Title

CellSpecks: A Software for Automated Detection and Analysis of Calcium Channels in Live Cells

Journal

Biophysical Journal, 115(11)

ISSN

0006-3495

Authors

Shah, Syed Islamuddin
Smith, Martin
Swaminathan, Divya
et al.

Publication Date

2018-12-01

DOI

10.1016/j.bpj.2018.10.015

Peer reviewed

Biophysical Journal
Article

Biophysical Society

CellSpecks: A Software for Automated Detection and Analysis of Calcium Channels in Live Cells

Syed Islamuddin Shah,[1] Martin Smith,[2] Divya Swaminathan,[3] Ian Parker,[3,4] Ghanim Ullah,[1,*] and Angelo Demuro[3,*]

[1]Department of Physics, University of South Florida, Tampa, Florida; [2]Pacific Biosciences, Menlo Park, California; [3]Department of Neurobiology and Behavior, and [4]Department of Physiology and Biophysics, University of California Irvine, Irvine, California

ABSTRACT To couple the fidelity of patch-clamp recording with a more high-throughput screening capability, we pioneered a, to our knowledge, novel approach to single-channel recording that we named "optical patch clamp." By using highly sensitive fluorescent Ca^{2+} indicator dyes in conjunction with total internal fluorescence microscopy techniques, we monitor Ca^{2+} flux through individual Ca^{2+}-permeable channels. This approach provides information about channel gating analogous to patch-clamp recording at a time resolution of ~2 ms with the additional advantage of being massively parallel, providing simultaneous and independent recording from thousands of channels in the native environment. However, manual analysis of the data generated by this technique presents severe challenges because a video recording can include many thousands of frames. To overcome this bottleneck, we developed an image processing and analysis framework called CellSpecks capable of detecting and fully analyzing the kinetics of ion channels within a video sequence. By using randomly generated synthetic data, we tested the ability of CellSpecks to rapidly and efficiently detect and analyze the activity of thousands of ion channels, including openings for a few milliseconds. Here, we report the use of CellSpecks for the analysis of experimental data acquired by imaging muscle nicotinic acetylcholine receptors and the Alzheimer's disease-associated amyloid β pores with multiconductance levels in the plasma membrane of *Xenopus laevis* oocytes. We show that CellSpecks can accurately and efficiently generate location maps and create raw and processed fluorescence time traces; histograms of mean open times, mean close times, open probabilities, durations, and maximal amplitudes; and a "channel chip" showing the activity of all channels as a function of time. Although we specifically illustrate the application of CellSpecks for analyzing data from Ca^{2+} channels, it can be easily customized to analyze other spatially and temporally localized signals.

INTRODUCTION

Ca^{2+} is a highly specific universal second messenger that regulates numerous cellular functions and plays a key role in many diseases (1–8). In physiological conditions, Ca^{2+} is tightly regulated by ion channels, pumps, and buffering proteins (8–10). Thanks to the recent progress in imaging technology and the development of more efficient Ca^{2+}-sensitive dyes, it is now possible to image Ca^{2+} signals inside intact cells with high spatiotemporal resolution, providing the ability to monitor Ca^{2+} flow across single plasma membrane Ca^{2+}-permeable channels including voltage-sensitive N-type Ca^{2+} channels (11–14), L-type Ca^{2+} channels in cardiac muscle, dihydropyridine-sensitive voltage-gated $Ca_V1.2$ channels (15), ligand-gated muscle nicotinic acetylcholine receptors (nAChRs) (16), inositol 1,4,5-trisphosphate (IP_3) receptors (IP_3Rs) (17), and membrane pores formed by soluble amyloid β oligomers (Aβ42) having multiple conductance levels (11–18). Surprisingly, a common feature of all these channels is their lack of motility because the fluorescent signal generated at a specific site can be imaged for many tens of second, suggesting a stationary feature during channel activity (12,13).

We have pioneered an imaging technique called "optical patch clamp," a massively parallel two-dimensional (2D) optical approach, capable of simultaneously and independently monitoring the functions of several thousand Ca^{2+}-permeable channels at single-channel resolution (19). Binding of Ca^{2+} to cytosolic fluorescent Ca^{2+} indicators generates fluorescent signals (single-channel calcium fluorescent transients; SCCaFTs) that closely track the opening and closing of ion channels (14). The presence of these bright spots on the 2D video sequence captured by total internal reflection fluorescence microscope (TIRFM) can be identified manually by visual inspection to infer channel

Submitted June 21, 2018, and accepted for publication October 22, 2018.

*Correspondence: gullah@usf.edu or ademuro@uci.edu

Syed Islamuddin Shah and Martin Smith contributed equally to this work.

Editor: James Sneyd.

https://doi.org/10.1016/j.bpj.2018.10.015

location and activity. However, the manual detection of these elementary events is extremely tedious and time consuming and does not lend itself to a comprehensive analysis of the stacks because it is nearly impossible to accurately determine the location of each of the hundreds or sometimes thousands of channels present in the image field. This is particularly difficult for channels with extremely low open probability (P_O) and short opening events in which the few recorded events are several thousand frames apart. This could result in an incomplete representation of the channel population behavior.

