each individual adsorption site, the events are sorted in order of which they occurred. The number of frames a particle is present (GroupLocat.dwell) and the time between events (GroupLocat.assoc) are counted. Based on the time between frames the number of frames are converted to units of time (ms). These are stored for each individual location in GroupLocat and also as an ensemble across the entire sample (Ensemble.Dwell, Ensemble.Assoc).

------------------INPUT------------------
GroupLocat-locations determined by Jixin's super-resolution or Joey's FindFinalLocat
e - global variable; use e.dataSpace, e.dataTime, e.datafreq
------------------OUTPUT------------------
GroupLocat.Dwell - single location dwell times
GroupLocat.Assoc - single location association times
Ensemble.Dwell - ensemble dwell times
Ensemble.Assoc - ensemble association times

Example useful output variables (post analysis/images):

Plot all found events on top of image of data

```
imagesc(sum(Data1,3))
hold on
for i=1:size(LocatStore,2);
    plot(LocatStore(1,i).PSFfinal(:,2),LocatStore(1,i).PSFfinal(:,1))
end
```

Plot all grouped locations on top of image of data

```
imagesc(sum(Data1,3))
hold on
for i=1:size(GroupLocat,2);
    plot(GroupLocat(1,i).Centroid(1,2),GroupLocat(1,i).Centroid(1,1))
end
```

Plot all initially found events (white), then filtered events (green) on top of image of data

```
imagesc(sum(Data1,3))
hold on
for i=1:size(LocatStore,2);
    plot(LocatStore(1,i).PSFData(:,2),LocatStore(1,i).PSFData(:,1),’wx’)
end

for i=1:size(LocatStore,2);
    plot(LocatStore(1,i).PSFfinal(:,2),LocatStore(1,i).PSFfinal(:,1),’gx’)
end
```

Analyze ensemble kinetics by cumulative distribution (other LRG function) for kinetics plot

```
[dd,id]=cumuldist(Ensemble.Dwell,unique(Ensemble.Dwell));
figure
plot(dd,id)
set(gca,'YScale','log');
```

Plot all individual site association cumulative distributions

```
figure
hold on
set(gca,'YScale','log');
for i=1:size(GroupLocat,2);
```

```
    if isempty(GroupLocat(1,i).Assoc)==0
      [dd,id]=cumuldist(GroupLocat(1,i).Assoc,unique(GroupLocat(1,i).Assoc));
        plot(dd,id)
    end
end
```

Other
Example test data of 10 frames of diffusing beads is included (test.mat). This test data has very high signal to noise ratio. Please contact any of the authors for more representative test data if needed.

References:

Chen, J.; Bremauntz, A.; Kisley, L.; Shuang, B.; Landes, C. F., Super-Resolution mbPAINT for Optical Localization of Single-Stranded DNA. *ACS Appl. Mater. Interfaces* **2013,** *5*, 9338-9343.

Kisley , L.; Chen, J.; Mansur, A. P.; Shuang, B.; Kourentzi, K.; Poongavanam, M. –V.; Chen, W. –H.; Dhamane, S.; Willson, R. C.; Landes, C. F. Unified super-resolution experiments and stochastic theory provide mechanistic insight into protein ion-exchange adsorptive separations. *Under Review.*

Parthasarathy, R., Rapid, accurate particle tracking by calculation of radial symmetry centers. *Nat. Meth.* **2012,** *9*, 724-726.

Shuang, B.; Chen, J.; Kisley, L.; Landes, C. F. Troika of Single Particle Tracking Programing: SNR Enhancement, Particle Identification, and Mapping *Phys. Chem. Chem. Phys*, **Accepted.**

Shuang, B.; Chen, J.; Kisley, L.; Landes, C. F. Euler Gradient Fitting for High Reliability in High Particle-Density Images. *In Preparation.*