MATLAB Clustering Classes

September 29, 2016

Michael J. Wester (wester@math.unm.edu)
SpatioTemporal Modeling Center
University of New Mexico Health Sciences Center

MATLAB Clustering Classes

- Best with MATLAB version 2014 or later
- Toolboxes used (matlab.codetools.requiredFilesAndProducts):

  Control System

  Curve Fitting

  Image Processing

  Statistics and Machine Learning
- Common interface to sets of functions
- Scriptable
- Availability:

  http://stmc.health.unm.edu/tools-and-data/
- See also **SuperCluster.m** which operates with a GUI

Definitions

class: encapsulation of shared functionality (methods) and data for performing some process (computer code)

fluorophore: fluorescent probe activated by light emission (blinking)

localization: true location of a probe (often inferred) over a time interval (super-resolution)

observation: observed location of a probe at some instant of time (super-resolution)

MATLAB Clustering Classes

- SimulateDomains.m *[Statistics]*

  simulations of spatial domains (clusters) of fluorophore localizations that exhibit distributions of observations representing blinks

- Clustering.m *[CurveFit, ImageProcess, Statistics]*

  spatial clustering algorithms:

  hierarchical, DBSCAN [4 versions], Getis based, Voronoi based

  spatial clustering statistics:

  pairwise distance, Hopkin's, Ripley's, bivariate Ripley's, dendrogram

- SRcluster.m *[ControlSystem, CurveFit, ImageProcess, Statistics]*

  a top-down clustering algorithm to collapse clusters of observations of blinking fluorophores into a single estimate of the true location of the fluorophore using a log-likelihood hypothesis test

- PairCorr.m *[ImageProcess, Statistics]*

  pair auto- and cross-correlation curves and statistics

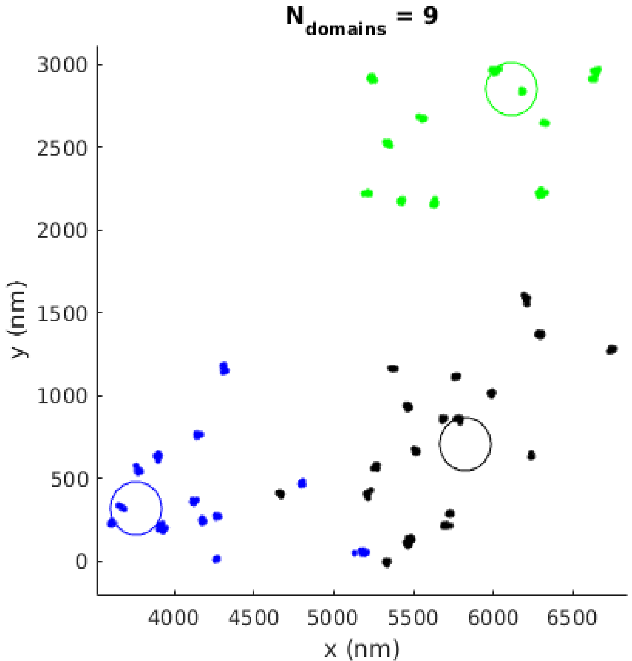
- ROITools.m

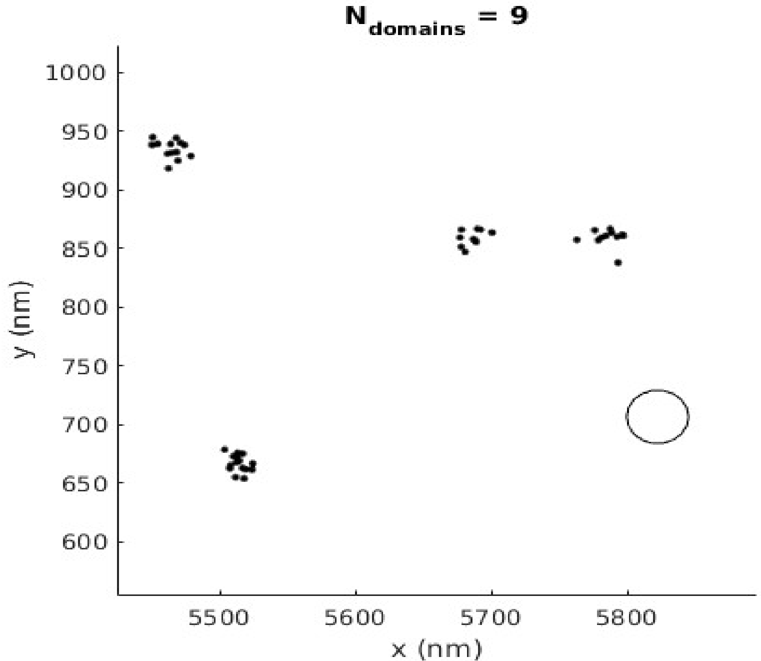
  region of interest selection tools for an arbitrary number of colors

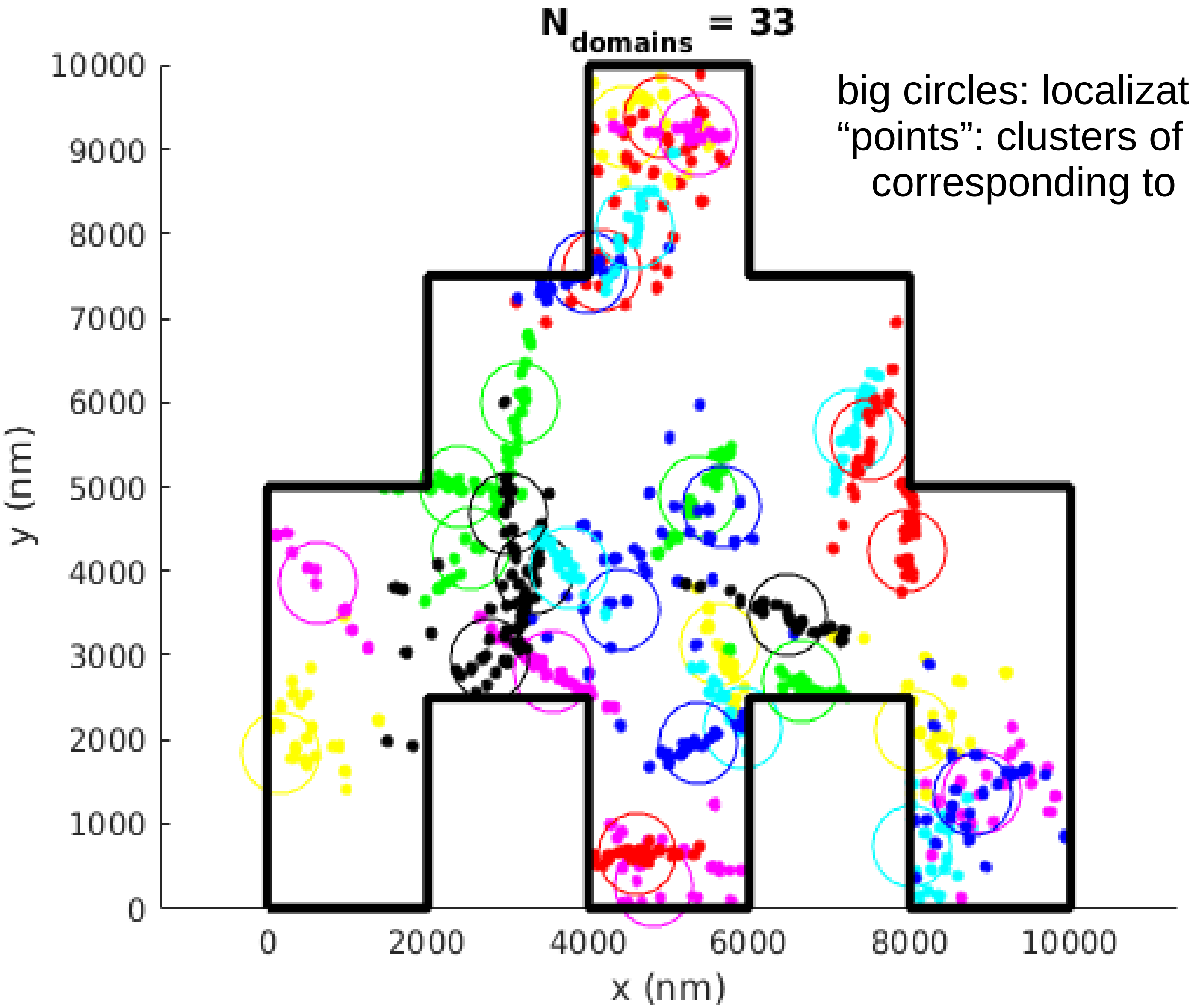
Many of these will work in 2D & 3D. Includes a collection of various sample drivers.

Domain Simulation

- clusters of localizations (Gaussian distribution)
- observations corresponding to each localization
- clusters can be elongated (σ_x ≠ σ_y) and rotated
- a mix of elongated and non-elongated clusters possible
- polygonal domain boundary can be provided
- 3D







big circles: localization cluster centers
“points”: clusters of observations
corresponding to localizations

Clustering

- Classification scheme such that objects in the same group or cluster are more similar to each other than to those outside.
- Similarity can be measured in many different ways, but a common one in biology is Euclidean distance separation.
- In this situation, the maximal distance between two points within a cluster (ε) is < the minimal distance between two points in different clusters.
- Typically, the user specifies the number of clusters or ε, although some algorithms purport to deduce the proper values.

Clustering Algorithms

- k-means (not in Clustering.m; **kmeans** in MATLAB)

  [number of clusters known]
- Hierarchical [standard]
- DBSCAN [density based]

  Daszykowski (with and w/o ε) [fast and stable]

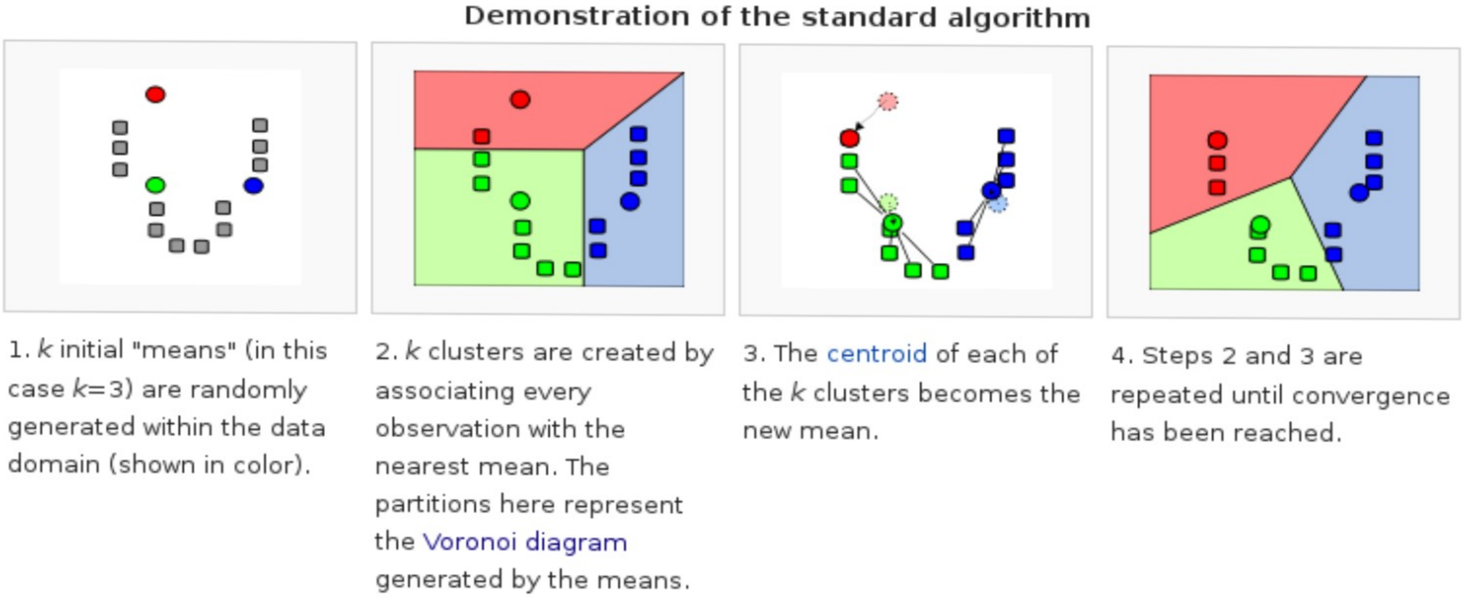
  Kovesi

  Pehlke

  Tran
- Getis based (2D only) [densely structured data]
- Voronoi based [experimental]

k-means Clustering

k-means clustering aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster. This results in a partitioning of the data space into Voronoi cells.
(https://en.wikipedia.org/wiki/K-means_clustering)



Demonstration of the standard algorithm

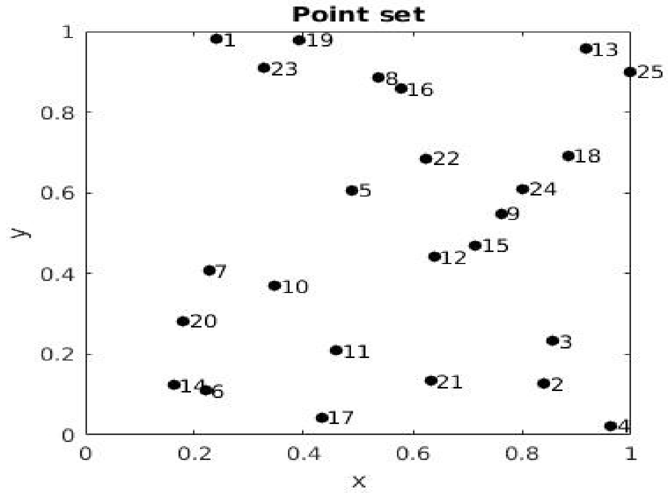
1. *k* initial "means" (in this case *k*=3) are randomly generated within the data domain (shown in color).

2. *k* clusters are created by associating every observation with the nearest mean. The partitions here represent the Voronoi diagram generated by the means.

