bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

StormGraph: An automated graph-based algorithm for quantitative clustering analysis of single-molecule localization microscopy data.

Joshua M. Scurll[1,2], Libin Abraham[2,3], Da Wei Zheng[1], Reza Tafteh[4], Keng C. Chou[4], Michael R. Gold[2,3], Daniel Coombs[1]

[1] Department of Mathematics and Institute of Applied Mathematics, 1984 Mathematics Road, University of British Columbia, Vancouver, British Columbia V6T 1Z2 Canada,

[2] Life Sciences Institute I[3] and Cell Research Groups, University of British Columbia, 2350 Health Sciences Mall, Vancouver, British Columbia V6T 1Z3 Canada,

[3] Department of Microbiology and Immunology, University of British Columbia, 2350 Health Sciences Mall, Vancouver, British Columbia V6T 1Z3 Canada,

[4] Department of Chemistry, University of British Columbia, Vancouver, British Columbia V6T 1Z1 Canada

Corresponding authors

Joshua M. Scurll and Daniel Coombs

Department of Mathematics and Institute of Applied Mathematics

1984 Mathematics Road, University of British Columbia

Vancouver, British Columbia V6T 1Z2 Canada

Phone: +1-604-446-0450 (JMS); +1-604-822-2859 (DC)

E-mail: jscurll.ubc@gmail.com; coombs@math.ubc.ca

Keywords: single-molecule localization microscopy, dSTORM, heterogeneous clustering, graph community detection, hierarchichal clustering, B cell receptor, co-localization.

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

Abstract

Clustering of proteins is crucial for many cellular processes and can be imaged at nanoscale resolution using single-molecule localization microscopy (SMLM). Existing cluster analysis methods for SMLM data suffer from major limitations, such as unsuitability for heterogeneous datasets, failure to account for uncertainties in localization data, excessive computation time, or inability to analyze three-dimensional data. To address these shortcomings, we developed StormGraph, an algorithm using graph theory and community detection to identify and quantify clusters in heterogeneous 2D and 3D SMLM datasets. StormGraph accounts for localization uncertainties and, by determining thresholds adaptively, it allows many heterogeneous samples to be analyzed using identical parameters. Consequently, StormGraph improves the potential accuracy, objectivity, and throughput of cluster analysis. Furthermore, StormGraph generates a hierarchical clustering, and it quantifies cluster colocalization for two-color SMLM data. We use simulated data to show that StormGraph is superior to existing algorithms. Finally, we demonstrate its application to two-dimensional B-cell antigen receptor clustering and three-dimensional intracellular LAMP-1 clustering.

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

Introduction

Single-molecule localization microscopy (SMLM) techniques, such as direct stochastic optical reconstruction microscopy (dSTORM) (1; 2) and photoactivated localization microscopy (PALM) (3), overcome the diffraction limit of conventional microscopy by acquiring many sequential images, each containing very few fluorescing labels. Individual labels can then be computationally super-resolved and precisely localized to generate a list of localization coordinates, often with estimated positional uncertainties (4; 5; 6). This is possible in both two and three dimensions (7; 8; 9; 10).

SMLM is commonly used to investigate nanoscale clustering of cell-membrane and intracellular proteins (11; 12; 13; 14; 15; 16; 17; 18; 19; 20), which usually exhibits both cell-to-cell and within-cell heterogeneity. Notwithstanding, clustering is frequently analyzed using spatial summary statistics that fail to capture the heterogeneity of clusters within a sample, such as Ripley's functions (21; 22). Instead, clusters can be individually quantified by using a clustering algorithm to assign localizations to specific clusters. However, using existing clustering algorithms, it is difficult to accurately and objectively analyze multiple heterogeneous samples. Subjective bias can result from algorithm parameter selection, or from selection of a small number of "representative samples" to analyze using slow or cumbersome algorithms. Failure to account for localization uncertainties can also make conclusions unreliable.

The most widely used clustering algorithms in SMLM literature, including Density-Based Spatial Clustering of Applications with Noise (DBSCAN) (23), identify clusters based on a user-specified minimum number of points within a user-specified radius. However, these parameters are difficult to select and, for heterogeneous samples, should be sample-specific. Recently, algorithms based on Voronoi diagrams, for example ClusterViSu, have been developed for 2D (24; 25) and 3D SMLM data (26). Importantly, however, none of the above clustering algorithms account for localization uncertainties. A pixelated variant of DBSCAN (27) partially addresses this deficiency but is limited to 2D datasets. Simulation-aided DBSCAN (28) offers a more complete strategy, but it remains limited by user-determined DBSCAN parameter selection. A Bayesian, model-based cluster identification method (29) uses localization uncertainties and has been extended to 3D (30), but it assumes that clusters are circular or spherical and it is prohibitively slow.

To address all of the limitations of existing cluster analysis methods described above, we developed StormGraph, a comprehensive graph-based clustering algorithm inspired by PhenoGraph (31), which was developed for single-cell cytometry data. StormGraph converts SMLM data into a graph using localization coordinates and their uncertainties to specify nodes and weighted edges. It then utilizes graph theory and community detection algorithms (32) to assign nodes to specific clusters. Crucially, StormGraph determines

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

key thresholds from the data for each region of interest (ROI) adaptively, using at most three user-specified parameters that can remain fixed across experiments, enabling unbiased comparison of results. Unlike the Bayesian method, StormGraph makes no assumptions about the shapes of clusters and it is at least an order of magnitude faster. Moreover, StormGraph has both 2D and 3D implementations and it can quantify cluster overlap for two-color SMLM data.

Furthermore, SMLM data often exhibits hierarchical clustering at multiple spatial scales. For example, SMLM revealed multiscale organization of RNA polymerase in *Escherichia coli* (33). To enable multiscale cluster analysis, StormGraph generates a hierarchical clustering. This is an advantage over existing methods, for which users must repeatedly change parameters to perform multiscale analysis. Notwithstanding, StormGraph also generates an appropriate single-level clustering.

Here, we describe StormGraph and its capabilities, and we use simulated data to compare its accuracy to that of DBSCAN and ClusterViSu. We then apply StormGraph to characterize nanoscale clustering of B-cell antigen receptors (BCRs) from heterogeneous 2D SMLM data. We also demonstrate StormGraph's ability to quantify 3D clusters of the lysosomal protein LAMP-1 and to quantify cluster overlap for two-color SMLM data. Because of cell-to-cell heterogeneity as well as non-uniformity of cellular compartments such as the plasma membrane, it is essential to compile SMLM data from ROIs from multiple cells for each experimental condition. To make this practical, we developed software to crop ROIs and batch process StormGraph analysis for multiple samples. We will make the software and source code freely available upon publication of this manuscript.

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

Results

The StormGraph algorithm

To identify clusters in SMLM data, dense localization neighborhoods must be identified. To this end, StormGraph first determines an ROI-specific length scale r_0 from the data using either of two methods (see Methods and Figure S1). One method, which seeks a balance between inter-localization and inter-cluster distances, is fully automatic but heuristic. The other is semi-automatic, using k-nearest neighbor (kNN) distances with user-defined k. The automatic method reduces user input but is designed primarily for data with very few dispersed localizations between clusters, whereas the kNN method is universally applicable.

Next, using the localizations as nodes (Figure 1a), StormGraph essentially constructs a weighted r_0-neighborhood graph (Figure 1b) as follows. Define

$$s_{ij} = \begin{cases} 1 - r_{ij}/r_0 & \text{if } r_{ij} \leq r_0, \\ 0 & \text{if } r_{ij} > r_0, \end{cases}$$

where r_{ij} is the Euclidean distance between nodes i and j. If localization coordinate uncertainties are unknown, StormGraph assigns to each node pair $\{i, j\}$ an edge of weight $W_{ij} = s_{ij}$. Otherwise, StormGraph uses the uncertainties to estimate $\langle s_{ij} \rangle$, the expectation of s_{ij}, from Monte Carlo simulations (Methods) and assigns $W_{ij} = \langle s_{ij} \rangle$.

At this stage, unclustered localizations are identified and removed by applying a threshold to the weighted node degree,

$$\deg(i) = \sum_{j \neq i} W_{ij},$$

a proxy for local density. In principle, nodes are classified as unclustered and removed if their degree falls below a data-dependent threshold (Figure 1c). StormGraph automatically determines this threshold from random point clouds using the user-defined significance parameter α (Methods). The value of α is the maximum probability of a completely randomly distributed localization being classified as clustered, a subjective choice for the user but typically 0.05. Figure S2 shows the effects of varying the parameters α and k. Removed localizations are reported as an algorithm output.

The graph is then regenerated using a new r_0 value determined automatically from only the retained nodes. If localization coordinate uncertainties are available, edges are subsequently pruned from the graph

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

to ensure that any pair of edges have at least an estimated 50% probability of co-occurring in the r_0-neighborhood graph for the unknown true localization positions (Methods). StormGraph then finds a hierarchy of node clusters (Figure 1e) using the multi-level Infomap algorithm (34), followed by additional cluster merging when warranted (Methods).

To obtain a single-level clustering from the hierarchy, we developed a novel, fast method motivated by the idea of consensus clustering (35; 36). Briefly, clusters are recursively divided into their coarsest constituent subclusters if they closely match the connected components of an alternative neighborhood graph (Methods). Optionally, the user may specify the minimum number of points per reported cluster (MinCluSize). As output, StormGraph provides the single-level and hierarchical cluster assignments of every localization. Combined with localization coordinates, this provides the information necessary to quantify individual cluster properties, such as area (Figure 1d). Our software automatically quantifies the single-level and coarsest-level clusterings.

Lastly, a common caveat of SMLM is multiple counting of single molecules, often causing single molecules to spuriously appear as clusters. This can be due to multiple labeling of single molecules or individual fluorophores yielding multiple localizations. Therefore, StormGraph includes optional functionality that uses localization uncertainties to reclassify as unclustered localizations any putative clusters that cannot be confidently distinguished from multiply counted single molecules (Methods).

Validation using simulated data and comparison to other algorithms

To compare StormGraph with DBSCAN and ClusterViSu, we simulated a wide variety of 2 µm × 2 µm ROIs containing isolated and heterogeneously aggregated circular nanoclusters (e.g. Figure 2a; Methods). Outside the clusters we added randomly distributed molecules. Individual simulated molecules were allowed to yield multiple localizations, each with a positional uncertainty sampled from a real dSTORM experiment. We tested both the automatic and kNN ($k = 10$, 15 or 20) methods for determining r_0 while maintaining $\alpha = 0.05$. We set a minimum cluster size of 5 localizations in both StormGraph and ClusterViSu. For DBSCAN, we tested 16 different parameter choices based on the underlying parameters used for data simulation, although such knowledge is generally unavailable for real data. To assess cluster assignments from each algorithm, we used normalized mutual information (NMI) (37) and mean F-measure (38). Higher values indicate superior performance.

StormGraph consistently outperformed ClusterViSu regardless of whether localization uncertainties were used and regardless of the method used to determine r_0 (Figures 2b–c and S3). DBSCAN's performance

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

was very sensitive to the choice of parameters and no single choice was suitable for all of the data (Figure S4), demonstrating its unsuitability for batch processing analysis of heterogeneous samples. StormGraph was generally superior to DBSCAN regardless of parameter choices among those tested. Moreover, StormGraph's performance was not very sensitive to varying k from 10 to 20, particularly when localization uncertainties were used (Figures 2b and S3).

For simulated data with nanoclusters of 50 nm radius, we were able to manually identify a level of clustering from StormGraph's cluster hierarchy that accurately recovered the ground-truth nanoclusters that composed larger aggregations (Figure S5). This demonstrates that StormGraph is able to identify meaningful clusters at multiple scales. Additionally, we performed tests using simulated data without multiple counting of single molecules and found that StormGraph still outperformed ClusterViSu and DBSCAN (Figure S6).

Finally, we compared StormGraph to Ripley's H-function (22) using simulated circular clusters. Ripley's H-function was biased towards the clusters containing the most points, as mathematically expected, and it did not provide an accurate measure of cluster radius (Figure S7). Conversely, StormGraph provided excellent estimates (Figure S7).

StormGraph quantifies heterogeneous B-cell receptor clustering from dSTORM data in 2D

To test StormGraph on real SMLM data, we used it to analyze the clustering of immunoglobulin M (IgM)-isotype B-cell antigen receptors (BCRs) on the cell membranes of B lymphocytes. IgM-BCRs are thought to exist in nanoclusters on resting B cells (12; 39) and form larger "microclusters" during B-cell activation induced by antigen engagement (40; 41). The exact changes in IgM-BCR arrangement are controversial, however (12; 39; 41).

Using dSTORM, we imaged fluorescently labeled IgM-BCRs on *ex vivo* murine splenic B cells that were either resting or treated with bivalent antibodies against the BCR's Igκ light chain, used as antigen surrogates. Localization coordinates and their associated uncertainties were computationally determined from the fluorescence data. We then used StormGraph ($\alpha = 0.05$, MinCluSize = 5 localizations) to batch process the analysis of IgM-BCR clustering in, respectively, 28 and 24 rectangular ROIs that were $> 1\ \mu m^2$, from separate cells, and entirely within cell boundaries (Figure 3a). We applied StormGraph's functionality to remove from the results any clusters of localizations that could not be confidently distinguished from overcounted single molecules (Methods).