Several automated detection packages are available for analysis of similar data, such as GMimPro for single fluorophores (20), ImageJ plugins for line-scan images like SparkMaster (21) and xySpark (22), FLIKA for Ca^{2+} puffs resulting from concerted gating of clustered IP_3Rs (23), and several algorithms for processing of Ca^{2+} sparks generated by clustered ryanodine receptors (24–30) and human PIEZO1-channels (31). However, the existing software packages are limited in their abilities to detect a large number of simultaneously active channels (sites) with multiple conductance levels and generate huge data sets about the location and gating kinetics of these channels. For example, in our experiments, typical imaged data consist of a high-temporal-resolution image stack recorded at ~500 frames per second in a 128 × 128 pixel multi-image (up to 15,000 frames) Metamorph (.stk) file. Automated detection of channel locations and their gating kinetics would allow for a thorough description of the behavior of every channel, reliably pinpointing all event locations regardless of size, frequency, and duration. Furthermore, statistical analysis can be customized and automated, providing a powerful tool for ion channels and single-molecule studies.

With these needs in mind, we developed an automatic detection algorithm coupled with graphical user interface and statistics modules. We named the resulting software CellSpecks, which is implemented in the Java programming language for speed, flexibility, and portability. CellSpecks is capable of automatic detection of the location and gating behavior of many ion channels in a Metamorph stack file or image files in formats such as JPG and TIFF, with tremendous efficiency over a manual approach, and is able to reliably identify subtle events in which manual identification is tedious if not impossible. As a result, analysis of imaging data from TIRFM experiments now takes only a fraction of the time needed for visual inspection (only a few minutes instead of many hours and days), and events previously not accessible manually or through other packages are made available for statistical analysis.

Here, we first use CellSpecks to analyze synthetic images sequences in which all events are known a priori to validate its accuracy in challenging but anticipated situations. After validating the performance and accuracy, we then illustrate the use of CellSpecks to process TIRFM image sequences of Ca^{2+}-permeable nAChR channels and individual pores formed by Aβ42 oligomers in the plasma membrane of *Xenopus laevis* oocytes. CellSpecks can efficiently generate location maps and create raw and processed fluorescence time traces; histograms of maximal amplitudes of Ca^{2+} release events, mean open times, mean close times, and P_O of all channels; and amplitudes and durations of all events in the image sequence. All data can be exported in ASCII file format for plotting and further analysis. In particular, time traces from all channels can be exported and idealized using the open source software TraceSpecks (32,33) for developing single-channel models. Thus, CellSpecks, together with high-resolution fluorescence microscopy, provides a powerful tool for characterizing and modeling ion channel behavior, using an unprecedented amount of data sets simultaneously recorded from thousands of channels in their native environment.

MATERIALS AND METHODS

The experimental data used to test CellSpecks derive from previously published work imaging the activity of nAChRs (13) and pores formed by Aβ42 oligomers (18). A brief description of these methods is given below.

Oocyte preparation and electrophysiology

Experiments were performed on defolliculated stage VI oocytes obtained from *X. laevis* (34). For experiments with muscle nAChRs, in vitro-transcribed complementary RNAs coding for α, β, γ, and δ subunits (in a molar ratio 2:1:1:1) were mixed to a final concentration of 0.1–1 μg/μL and microinjected (50 nL) into oocytes (16). After 3–5 days, the expression of nAChRs was monitored by recording currents evoked by bath application of acetylcholine (16). Insertion of functional Aβ42 pores into the oocyte's plasma membrane (18) was achieved by bath application of solution containing soluble oligomers prepared from human recombinant Aβ42 peptide. The solutions containing Aβ42 oligomers were delivered from a glass pipette with tip diameter ~30 μm positioned near the edge of the membrane footprint of the oocyte membrane on the cover glass.

Oocytes were injected ~1 h before imaging with fluo-4-dextran (MW ~10 kD; Ca^{2+} affinity ~3 μM) to a final intracellular concentration of ~40 μM. For imaging, oocytes were placed animal-hemisphere down in a chamber with its bottom made from a fresh-ethanol-washed microscope cover glass (type-545-M; Thermo Fisher Scientific, Waltham, MA) and were bathed in Ringer's solution (110 mM NaCl, 1.8 mM $CaCl_2$, 2 mM KCl, and 5 mM Hepes (pH 7.2)) at room temperature (~23°C) continually exchanged at a rate of ~0.5 mL/min by a gravity-fed superfusion system. The membrane potential was clamped at a holding potential of 0 mV using a two-electrode voltage clamp (Gene Clamp 500; Molecular Devices, San Jose, CA) and was stepped to more negative potentials up to −100 mV when imaging Ca^{2+} flux through the gating channels to increase the driving force for Ca^{2+} entry into the cytosol (16,18).