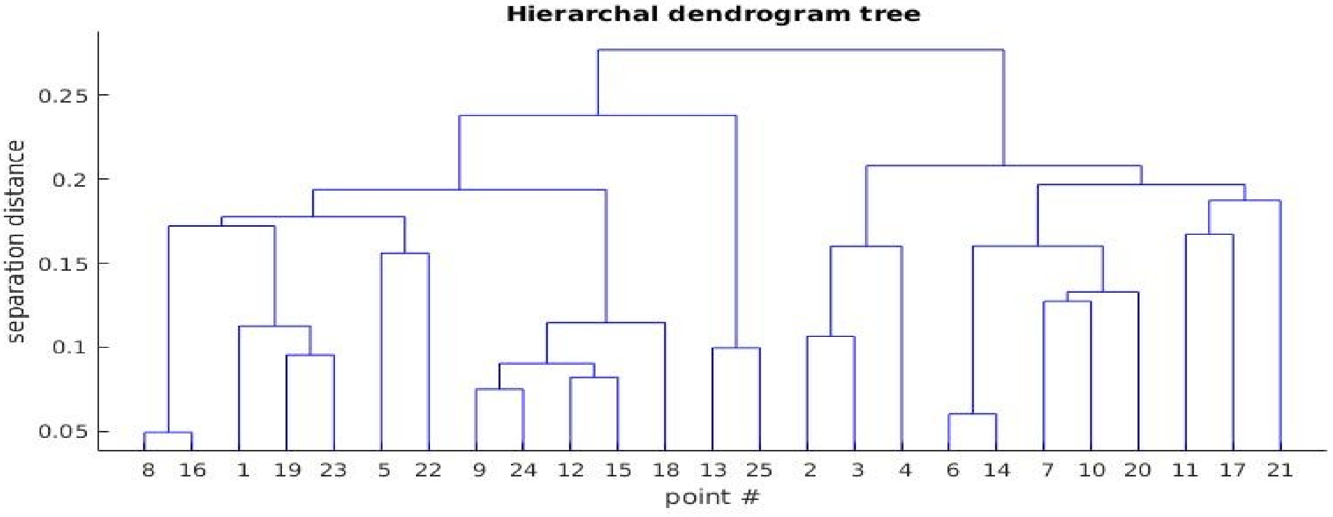
3. The centroid of each of the *k* clusters becomes the new mean.

4. Steps 2 and 3 are repeated until convergence has been reached.

Hierarchical Clustering
(MATLAB linkage function)



[stable on coordinate reordering]

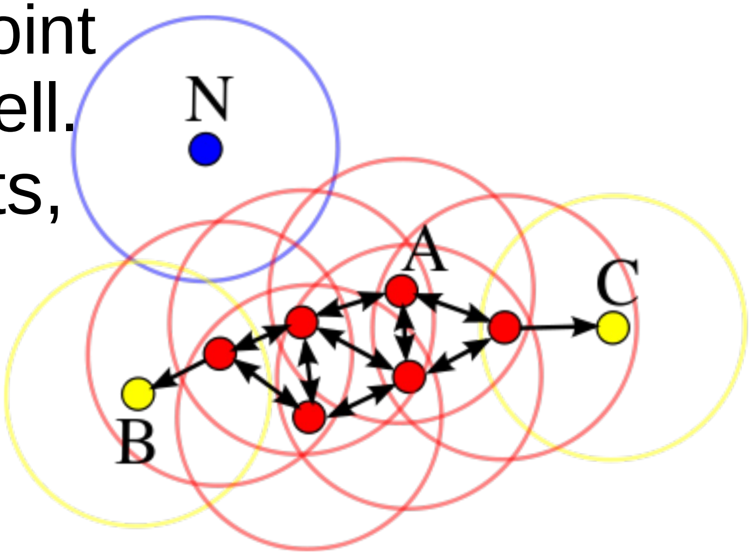


DBSCAN (Density-Based Spatial Clustering of Applications with Noise)

A cluster satisfies two properties (https://en.wikipedia.org/wiki/DBSCAN):

- all points within a cluster are mutually density-connected (density-reachable by a distance < ε from common intermediate points and the cluster has a sufficient number of points),
- if a point is density-reachable from any point of the cluster, it is part of the cluster as well.

Points are thus designated as core points, density-reachable points and outliers.
Noise is not part of any cluster.



Martin Ester, Hans-Peter Kriegel and Jörg Sander and Xiaowei Xu, ``A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise'', in Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96) edited by Evangelos Simoudis, Jiawei Han and Usama M. Fayyad, AAAI Press, 1996, 226--231 (ISBN:1-57735-004-9, DOI:10.1.1.71.1980).

DBSCAN implementations

(adjudged by **speed** and **stability** under coordinate reordering)

M. Daszykowski, B. Walczak and D. L. Massart, ``Looking for natural patterns in data. Part 1: Density-based approach'', *Chemometrics and Intelligent Laboratory Systems*, Volume 56, Issue 2, May 2001, 83—92. **[fast and stable under coordinate reordering]**

Peter Kovesi, Centre for Exploration Targeting, The University of Western Australia, 2013.

Carolyn Pehlke, SpatioTemporal Modeling Center, University of New Mexico, 2013.

Thanh N. Tran, Klaudia Drab and Michal Daszykowski, ``Revised DBSCAN algorithm to cluster data with dense adjacent clusters'', *Chemometrics and Intelligent Laboratory Systems*, Volume 120, Issue 92, January 2013, 92—96 (DOI: 10.1016/j.chemolab.2012.11.006).

Getis based Clustering
(developed by Carolyn Pehlke)

Getis statistic:

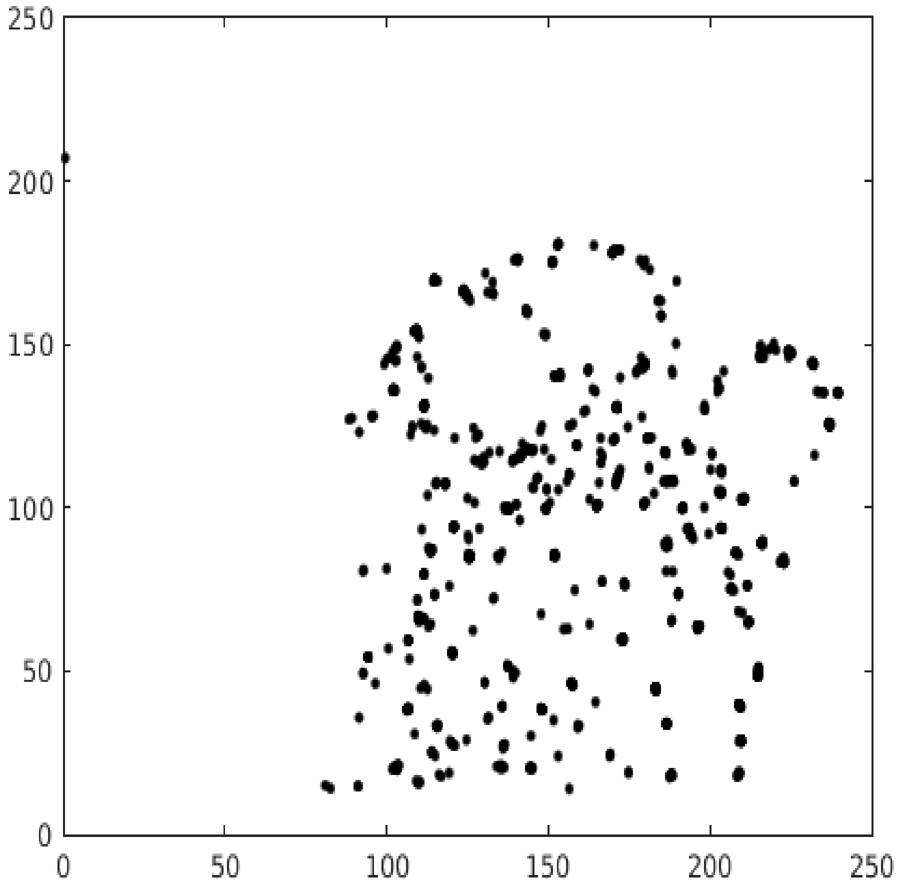
$$G_i(d) = \frac{\sum_{j \neq i} w_{ij}(d) x_j}{\sum_{j \neq i} x_j}$$

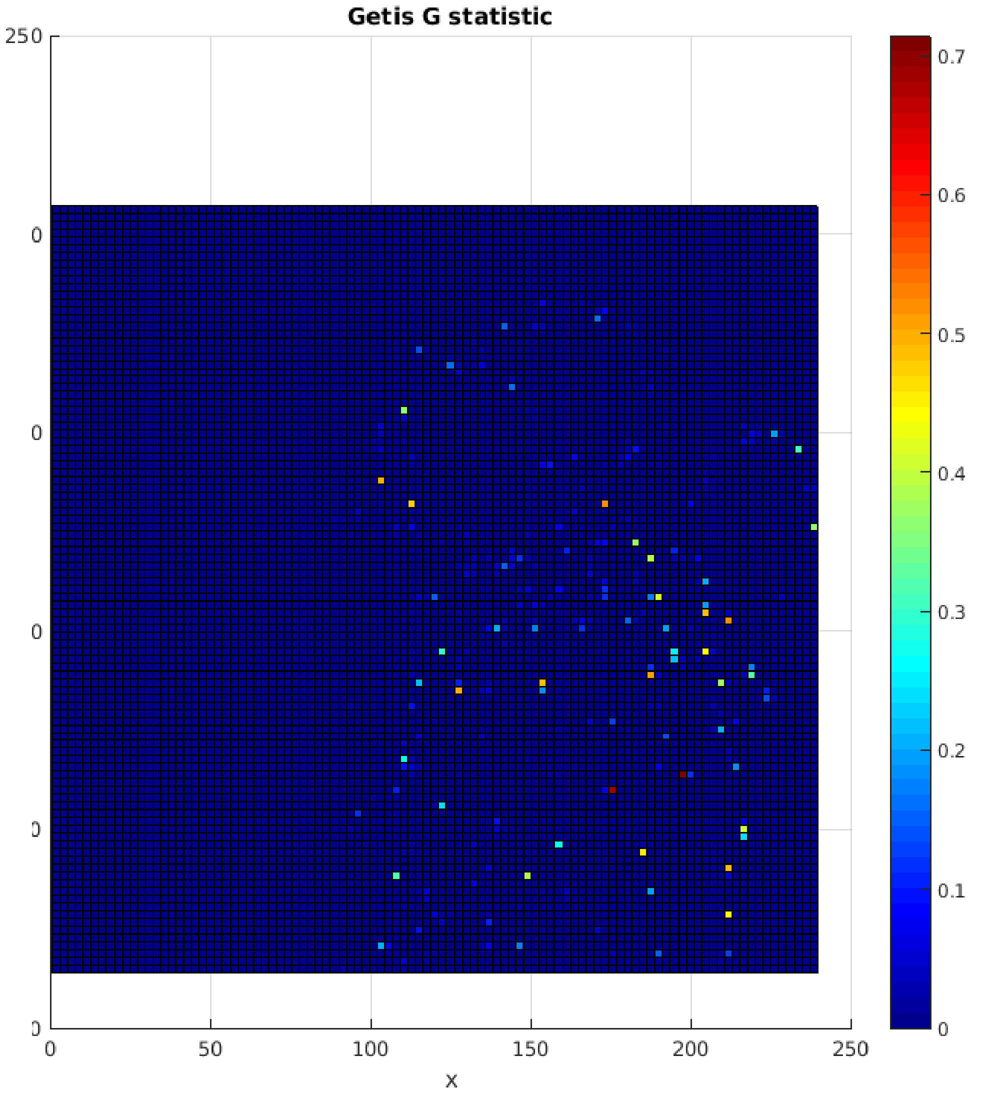
x_j is the intensity of the jth point (n points total)
w_ij(d) is a symmetric weight matrix that is a function of the distance d
that i is separated from j

J. K. Ord and Arthur Getis, ``Local Spatial Autocorrelation Statistics: Distributional Issues and an Application'', *Geographical Analysis*, Volume 27, Number 4, October 1995, 286--306.

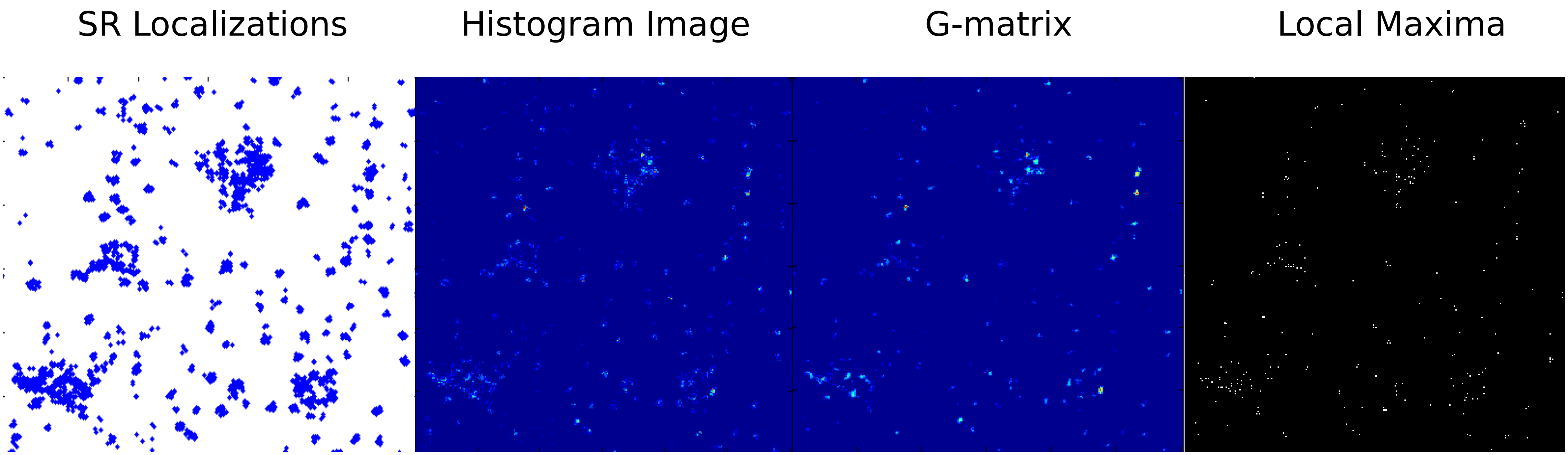
Michelle S. Itano, Matthew S. Graus, Carolyn Pehlke, Michael J. Wester, Ping Liu, Keith A. Lidke, Nancy L. Thompson, Ken Jacobson and Aaron K. Neumann, ``Super-resolution imaging of C-type lectin spatial rearrangement within the dendritic cell plasma membrane at fungal microbe contact sites'', *Frontiers in Physics, section Membrane Physiology and Membrane Biophysics*, Volume 2, Number 46, August 2014, 1—17 (DOI: 10.3389/fphy.2014.00046).