Using StormGraph's single-level clustering results, we compared cluster areas between conditions. Using

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

$k = 15$, we found that the mean area of IgM-BCR clusters was significantly larger on anti-Igκ-treated cells than on resting cells (Figure 3b(i), $p < 10^{-5}$), as expected. This difference was mainly due to an increase in the size and frequency of clusters > 6000 nm^2, rather than a uniform increase in cluster areas (Figure 3b(ii)). In fact, the majority of clusters present on anti-Igκ-treated cells were small multimers that were comparable to, or even smaller than, the IgM-BCR clusters on untreated cells. The automatic (no k value) implementation of StormGraph yielded consistent conclusions (Figure S8).

Large BCR clusters have been associated with chronic BCR signaling in a subset of activated B-cell like (ABC) diffuse large B-cell lymphomas (DLBCLs). Diffraction-limited microscopy revealed large IgM-BCR microclusters in the absence of any stimulus on the ABC DLBCL cell lines HBL-1 and TMD8 but not on the Burkitt's lymphoma cell line BJAB (42). To further investigate this observation, we used dSTORM to image IgM-BCRs on HBL-1, TMD8 and BJAB cells. We then batch processed StormGraph analysis of, respectively, 39, 33 and 81 ROIs > 1 μm^2 (Figure 3c), which contained between 5×10^3 and 3×10^5 localizations.

Using $k = 15$, StormGraph revealed that the mean areas of IgM-BCR clusters on HBL-1 and TMD8 cells were significantly larger than on BJAB cells ($p < 10^{-4}$ and $p < 10^{-14}$ respectively; Figure 3d(i)). Interestingly, the relative distributions of IgM-BCR cluster areas on BJAB and HBL-1 cells resembled those for resting and anti-Igκ-treated B cells, respectively. Both BJAB and HBL-1 cells had many small IgM-BCR clusters, but HBL-1 displayed a notable increase in the size and frequency of large clusters exceeding 10^4 nm^2. In contrast, TMD8 cells displayed an overall increase in cluster areas compared to BJAB (Figure 3d(ii)). Again, the automatic implementation of StormGraph yielded similar results (Figure S8). Our observations reveal that IgM-BCR organization can differ substantially between DLBCL cell lines of the same ABC subtype. Furthermore, assuming that BCR signaling is mostly due to large clusters, the presence of many small IgM-BCR clusters suggests that only a fraction of IgM-BCRs participate in chronic BCR signaling in HBL-1 cells, unless the clusters are highly dynamic.

Finally, we repeated all StormGraph analyses ignoring localization uncertainties. Although the magnitude and statistical significance of our results were altered, the overall conclusions were unchanged (Figure S8). Importantly, this demonstrates that StormGraph can still detect differences in clustering when localization uncertainties are unavailable.

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

StormGraph is robust to changes in global density of SMLM localizations

Because the average density of SMLM localizations can vary between ROIs, a clustering algorithm must not depend on the average localization density if batch processing is to be implemented and clustering results are to be compared across samples with different localization densities. We verified that StormGraph is robust in this regard by repeating StormGraph analysis after randomly removing 0%, 25%, 50% or 75% of the localizations from a dSTORM ROI containing heterogeneous clusters (Figure 4a). Although small, low-density clusters were eventually lost, the identification and area quantification of large, unambiguous clusters was robust, and the overall distribution of cluster areas was not significantly impacted ($p > 0.05$; Figure 4b).

We also tested StormGraph's sensitivity to random noise by artificially adding random localizations (with uncertainties) to the same ROI (Figure S9). StormGraph's ability to detect all but small, low-density clusters was again robust, and its overall sensitivity to random noise was minimized by including localization uncertainties and using the kNN method to determine r_0. This implementation with $k = 15$ resulted in no statistically significant ($p < 0.05$) change in the distribution of cluster areas until the ratio of true localizations to artificial localizations was < 2.

Two-color analysis of cluster overlap

To quantify colocalization of differently colored (e.g. red and blue) clusters in two-color SMLM data, our software quantifies the total area of overlap divided by each of the following: (1) total red cluster area; (2) total blue cluster area; and (3) total area covered by clusters of either color, yielding the Jaccard index (43) (Figure 5d). Our software also reports analogous quantities using numbers of localizations instead of areas (not shown). To estimate the maximal experimentally observable colocalization, colocalization analysis should first be applied to the same molecular species labeled with two different probes. This rarely yields 100% colocalization for several reasons, including differing affinities of antibody-fluorophore conjugates, differing photophysical properties of fluorophores, and the inability of two probes to occupy the same binding site.

To demonstrate cluster overlap analysis by StormGraph, we performed such a positive control experiment by simultaneously labeling cell-surface IgG-BCRs on murine A20 B cells with anti-IgG antibodies conjugated to either Alexa Fluor 647 (AF647) or Cy3B fluorophores. These antibodies were bivalent, thus inducing formation of large clusters prior to cell fixation. Both color channels were imaged using dSTORM and aligned using custom MATLAB code to correct for chromatic aberrations. We then analyzed multiple ROIs

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

using StormGraph (Figure 5). On average, we found 79% overlap of the IgG-AF647 clusters with the IgG-Cy3B clusters and 66% overlap of the IgG-Cy3B clusters with the IgG-AF647 clusters (Figure 5e). This difference is likely due to differing qualities of the AF647- and Cy3B-conjugated antibodies. The Jaccard index cannot exceed either one-sided overlap score, and we obtained an average Jaccard index of 0.5. In a similar experiment but for tubulin, ClusterViSu obtained ∼40% overlap of each probe with the other. This shows that StormGraph performs well as part of a pipeline for analyzing cluster colocalization by SMLM.

Clustering in three dimensions

To extend StormGraph to 3D, we considered some particular features of 3D SMLM. StormGraph implicitly assumes that all dimensions should be weighted equally during graph construction, but most 3D SMLM techniques achieve lower axial resolution than lateral resolution. Therefore, StormGraph pre-processes the data, for cluster identification but not subsequent quantification, by rescaling the axial (z) dimension so that average axial and lateral positional uncertainties, when known, become equal. Furthermore, 3D SMLM localizations are often concentrated around a focal plane, causing their axial distribution to be nonuniform. Accordingly, StormGraph uses the parameter α to obtain a z-dependent node-degree threshold from random point clouds with normally distributed z-coordinates (Methods). This provides a clear advantage over DBSCAN, which is unable to adapt to axial variation in localization density. For situations with localizations distributed uniformly in z, StormGraph still retains the option to use a constant threshold instead.

We compared the performances of StormGraph and DBSCAN in 3D using simulated 3D data (Methods). As in 2D, we found that, overall, StormGraph was superior to DBSCAN regardless of parameter choices (Figure S10). We also performed 2D clustering of the xy-projections of our simulated 3D data. To achieve results with DBSCAN that were comparable between 2D and 3D, it was inevitably necessary to use different parameter values. In contrast, StormGraph required no changes to parameters, thus making it easy to switch between 2D and 3D analyses. Nonetheless, including the z-component of 3D data improved clustering accuracy (Figure S10). This is because localizations and clusters that are separated only in z are indistinguishable in the xy-projection.

To illustrate StormGraph's application to 3D SMLM data, we used dSTORM to image intracellular lysosomal-associated membrane protein 1 (LAMP-1). We simultaneously immunostained LAMP-1 in B16 melanoma cells with two different labels, AF647 and Cy3B, and applied StormGraph ($k = 15$, $\alpha = 0.1$, MinCluSize = 5 localizations) to a 3D ROI with axial variation in localization density and known localization uncertainties (Figure 6a–b). StormGraph detected 363 LAMP-1 AF647 clusters and 129 LAMP-1 Cy3B

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

clusters (Figure 6c-d). The AF647 clusters had volumes ranging from 1.5×10^3 nm^3 to 7.1×10^7 nm^3 with a median of 3.5×10^5 nm^3, and Cy3B clusters had volumes ranging from 3.1×10^3 nm^3 to 3.7×10^7 nm^3 with a median of 9.0×10^5 nm^3 (Figure 6e). The discrepancy in cluster volumes was likely caused by variance in labeling or probe detection. Indeed, we detected almost four times as many AF647 localizations as Cy3B localizations (9.0×10^4 versus 2.5×10^4). Hence, when performing one-color SMLM, choosing the optimal fluorescent label can improve cluster detection and quantification.

Additionally, we computed volumetric overlap between AF647 clusters and Cy3B clusters (Figure 6f–g). To our knowledge, our software is the first to offer this functionality for two-color, 3D SMLM data. We found that 31% of the total AF647 cluster volume overlapped with Cy3B clusters, whereas 50% of the total Cy3B cluster volume overlapped with AF647 clusters. The comparatively low overlap of AF647 with Cy3B was likely due to weaker labeling or detection with Cy3B than AF647. The Jaccard index was 0.24. In sum, our results for LAMP-1 clearly show that StormGraph can identify and quantify clusters of localizations in 3D SMLM ROIs and, furthermore, that it can detect overlap between 3D clusters in two-color data despite experimental limitations.

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

Discussion

By converting 2D or 3D SMLM localization data into a neighborhood graph, StormGraph leverages concepts from graph theory, especially community detection, to assign localizations to individual clusters that can be quantified. It enables analysis of clustering at multiple scales within datasets by generating a hierarchical clustering, but it also provides a single-level clustering to simplify interpretation of results. The StormGraph can be run in MATLAB using either a script or a simple graphical user interface. The software automatically quantifies clusters and it includes MATLAB functions for data visualization in 2D or 3D. *The software and a user manual are available from the authors on request.*

StormGraph has three optional, user-definable parameters. If by visual inspection the vast majority ($>$ approximately 90%) of localizations are organized into clear, well-separated clusters, then all three user-definable parameters could be discarded from StormGraph. Otherwise, we provide guidelines for their selection in the Methods. Each parameter specifies either a number of localizations (k and MinCluSize) or a probability (α) and can therefore be set without any knowledge of the scale or density of the localization data. StormGraph then adaptively determines scale- and density-dependent thresholds from the data using k, or automatically without k, and using α respectively. This allows disparate datasets to be analyzed using identical parameters, which increases both the objectivity and, by means of batch processing, the potential throughput of cluster analysis. This is in sharp contrast to DBSCAN, where two user-defined parameters explicitly define a threshold density.

Another distinguishing feature of StormGraph is its full utilization of individual localizations' positional uncertainties, when available. This is important because these uncertainties locally determine the minimum scale at which clusters can be reliably resolved. Only StormGraph and Bayesian methods (29; 30) have this feature in addition to being suitable for batch processing of heterogeneous ROIs. However, StormGraph has significant advantages over the Bayesian methods. First and foremost, it has superior computational efficiency. On a standard desktop computer, StormGraph analyzed a 2D ROI containing 5,349 localizations in less than 40 seconds and a 2D ROI containing 26,941 localizations in less than 3 minutes. The largest ROIs that we analyzed using StormGraph contained more than 10^5 localizations. On the other hand, the 2D Bayesian method took $\sim$50 minutes to analyze the 5,349-localization ROI and it failed to analyze the 26,941-localization ROI due to memory limitations. The second notable advantage of StormGraph is that, unlike the Bayesian methods, it makes no assumptions about the shapes of clusters. These advantages make StormGraph widely applicable and make it the standout choice for cluster analysis of SMLM data.

Using simulated data, we demonstrated that StormGraph is superior to both DBSCAN and the most

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

recent algorithm based on Voronoi diagrams, ClusterViSu, at assigning localizations to clusters. We also applied StormGraph to actual dSTORM data. This revealed the presence of many small BCR clusters, in addition to the expected large clusters, on the cell membranes of activated B cells. Our discoveries here highlight the importance of being able to objectively analyze nanoscale protein clustering in heterogeneous samples. By providing improved, high-throughput quantitative characterization of nanoscale receptor clustering, StormGraph should enable new insights into the relationship between receptor clustering and receptor signaling.

It should be noted that measurement errors generally cause clusters to appear slightly larger than the true underlying molecular clusters, and they can also cause over- or under-estimation of cluster overlap for two-color data. StormGraph does not correct for this during cluster quantification. However, for data with approximately Gaussian clusters, mathematical correction methods (44) could be applied to StormGraph's clusters prior to quantification. Nonetheless, we believe that StormGraph will advance cluster analysis in the SMLM field thanks to its generality, its ability to utilize information about localization uncertainties, and its potential to increase the throughput of single-molecule localization cluster analysis via batch processing of heterogeneous datasets.

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

Methods

Calculation of the length scale r_0

(1) The fully automatic, heuristic method

To automatically determine a length scale r_0 without user input, we implement a variation of the elbow method heuristic. For values of ε ranging from 0 to a sufficiently large value based on the optimal affinity scale stated by Arias-Castro (45), we construct the ε-neighborhood graph for the data. We then plot the number of connected components (including singletons) against ε. This must be monotonically decreasing and typically bears resemblance to a decaying exponential or logistic function. As ε increases, an "elbow" region occurs as rapid linking of nodes within clusters at small values of ε transitions to slower linking of distinct clusters and dispersed nodes at larger values of ε. Eventually all nodes would belong to a single connected component.