TIRFM imaging

Imaging was accomplished by using a custom-built TIRF microscope system based around an Olympus IX 71 microscope (Olympus, Tokyo, Japan) equipped with an Olympus 60× TIRFM objective (NA = 1.45) (16,18). Fluorescence excited by a 488 nm laser was imaged using an electron-multiplied charged coupled device camera (Cascade 128+; Roper Scientific, Vianen, the Netherlands) at full resolution (128 × 128 pixel: 1 pixel = 0.33 μm at the specimen) at a rate of 500 s^{-1}. Image data were

acquired using the MetaMorph software package (Universal Imaging, Westchester, PA).

Event detection algorithm and automated analysis

CellSpecks has a menu-driven interface for reading and processing video stacks and individual images, as well as the ability to illustrate various steps and results of the algorithm, allowing the experimenter to confirm the effectiveness of the algorithm for their particular data set. Third-party libraries capable of returning pixel byte arrays for these video/image stacks are used to feed image data into the program that convert an image stack from its native format to a stack data structure. The stack is then cloned (copied) and modified a number of times along the way to preserve processed data that will later help the experimenter understand how the algorithm produced the results.

Fig. 1 summarizes the steps, including noise detection, signal isolation from noise, event attribution, and analysis from time trace of a given channel. After cloning the initial input movie frames or TIFF images, the program takes each pixel in the image over time and assigns it to an individual thread to generate blurred signal, noise, and signal stacks from the Original stack. Flowchart 1 in the Supporting Materials and Methods lists various steps involved in computing the Noise stack from the Original stack. These steps are explained below.

First, the Original stack is copied to a temporary stack called Temp. Each pixel in the Temp stack is then replaced by the average of the intensities of the same pixel over time (51 frames) by the application of a one-dimensional flat kernel of length 51 (a moving box filter with a window size of 51 frames), with the center of the kernel being the intensity value to be modified. Appropriate boundary conditions are applied at the boundary (both in time and space) pixels. As a result, a modified stack (Blur) is generated for each pixel. Next, a mode value for each pixel over time in the Blur stack is determined. This is followed by computing the SD of the points in the Original signal that are below the mode of this pixel. A noise threshold is set at 2.5 $\times$ standard deviation + mode. All intensity values over time of this pixel in the Original stack that are greater than the noise threshold are replaced with the mode value to get what is called the Noise stack. Subtracting the Noise stack from the Original image stack generates a Signal stack. The Signal stack is then used to identify all Ca^{2+} channels along with their location, open and close events, P_O, and maximal amplitudes. Various steps involved in event detection, channel locations, and assigning events to different channels are listed in Flowchart 2 in the Supporting Materials and Methods and described below.

First, a three-dimensional array of size X $\times$ Y $\times$ Time (where X and Y are the dimensions (number of pixels along the x and y axes) of the image and Time is the number of images (frames) in the data) called *EventPartStack* is created to hold each pixel's data (intensity) over time. Each element of this array also contains variables to link each nonzero intensity pixel (a pixel with no signal is represented by a value of zero) to four neighboring pixels in the current frame (at time t) and two neighbors in the frames before (time $t - 1$) and after (time $t + 1$) the current frame. Each nonzero element of the *EventPartStack* array is linked to all of its nonzero six neighbors. Now, for each nonzero value of *EventPartStack*, events containing links to the nonzero neighbors of the current pixel are recursively generated unless a zero-valued pixel is reached and are added to the probable event list (*probEventlist*) array. The size of the *probEventlist* array now contains all probable events for all the channels in the given data set. This *probEventlist* is further screened for events whose duration is $<$10 frames or whose cumulative intensity is $<$50 (intensity units), and a final list of events called *Eventlist* is generated.

To accurately place these channels, a weight array is generated as a 2D array (of image size) containing the sum of pixel values in the Signal over time for that pixel. The weight array is then normalized by the highest pixel value in the weight array. Nonzero values in weight array are interpreted as likely channel locations and added to a list of channels with no events.

The events are then added to channels one by one, either to the closest known channel if there exists one with which it overlaps or to a new channel if no current channel overlaps. Channels are allowed to have floating point coordinates, somewhat complicating the concept of overlapping, especially when each event covers an area often no more than two pixels across. The nearest neighboring channel of each event is determined either by matching the coordinates of the event and channel or by calculating the overlap likelihood in case the coordinates do not have a one-to-one match (see Channel Locations in the Supporting Materials and Methods). The value of overlap likelihood is a combination of the inverse of the distance and the relative intensity of the event at the location of the nearest-neighbor channel. Thus, if the center of the channel does not overlap the event at all, the overlap likelihood is zero, and any positive overlap value decreases to zero as the distance increases. Finally, channels containing no events are removed. At the end of this process, channel locations are adjusted for the actual