Getis Heat Map





Finding the Search Radius



SR Localizations

Histogram Image

G-matrix

Local Maxima

1. Find a maximum distance *d*, using Ripley's L-function, to estimate the radius of maximum aggregation for each ROI.

2. Construct a series of binary weight matrices based on a range of "cutoff" values between 0 and *d*.

3. Convert SR localizations to a histogram image.

4. Perform the Getis G calculation on the binarized image for each weight matrix. This results in a three dimensional matrix of G values.

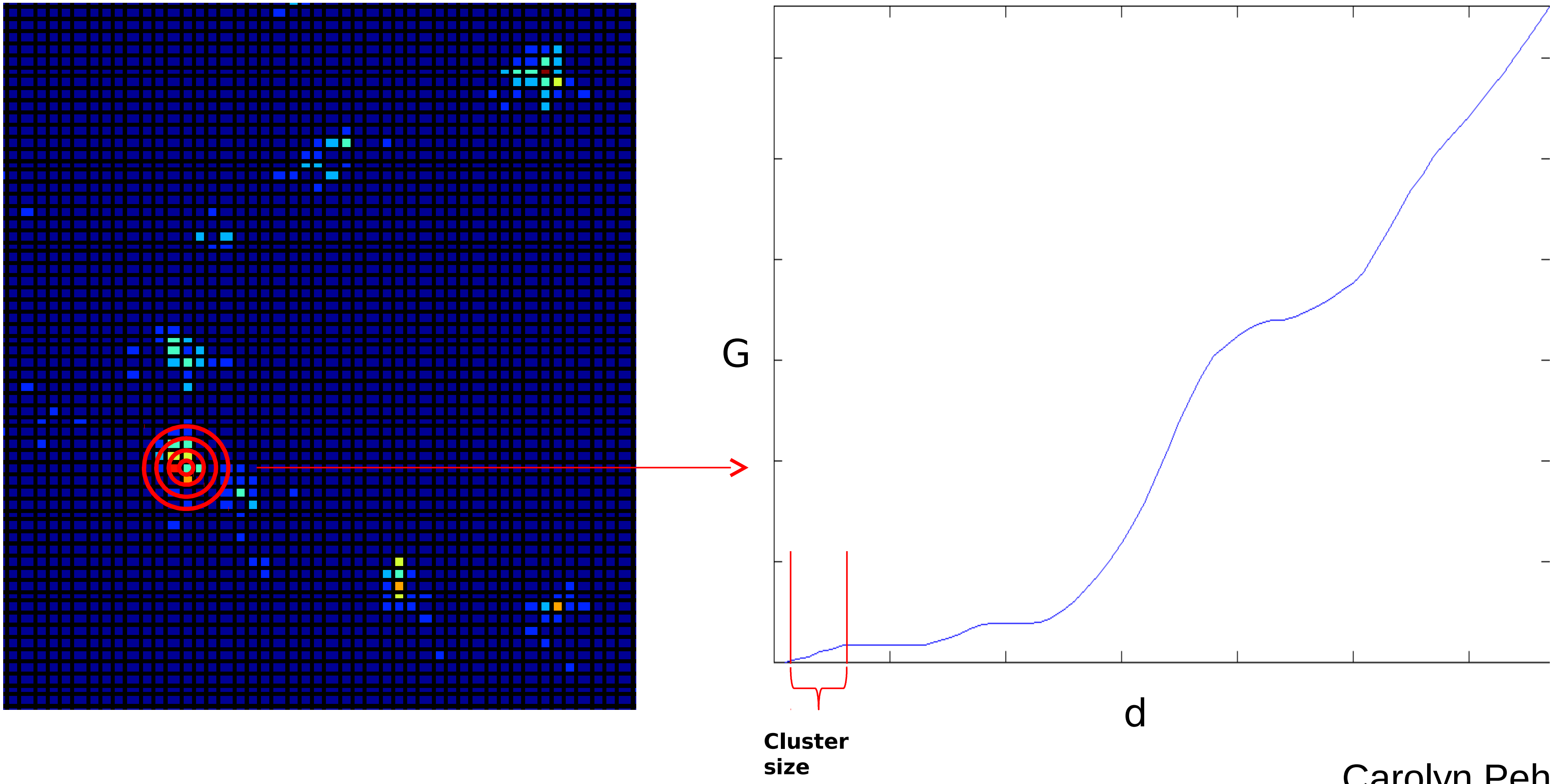
5. Create a sum projection of the 3D matrix of G values (G-matrix).

6. Find local maxima of G-matrix to use as seed points.

Carolyn Pehlke

7. For each local maxima, construct a G vs. radius curve. The first critical point indicates the approximate domain size.

8. Each local maxima is used as a seed point to build clusters. Pixels within radius r of a seed point are combined into clusters, and overlapping clusters are combined.



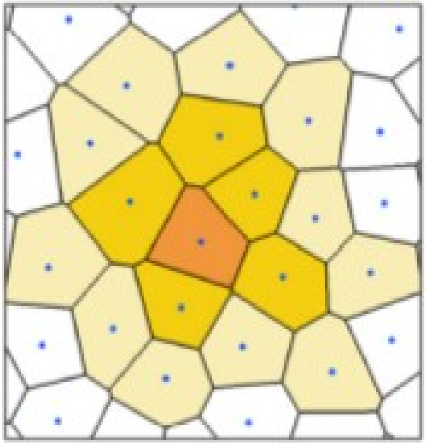
Carolyn Pehlke

Getis based Clustering

- Getis statistic produces an intensity heat map yielding seed points
- Estimate local length scale of clustered structures via the first critical point in G vs r plots for each seed point
- r_crit acts as ε for DBSCAN style clustering
- Good for densely structured data

Voronoi based Clustering

A Voronoi diagram or tesselation is a partitioning of the plane into regions, each containing one seed point, such that each boundary edge segment is equidistant from the nearest seed points.



Choose rank n polygons with density > alpha*(density_average) (alpha = 2 is a typical choice).

Florian Levet, Eric Hosy, Adel Kechkar, Corey Butler, Anne Beghin, Daniel Choquet and Jean-Baptiste Sibarita, ``SR-Tesseler: a method to segment and quantify localization-based super-resolution microscopy data'', *Nature Methods*, Volume 12, Number 11, 2015, 1065—1071 (DOI:10.1038/NMETH.3579).

Florian Levet et al.

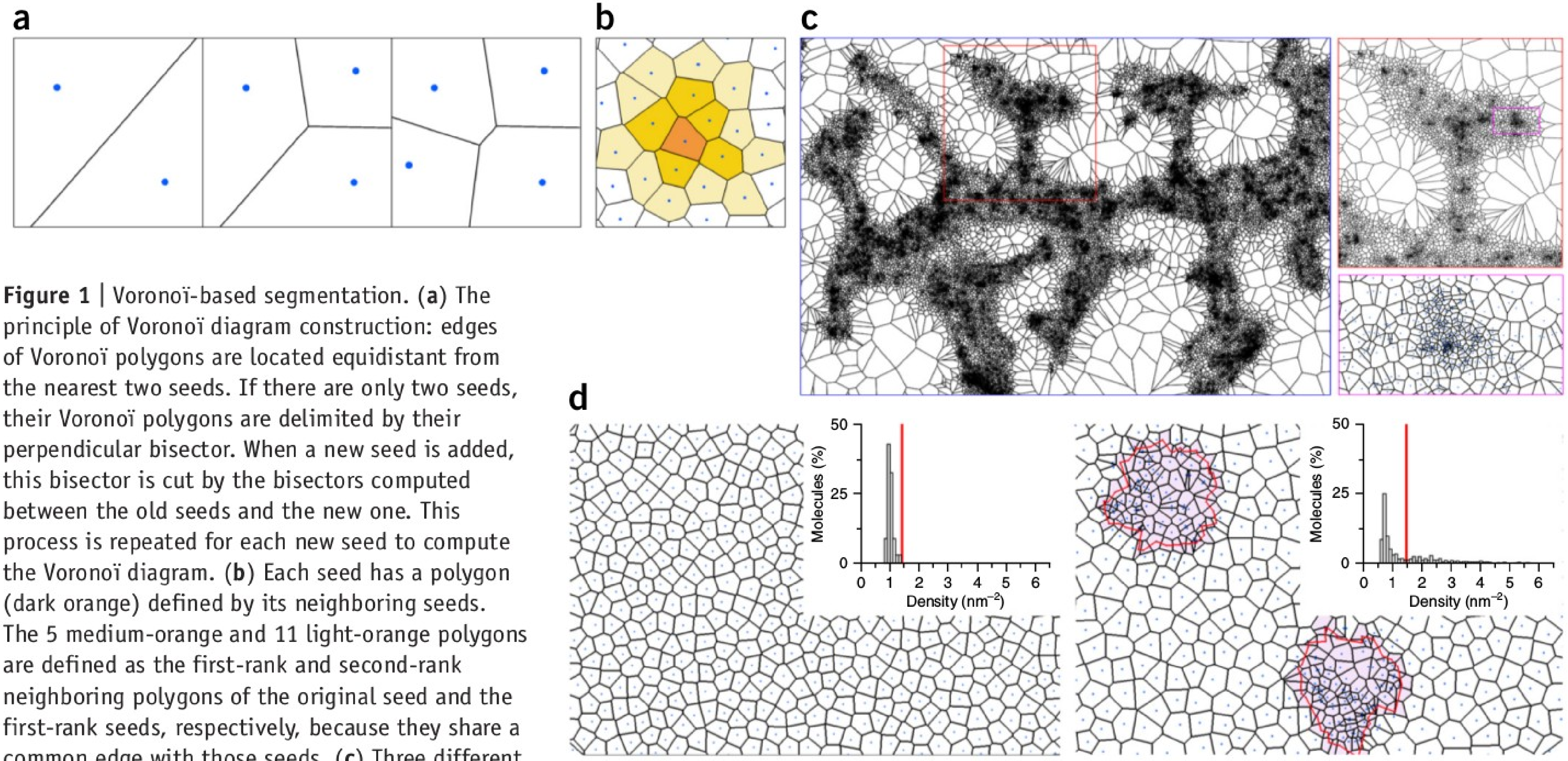
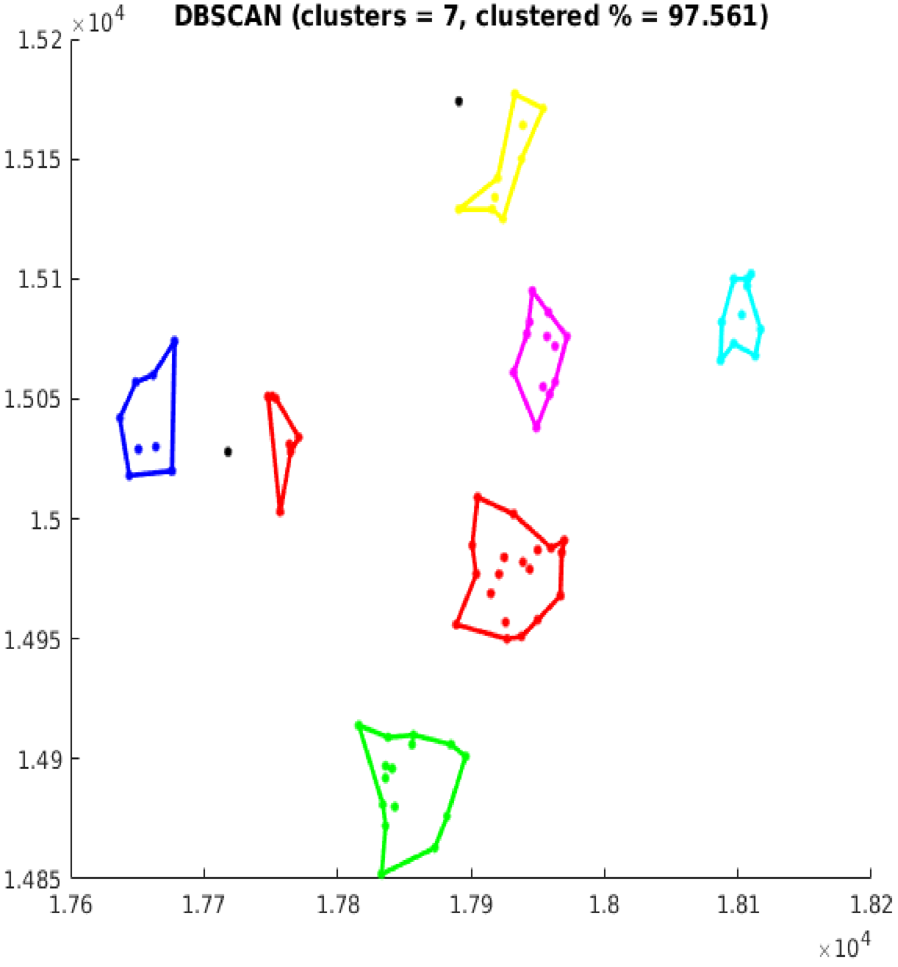
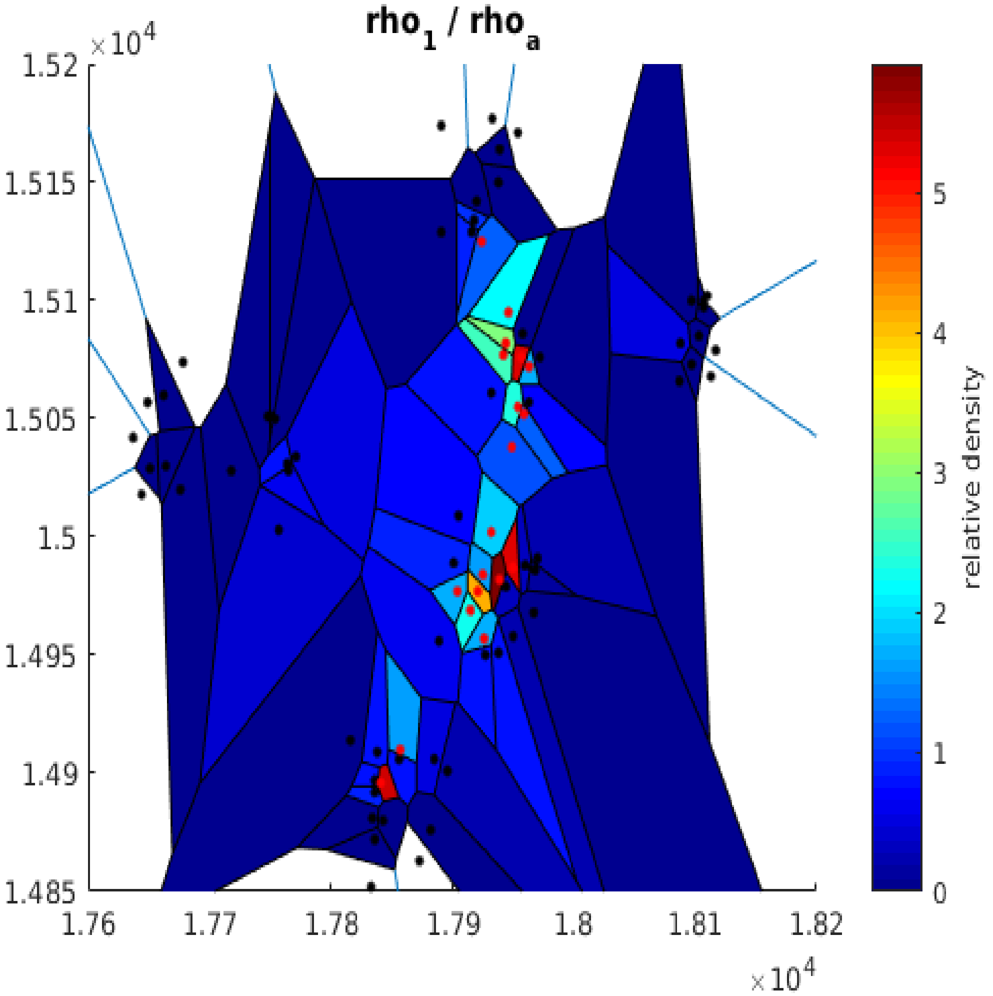