Sometimes, a natural number of clusters will be evident as a horizontal (i.e. constant) plateau occurring at > 1 connected component in this plot. In such cases, we find the plateau corresponding to the largest fold increase in the area or volume of the ε-neighborhood. Let ε_1 be the value of ε at the start of this plateau, and let $\varepsilon_2 = 2^{1/d}\varepsilon_1$, where d is the dimensionality of the data, be chosen such that the ε_2-neighborhood is twice the area or volume of the ε_1-neighborhood. If the ε_1- and ε_2-neighborhood graphs have the same number of connected components, then we set $r_0 = \varepsilon_2$ (Figure S1).

Otherwise, we fit a curve $f(\varepsilon)$ to the number of connected components versus ε (Figure S1). We choose $f(\varepsilon)$ to be the sum of a constant b and either one or two generalized logistic functions of the form

$$L(\varepsilon) = \frac{a}{(1 + \exp(s(\varepsilon - \varepsilon_0)))^{1/\nu}},$$

where $b \geq 0$, $a \geq 0$, $s \geq 0$, $\nu > 0$, and ε_0 are coefficients to be fit. To avoid overfitting, we only include the second logistic function if it yields a substantial improvement in the goodness of fit and we restrict its allowable values of ν. The elbow of this curve is not mathematically well defined, but intuitively it is related to the concavity: the curve achieves maximum (positive) concavity as it approaches the elbow region, and then its concavity decreases as it traverses the elbow region. StormGraph chooses the length scale r_0 to be towards the end of the elbow region as follows. Let $\varepsilon_{\max}$ be the value of ε at which $f''(\varepsilon)$, the concavity of $f(\varepsilon)$, is maximized. StormGraph sets r_0 to be the value of $\varepsilon > \varepsilon_{\max}$ where $f''(\varepsilon)$ first falls below 2% of its maximum value (Figure S1).

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

When localization uncertainties are available in the data, they are initially excluded when utilizing the elbow method to set the initial length scale r_0, which is used for classifying localizations as either clustered or unclustered. The uncertainties are subsequently taken into account during the final use of the elbow method, which sets the value of r_0 that is used for construction of the final graph following elimination of unclustered localizations. Specifically, the graph in which we count the number of connected components for a given ε is constructed from Monte Carlo simulated realizations of the data with two nodes connected to each other by an edge if and only if they are within a distance ε of each other in at least 75% of the Monte Carlo simulations. Note that edge weights are not relevant here because they do not affect the number of connected components.

(2) The kNN method

To determine the length scale r_0 for a selected ROI using a k-nearest neighbors (kNN) approach, StormGraph first finds the distance of every point in the ROI to its kth nearest neighbor. If localization uncertainties are available in the data, this is performed for 100 Monte Carlo simulated realizations of the data, and the 95% confidence level for the kth nearest neighbor distance is obtained for every localization. The distribution of kth nearest neighbor distances is also obtained for Monte Carlo simulations of random data with the same global average point density as the ROI. A histogram of kth nearest neighbor distances should initially increase more rapidly for clustered data than for random data, but the histograms for clustered and random data will eventually intersect each other (Figure S1). Inspired by the automated version of ClusterViSu (25), StormGraph defines r_0 as the distance at which these histograms of kth nearest neighbor distances first intersect. Points closer than r_0 to their k^{th} nearest neighbor are more likely to exist in clustered data, while points farther than r_0 from their k^{th} nearest neighbor are more likely to exist in random data. Moreover, points in clusters will tend to have more than k neighbors within a distance r_0, while randomly distributed points will tend to have fewer than k neighbors within a distance r_0. However, if this first histogram intersection occurs after the median of the random data's histogram, this indicates that, on average, the real data is actually more dispersed than the random data, and in this case StormGraph defines r_0 simply as the median of the random data's k^{th} nearest neighbor distances.

Simulating multiple data realizations and calculation of graph edge weights

StormGraph uses Monte Carlo simulations to simulate multiple realizations of the data by resampling each localization's coordinates. The new x, y and, if applicable, z coordinates for a particular localization are

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

drawn independently from normal distributions centered at the original observed localization position. The standard deviations are equal to the corresponding uncertainties recorded in the data. StormGraph then determines the graph edge weights $W_{ij} = \langle s_{ij} \rangle$ from the Monte Carlo simulations by calculating $\langle s_{ij} \rangle$ to be the mean of the simulated values of s_{ij} for each specific node pair $\{i, j\}$.

Thresholding of node degrees to eliminate unclustered nodes

Setting $\alpha = 1$ skips the thresholding step altogether, allowing all nodes to be considered for clustering. Otherwise, to set the node-degree threshold, StormGraph first constructs r_0-neighborhood graphs with edge weights s_{ij} for simulated random point clouds with the same global average point density as the SMLM data. For 2D data (and for 3D data with uniform axial acquisition), the random points are uniformly distributed in x and y (and z). Then StormGraph sets the degree threshold as the $((1 - \alpha) \times 100)^{\text{th}}$ percentile of the aggregated degree distribution of the random simulations. For 3D data with localizations concentrated around a focal plane, StormGraph simulates random data with z-coordinates that are distributed normally with the same interquartile range as the data. StormGraph then obtains a z-dependent node-degree threshold by fitting a Gaussian curve to node degree versus z for the simulated random points and finding the $(1 - \alpha) \times 100\%$ confidence upper bound curve. Thus, for both 2D and 3D data, an expected $\alpha \times 100\%$ of nodes in any of the random simulations would have degrees exceeding the threshold.

For actual data, because the edge weights are calculated by averaging s_{ij} over Monte Carlo simulations, the number of localizations that would be classified as clustered in random data would usually be less than $\alpha \times 100\%$. Hence, this averaging using localization uncertainties reduces the detection of spurious, small clusters arising from random spatial fluctuations in density.

If localization uncertainties are not known, then we take a different approach to reduce detection of spurious clusters. Preliminary clusters are defined using a community detection algorithm. A node is then classified as unclustered if it meets any of the following four criteria: (1) it belongs to a preliminary cluster whose mean degree is below the threshold; (2) its own degree is below the threshold and is also a lower outlier ($<$ lower quartile (LQ) $- 1.5 \times$ interquartile range (IQR)) for its preliminary cluster; (3) its own degree passes the threshold but is a strong lower outlier ($<$ LQ $- 3 \times$ IQR) for its preliminary cluster; (4) its own degree is less than half of the threshold. The first criterion provides robustness by spatially averaging node degrees over small areas. This prevents the inclusion of spurious, small clusters. The other three criteria prevent the inclusion of nodes that are visually separate from a cluster but still within a distance r_0 of one.

To avoid biases arising from the choice of algorithm used for the preliminary clustering, StormGraph

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

performs this twice, independently, using two different community detection algorithms, and it then classifies nodes as unclustered if either method does. The two algorithms used are the two-level version of Infomap (46) and the Louvain method (47), which are two of the top performing community detection algorithms (32). Infomap is an information theoretic algorithm based on flow on the graph, while the Louvain method is one of several algorithms that aims to maximize a property of the graph called "modularity". See Supplementary Note 1 for further technical details.

Edge pruning

When localization uncertainties are used in the StormGraph algorithm, we prune edges from the final graph that is constructed from only the nodes that are retained after thresholding node degrees. To do this, we delete every edge that has nonzero s_{ij} in less than 75% of the Monte Carlo simulations that were used to calculate the edge weights. This guarantees that any pair of retained edges have at least an estimated 50% probability of co-occurring in the r_0-neighborhood graph for any realization of the data, and the unknown true localization positions is one possible realization. In turn, this prevents the linking of clusters that are disconnected in most realizations of the r_0-neighborhood graph but connected in the average graph.

Merging clusters at the top of the multi-level Infomap hierarchy

To facilitate the identification and quantification of particularly large clusters, StormGraph creates an additional level at the top of the multi-level Infomap cluster hierarchy, if possible, by merging sufficiently interconnected clusters. It is natural to consider the connected components of a graph to be the clusters at the coarsest level of a cluster hierarchy. We therefore use this concept to define the top level of StormGraph's cluster hierarchy by merging Infomap clusters that form connected components. However, due to the uncertainties in SMLM data, StormGraph only merges clusters if they form stable connected components, which we define as connected components that would remain connected following the random removal or displacement of any one node. Oftentimes, this step results in no merging of clusters and so no additional level of clustering is created.

Algorithm to obtain single-level clustering from cluster hierarchy

Although various methods exist to select one level from a cluster hierarchy, for example silhouette scores (48) and the gap statistic (49), existing methods are either very computationally intensive or otherwise

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

incompatible with StormGraph. We therefore developed our own fast algorithm to obtain a single-level clustering from the cluster hierarchy output by StormGraph, which we describe here.

The hierarchical clustering output by StormGraph is generated from an r_0-neighborhood graph. An alternative type of graph commonly used for clustering problems is the symmetric k-nearest neighbor (kNN) graph, in which two nodes are connected by an edge if either of them is among the k nearest neighbors of the other. A related graph is the mutual kNN graph, a subgraph of the symmetric kNN graph, in which two nodes are connected by an edge if and only if each node is among the k nearest neighbors of the other. One simple clustering algorithm would be to identify the connected components in a symmetric kNN graph or in a mutual kNN graph, where k is an adjustable parameter.

In a symmetric kNN graph, it is guaranteed that every node has at least k edges. However, as k increases, nodes in low-density regions between two distinct clusters quickly become connected to both clusters, while the high-density regions inside the clusters may remain fragmented into multiple connected components until higher values of k. A mutual kNN graph, in which every node is guaranteed to have at most k edges, more faithfully represents such clusters by preventing nodes in low-density regions from making too many connections. However, mutual kNN graphs often suffer from having singletons and small connected components due to the weak connectivity. We therefore chose to combine the concepts of both the symmetric kNN and mutual kNN graphs.

For a set of points V and positive integers M and $K > M$, we define $G_{M,K}(V)$ to be the union of the symmetric MNN graph and the mutual KNN graph for vertices V. This is still a subgraph of the symmetric KNN graph, but it has stronger connectivity than the mutual KNN graph by guaranteeing that every node has at least M edges, which in turn ensures that $G_{M,K}(V)$ contains no connected components with fewer than $(M+1)$ nodes.

For each cluster at the top level of the cluster hierarchy, StormGraph decides whether to split the cluster into its subclusters at the next level down in the hierarchy according to the algorithm described below. If the split is rejected, then StormGraph keeps the current cluster and does not examine any of the finer levels of the hierarchy within that cluster. If the split is accepted, then this process is repeated recursively for each of the newly accepted subclusters. A split is automatically rejected if more than 1% of the points in the cluster belong to subclusters with fewer than the minimum number of points, specified by the user, that constitute a cluster.

Let V be the set of nodes in a cluster C, let $A = \{C_1, C_2, \ldots, C_n\}$ be the set of n subclusters of C at the next finest level of the cluster hierarchy, and let $B(M,K) = \{C'_1, C'_2, \ldots, C'_{n'}\}$ be the set of n' connected

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

components of the graph $G_{M,K}(V)$. StormGraph decides whether to split cluster C into its constituent subclusters A using the following algorithm:

1. Construct $G_{2,K}(V)$ for all integers $K \in \{6, \ldots, K_1\}$, where K_1 is the smallest integer such that $G_{2,K_1}(V)$ is connected. We empirically chose the minimum value of K to be 6 because this usually results in randomly distributed points forming a single connected component.

2. Find the value of K for which $B(2, K)$ is most similar to A according to some measure of similarity. Denote this value of K by K^*.

3. Split cluster C into subclusters A if the similarity between A and $B(2, K^*)$ is greater than both a threshold similarity and the similarity between C and $B(2, K^*)$.

The most obvious choices for a similarity measure to score the similarity between two clusterings of the nodes V are normalized mutual information (NMI) (37) and mean F-measure (38). We require a similarity measure that is defined even if one of the clusterings being compared consists of only a single cluster. This eliminates NMI as a suitable choice, so we use mean F-measure.

Let $F(A, B)$ denote the similarity of clustering A to clustering B as measured by the mean F-measure. The F-measure or F_1 score for a binary classification problem in which a cluster C_i is compared to a reference cluster C'_i (usually the ground-truth cluster that the cluster C_i, found by a clustering algorithm, is supposed to recover) is defined as the harmonic mean of precision (P) and recall (R):

$$F_1(C_i, C'_i) = \frac{2 \cdot P(C_i, C'_i) \cdot R(C_i, C'_i)}{P(C_i, C'_i) + R(C_i, C'_i)}.$$

The precision $P(C_i, C'_i)$ is the fraction of C_i that belongs to C'_i, and the recall $R(C_i, C'_i)$ is the fraction of C'_i that belongs to C_i. The mean F-measure $F(A, B)$ is then defined as the weighted arithmetic mean of the maximum F-measures for each of the clusters C'_i in B:

$$F(A, B) = \frac{\sum_{i=1}^{n'} |C'_i| \max_{1 \leq j \leq n}\{F_1(C_j, C'_i)\}}{\sum_{i=1}^{n'} |C'_i|},$$

where $|C'_i|$ denotes the number of points in C'_i.