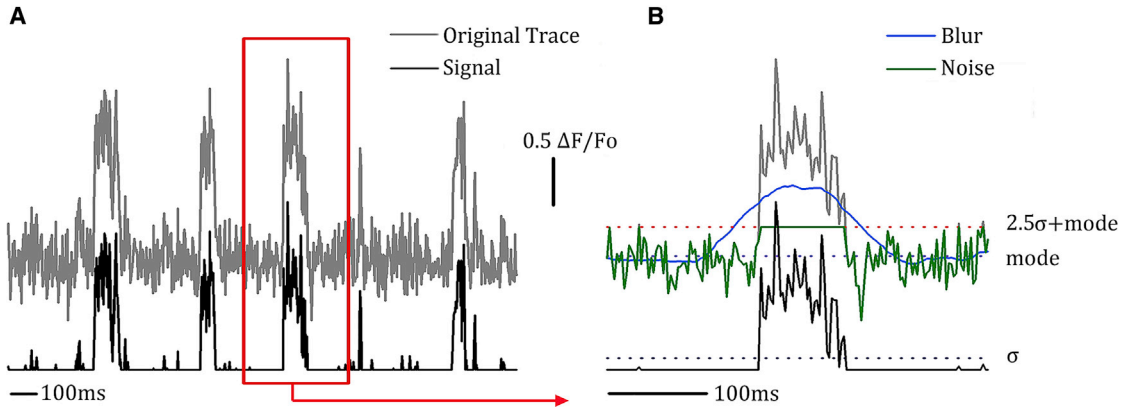

FIGURE 1 Illustration of CellSpecks algorithm processing steps: separation of noise from signal of nAChRs channel activity. (*A*) shows fluorescence trace from a single pixel (*gray*) and the final signal trace (*black*) calculated by CellSpecks. (*B*) illustrates the processing steps required to go from the original (*gray*) trace to the signal (*black*) trace. First, temporal smoothing creates a blur (*blue*) trace. A mode value is calculated for the entire blur trace (*purple dotted line*). SD, σ, is calculated for all fluorescence points from the original trace that are below the mode value (*dark blue dotted line*). Noise threshold is set as 2σ + mode (*red dotted line*) and a noise trace generated (*green*). Subtracting noise (*green trace*) from the original trace (*gray*), the signal trace (*black*) is obtained. In this figure, all time-varying quantities (original trace, blur, noise, signal) are plotted with solid lines, whereas time-invariant variables (mode, σ, noise threshold) are plotted with dotted lines.

events they contain, and any channels found to be within the same pixel are combined. Further details about assigning events to channels and identifying channel locations are given in the Channel Locations section of the Supporting Materials and Methods.

By the end of the detection and association process, a list of channel locations has been produced; within each channel is a list of known events, and in each event is a list of known event parts (start and end of an event) as shown in Fig. 2. The times between these events are averaged to give a mean close time, and the durations of these events averaged to give a mean open time. The total open time divided by the total sample time gives the P_O. Maximal amplitude is the highest peak in the channel after a one-dimensional smoothing kernel of length three (moving box average with a window size 3) is applied to the intensities over time to filter high-frequency noise. The program also saves the durations and amplitude of all events of all channels in the stack or image sequence. Descriptive statistics as well as intermediate stack, baseline, signal, and noise information can be viewed in the CellSpecks graphical user interface (Fig. 2). Channel locations are visually reported for one-click, intensity versus time graph confirmation, and the sampling for these traces can be modified to select an average or maximum of pixel values over a user-specified channel. These channel locations may also be threshold by maximal intensity. Statistical information, time-trace data (as modified by user-selected sampling parameters), location maps, and channel chips (a surface (raster) plot showing the gating of all channels over time) may then be exported using the interface as well.

The resulting program has been tested on Mac operating system X, Ubuntu Linux, and Windows XP-7 and using the Sun Java 1.6 Jave run time environment (JRE), Apple Java 1.5 and 1.6 JRE, SoyLatte Java 1.6 JRE, and the OpenJDK Java 1.6 JRE. The precompiled JAR file, User Manual, sample stack file, and sample images are included with this article as Data S1. A video explaining different steps of using the software the software is also provided as Video S1. The source code for the software is available upon request from the authors. In trial runs against previously examined stacks, CellSpecks is capable of processing a 43 × 43 pixel × 15,000 image stack in ~43 s on a 1.73 GHz Intel Core i7-820QM with 8 gigabytes DDR3 random-access memory. The peak random-access memory usage during the detection process is ~7.5 GB on Windows 7. This includes opening the images, feeding them into integer arrays, and executing the detection algorithms. Fig. 2 shows typical confirmation images produced by the program.

RESULTS

In the first part of this section, we use the time traces, locations, and various statistics from all channels in the image sequence to verify that the output from the algorithm matches the ground truth of synthetic data.