Figure 1 | Voronoï-based segmentation. (**a**) The principle of Voronoï diagram construction: edges of Voronoï polygons are located equidistant from the nearest two seeds. If there are only two seeds, their Voronoï polygons are delimited by their perpendicular bisector. When a new seed is added, this bisector is cut by the bisectors computed between the old seeds and the new one. This process is repeated for each new seed to compute the Voronoï diagram. (**b**) Each seed has a polygon (dark orange) defined by its neighboring seeds. The 5 medium-orange and 11 light-orange polygons are defined as the first-rank and second-rank neighboring polygons of the original seed and the first-rank seeds, respectively, because they share a common edge with those seeds. (**c**) Three different magnification views of a Voronoï diagram built from an experimental GluA1-mEOS2 PALM data set, showing a dendrite (blue outline), a spine (red outline) and a cluster (magenta outline). (**d**) Automatic segmentation of the Voronoï diagram on the basis of the first-rank density, with a spatially uniform distribution to which a threshold of twice the average localization density in the image was applied (red threshold in inset) (left) and a nonuniform data set analyzed with the same threshold, where selected polygons were merged (right).

DBSCAN vs Voronoi





Example Clustering Driver

```
XY = load('...');   % nm
E = 30;             % nm
minPts = 3;

c = Clustering();
algorithm = 'Hierarchal';
[nC, C, centers, ptsI] = c.cluster(algorithm, XY, E, minPts);
fprintf('number of clusters = %d\n', nC);
results = c.clusterStats(XY, C, centers)
clusterFig = c.plotClusters(XY, C, centers, ptsI, algorithm);
showm(clusterFig);
```

Hopkins' Statistic

The Hopkins' statistic (H) tests for complete spatial randomness of a probe pattern by comparing nearest neighbor distances from random points and randomly chosen probes. If the number of probes in the set S (an image) is n, choose m << n random sampling locations s_j and probes p_j, then compute

U = Σ(d^2(s_j, S), j = 1 .. m)   [random sampling locations wrt all probes]
W = Σ(d^2(p_j, S), j = 1 .. m)   [random probes wrt all probes]

where

d(p_j, S) = min{ d(p_j, p_k) for all p_k in S }

and d(p_j, p_k) = || p_j - p_k || is the distance between p_j and p_k.
The Hopkins' Statistic is defined as

H = U / (U + W)

and will lie in the interval [0, 1].

For good results, H should be computed multiple times for a single image.

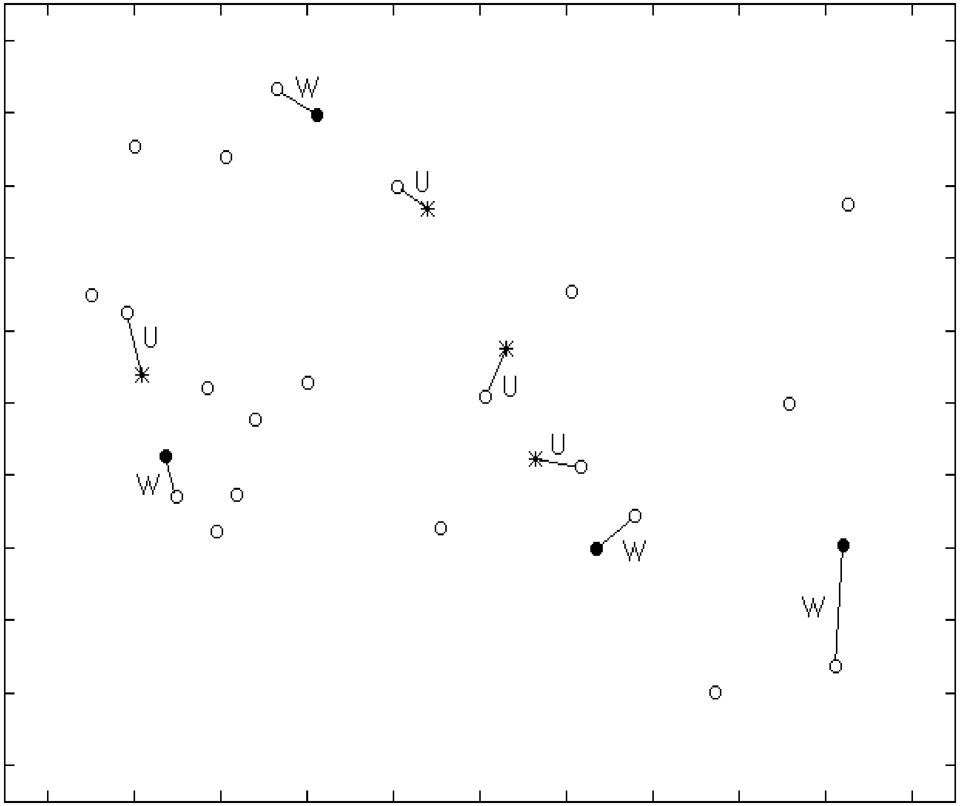
H = 0   for uniformly distributed probes
H = 1/2 for completely random probes
H = 1   for completely clustered probes

Hopkins' Statistic

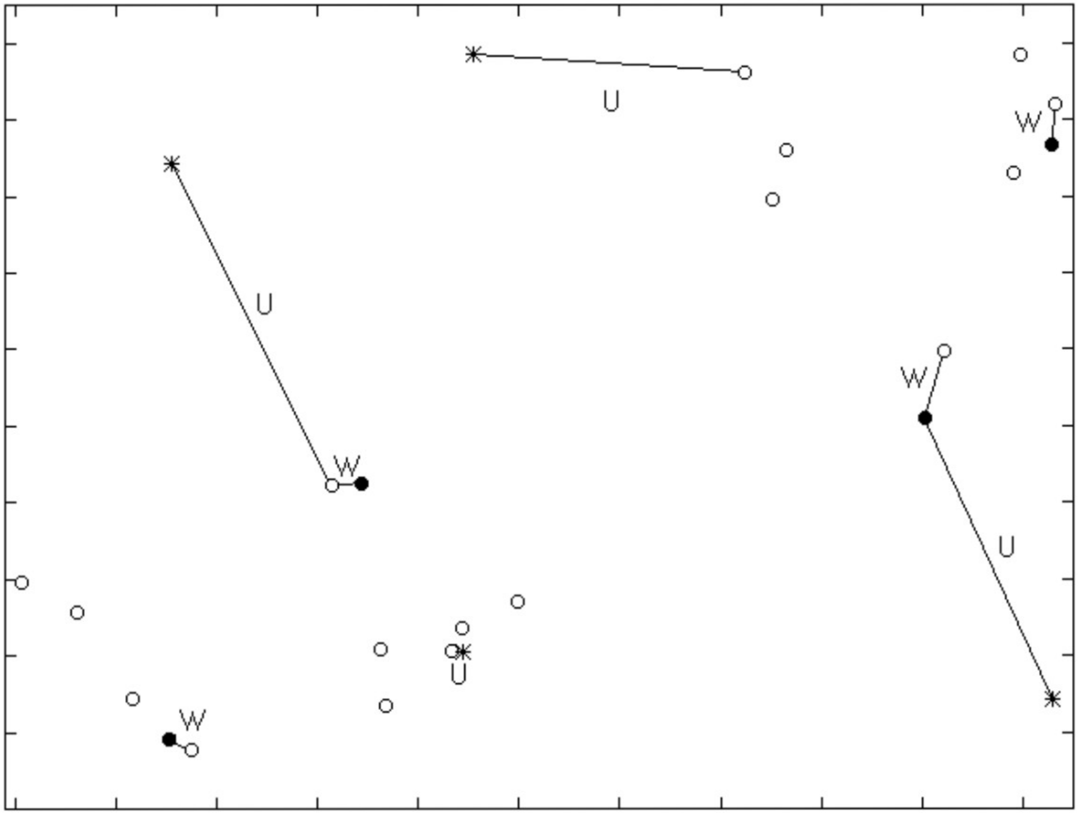
U = Σ(d^2(s_j, S), j = 1 .. m)   [random sampling locations wrt all probes]
W = Σ(d^2(p_j, S), j = 1 .. m)   [random probes wrt all probes]

H = U / (U + W)

H = 0.49



H = 0.73



http://www.vub.ac.be/fabi/multi/pcr/chaps/chap8.html

Ripley's Statistics

Ripley's K analysis tests for clustering and co-clustering by comparing the average number of probes in a disk of radius r about each of the probes with the average density of probes (λ = n/A) over the region considered.

K(r) = 1/n Σ(Σ I_ij(r), j != i), i = 1 .. n) / λ   =   A/n^2 Σ(Σ I_ij(r), j != i), i = 1 .. n)

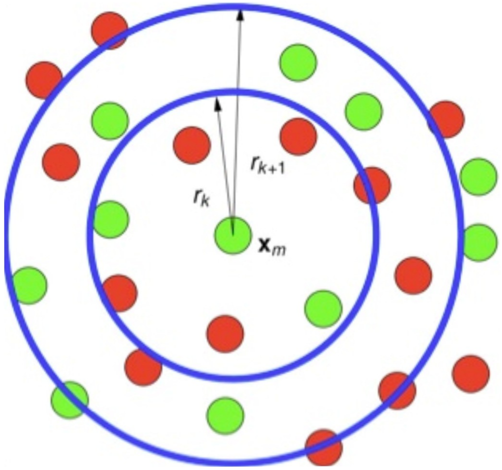
where n is the number of probes in a domain of area A, and I_ij(r) = 1 if the distance d(p_i, p_j) < r, otherwise 0. The expected value for randomly distributed probes is π r^2.

This counts the number of points encircled by concentric disks centered on each probe normalized by the average density of the region. This can be linearized as

L(r) = sqrt(K(r) / π)

so that

L(r) < r   probes are less clustered than random
L(r) = r   probes are clustered as in a random distribution
L(r) > r   probes are more clustered than random



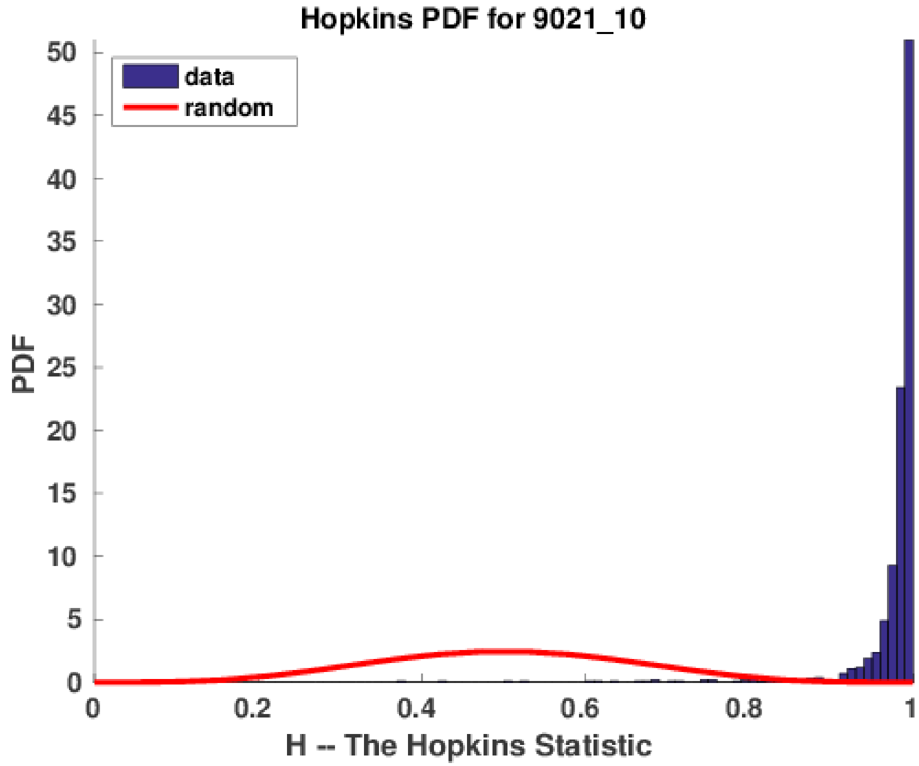
In a bivariate analysis (where two different sets of probes in the same region are considered), the L function acts as above except that now the cluster size refers to clusters of co-mingled probes from the two sets.