The mean F-measure is not symmetric, i.e. $F(A, B) \neq F(B, A)$, which is not desirable in our situation where we wish to compare two clusterings, neither of which is necessarily ground-truth. To avoid having to choose one of the clusterings A and B to be the reference, we define a symmetric similarity measure,

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

$\overline{F}(A, B)$, as the arithmetic mean of $F(A, B)$ and $F(B, A)$:

$$\overline{F}(A, B) = \frac{1}{2}\left(F(A, B) + F(B, A)\right).$$

This is the similarity measure that we use in our algorithm for obtaining a single-level clustering from the hierarchy. It ranges from 0 to 1, and $\overline{F}(A, B) = 1$ if and only if A and B are identical. We impose a minimum similarity score of $\overline{F}_{\min} = 0.8$ for a cluster split to be considered. Thus, we split cluster C into its highest level of subclusters, A, if A is at least 80% similar to $B(M, K^*)$ and is also a closer match to $B(M, K^*)$ than the single, unified cluster C is. The 80% similarity threshold prevents the fragmentation of a cluster if there is not substantial consensus between the two independent subclusterings. This threshold could be tuned to make it more or less difficult to split a cluster into finer levels of subclusters. In particular, a threshold of $\overline{F}_{\min} = 1$ would demand perfect agreement between the subclusters of C and the alternative, independent clustering $B(M, K^*)$ for the subclusters to be accepted as a better clustering of V than a single cluster. We chose a threshold of 0.8 to allow some leniency.

Identifying clusters that can be confidently distinguished from multiply counted single molecules

Localizations arising from multiply counted single molecules may be falsely identified as clusters. As an optional step during StormGraph analysis, clusters of localizations that cannot be distinguished with high confidence, due to their positional uncertainties, from multiply counted single molecules can be identified and subsequently reclassified as unclustered (cluster label 0). To do this, StormGraph checks each cluster systematically as follows.

First, for each pair of localizations, $\mathbf{X^i} = (x^i, y^i, z^i)$ and $\mathbf{X^j} = (x^j, y^j, z^j)$, in the cluster, let $\mathbf{Y^{ij}} = \mathbf{X^i} - \mathbf{X^j}$ be their vector difference, and let $\mathbf{\Sigma^{ij}}$ be the covariance matrix for the coordinates of $\mathbf{Y^{ij}}$. The off-diagonal elements of $\mathbf{\Sigma^{ij}}$ are assumed to all be 0 (i.e. the uncertainty in each coordinate of a localization is assumed to be independent of its other coordinates). Assuming each molecule to be approximated by a point particle of zero size, the mth diagonal element of $\mathbf{\Sigma^{ij}}$ is $V_m^{ij} = {\sigma_m^i}^2 + {\sigma_m^j}^2$, where σ_m^i denotes the standard deviation for the uncertainty in the mth coordinate of localization i, as given by the input data.

This assumes that the true position is identical for all localizations originating from the same molecule. In practice, the fluorophore positions may be different from the actual molecule positions. For example, when molecules are detected using antibodies, the fluorophore conjugated to the antibody may be located

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

as much as 10 nm away from the antibody's binding site. In addition, if each molecule can be labeled by more than one fluorophore, then the true positions of localizations originating from a single molecule will not only be different from the actual molecule but also from each other. If the sizes of the molecule and fluorescent label are not negligible, they can be approximately taken into account in the following way. For mathematical simplicity, we approximate the uncertainty due to the molecule and label size as an isotropic Gaussian distribution with variance $(r/3)^2$, where r is the effective radius of the molecule and fluorescent label combined, which is specified by the user based on underlying biophysical knowledge. We then add this variance term twice (once each for localizations $\mathbf{X^i}$ and $\mathbf{X^j}$) to each of the diagonal elements in $\mathbf{\Sigma^{ij}}$. For our simulated data, this was not necessary as the true position of every localization was at the centre of a simulated molecule. For our BCR dSTORM data, we used $r = 8$ nm.

Next, we construct the statistic $Z^{ij} = \sum_{m=1}^{d} {Y_m^{ij}}^2 / V_m^{ij}$ for each pair of localizations, where d is the number of dimensions (2 or 3) and Y_m^{ij} denotes the mth coordinate of the vector $\mathbf{Y^{ij}}$. If two localizations $\mathbf{X^i}$ and $\mathbf{X^j}$ have the same true position, then Z^{ij} is chi-squared distributed with d degrees of freedom. We then look for pairs of localizations for which Z^{ij} exceeds a desired quantile of the appropriate chi-squared distribution, indicating confidence that they originated from different molecules. Because we are testing multiple pairs of localizations for significance, we correct for multiple hypothesis testing using the Šidák correction. If we desire a significance level of $1 - q$, then we look for pairs of localizations for which Z_{ij} exceeds the $\left(q^{1/N}\right)^{\text{th}}$ quantile of the chi-squared distribution with d degrees of freedom. Here, N is the number of localizations in the cluster. Even though there are $N(N-1)$ pairs of localizations, the null hypotheses are that each localization originated from the same molecule as all other localizations in the cluster, and so there are only N hypotheses to test. By default, StormGraph uses a significance level of 0.05, so it uses the $\left(0.95^{1/N}\right)^{\text{th}}$ quantile. Finally, since a cluster must always contain at least three localizations (we do not consider pairs of localizations to be clusters), StormGraph increases confidence further by demanding that at least two localizations are each, probabilistically, sufficiently far from at least two other localizations. This way, a single outlying localization within a cluster is not sufficient on its own to qualify the cluster as containing multiple molecules with high confidence.

Guidelines for StormGraph parameter selection

StormGraph has three user-controllable parameters. The parameter α controls the node-degree threshold used to identify and remove unclustered nodes prior to clustering. For data that does not suffer from overcounting of molecules, or for which overcounting has already been corrected, α is effectively the maximum

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

false positive rate (FPR) for classifying localizations as clustered if all localizations in a random distribution should be classified as unclustered. When overcounting is present in the data, the FPR may be greater than α. Nevertheless, for any given $\alpha < 1$, StormGraph takes steps to minimize the FPR as far as possible. Hence, we suggest setting α as the maximum fraction of localizations that the user would accept as being clustered if they were completely randomly distributed. For most applications, we recommend $\alpha = 0.05$, the default value. Larger values of α may be suitable if the user is already confident that the localizations are strongly clustered but there is large variation in the density of clusters. For example, $\alpha = 0.5$ would simply demand that clusters are at least as dense as the average density of a random distribution, but this could result in as many as 50% of localizations in a random distribution qualifying as clustered. Alternatively, the user can choose to skip the thresholding step and instead allow all localizations to be possibly assigned to clusters by setting $\alpha = 1$, which ultimately removes all use of α from the StormGraph algorithm.

The optional parameter k specifies the number of nearest neighbors to use when calculating the graph neighborhood radius r_0. The value of k, if set, is the minimum (respectively maximum) number of neighbors that most clustered (respectively unclustered) localizations should have. It should be smaller than the number of localizations in a typical cluster, but preferably larger than the estimated number of times that a typical single molecule might blink. These values can be estimated by visual inspection of localization clusters within cell boundaries and on the coverslip outside of cells. Increasing k, and consequently r_0, can influence the exact placement of cluster boundaries, and hence cluster quantification, by allowing more low-density localizations on the periphery of clusters to be included in the clusters. This highlights the inherent ambiguity in clustering problems, which results from the lack of a clear definition of a cluster. Nonetheless, we found values of k between 10 and 20 to be generally appropriate. Alternatively, if k is not set, StormGraph will determine r_0 heuristically without any user input.

Finally, the user can optionally set the minimum number of localizations that a cluster must contain, MinCluSize. One possible strategy for setting its value is to investigate background regions outside of cells, where most clusters of localizations are likely to be due to individual fluorescent labels stuck to the coverslip, and assess how many localizations are typical of these apparent clusters. However, because StormGraph provides an option to use localization uncertainties to identify and reclassify localization clusters that could have arisen just from overcounting of single molecules, clusters that could be due to single molecules can be automatically removed from analysis without the need for a minimum cluster size parameter. Note that StormGraph requires all clusters to contain at least three localizations, even if MinCluSize is not set.

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

Computational approximations in StormGraph

In order to improve computational efficiency, StormGraph includes some computational approximations. Firstly, neighborhood searches about each node are performed using the MATLAB function "rangesearch", which uses a k-d tree, as this is faster than computing distances between all pairs of nodes. Without uncertainties in localization positions, rangesearch is implemented with a search radius of r_0. However, when Monte Carlo simulations are used to perturb localization positions using their uncertainties, it is inefficient to perform rangesearch for every simulation. Instead, we perform rangesearch just once, using an expanded search radius, to identify candidate edges for the graph. StormGraph then calculates expected edge weights only for the candidate edges. Since the computational time for rangesearch increases as the search radius increases, we chose ($r_0 + 6 \times$ mean localization uncertainty) as the expanded search radius because most pairs of nodes separated by distances greater than this would have only negligible or zero edge weights anyway. Increasing the search radius further would not only make rangesearch slower, but it could also add more edges to the graph and consequently increase the computational cost of community detection, even though the additional edges would be mostly negligible.

Secondly, StormGraph limits nodes to having no more than 500 neighbors in the graph. This is to prevent extremely dense, large clusters from dramatically slowing down community detection, since the computational time required by Infomap scales with the number of edges in the graph. In practice, for reasonably chosen values of k, e.g. in the range from 10 to 20, and for r_0 values determined heuristically, very few nodes, if any, in most datasets should have this many neighbors.

Lastly, we note that StormGraph is not deterministic, meaning that it can give slightly different results each time that it is run. This is for two reasons. The first reason is because StormGraph uses Infomap or the Louvain method to perform community detection. Infomap seeks to optimize the map equation and the Louvain method seeks to optimize modularity. In both cases, the full optimization problem is NP-hard. Therefore, both methods take a greedy approach to the optimization, which generally finds a local, but not necessarily global, optimum. They then select the best optimum from multiple iterations started from random initiations. In StormGraph, the default number of iterations used for finding the final cluster hierarchy is 50. Results can be improved at the expense of increasing computational time by increasing the number of iterations. Conversely, computational time can be reduced at the expense of cluster accuracy by decreasing the number of iterations. The second reason for slight variability in results is the use of Monte Carlo simulations by StormGraph. This variability can be decreased, again at the expense of increasing computational cost, by increasing the number of Monte Carlo simulations.

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

The non-deterministic nature of StormGraph is only a minor drawback, as variability in clustering results for a single dataset is small. To demonstrate this, we repeatedly applied StormGraph using identical settings to a heterogeneous dSTORM ROI containing visually ambiguous clusters. We did this in both 2D and 3D and for both the automatic and kNN methods for determining r_0, each time generating 11 StormGraph repeats. We then assessed the similarity of cluster assignments from each of the last 10 repeats to the first one using NMI, which can range from 0 to 1. We always achieved NMI > 0.94, indicating very high similarity (Figure S11).

Simulating SMLM data in 2D and 3D

In both 2D and 3D, we distributed 3000 molecules into circular nanoclusters with a fixed radius, r, and fixed molecular density, ρ. Each molecule was assigned uncertainties, which were sampled randomly from a real dSTORM dataset, in its x-, y- and (for 3D) z-coordinates and a number of blinks, which was drawn from a geometric distribution (50) supported on $\{1, 2, 3, ...\}$ with success probability parameter λ. Within each nanocluster, molecules were distributed uniformly at random, and for each molecule the observed localizations (blinks) were drawn from a normal distribution with mean equal to the molecule's position and standard deviations equal to the uncertainties assigned to the molecule. Every observed localization was assigned the same uncertainties as its associated molecule. The total number of nanoclusters, N_{nano}, was determined by the total number of molecules in clusters (3000) and the density, ρ, of molecules within clusters.

The nanoclusters were positioned inside a 2 µm × 2 µm ROI in 2D or a 2 µm × 2 µm × 1 µm ROI in 3D such that some existed as isolated nanoclusters and others were randomly aggregated into larger clusters according to the following process, which was adapted from a Dirichlet process: for i from 1 to N_{nano}, draw a random number from the uniform distribution on [0,1]; if it is less than or equal to $((p + 10)/(p + i - 1))^q$ for positive integers p and q, then place the ith nanocluster away from existing clusters; otherwise, add the ith nanocluster to a randomly selected existing cluster, excluding the first 10 nanoclusters that were placed. If a nanocluster was added to an existing cluster, it was placed such that its centre was exactly a distance $2r$ from the centre of another nanocluster in the same aggregate cluster, and without overlapping with any other existing nanoclusters in the aggregate cluster.

This process ensures that there are at least 10 isolated nanoclusters and a variable number of larger aggregate clusters of variable size, thus creating heterogeneous clusters. The heterogeneity is controlled by the parameters p and q. In our simulations, we fixed $p = 5$ and varied q from 1 to 5, with larger values of q

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

resulting in larger (and fewer) cluster aggregates. Outside of the clusters, we added molecules uniformly at random at a specified average density, and the number and positions of observed localizations corresponding to each of these background molecules were drawn from geometric and normal distributions respectively, as described for the in-cluster localizations.

If the simulations were performed in 3D, points were then randomly removed such that the probability of a localization being observed in the final simulated data decayed according to a Gaussian profile as the axial distance from a central focal plane increased. This was to imitate the realistic scenario for most 3D SMLM techniques in which fluorescent blink events are more likely to be collected and localized the closer they are to the focal plane.