Synthetic data validation

To test the accuracy of the algorithm, we generated a synthetic data set of 50 channels randomly distributed on a grid of 128 × 128 pixels for a total duration of 2 s in a sequence of 1000 TIFF images. We deliberately randomized the opening and closing of these channels as well as their open and close times to mimic experimental conditions. To keep channel flux tractable, an amplitude of zero (intensity units) represents a closed channel, whereas an open channel was assigned an amplitude of 200 (intensity units) in line with the frequently observed intensity values when the channel is open. To make the channel signal realistic, noise derived from normal distribution with a mean of 5 and SD of 2 was superimposed on each channel's time trace as well as all other pixels in the image frame. The observed spread of Ca^{2+} from the channel to the surrounding area is mimicked by allowing the fluorescence from an open

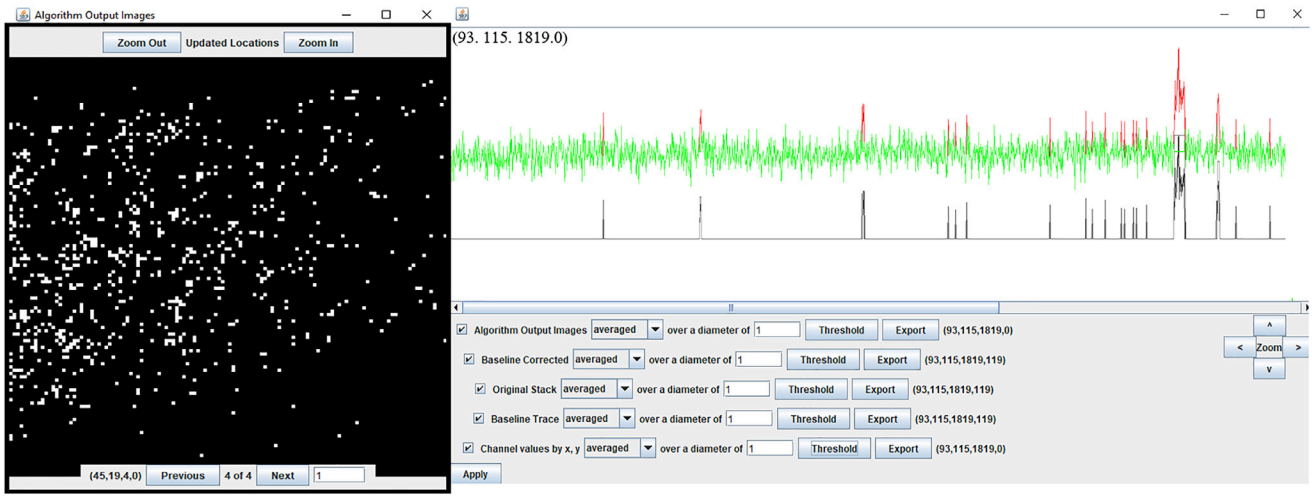

FIGURE 2 CellSpecks graphical user interface. The CellSpecks interface includes the capacity to audit the detection and analysis process for quality and accuracy. The map of detected channels is shown in the Algorithm Output Images windows (*left*). Map is generated from a 5000 frames stack captured by imaging a 40 × 40 μm^2 region of an oocyte plasma membrane expressing muscle ($\alpha\beta\gamma\delta$) nAChRs. Channels are represented by bright pixels at their determined location. The brightness of a pixel corresponds to the maximal amplitude event generated by the channel. Clicking on any pixel brings up the trace window (*right*), displaying the time traces such as raw (*red*), baseline or noise (*green*), and processed or signal traces (*black*) of the channel at that location. Results of any modification are displayed in the Algorithm Output Images. Information such as mean open and close times, P_O, peak amplitudes, lifetimes and amplitudes of all events for all channels, channel chip showing the activity of all channels detected as a function of time, and channel location maps for all channels can be exported as ASCII files for publication-quality plots and further analysis.

channel's location to spread to the nearest neighbor pixels. Code used to generate synthetic data is available on request from the authors. Further details about the algorithm generating the synthetic data are given in the Supporting Materials and Methods.

We tested our algorithm for signal/noise ratios (SNRs) in the range of 5–40. CellSpecks was able to correctly identify channel locations, number of channels, and all opening events along with their open and close times as well as mean open and close times and P_O for all SNRs > 5. Fig. 3 compares the results generated by CellSpecks for the synthetic data set with the actual statistics of the data. As is evident from Fig. 3, *A* and *B*, the distributions for mean open and close times predicted by CellSpecks (*second row*) are in close agreement with the true distributions (*top row*). Similarly, the open probabilities estimated by CellSpecks (*bottom*) are in close agreement with the true values (*top*) (Fig. 3 *C*). The slight discrepancy in P_O distribution is due to the fact that CellSpecks excludes partial open or close events toward the end of the recording, whereas such events were included in the true statistics. The gating state of all channels at a given time step (frame number) given by CellSpecks is saved as a channel-chip representation (Fig. 3 *D*). The kinetics of all 50 channels given by CellSpecks is remarkably close to the true values. An example of synthetic channel trace generated with an SNR of 5 is shown in Fig. 3 *E*, whereas the time trace identified by CellSpecks is shown in Fig. 3 *F*. A comparison of the two traces shows that CellSpecks is capable of identifying highly noisy channels along with open/close events precisely.