Hopkins' and Ripley's Statistics

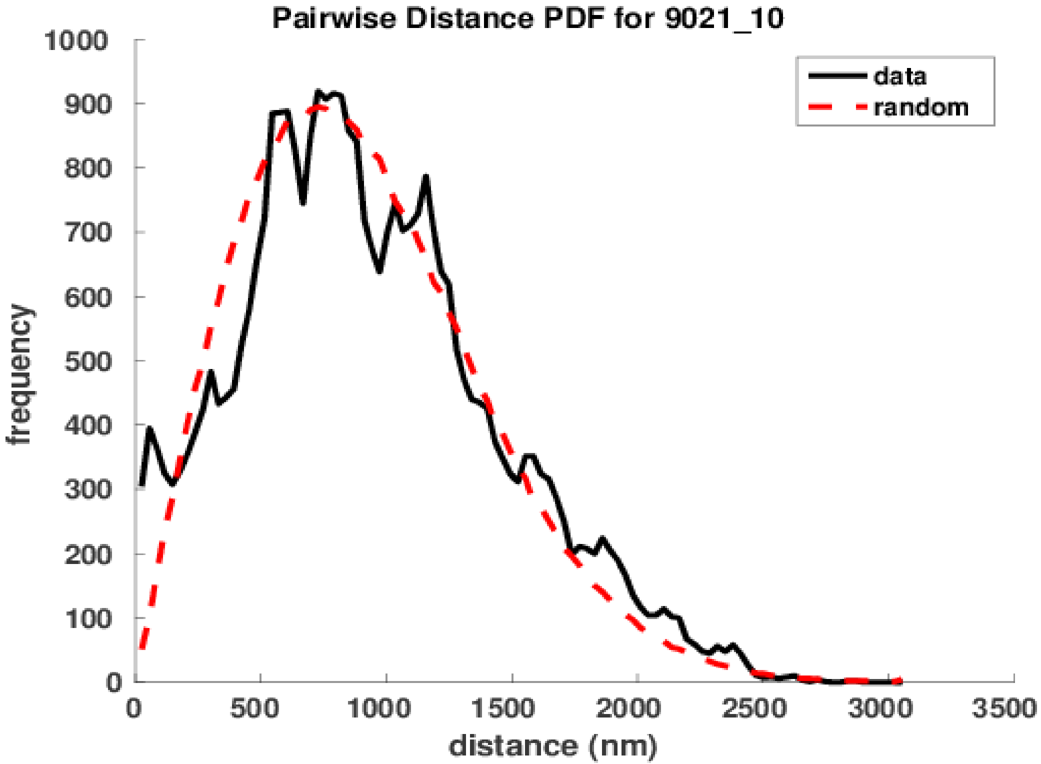
Jun Zhang, Karin Leiderman, Janet R. Pfeiffer, Bridget S. Wilson, Janet M. Oliver and Stanly L. Steinberg, ``Characterizing the Topography of Membrane Receptors and Signaling Molecules from Spatial Patterns Obtained using Nanometer-scale Electron-dense Probes and Electron Microscopy'', Micron, Volume 37, Issue 1, January 2006, 14—34 (DOI:10.1016/j.micron.2005.03.014).

Dendrograms (for estimating cluster ε):

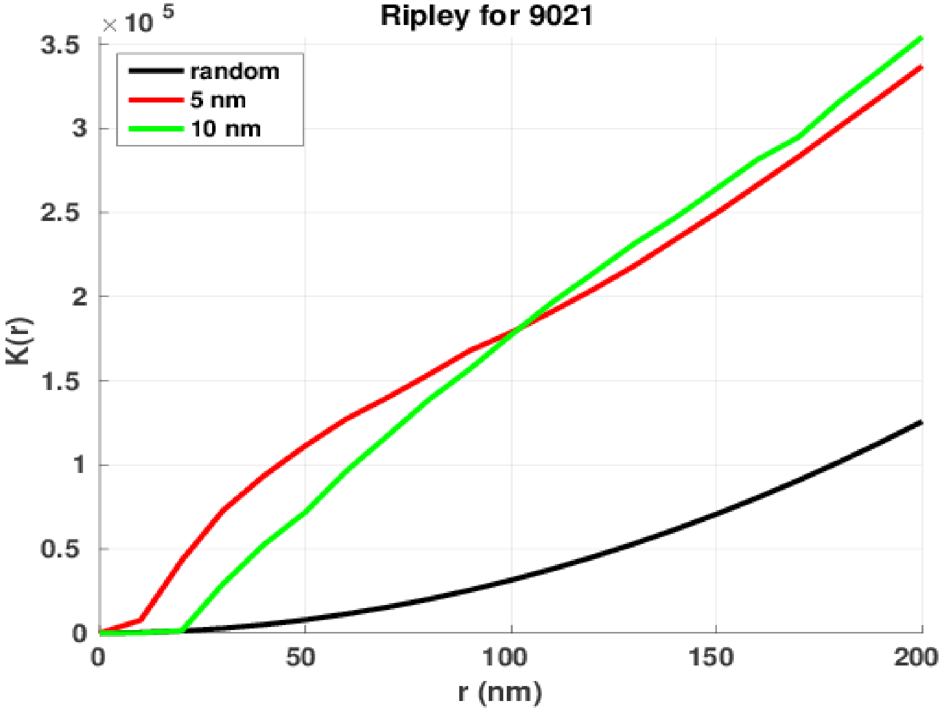
Flor A. Espinoza, Janet M. Oliver, Bridget S. Wilson and Stanly L. Steinberg, ``Using Hierarchical Clustering and Dendrograms to Quantify the Clustering of Membrane Proteins'', *Bulletin of Mathematical Biology*, Volume 74, Issue 1, January 2012, 190—211 (PMID: 21751075, PMCID: PMC3429354).



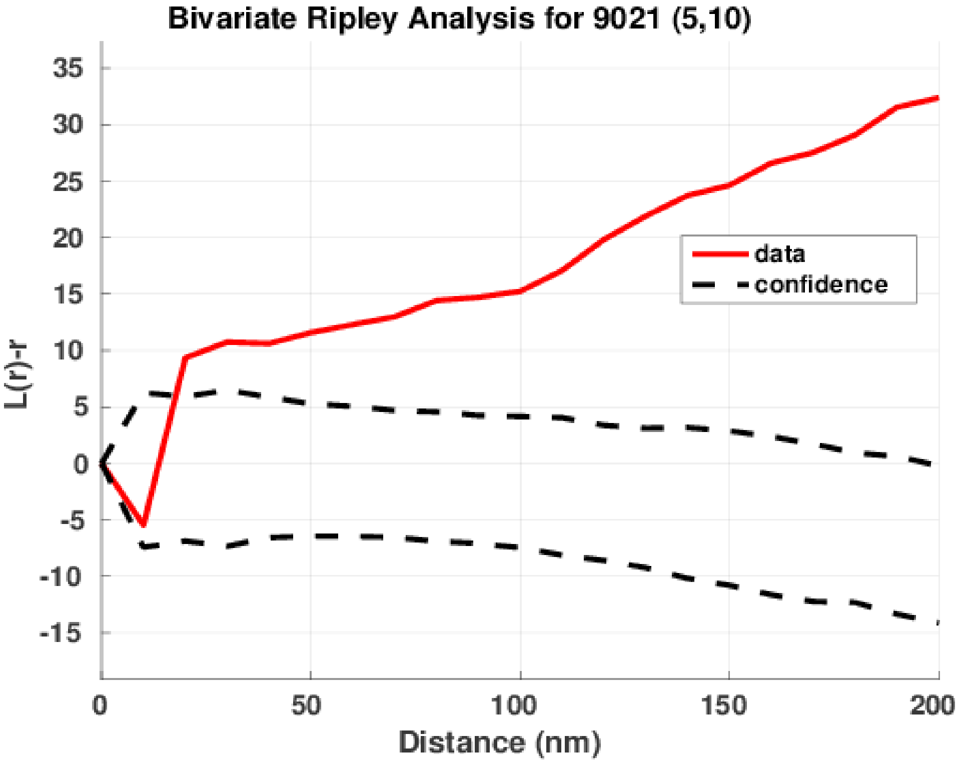
Hopkins PDF for 9021_10



Pairwise Distance PDF for 9021_10



Ripley for 9021



Bivariate Ripley Analysis for 9021 (5,10)

H-SET (Hierarchical Single Emitter hypothesis Test)

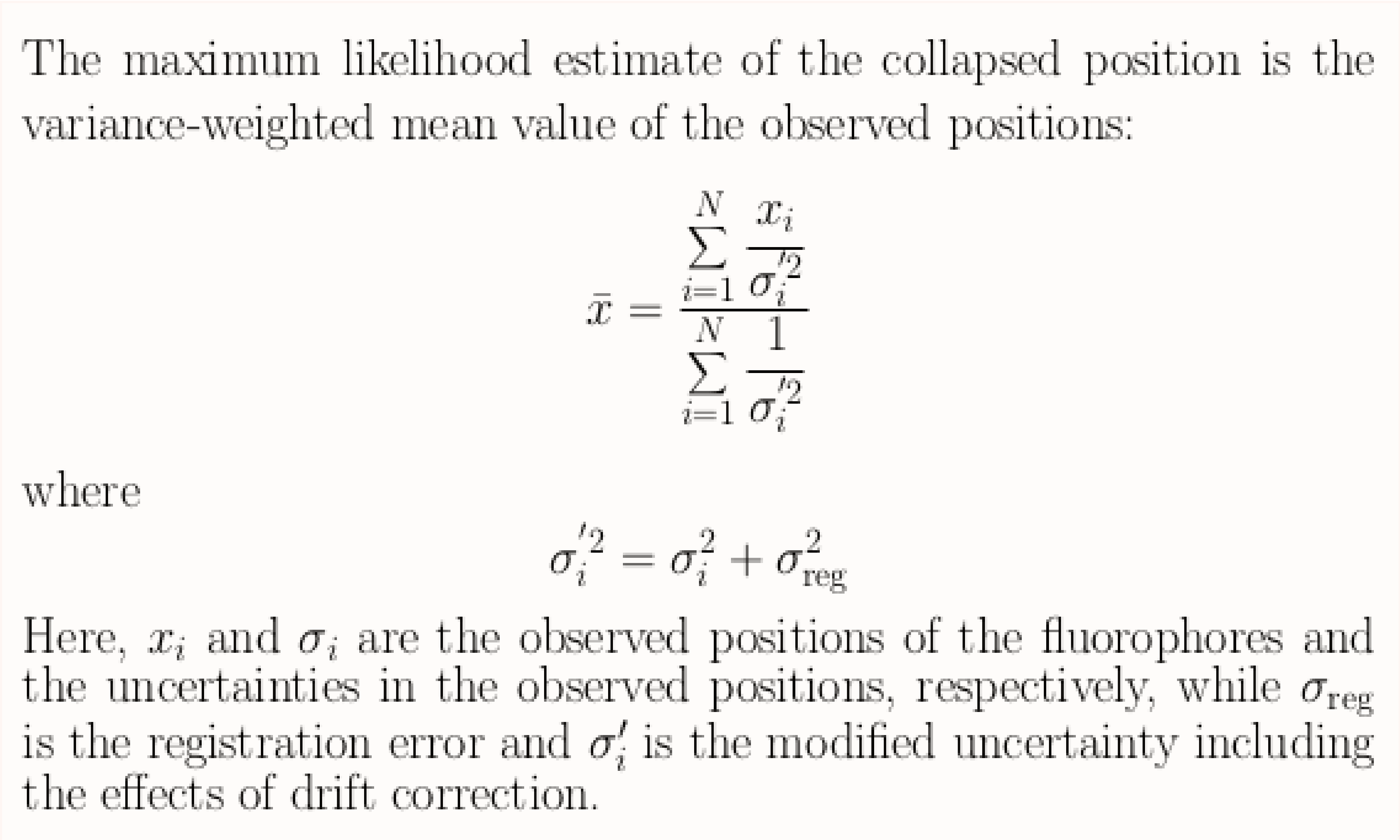
A top-down clustering algorithm to collapse clusters of observations of blinking fluorophores into a single estimate of the true location (localization) of the fluorophore using a log-likelihood hypothesis test.

Jia Lin, Michael J. Wester, Matthew S. Graus, Keith A. Lidke and Aaron K. Neumann, ``Nanoscopic cell wall architecture of an immunogenic ligand in *Candida albicans* during antifungal drug treatment'', *Molecular Biology of the Cell*, Volume 27, Number 6, March 15, 2016, 1002—1014 (DOI: 10.1091/mbc.E15-06-0355, PMID: 26792838).

H-SET

- If observations are clustered directly, typically there is a peak at the localization precision.
- This is due to overcounting of the probes, where single localizations are being expressed by multiple observations.
- A more sophisticated collapsing algorithm (reversible-jump Markov Chain Monte Carlo [RJMCMC]) is being developed by Mohamad Fazel and Keith Lidke, which will avoid creating artificial small clusters.

H-SET



The maximum likelihood estimate of the collapsed position is the variance-weighted mean value of the observed positions:

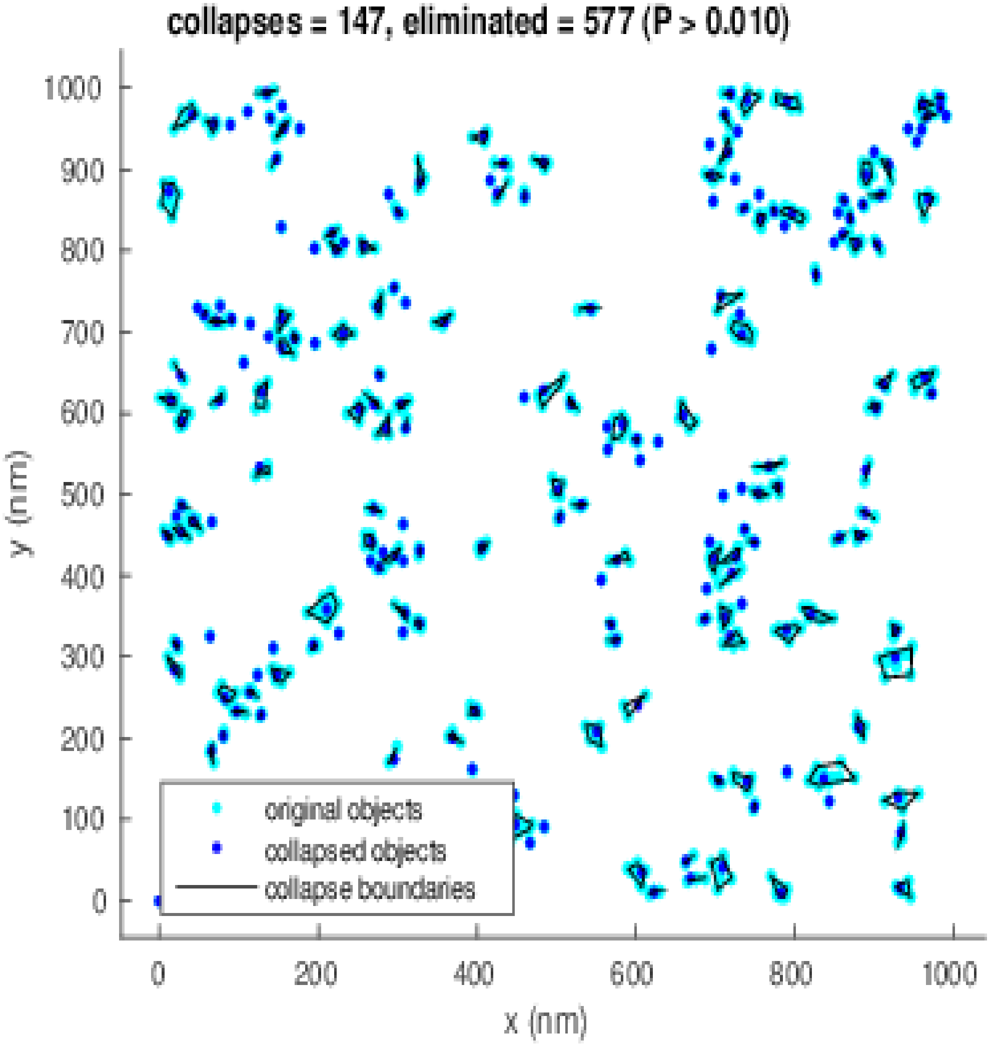
$$\bar{x} = \frac{\sum_{i=1}^{N} \frac{x_i}{\sigma_i'^2}}{\sum_{i=1}^{N} \frac{1}{\sigma_i'^2}}$$