We generated 64 2D datasets with multiple blinking of molecules (e.g. Figures 2a(i) and S5) by varying the following parameters: (1) the radius of the nanoclusters (20 nm, 30 nm or 50 nm), (2) the density of clustered molecules (0.01 nm^{-2} or 0.02 nm^{-2}), (3) the density of the random molecules (1%, 5%, 10%, 20% or 40% of the within-cluster molecular density), (4) the average number of blinks per molecule (4/3, 2 or 4; these values provide examples ranging from cases in which most molecules blink only once to cases where the molecules could be bivalent and labeled by fluorophores that blink on average twice, which is typical for the photoactivatable fluorophore mEos2 (4; 51)), and (5) the propensity for nanoclusters to coalesce into larger aggregate clusters (parameter q).

We generated 130 3D datasets analogously but using within-cluster molecular densities of 1×10^{-4} nm^{-3} and 2×10^{-4} nm^{-3}. In 3D, we used nanoclusters of radii 30 nm and 50 nm, and we used densities of random, unclustered molecules equal to 1%, 5%, 10% or 20% of the within-cluster molecular density. At 20%, clusters were barely visible in 2D projections of the simulated 3D data onto the xy-plane.

Running ClusterViSu on simulated data

The ClusterViSu algorithm consists of running a series of two functions provided as part of its source code, specifically the functions "VoronoiMonteCarlo" and "VoronoiSegmentation". However, the authors did not provide a script for running ClusterViSu. Hence, for users with zero programming expertise, it can only be run using a graphical user interface that requires each file to be loaded and analyzed separately. Also, ClusterViSu outputs the bounding polygon for each detected cluster but not the actual cluster assignments of the localizations, which we needed to compute NMI and mean F-measure scores for assessing the performance of cluster assignment. Therefore, we wrote our own custom MATLAB script (available upon request) to run and batch process ClusterViSu from its source code and subsequently determine the cluster assignments of

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

the localizations. In addition, we found that ClusterViSu prefers input ROIs to be at least 18 μm × 18 μm, so we rescaled our 2 μm × 2 μm simulated data by a factor of 9, which drastically improved ClusterViSu's performance, at least in terms of computational time.

Furthermore, we only included ClusterViSu results for simulated datasets on which ClusterViSu analysis completed in under 2 hours. This resulted in 15 out of 64 simulated datasets being excluded from our summary of test results for ClusterViSu, but these 15 datasets were still included for assessing StormGraph and DBSCAN. However, these 15 datasets were excluded in Figures 2c and S3b, where NMI or mean F-measure results for StormGraph and DBSCAN are shown as a ratio to the NMI or mean F-measure results for ClusterViSu.

Functionalization of glass coverslips for cell adherence

Glass coverslips were cleaned and functionalized as previously described (52). Briefly, acid-cleaned glass coverslips (Marienfeld #1.5H, 18 mm × 18 mm; catalogue #0107032, Lauda-Königshofen, Germany) were incubated with 0.01% poly-L-lysine (Sigma-Aldrich; catalogue #P4707) or 0.25 μg/cm^3 of the non-stimulatory M5/114 anti-MHCII monoclonal antibody (Millipore; catalogue #MABF33) or 2 μg/cm^2 fibronectin (Sigma Aldrich; catalogue #F4759) for at least 3 h at 37 °C. The slides were then washed with phosphate-buffered saline (PBS) prior to being used for experiments.

Monovalent Fab fragments and antibodies

The anti-mouse-Igκ antibody for clustering BCRs was purchased from Southern Biotech (Birmingham, AL; catalogue #1050-01). AF647-conjugated anti-mouse-IgM Fab fragments (catalogue #115-607-020) and AF647-conjugated anti-human-IgM Fab fragments (catalogue #109-607-043) were from Jackson ImmunoResearch Laboratories (West Grove, PA). All Fab fragments were routinely tested for aggregation using dynamic light scattering (Zetasizer Nano) and unimodal size distributions were observed. Anti-LAMP-1 antibody was purchased from Abcam (catalogue #ab24170). AF647-conjugated goat anti-mouse-IgG (catalogue #A21236) and AF647-conjugated goat anti-rabbit-IgG (catalogue #A21244) were purchased from ThermoFisher Scientific. Goat anti-mouse-IgG (Jackson ImmunoResearch Laboratories; catalogue #115-005-008) and goat anti-rabbit-IgG (Jackson ImmunoResearch Laboratories; catalogue #111-001-008) were conjugated to Cy3B using a Pierce antibody conjugation kit (catalogue #44985).

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

Cell labeling for dSTORM

(1) Murine splenic B cells

Animal protocols were approved by the University of British Columbia and all animal experiments were carried out in accordance with institutional regulations. Splenic B cells were obtained from 6- to 10-week old C57BL/6 mice (Jackson Laboratory) of either sex using a B-cell isolation kit (Stemcell Technologies; catalogue #19854) to deplete non-B cells. To induce IgM-BCR clustering, 5×10^6 *ex vivo* splenic B cells/mL were stimulated with 20 μg/mL anti-Igκ in PBS for 10 min at 37 °C. A similar volume of PBS was added to control samples (resting B cells). All subsequent procedures were performed at 4 °C. Cells were washed three times with ice-cold PBS, and IgM-BCRs on the cell surface were labeled using AF647-conjugated, monovalent anti-mouse-IgM Fab fragments for 15 min. These Fab fragments bind to the constant region of the μ heavy chain of IgM-BCRs, which is distinct from sites on the IgM-BCR that the anti-Igκ treatment antibody binds to. Following multiple PBS washes, cells were settled onto pre-cooled anti-MHCII-functionalized coverslips for 10 min and subsequently fixed with PBS containing 4% paraformaldehyde and 0.2% glutaraldehyde for 90 min. The coverslips were washed thoroughly with PBS and fiducial markers (100 nm diameter; ThermoFisher Scientific, catalogue #F8799) were allowed to settle onto the coverslip overnight at 4 °C. Unbound fiducial markers were removed by PBS washes and the stuck particles were used for real-time drift stabilization (53).

(2) Human and murine B-lymphoma cell lines

A20 and BJAB B-lymphoma cells were obtained from American Type Culture Collection (ATCC). HBL-1 cells were obtained from Dr. Izidore S. Lossos, Sylvester Comprehensive Cancer Center, University of Miami (Miami, FL). TMD8 cells were a gift from Dr. Neetu Gupta, Lerner Research Institute, Cleveland Clinic (Cleveland, OH). All B-cell lines were cultured in RPMI-1640 (Life Technologies; catalogue #21870-076), supplemented with 10% heat-inactivated fetal bovine serum, 2 mM L-glutamine, 50 μM β-mercaptoethanol, 1 mM sodium pyruvate, 50 U/mL penicillin, and 50 μg/mL streptomycin (complete medium). All cell lines were authenticated by STR DNA profile analysis.

All staining procedures were performed at 4 °C. Cell-surface IgM-BCRs on BJAB, HBL-1 and TMD8 cells were labeled using AF647-conjugated anti-human-IgM Fab fragments for 15 min. Cell-surface IgG-BCRs on A20 cells (ATCC) were labeled using both AF647-conjugated anti-mouse-IgG and Cy3B-conjugated anti-mouse-IgG at 1:1 stoichiometry for 15 min. Fc receptors on A20 cells were blocked prior to staining using the 2.4G2 rat anti-Fcγ receptor monoclonal antibody. Cells were washed in PBS and subsequently fixed with

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

ice-cold PBS containing 4% paraformaldehyde and 0.2% glutaraldehyde for 60 min. Following multiple PBS washes, the cells were settled onto pre-cooled poly-L-lysine-coated coverslips for 15 min and subsequently fixed again for 30 min. The coverslips were washed thoroughly with PBS and fiducial markers were added and incubated overnight at 4 °C.

(3) B16 melanoma cell lines

B16F1 melanoma cells (ATCC) were grown in RPMI-1640 complete medium. Approximately 3×10^4 cells were seeded on fibronectin-coated coverslips for 1 h and fixed with PBS containing 4% paraformaldehyde for 30 min. Cells were permeabilized with 0.1% Triton X-100 for 10 min, washed with PBS, and incubated for 30 min at room temperature (RT) with Image-IT FX Signal Enhancer (Life Technologies, catalogue #I36933) to neutralize surface charge. Cells were washed briefly in PBS and then incubated with BlockAid blocking solution (Life Technologies; catalogue #B10710) for 1 h at RT. The cells were incubated with anti-LAMP-1 antibody (diluted in BlockAid) for 4 h at RT. Following PBS washes, cells were incubated with both AF647-conjugated anti-rabbit-IgG and Cy3B-conjugated anti-rabbit-IgG at 1:1 stoichiometry for 90 min. Cells were washed in PBS and subsequently fixed again with 4% paraformaldehyde for 10 min. The coverslips were washed thoroughly with PBS and fiducial markers were added and incubated overnight at 4 °C.

dSTORM

Imaging was performed using a custom-built microscope with a sample drift-stabilization system that has been described previously (53; 54). Briefly, three lasers were used in the excitation path. These were a 639 nm laser (Genesis MX639, Coherent) for exciting the AF647, a 532 nm laser (Laser quantum, Opus) for exciting the photo-switchable Cy3B, and a 405 nm laser (LRD 0405, Laserglow Technologies) for reactivating the AF647 and Cy3B. All three lasers were coupled into an inverted microscope equipped with an apochromatic TIRF oil-immersion objective lens (60x; NA 1.49; Nikon). The emission fluorescence was separated using appropriate dichroic mirrors and filters (Semrock) (53; 54), and detected by EM-CCD cameras (Ixon, Andor). A feedback loop was employed to lock the position of the sample during image acquisition using immobile fiducial markers. Sample drift was controlled to be less than 1 nm laterally and 2.5 nm axially.

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

dSTORM image acquisition and reconstruction

Imaging was performed in an oxygen-scavenging GLOX-thiol buffer consisting of 50 mM Tris-HCl, pH 8.0, 10 mM NaCl, 0.5 mg/ml glucose oxidase, 40 μg/ml catalase, 10% (w/v) glucose and 140 mM 2-mercaptoethanol (55). The coverslip with attached cells was mounted onto a depression slide filled with imaging buffer and sealed with Twinsil two-component silicone-glue (Picodent; catalogue #13001000).

For SMLM imaging, a laser power density of 1 kW/cm^2 for the 639 nm and 532 nm lasers was used to activate the AF647 and Cy3B, respectively. For each sample, 4×10^4 images were acquired for each color channel at 50 Hz. Localization coordinates and their associated uncertainties were computationally determined simultaneously by fitting a function to the intensity profile of each fluorescence event using MATLAB (Figure S12), as described previously (54). Expressed as standard deviations, lateral uncertainties were typically < 10 nm while axial uncertainties were typically < 40 nm (Figure S12).

For two-color SMLM, image acquisition was performed sequentially for each color with AF647 imaged first to prevent photobleaching by the Cy3B excitation laser. Two-color SMLM images were acquired using a beam splitter with appropriate filters to direct each signal to one of two independent cameras. Alignment of these two colors was carried out using $\sim 4 \times 10^4$ images of fluorescent beads simultaneously recorded at various positions to find an optimal geometric transformation. The resulting color-alignment error is $\sim$10 nm root mean squared.

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

Acknowledgements

We thank Alejandra Herrera-Reyes for helpful discussions and code, Ki Woong Sung for preliminary computational work, Dr. Vivian Qian Liu for assistance with dSTORM data fitting, Dr. Neetu Gupta for TMD8 cells, Dr. Izidore S. Lossos for HBL-1 cells, and Dr. David R.L. Scriven for helpful discussion. This work was supported by funding from the Natural Science and Engineering Research Council of Canada (Discovery Grant RGPIN-2015-04611 to DC, an Undergraduate Student Research Award to DWZ, Discovery Grant RGPIN-2014-03581 to KCC, and Discovery Grant RGPIN-2017-04862 to MRG), the Canadian Cancer Society Research Institute (Innovation Grant 704254 to DC and MRG), the Canadian Institutes of Health Research (PJT-19426 to MRG), and Canada Foundation for Innovation (to KCC).

Author contributions

JMS conceived the project, developed and tested the StormGraph algorithm and software, proposed experiments, performed data analysis, produced figures, and wrote the manuscript. LA proposed and performed experiments, wrote experimental methods, produced figures, and provided essential feedback about the algorithm and software. DWZ assisted with software development, simulation of data, and algorithm testing. RT built the dSTORM microscope and assisted with dSTORM data fitting and processing. KCC provided code for fitting dSTORM localizations and aligning two-color dSTORM data. MRG and DC supervised the project and wrote the manuscript. All authors approved the final manuscript.

Competing interests statement

The authors declare no competing interests.

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

References

[1] Heilemann, M. *et al.* Subdiffraction-resolution fluorescence imaging with conventional fluorescent probes. *Angew Chem Int Ed Engl* **47**, 6172–6176 (2008).

[2] van de Linde, S. *et al.* Direct stochastic optical reconstruction microscopy with standard fluorescent probes. *Nat Protoc* **6**, 991–1009 (2011).

[3] Betzig, E. *et al.* Imaging intracellular fluorescent proteins at nanometer resolution. *Science* **313**, 1642–1645 (2006).