Although CellSpecks can accurately identify channels and their associated events for SNRs > 5, for data with an SNR ≤ 5, we found that it misses some channels as well as associated events. Fig. 3 *G* shows a comparison of the number of channels (*left*) and all events (*right*) identified by CellSpecks (*green bars*) with the actual values (*red bars*) as a function of SNR in the records. CellSpecks missed two channels for SNR $= 5$ but was able to identify all channels for SNR > 5. Similarly, CellSpecks missed 28 events for SNR $= 5$ while identifying all events in images with a higher SNR. Fig. 3 *H* shows a comparison between channel locations identified by CellSpecks (*green circles* for SNR $= 5$ and *blue cross* for SNR $= 10$) and actual locations (*yellow circles* for SNR $= 5$ and *red circles* for SNR $= 10$). As is evident, the channel location map identified by CellSpecks for SNR $= 10$ matches exactly with the actual locations. The same is the case for SNR $= 5$ except that CellSpecks missed two channels. It is worth mentioning that the SNR in TIRFM using fluorescence dye Fluo-4 is close to 8 and higher for Cal-520 (35). CellSpecks did not detect any events in control stacks with no events.

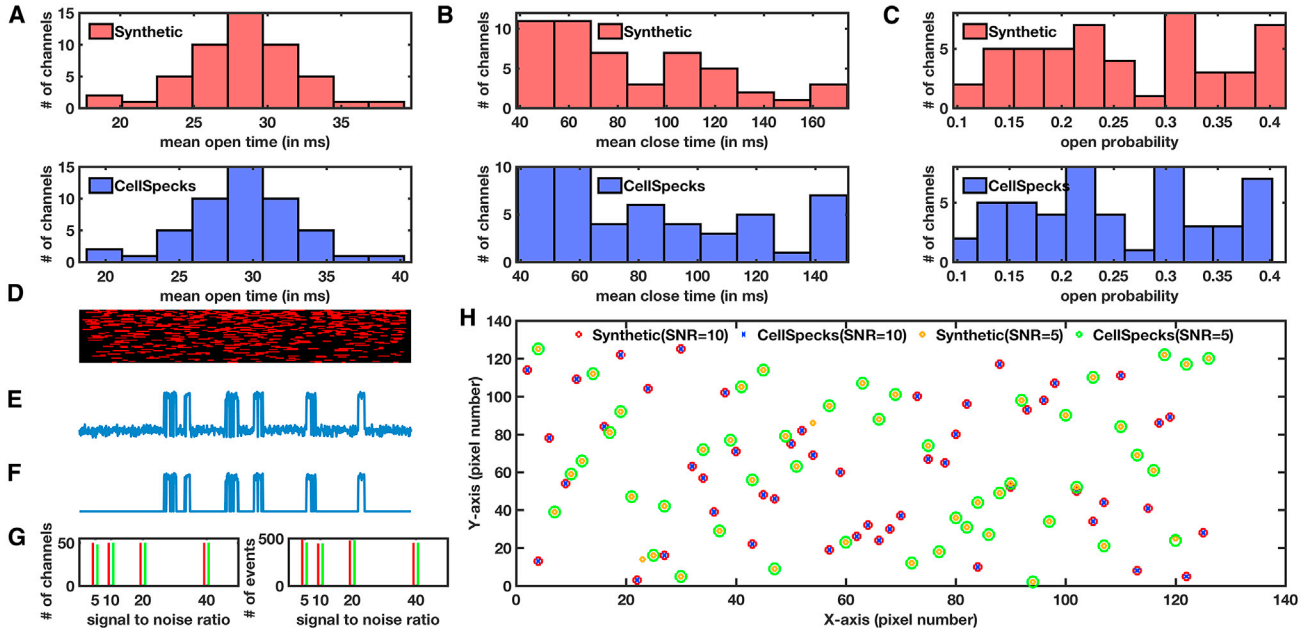

FIGURE 3 CellSpecks closely reproduces the channel locations and gating kinetics of simulated functional channels with random location and gating kinetics. Distributions of mean open times (*A*), mean closed times (*B*), and mean P_O (*C*) of true values (*first row*) and values estimated by CellSpecks (*second row*) for all channels are shown. (*D*) Channel-chip (image exported directly from CellSpecks) representation in which the horizontal axis is time (2 s total) and the vertical axis is the channel number (1–50)—i.e., each horizontal line represents one channel. The red and black represent the open and closed states of the channel, respectively. (*E*) The actual noisy (SNR $= 5$) time trace for a single channel and (*F*) the trace identified by CellSpecks are shown. (*G*) The number of channels (*left panel*) and number of events (*right panel*) identified by CellSpecks (*green bars*) versus the actual values (*red bars*) as functions of SNR. (*H*) Channel location maps from CellSpecks (*green circles* for SNR $= 5$ and *blue* $\times$ for SNR $= 10$) and true locations (*yellow circles* for SNR $= 5$ and *red circles* for SNR $= 10$).

We remark that event identification is accurate for all but extremely long events lasting more than 50% of the total duration of the recording. In cases in which event duration approaches half of the total duration of the experiment, the amplitude of the event can be mistakenly considered to be the baseline. Note that the intensity during the event will have to be almost constant for this failure to happen. Although unlikely for data without potentially compromising flaws, it could be an issue in data sets that have been cropped to represent a very short total duration.