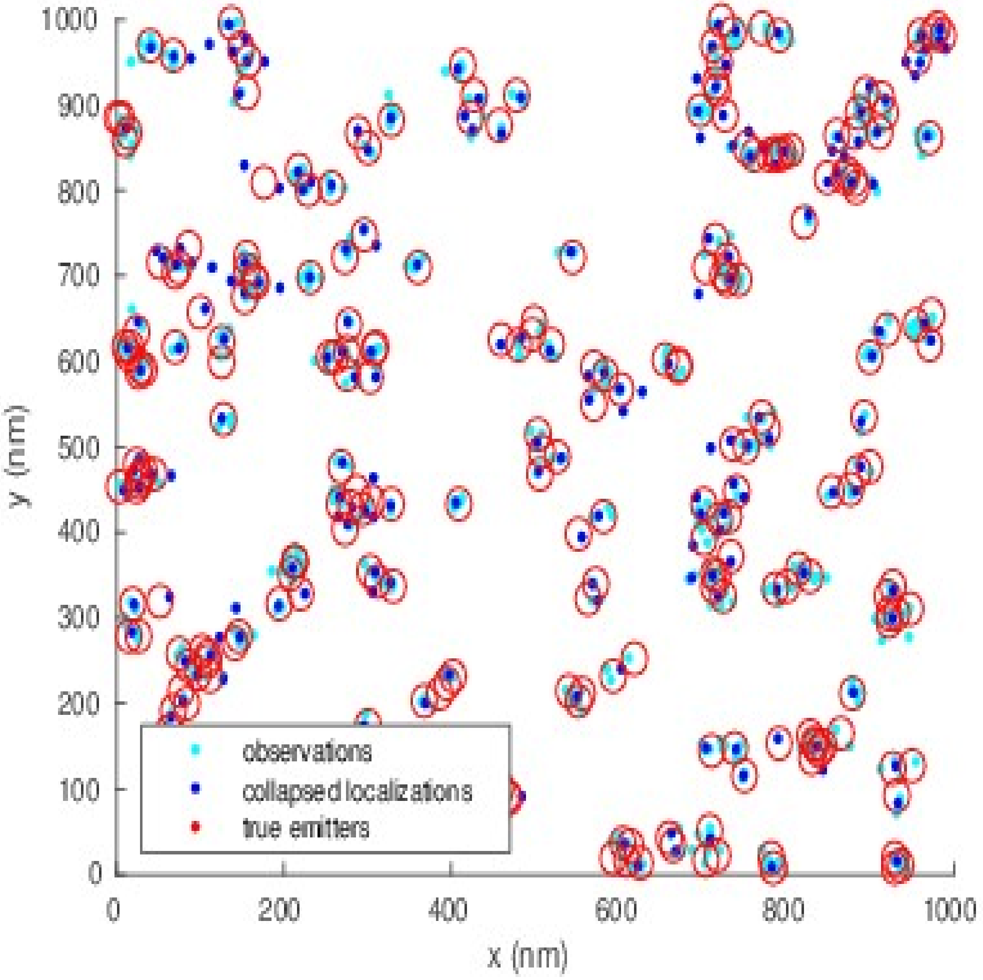
where

$$\sigma_i'^2 = \sigma_i^2 + \sigma_{\text{reg}}^2$$

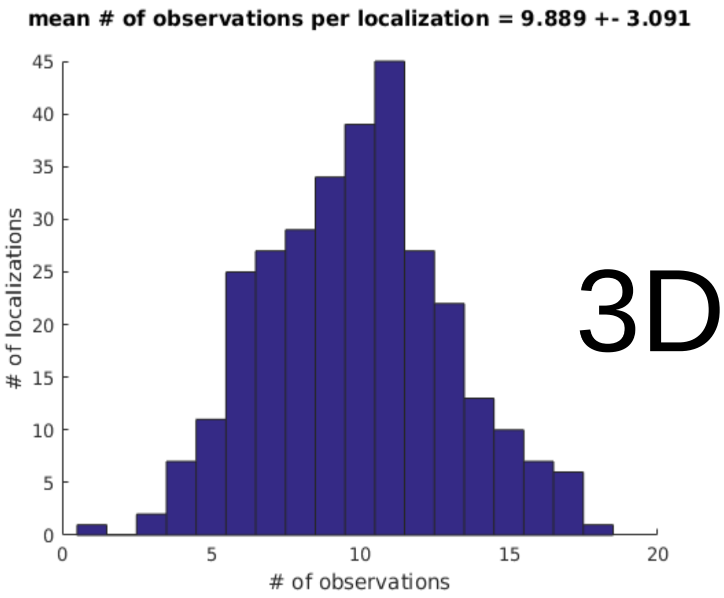
Here, x_i and σ_i are the observed positions of the fluorophores and the uncertainties in the observed positions, respectively, while σ_{reg} is the registration error and σ_i' is the modified uncertainty including the effects of drift correction.

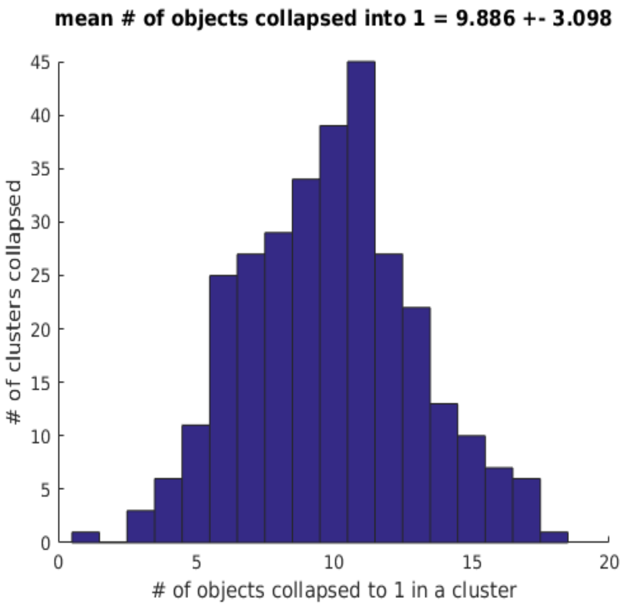
H-SET

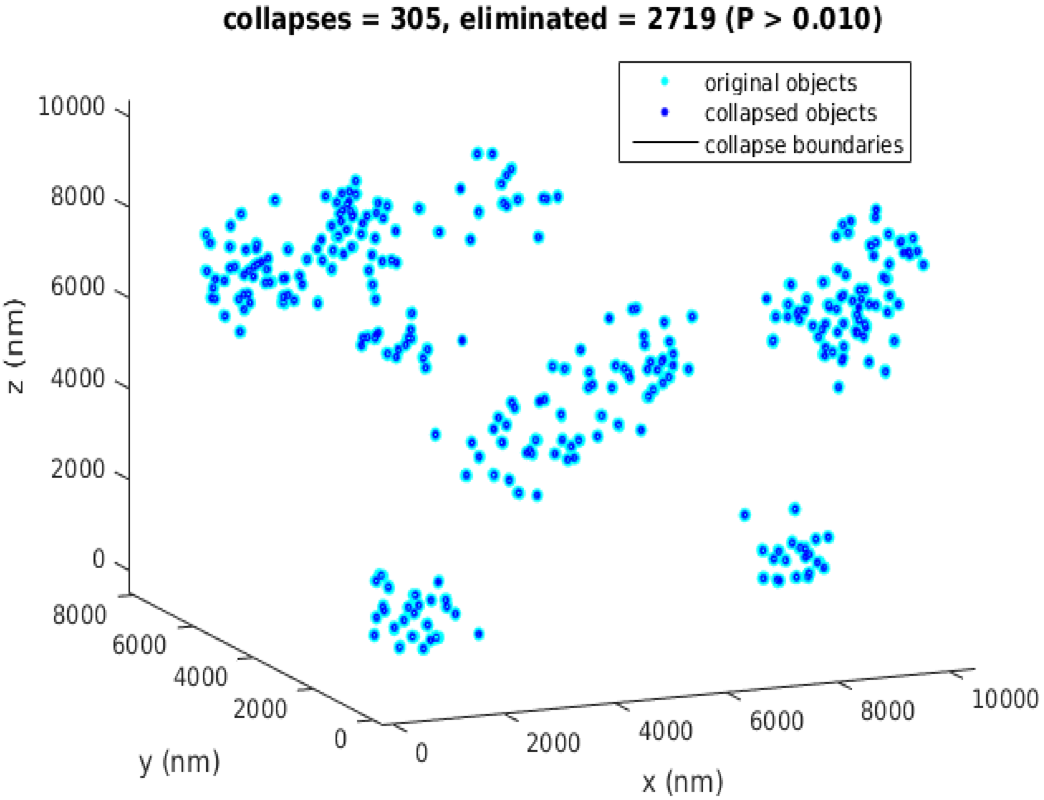

collapses = 147, eliminated = 577 (P > 0.010)
y (nm)
x (nm)
0
100
200
300
400
500
600
700
800
900
1000
0
200
400
600
800
1000
original objects
collapsed objects
collapse boundaries

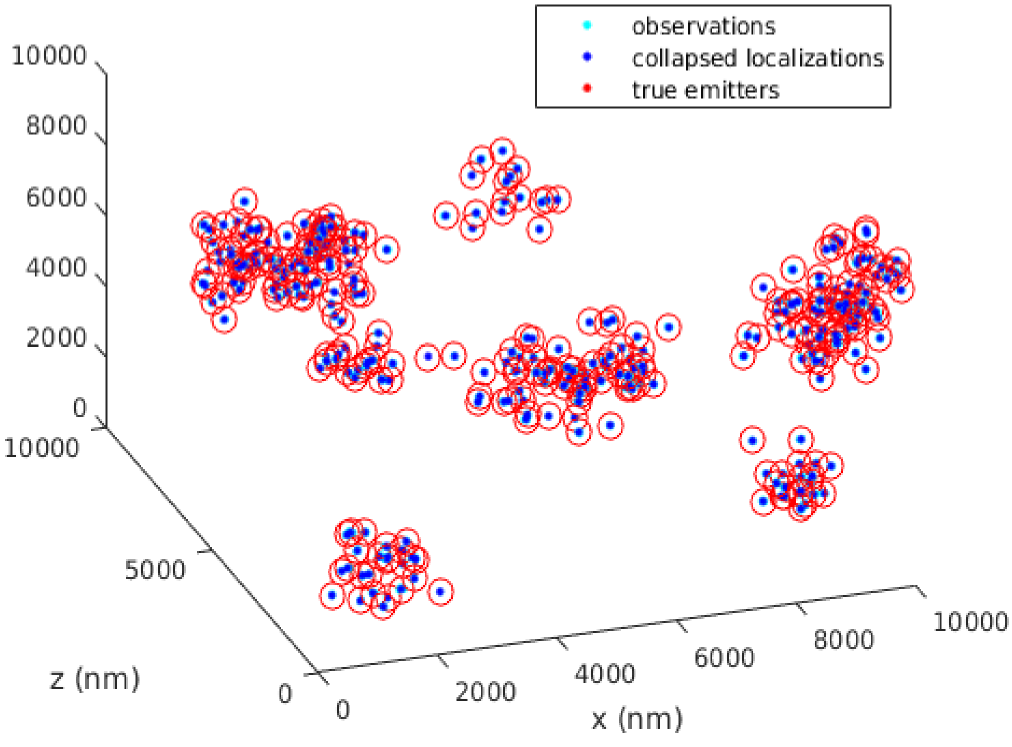

y (nm)
x (nm)
0
100
200
300
400
500
600
700
800
900
1000
0
200
400
600
800
1000
observations
collapsed localizations
true emitters