[4] Shivanandan, A., Deschout, H., Scarselli, M. & Radenovic, A. Challenges in quantitative single molecule localization microscopy. *FEBS Letters* **588**, 3595–3602 (2014).

[5] Thompson, R., Larson, D. & Webb, W. Precise nanometer localization analysis for individual fluorescent probes. *Biophys. J.* **82**, 2775–2783 (2002).

[6] Mortensen, K., Churchman, L., Spudich, J. & Flyvbjerg, H. Optimized localization analysis for single-molecule tracking and super-resolution microscopy. *Nat. Methods* **7**, 377–381 (2010).

[7] Juette, M. *et al.* Three-dimensional sub–100 nm resolution fluorescence microscopy of thick samples. *Nat. Methods* **5**, 527–529 (2008).

[8] Huang, B., Wang, W., Bates, M. & Zhuang, X. Three-dimensional super-resolution imaging by stochastic optical reconstruction microscopy. *Science* **319**, 810–813 (2008).

[9] Pavani, S. *et al.* Three-dimensional, single-molecule fluorescence imaging beyond the diffraction limit by using a double-helix point spread function. *Proc. Natl. Acad. Sci.* **106**, 2995–2999 (2009).

[10] Aquino, D. *et al.* Two-color nanoscopy of three-dimensional volumes by 4Pi detection of stochastically switched fluorophores. *Nat. Methods* **8**, 353–359 (2011).

[11] Owen, D. *et al.* PALM imaging and cluster analysis of protein heterogeneity at the cell surface. *J. Biophotonics* **3**, 446–454 (2010).

[12] Mattila, P. *et al.* The actin and tetraspanin networks organize receptor nanoclusters to regulate B cell receptor-mediated signaling. *Immunity* **38**, 461–474 (2013).

[13] Grove, J. *et al.* Flat clathrin lattices: stable features of the plasma membrane. *Molecular Biology of the Cell* **25**, 3581–94 (2014).

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

[14] Pollitt, A. *et al.* Syk and Src family kinases regulate C-type lectin receptor 2 (CLEC-2)-mediated clustering of podoplanin and platelet adhesion to lymphatic endothelial cells. *The Journal of Biological Chemistry* **289**, 35695–710 (2014).

[15] Hartley, J. *et al.* Super-resolution imaging and quantitative analysis of membrane protein/lipid raft clustering mediated by cell-surface self-assembly of hybrid nanoconjugates. *Chembiochem* **16**, 1725–9 (2015).

[16] Roh, K.-H., Lillemeier, B. F., Wang, F. & Davis, M. M. The coreceptor CD4 is expressed in distinct nanoclusters and does not colocalize with T-cell receptor and active protein tyrosine kinase p56lck. *Proceedings of the National Academy of Sciences* **112**, E1604–E1613 (2015).

[17] Gao, L., Chen, J., Gao, J., Wang, H. & Xiong, W. Super-resolution microscopy reveals the insulin-resistance-regulated reorganization of GLUT4 on plasma membranes. *Journal of Cell Science* **130**, 396–405 (2017).

[18] Rahbek-Clemmensen, T. *et al.* Super-resolution microscopy reveals functional organization of dopamine transporters into cholesterol and neuronal activity-dependent nanodomains. *Nature Communications* **8**, 740 (2017).

[19] Eryilmaz, M. *et al.* Localization microscopy analyses of MRE11 clusters in 3D-conserved cell nuclei of different cell lines. *Cancers* **10**, 25 (2018).

[20] Gao, J. *et al.* Revealing the cellular localization of STAT1 during the cell cycle by super-resolution imaging. *Scientific Reports* **5**, 9045 (2015).

[21] Ripley, B. Modelling spatial patterns. *J. R. Stat. Soc. Series B Stat. Methodol.* **39**, 172–192 (1977).

[22] Kiskowski, M., Hancock, J. & Kenworthy, A. On the use of Ripley's K-function and its derivatives to analyze domain size. *Biophys J.* **97**, 1095–1103 (2009).

[23] Ester, M., Kriegel, H., Sander, J. & Xu, X. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining.*, 226–231. KDD-96 (AAAI Press, 1996).

[24] Levet, F. *et al.* SR-Tesseler: a method to segment and quantify localization-based super-resolution microscopy data. *Nature Methods* **12**, 1065–1071 (2015).

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

[25] Andronov, L., Orlov, I., Lutz, Y., Vonesch, J. & Klaholz, B. ClusterViSu, a method for clustering of protein complexes by Voronoi tessellation in super-resolution microscopy. *Sci. Rep.* **6** (2016).

[26] Andronov, L. *et al.* 3DClusterViSu: 3D clustering analysis of super-resolution microscopy data by 3D Voronoi tessellations. *Bioinformatics* **34**, 3004–3012 (2018).

[27] Mazouchi, A. & Milstein, J. Fast Optimized Cluster Algorithm for Localizations (FOCAL): a spatial cluster analysis for super-resolved microscopy. *Bioinformatics* **32**, 747–754 (2016).

[28] Nan, X. *et al.* Single-molecule superresolution imaging allows quantitative analysis of RAF multimer formation and signaling. *Proc Natl Acad Sci USA* **110**, 18519–18524 (2013).

[29] Rubin-Delanchy, P. *et al.* Bayesian cluster identification in single-molecule localization microscopy data. *Nature Methods* **12**, 1072–1076 (2015).

[30] Griffié, J. *et al.* 3D Bayesian cluster analysis of super-resolution data reveals LAT recruitment to the T cell synapse. *Sci. Rep.* **7** (2017).

[31] Levine, J. *et al.* Data-driven phenotypic dissection of aml reveals progenitor-like cells that correlate with prognosis. *Cell* **162**, 184–197 (2015).

[32] Fortunato, S. Community detection in graphs. *Physics Reports* **486**, 75–174 (2010).

[33] Endesfelder, U. *et al.* Multiscale spatial organization of RNA polymerase in *Escherichia coli*. *Biophys J.* **105**, 172–181 (2013).

[34] Rosvall, M. & Bergstrom, C. Multilevel compression of random walks on networks reveals hierarchical organization in large integrated systems. *PLoS ONE* **6**, e18209 (2011).

[35] Strehl, A. & Ghosh, J. Cluster ensembles – a knowledge reuse framework for combining multiple partitions. *J. Mach. Learn. Res.* **3**, 583–617 (2002).

[36] Lancichinetti, A. & Fortunato, S. Consensus clustering in complex networks. *Scientific Reports* **2** (2012).

[37] Danon, L., Díaz-Guilera, A., Duch, J. & Arenas, A. Comparing community structure identification. *J. Stat. Mech.* 09008 (2005).

[38] Aghaeepour, N. *et al.* Critical assessment of automated flow cytometry data analysis techniques. *Nat. Methods* **10**, 228–238 (2013).

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

[39] Maity, P. *et al.* B cell antigen receptors of the IgM and IgD classes are clustered in different protein islands that are altered during B cell activation. *Sci. Signal.* **8**, ra93 (2015).

[40] Depoil, D. *et al.* CD19 is essential for B cell activation by promoting B cell receptor-antigen microcluster formation in response to membrane-bound ligand. *Nature Immunology* **9**, 63–72 (2008).

[41] Harwood, N. E. & Batista, F. D. Early events in B cell activation. *Annual Review of Immunology* **28**, 185–210 (2010).

[42] Davis, R. *et al.* Chronic active B-cell-receptor signalling in diffuse large B-cell lymphoma. *Nature* **463**, 88–92 (2010).

[43] Jaccard, P. The distribution of the flora of the alpine zone. *New Phytologist* **11**, 37–50 (1912).

[44] Shivanandan, A., Unnikrishnan, J. & Radenovic, A. Accounting for limited detection efficiency and localization precision in cluster analysis in single molecule localization microscopy. *PLoS ONE* **10**, e0118767 (2015).

[45] Arias-Castro, E. Clustering based on pairwise distances when the data is of mixed dimensions. *IEEE Transactions on Information Theory* **57**, 1692–1706 (2011).

[46] Rosvall, M. & Bergstrom, C. Maps of random walks on complex networks reveal community structure. *Proceedings of the National Academy of Sciences* **105**, 1118–1123 (2008).

[47] Blondel, V., Guillaume, J., Lambiotte, R. & Lefebvre, E. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment* **10**, 10008 (2008).

[48] Rousseeuw, P. J. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics* **20**, 53–65 (1987).

[49] Tibshirani, R., Walther, G. & Hastie, T. Estimating the number of clusters in a data set via the gap statistic. *J. R. Statist. Soc. B* **63**, 411–423 (2001).

[50] Nino, D., Rafiei, N., Wang, Y., Zilman, A. & Milstein, J. Molecular counting with localization microscopy: A Bayesian estimate based on fluorophore statistics. *Biophysical Journal* **112**, 1777–1785 (2017).

[51] Fricke, F., Beaudouin, J., Eils, R. & Heilemann, M. One, two or three? Probing the stoichiometry of membrane proteins by single-molecule localization microscopy. *Scientific Reports* **5** (2015).

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

[52] Abraham, L. *et al.* Limitations of Qdot labelling compared to directly-conjugated probes for single particle tracking of B cell receptor mobility. *Scientific Reports* **7** (2017).

[53] Tafteh, R. *et al.* Real-time 3D stabilization of a super-resolution microscope using an electrically tunable lens. *Opt. Express* **24**, 22959–22970 (2016).

[54] Tafteh, R., Scriven, D. R. L., Moore, E. D. W. & Chou, K. C. Single molecule localization deep within thick cells; a novel super-resolution microscope. *J. Biophoton.* **9**, 155–160 (2016).

[55] Dempsey, G. T., Vaughan, J. C., Chen, K. H., Bates, M. & Zhuang, X. Evaluation of fluorophores for optimal performance in localization-based super-resolution imaging. *Nature Methods* **8**, 1027–1036 (2011).

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

Figures and figure legends

Figure 1.

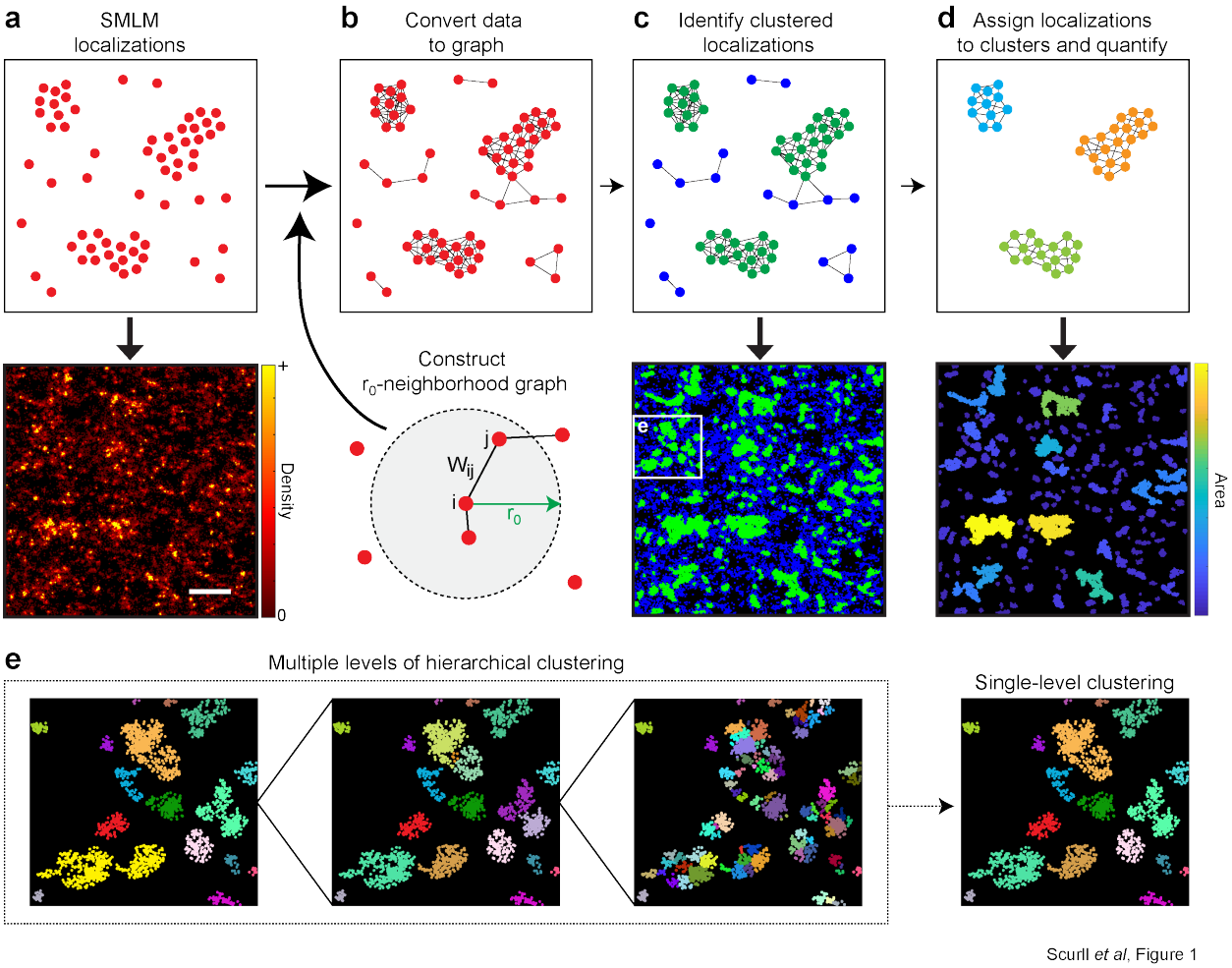

Figure 1: **Schematic illustration of StormGraph's workflow. (a–d)** SMLM localizations are used as nodes (a) and converted into a weighted graph (b). This graph is based on r_0-neighborhood graphs, where edges connect each node to all other nodes within a distance r_0. Two nodes i and j are connected by an edge of weight W_{ij}, which describes their similarity based on the distance between them and, if known, their positional uncertainties. Nodes are classified as either clustered (green) or unclustered (blue) based on their node degree, i.e. sum of adjacent edge weights, (c). A new graph is constructed from only the clustered nodes, which are then assigned to specific clusters using a community detection algorithm (d). Cluster properties (e.g. area) can then be quantified. The bottom panels in (a), (c), and (d) illustrate each step for an actual SMLM region of interest (scale bar = 500 nm). **(e)** StormGraph identifies a hierarchy of clusters at multiple scales and then generates an appropriate single-level clustering from the hierarchy. Shown are three different levels from the cluster hierarchy for the region in the white box in the lower panel of (c), along with the single-level clustering for this region. Colors distinguish different clusters.