CellSpecks works equally well on images and stacks in which the noise in the signal is not uniform. We tested it on images corrupted with noise in which the power spectral density varies inversely with frequency (often called flicker or "pink" noise) (36) generated using a power-law noise-generation algorithm (37). The software detects the channel locations and open/close events with the same accuracy as it gives for images with white noise (data not shown).

Detection of muscle nAChR channel activity in *Xenopus* oocytes

Next, we used CellSpecks to detect channels from experimental data obtained by imaging nAChR activity in *Xenopus* oocytes. As previously shown, TIRFM of Ca^{2+} imaging of membrane regions in oocytes expressing nAChRs revealed numerous transient fluorescence "flashes" (SCCaFTs) in the presence of nicotinic agonists when the membrane is hyperpolarized to increase the driving force for Ca^{2+} influx (14,19,38). Manual analysis of these data revealed a sparse distribution of nAChRs throughout the image field (14). When we used CellSpecks to detect channels and their events in the same data set, the number of active sites detected was ~3 times larger than the number detected manually. Moreover, CellSpecks required only a few minutes to detect ~850 channels (5178 total events), substantially less than the many hours normally needed for manual inspection. The location map of all the channels and examples of processed fluorescence time traces from three channels given by CellSpecks are shown in Fig. 4, *A* and *B*, respectively. We would like to point out that the ability of CellSpecks to segregate the two nearby channels can be limited in situations in which two channels overlap at two adjacent pixels with both channels displaying a long opening time. We have previously estimated that the spatial spread of an SCCaFT generated by the opening of nAChRs measured as the full width at half-maximal amplitude has a spatial spread of ~500 nm (13). In our experiments, controlled level of expression can easily overcome this problem because the channels imaged so far do not display tendencies to cluster but a clear random distribution that is normally achieved in relatively low channel overexpression.

The 850 channels detected by CellSpecks included 100% of those detected by hand in experimental data sets. The remaining channels were checked manually in experimental stacks and found to contain at least one centroid uniquely centered at that location without overlapping nearby channels (occasionally in neighboring pixels) at all. As seen in Fig. 5 *C*, the P_O distribution indicates that the vast majority of channels detected by CellSpecks are those that have small P_O with infrequent and short open events separated by long close events, affirming the ability of CellSpecks to detect channels with low P_O and events with extremely short lifetimes.

Automated analysis of channel properties

Analysis of the large amount of information provided by simultaneously analyzing the behavior of a few thousands ion channels is extremely challenging using available data analysis software such as R (GNU), Origin (OriginLab), or

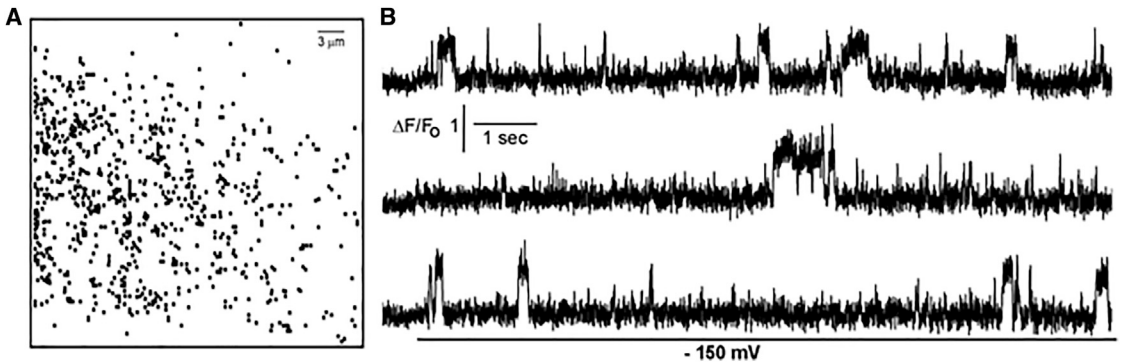

FIGURE 4 Detection of hundreds of individual nAChR channels by CellSpecks. (*A*) A channel map showing the locations of 850 nAChRs channels within a $40 \times 40\ \mu m^2$ membrane patch is shown. The map was generated automatically by CellSpecks after identifying coordinates of all channel sites through a 25 s imaging period, during which the oocyte was polarized to −150 mV in the presence of 1 μM acetylcholine. (*B*) An example of single-channel recordings (SCCaFTs) resulting from Ca^{2+} influx during channel opening when the membrane was hyperpolarized to −150 mV. Traces are obtained by monitoring fluorescence from regions of interest (one pixel, corresponding to a $0.33 \times 0.33\ \mu m^2$ of plasma membrane) centered on three of the channel locations shown in (*A*).