3D simulation

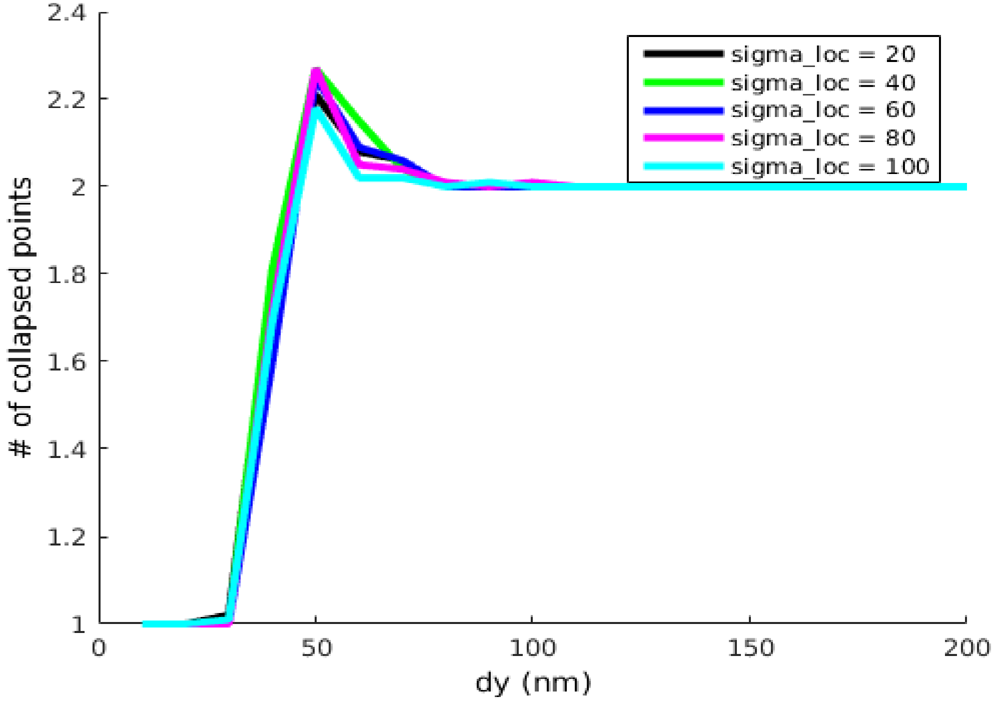


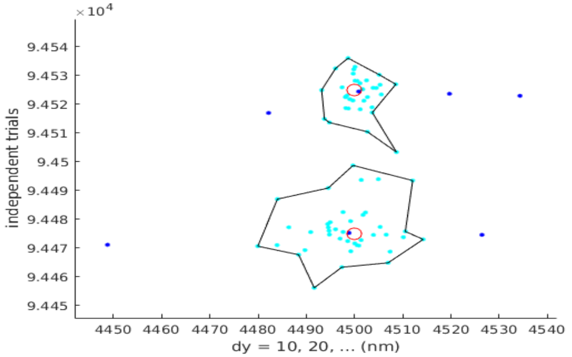


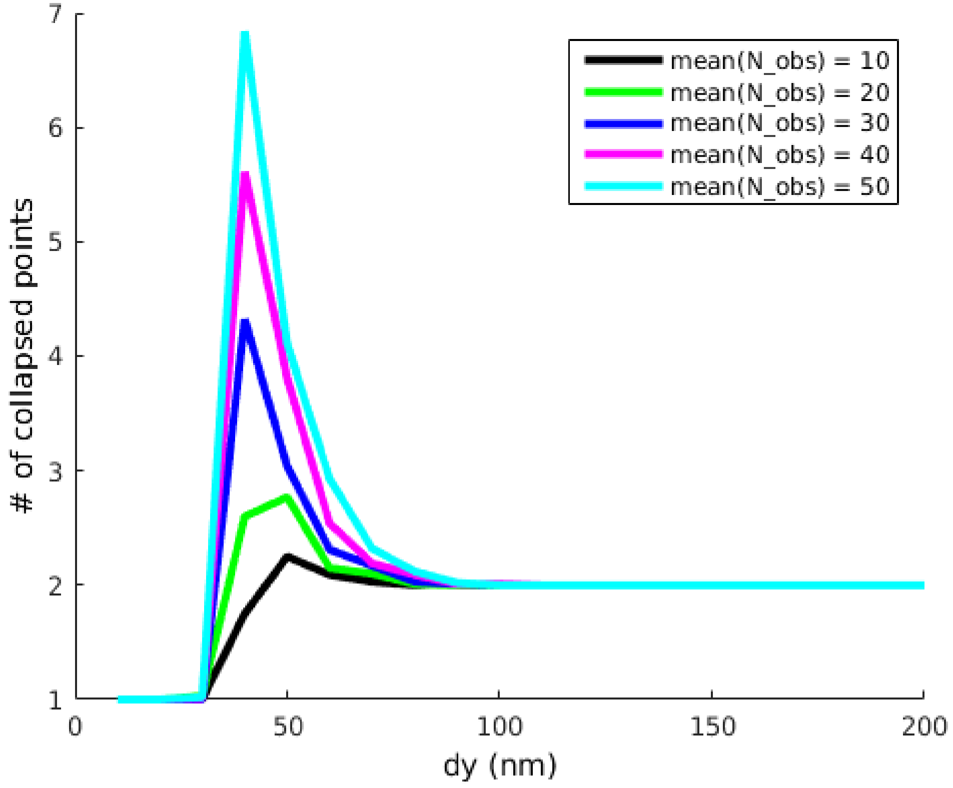


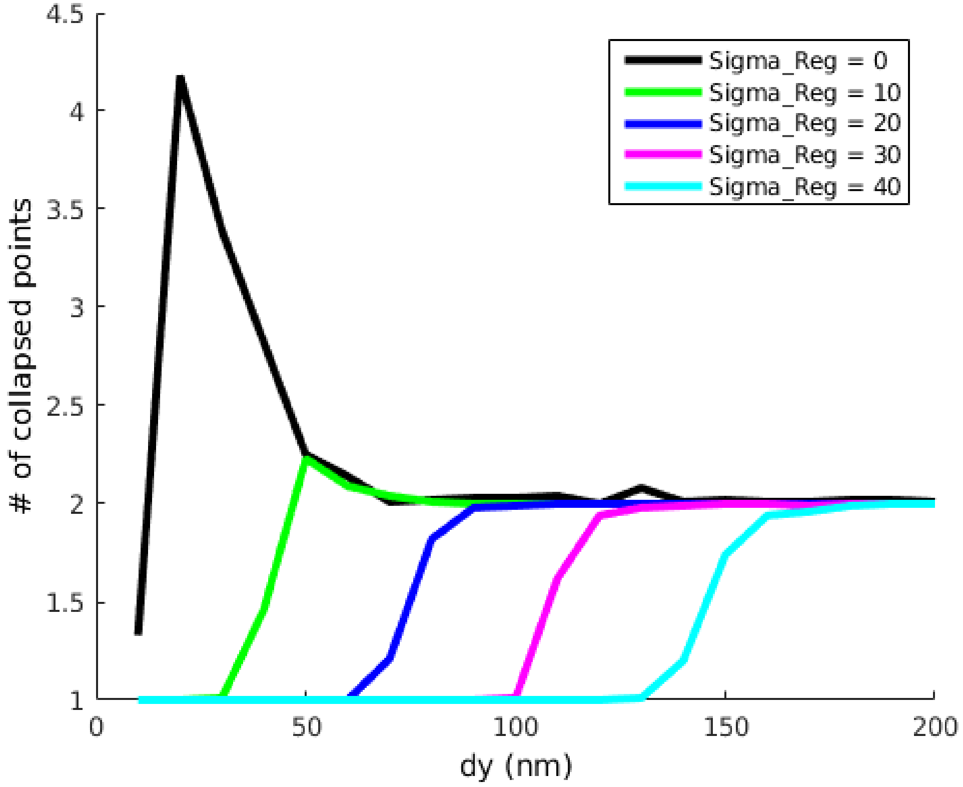


SR Collapse Studies
of various parameters for two clusters of observations at varying separations









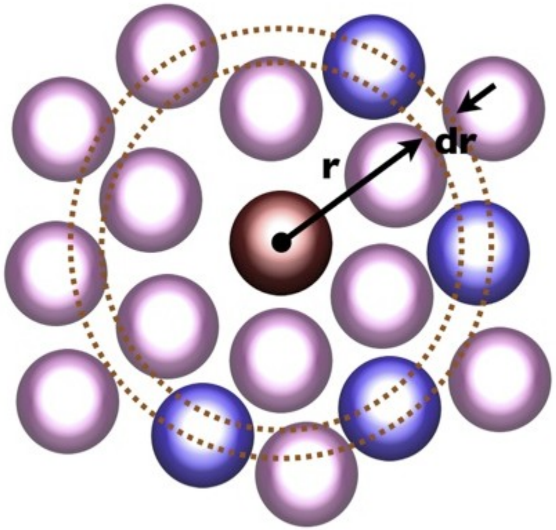
Pair Auto- and Cross-Correlation

The pair auto-correlation function (or radial distribution function) g(r) of a system of particles describes how density varies as a function of distance from a reference particle. If ρ is the average number density of particles, then the local time-averaged density at a distance r from any origin is ρ g(r). This simplified definition holds for a homogeneous and isotropic system.

Simply, g(r) measures the probability of finding a particle at a distance of r away from a given reference particle, relative to random. The general algorithm involves determining how many particles are enclosed within a shell or inner radius r and outer radius r + dr from the particle.

The pair correlation is determined by calculating the distance between all particle pairs and binning them into a histogram, which is then normalized with respect to a random distribution.

The pair cross-correlation function c(r) is similar to g(r), except that now the density distribution is calculated for particles of one color (or label) with respect to particles of a second color (or label).



https://en.wikipedia.org/wiki/Radial_distribution_function

Pair Auto- and Cross-Correlation

g(r) and c(r) report the increased probability of finding a second localized signal a distance r away from a given localized signal in super-resolution images, computed via fast Fourier transforms of the images.

Auto-Correlation:

$g(r) = < \rho(R)\ \rho(R - r) > / \rho^2$ →

$$g(\vec{r}) = \frac{FFT^{-1}(|FFT(I)|^2)}{\rho^2 N(\vec{r})}$$

Cross-Correlation:

$c(r) = < \rho_1(R)\ \rho_2(R - r) > / (\rho_1\ \rho_2)$ →

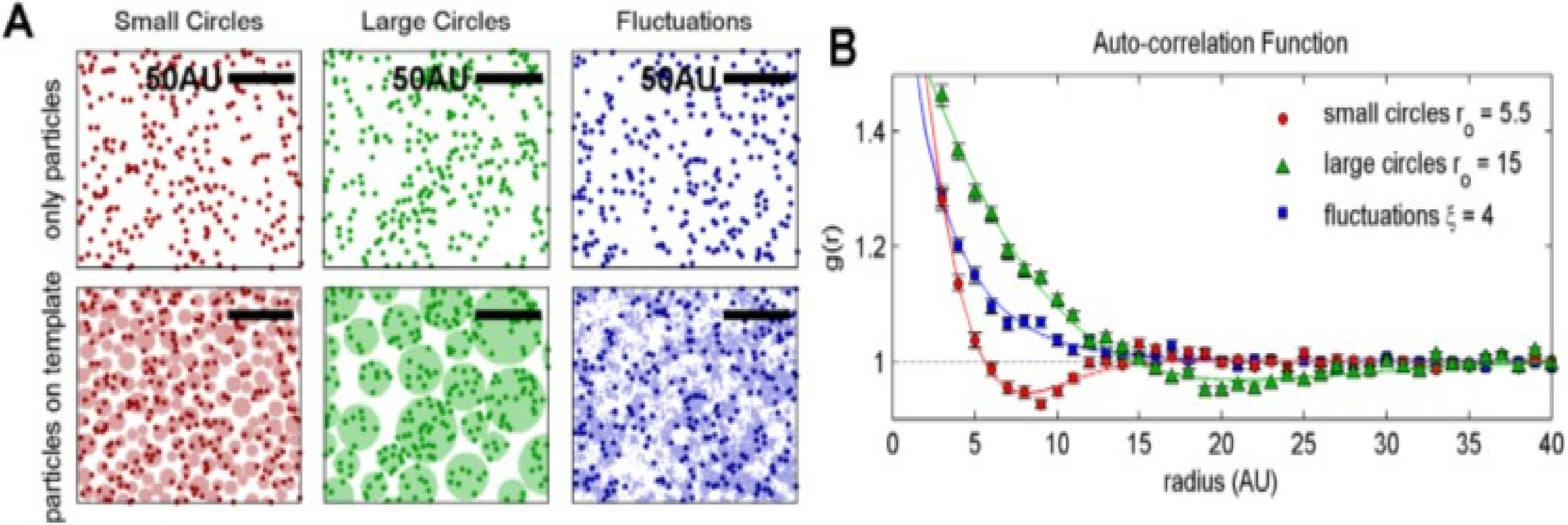
$$c(\vec{r}) = \mathrm{Re}\left\{ \frac{FFT^{-1}(FFT(I_1) \times conj[FFT(I_2)])}{\rho_1 \rho_2 N(\vec{r})} \right\}$$

where $\rho = < \rho(r) >$, $\rho_1 = < \rho_1(r) >$, $\rho_2 = < \rho_2(r) >$, and the average is over all positions R in the image.

$$N(\vec{r}) = FFT^{-1}(|FFT(W)|^2)$$

In this definition, g(r) = 1 represents a random distribution. Often, it can be assumed that g(r) is symmetric to rotations, and so can be averaged over angles.

Pair Auto- and Cross-Correlation



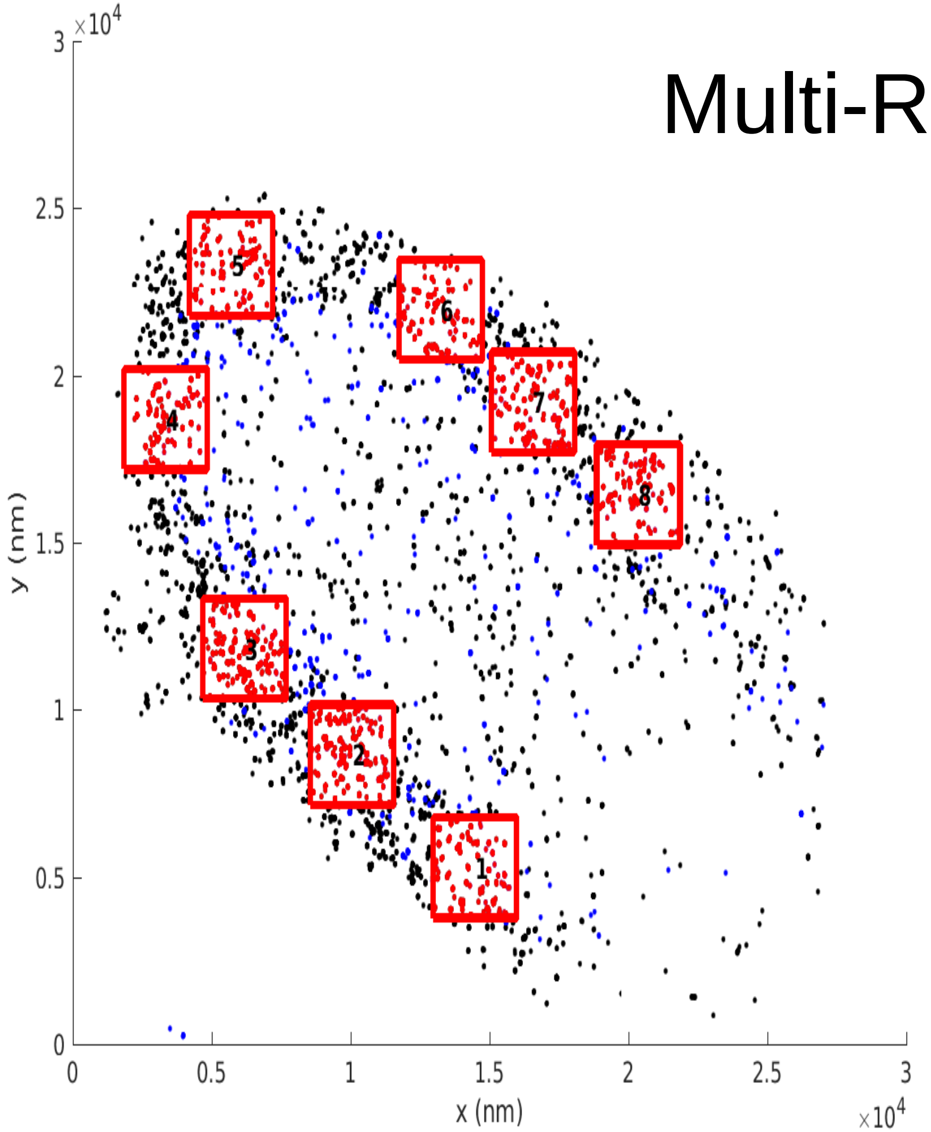
Fitting the g/c(r) curve produces estimates of cluster and localization sizes and densities. Able to combine results for multiple rectangular ROIs of various sizes.