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

Figure 2.

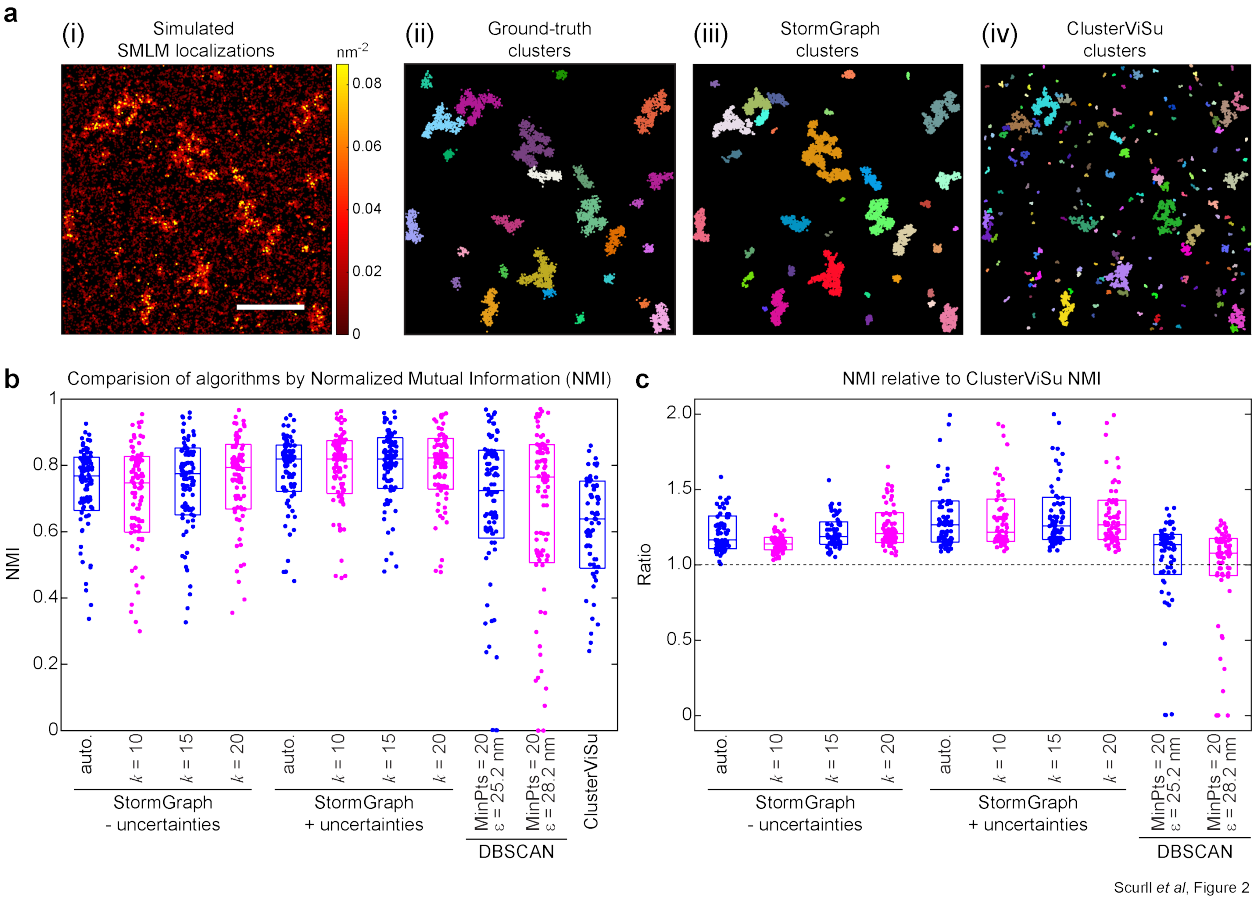

Figure 2: **StormGraph consistently outperforms ClusterViSu and DBSCAN on simulated data.** **(a)** (i) An example simulated dataset (color bar = density, scale bar = 500 nm), (ii) its ground-truth clusters, and cluster assignment results for (iii) StormGraph and (iv) ClusterViSu. Colors distinguish distinct clusters. **(b)** Accuracy of assigning data points to clusters as assessed by normalized mutual information (NMI; 1 = perfect). A total of 64 simulated datasets were analyzed using StormGraph, ClusterViSu, and DBSCAN. StormGraph was run either with (+) or without (-) localization uncertainties. The value of r_0 used by StormGraph was determined using the heuristic method (auto.) or the k-nearest neighbor method with k = 10, 15 or 20. DBSCAN was implemented using 16 different selections of its two parameters, MinPts and ε, of which the two best-performing are shown here. ClusterViSu results are only shown for the 49 datasets on which the analysis was completed in under 2 h. Boxes show medians and interquartile ranges. **(c)** The same results as in panel (b) are displayed normalized to ClusterViSu's performance for each of the 49 simulated datasets for which analysis by ClusterViSu was completed in under 2 h. StormGraph was consistently more accurate than ClusterViSu at assigning points to clusters, indicated by ratios > 1.

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

Figure 3.

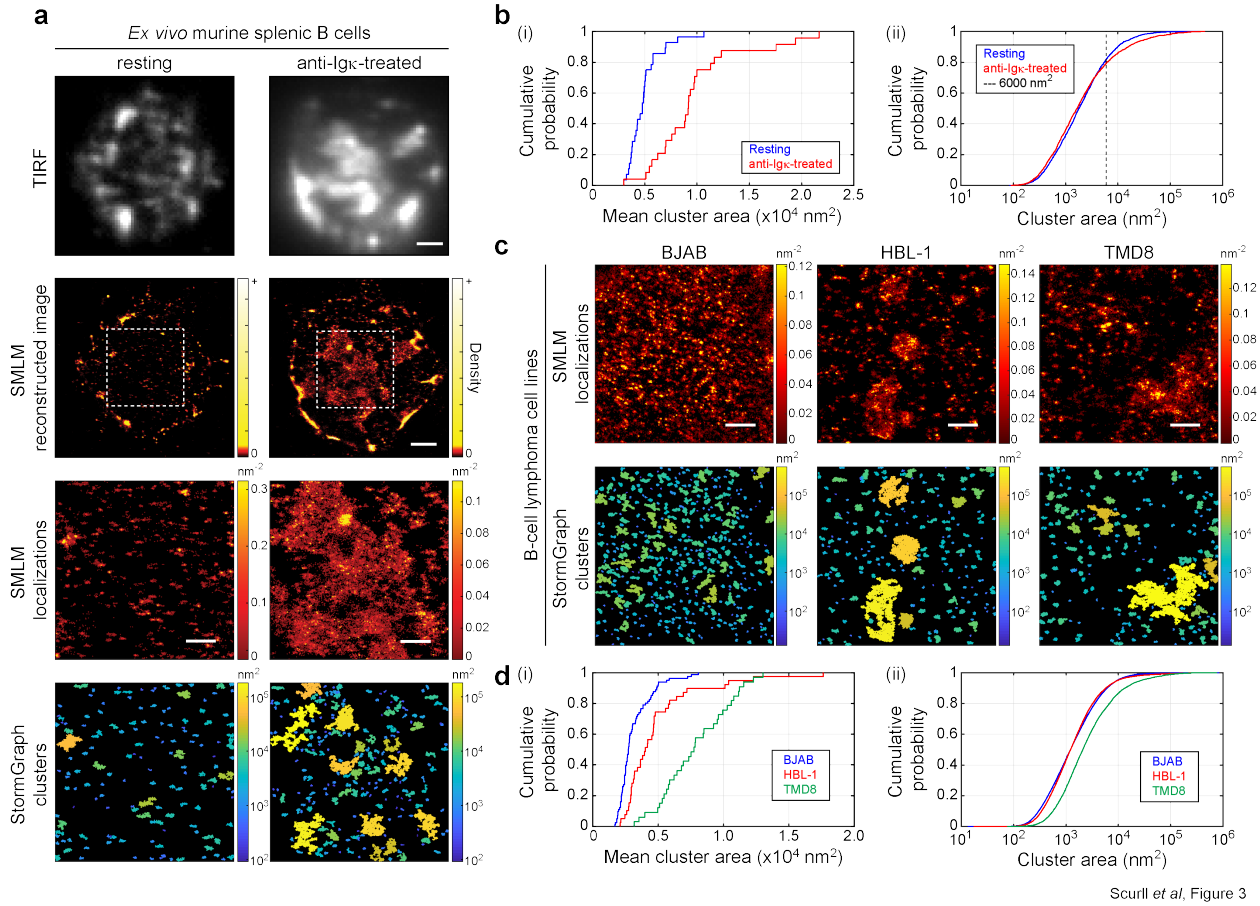

Figure 3: **StormGraph analysis of IgM-isotype B-cell antigen receptors (IgM-BCRs) on *ex vivo* murine splenic B cells (a–b) and human B-lymphoma cell lines (c–d) imaged using dSTORM. (a)** StormGraph analysis of IgM-BCRs imaged by dSTORM on *ex vivo* murine splenic B cells that were either resting (left panels) or treated with bivalent anti-Igκ antibodies (anti-Igκ-treated; right panels). Top row: Total internal reflection fluorescence microscopy images taken prior to dSTORM imaging of the same cells. Scale bar = 1 μm. Second row: Reconstructed images of IgM-BCRs from dSTORM localizations. Scale bar = 1 μm. Third row: IgM-BCR dSTORM localizations in the ROIs shown in the second row. Scale bar = 500 nm, color bar = density (nm^{-2}). Bottom row: Clusters identified by StormGraph, colored by their areas (nm^2). **(b)** Cumulative distribution functions (CDFs) for cluster areas in ROIs from 28 resting (blue) and 24 anti-Igκ-treated (red) *ex vivo* murine splenic B cells. Panel (i) shows the mean cluster area in each ROI. Panel (ii) shows all cluster areas from all ROIs. The increased mean cluster areas in anti-Igκ-treated cells compared to resting cells is due to clusters larger than ∼6000 nm^2. **(c)** StormGraph analysis of IgM-BCRs imaged by dSTORM on resting BJAB, HBL-1 and TMD8 cells. Top: ROIs containing IgM-BCR dSTORM localizations from one Burkitt's lymphoma (BJAB) and two ABC DLBCL (HBL-1 and TMD8) cells. Scale bar = 500 nm, color bar = density (nm^{-2}). Bottom: Clusters identified by StormGraph, colored by their areas (nm^2). **(d)** CDFs for cluster areas in ROIs from 81 BJAB (blue), 39 HBL-1 (red), and 33 TMD8 (green) cells. Panel (i) shows the mean cluster area in each ROI. Panel (ii) shows the areas of all clusters in all ROIs. The larger mean area of clusters on HBL-1 cells than on BJAB cells is due to small numbers of very large clusters. All StormGraph results shown here were generated using localization uncertainties and $k = 15$.

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

Figure 4.