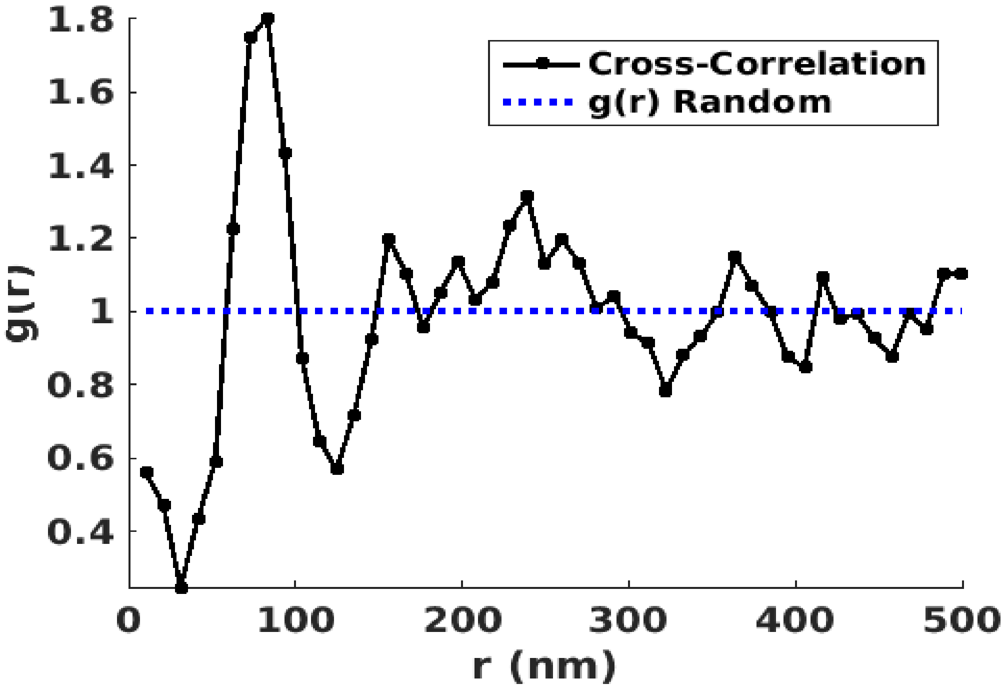
Sarah L. Veatch, Benjamin B. Machta, Sarah A. Shelby, Ethan N. Chiang, David A. Holowka and Barbara A. Baird, ``Correlation Functions Quantify Super-Resolution Images and Estimate Apparent Clustering Due to Over-Counting'', *PLoS ONE*, Volume 7, Issue 2, February 2012, 1—13.

Prabuddha Sengupta, Tijana Jovanovic-Talisman, Dunja Skoko, Malte Renz,Sarah L. Veatch and Jennifer Lippincott-Schwartz, ``Probing protein heterogeneity in the plasma membrane using PALM and pair correlation analysis'', *Nature Methods*, Volume 8, Number 11, November 2011, 969—975.

Multi-ROI Pair Cross-Correlation

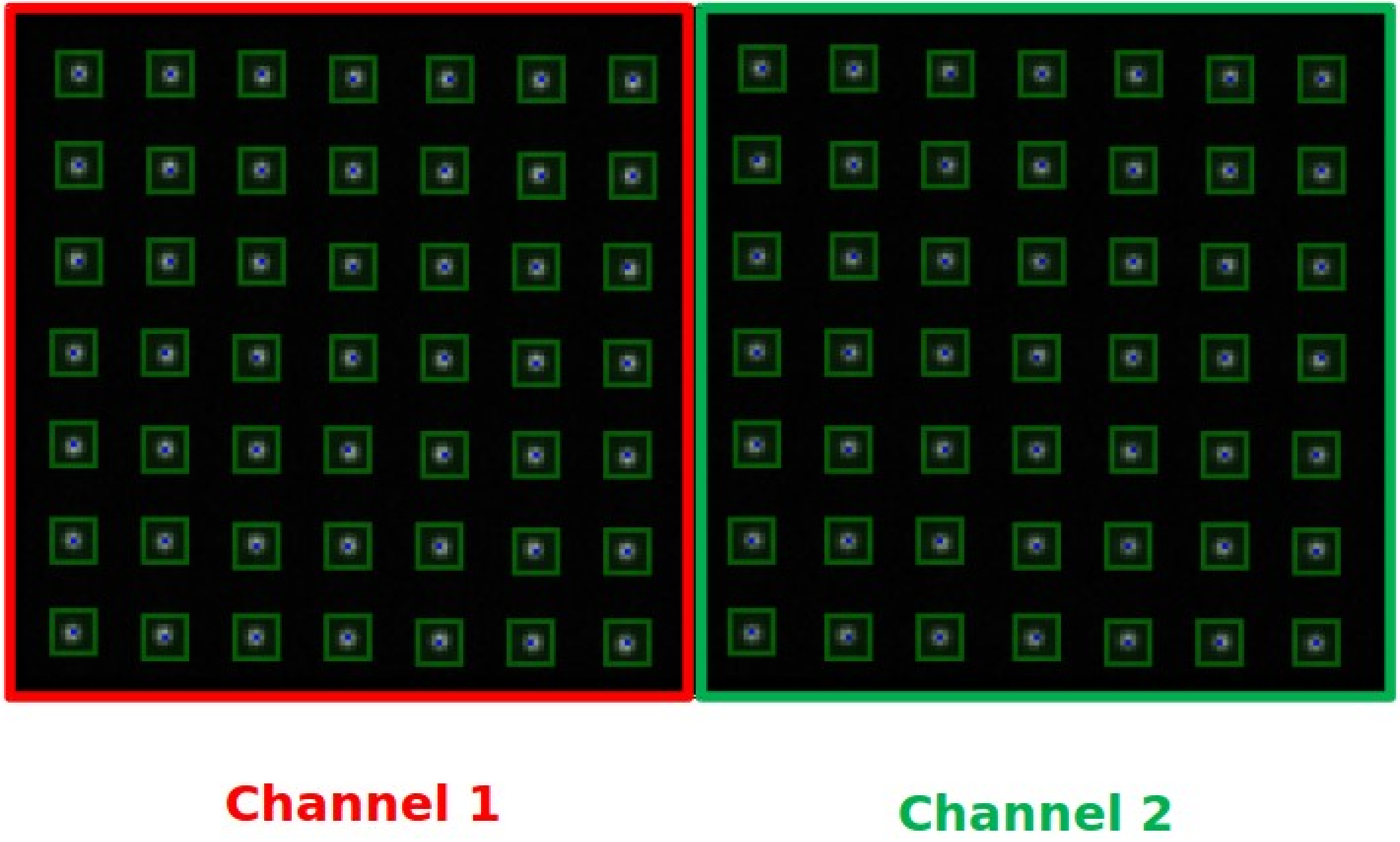
(Farzin Farzam, Keith Lidke)





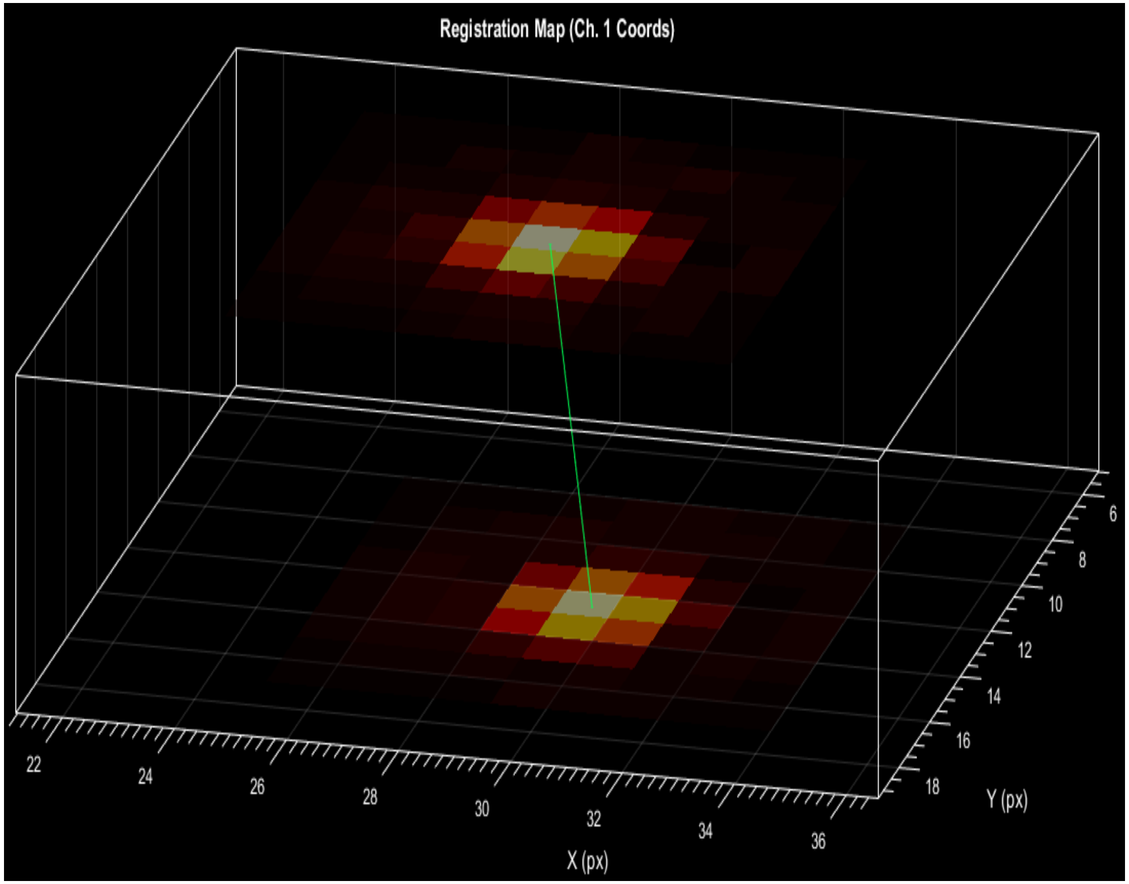
Channel Alignment

NanoGrid with lamp transmission light

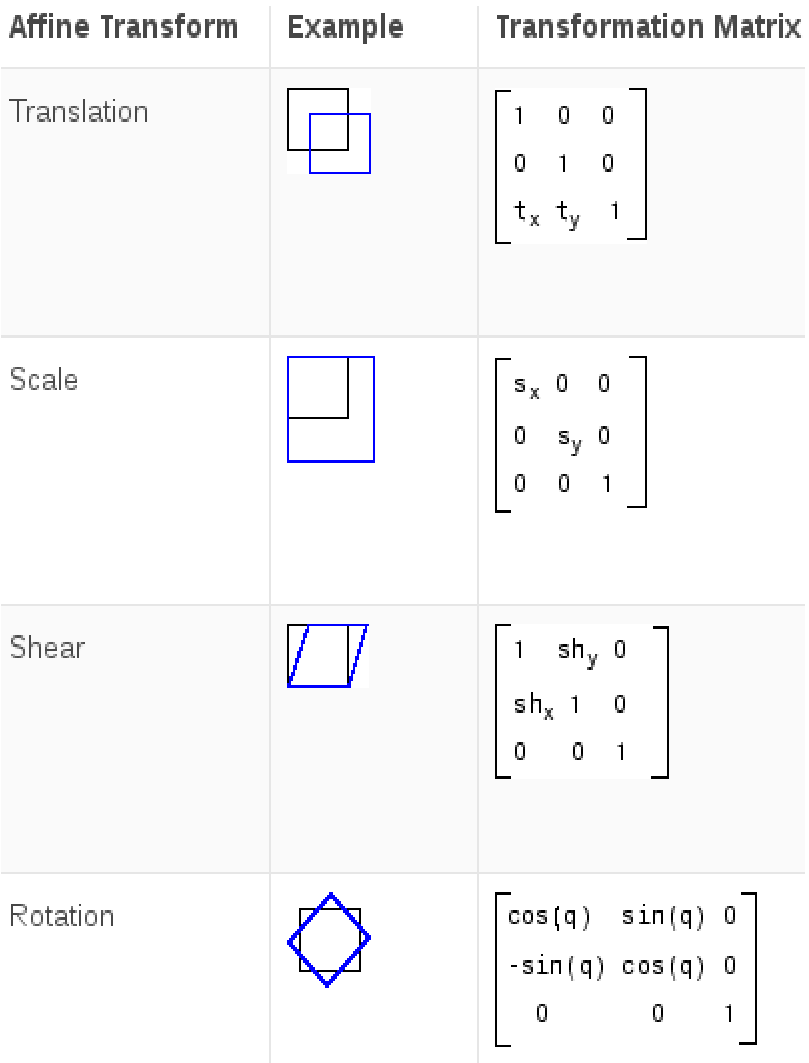


Adapted from Keith A. Lidke, University of New Mexico (BPS 2016)

Channel Alignment



Local affine transform produces small residual errors



Affine Transform	Example	Transformation Matrix
Translation		$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ t_x & t_y & 1 \end{bmatrix}$
Scale		$\begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$
Shear		$\begin{bmatrix} 1 & sh_y & 0 \\ sh_x & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
Rotation		$\begin{bmatrix} \cos(q) & \sin(q) & 0 \\ -\sin(q) & \cos(q) & 0 \\ 0 & 0 & 1 \end{bmatrix}$

(MathWorks)

Adapted from Keith A. Lidke, University of New Mexico (BPS 2016)

Acknowledgments

Keith Lidke

Stanly Steinberg

Flor Espinoza

Carolyn Pehlke

Matt Graus

Jia Lin

Aaron Neumann

Samantha Schwartz

Zsolt Sebestyen

Adam Halasz