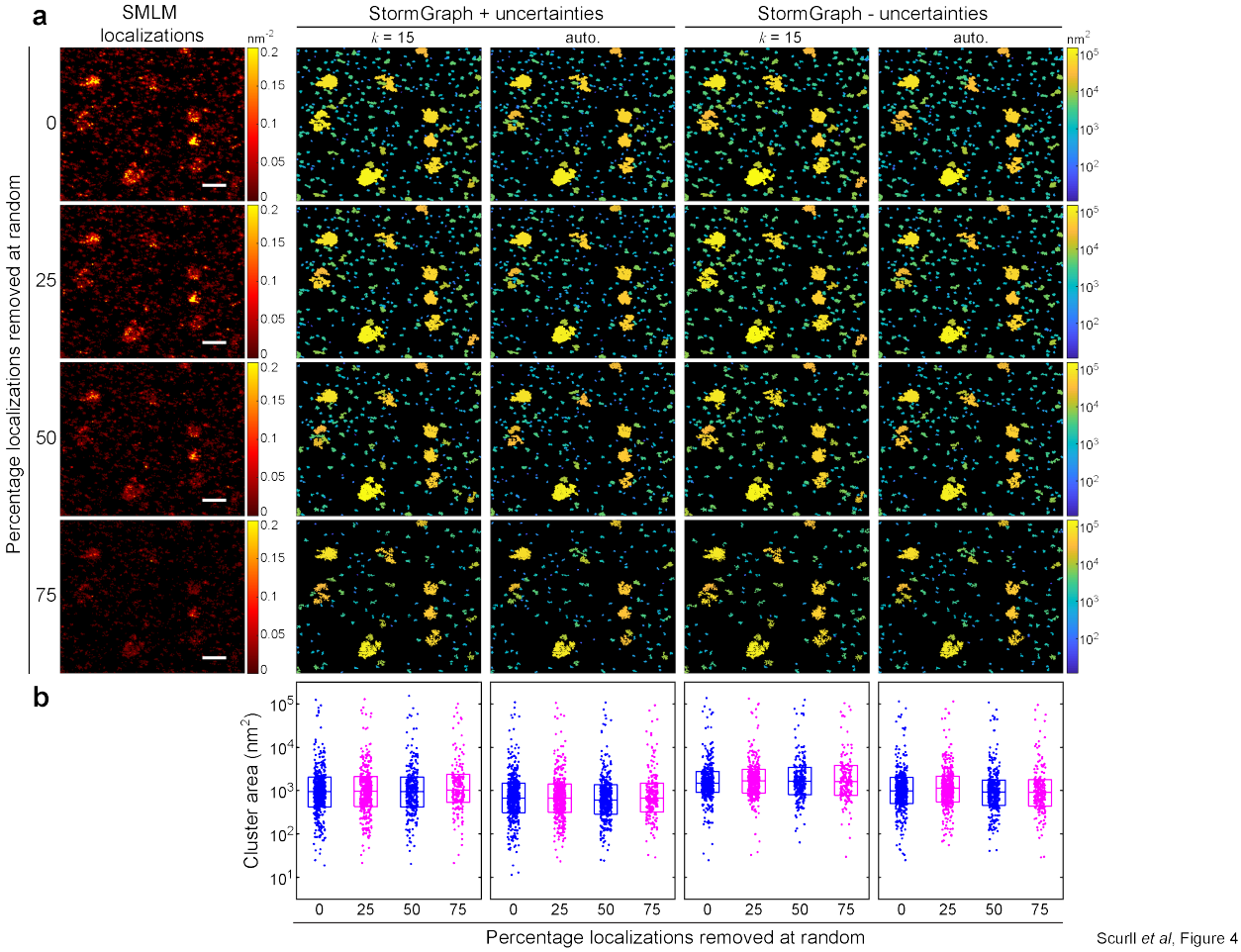

Figure 4: **StormGraph results are not sensitive to the global average density of localizations. (a)** StormGraph cluster results (color bar = cluster area (nm^2)) for a dSTORM dataset with 0%, 25%, 50% or 75% of the localizations randomly removed (left panels, color bar = density (nm^{-2})). Localization uncertainties were either used (+) or not used (-) by StormGraph during clustering. The value of r_0 was set using either the kNN method with $k = 15$ or the heuristic method (auto.). Scale bar = 500 nm. **(b)** Cluster areas quantified by each implementation of StormGraph for each of the four datasets, showing that the distribution of quantified cluster areas was not significantly affected by the random removal of localizations ($p > 0.05$ as determined by two-sample Kolmogorov-Smirnov tests). Boxes show medians and interquartile ranges.

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

Figure 5.

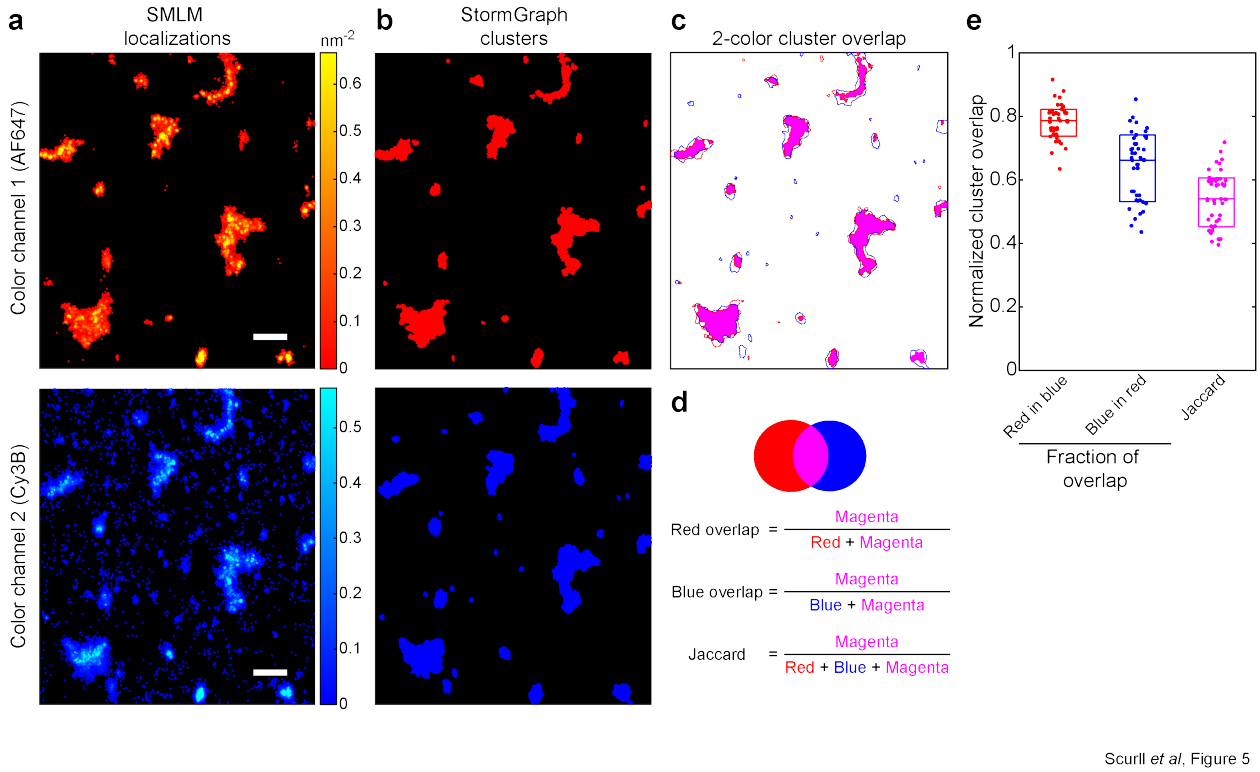

Figure 5: **Two-color cluster overlap analysis using StormGraph.** **(a)** IgG-isotype B-cell antigen receptors (IgG-BCRs) on A20 B cells were labeled simultaneously with bivalent anti-IgG antibodies that were conjugated to either AF647 (top panel; red) or Cy3B (bottom panel; pseudo-colored blue) and imaged using dSTORM. Bivalent antibodies were used to induce clustering, since each antibody can bind to up to two IgG-BCRs. Figure shows the IgG-BCR dSTORM localizations in a region of interest (ROI) from one representative cell analyzed using StormGraph. Scale bar = 500 nm, color bars = density (nm^{-2}). **(b)** Binary images of the AF647 (top) and Cy3B (bottom) clusters identified by StormGraph in the ROI shown in (a). **(c)** Merged image of the outlines of the AF647 clusters (red) and Cy3B clusters (blue) identified by StormGraph, with the overlapping area colored in magenta. **(d)** Pictorial description of the three area-based cluster overlap scores calculated by StormGraph, in the same order as the columns in panel (e). **(e)** Cluster overlap scores calculated using the formulae in panel (d) for 31 StormGraph-analyzed ROIs from multiple A20 cells imaged in the same experiment. Each ROI contributes one dot to each column. Boxes show medians and interquartile ranges. These scores determine the maximum observable overlap that could be expected for clusters of IgG-BCRs and a different molecule labeled using these same two fluorophores on A20 cells, imaged using the same imaging setup and analyzed by StormGraph.

bioRxiv preprint doi: https://doi.org/10.1101/515627; this version posted January 9, 2019. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

Figure 6.

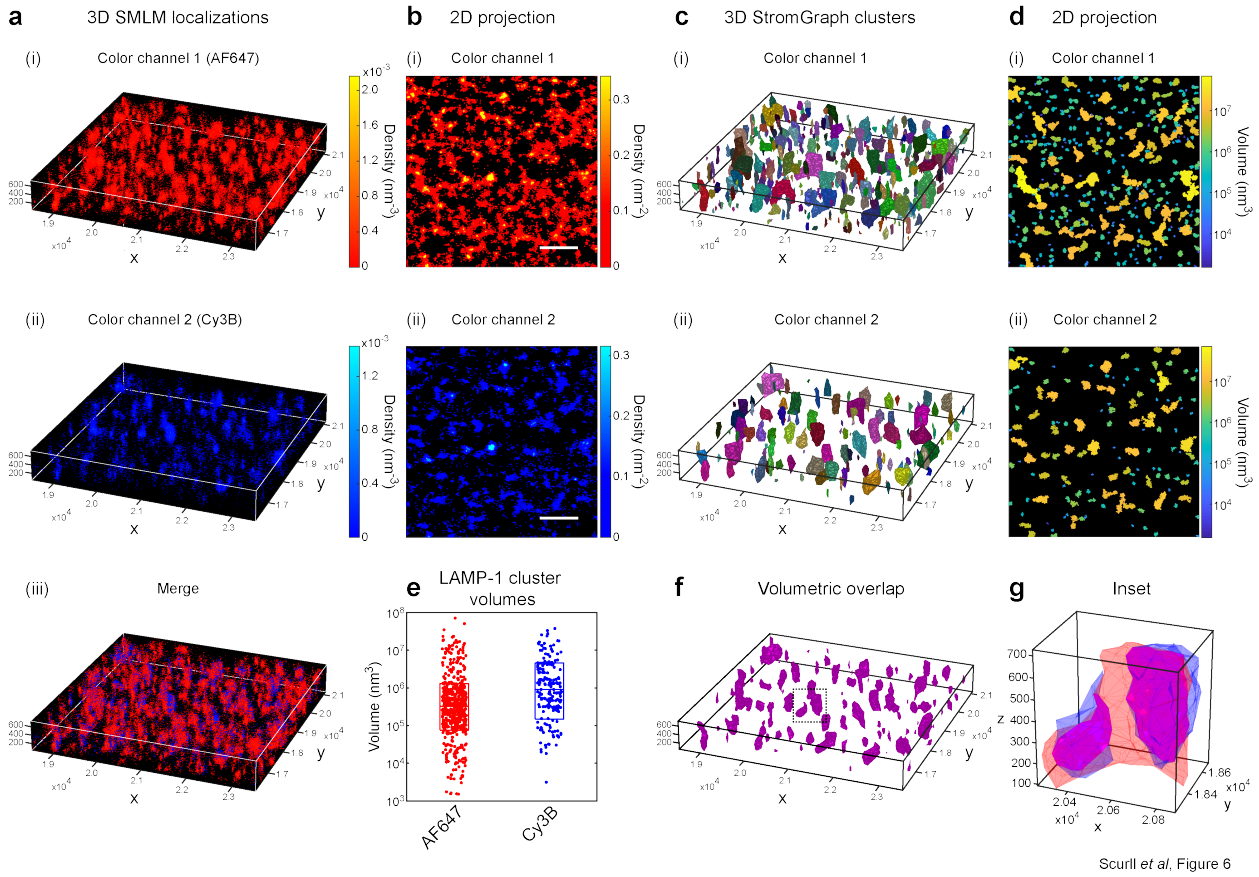

Figure 6: **Analysis of 3D SMLM data using StormGraph. (a)** Localizations of intracellular LAMP-1 detected by simultaneous staining with anti-LAMP-1 antibodies that were visualized using the same secondary antibody labeled with either (i) AF647 (red) or (ii) Cy3B (pseudo-colored blue). A murine B16 melanoma cell was imaged by two-color, 3D dSTORM and a 5 µm × 5 µm × 700 nm ROI was selected for analysis. Color bars = density (nm^{-3}). (iii) Merge. **(b)** 2D projections of the (i) AF647 and (ii) Cy3B localization data onto the xy-plane. Color bars = density (nm^{-2}). Scale bar = 1 µm. **(c)** (i) AF647 and (ii) Cy3B clusters found by StormGraph using localization uncertainties, $k = 15$, $\alpha = 0.1$, and a minimum cluster size of 5 localizations. Clusters of localizations that could not be confidently distinguished from a single, multiply counted fluorescent probe were removed. Colors distinguish different clusters. **(d)** 2D projections of the clusters in panel (c) colored by their volumes (nm^3). **(e)** All volumes of AF647 and Cy3B clusters detected by StormGraph, as in (d). Boxes show medians and interquartile ranges. **(f)** Overlapping volumes (magenta) of the AF647 and Cy3B clusters in panel (c). **(g)** Enlarged region showing overlap (magenta) between one AF647 cluster (red) and two Cy3B clusters (pseudo-colored blue).