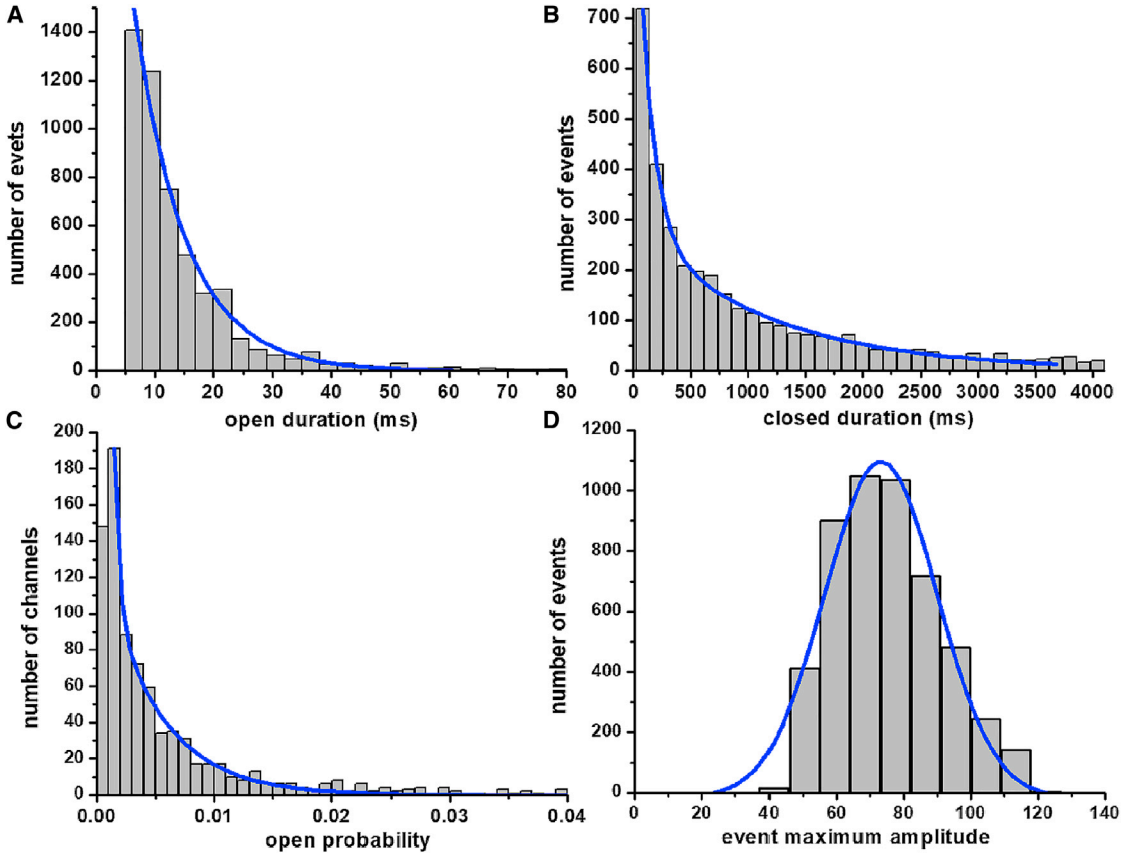

FIGURE 5 Automated analysis of channel parameters. After locating the active sites (850) in the image field (Fig. 4 *A*), CellSpecks can automatically generate analyses of the fluorescence events (5187) measured during the record, calculating the open times, close times, amplitudes for all events, and mean P_O for each corresponding region of interest. These data sets are then used to statistically analyze channel populations. (*A*) Distribution of the events duration for all of the detected nAChR channels shown in Fig. 4 *A*. Data are fitted by single exponential decay (*solid line*) with the time constant 8.6 ms. (*B*) Distribution of the corresponding closed times (intervals between events) fitted by a double exponential decay function (*solid line*) with time constants of 123.8 and 1191.3 ms. (*C*) Distribution of the mean open probabilities calculated for the corresponding channels. Data are fitted by a double exponent decay function with P_{O1} of 0.00041 and P_{O2} of 0.0047. (*D*) Plot displays the maximal events amplitude distribution obtained measuring the peak fluorescence for each detected event. Distribution are fitted by a Gaussian function with a peak amplitude of 73 ΔF/Fo $\times$ 100 (*solid line*). To see this figure in color, go online.

software designed for analysis of electrophysiological single-channel data. As described in the Materials and Methods, the statistical analysis module contained in CellSpecks is capable of generating statistical records of the parameters characterizing ion channel behavior. For example, in Fig. 5, we show the distributions of open durations (Fig. 5 *A*), close durations (Fig. 5 *B*), and the maximal amplitudes (Fig. 5 *D*) of the SCCaFTs detected in Fig. 4 *A*. The distribution of mean P_O of all channels is shown in Fig. 5 *C*. The ability of CellSpecks to store the lifetimes and amplitudes of all events for all channels detected (Fig. 5) makes the more complex analysis such as cross correlation studies between any of the measurable channel parameters a lot easier (18).

Detection of Aβ42 pore activity

The first version of CellSpecks was developed and used to investigate gating properties of Aβ42 pores in *X. laevis* oocytes (18), allowing the discovery of a large population of Aβ42 pores with very low P_O otherwise overlooked and underestimated by visual inspection. Moreover, CellSpecks is not only capable of correctly and efficiently characterizing the behavior of channels with single permeability level but can also characterize channels with multiple Ca^{2+} permeability levels. As an example, we applied CellSpecks to TIRFM imaging data of Ca^{2+}-permeable plasma membrane pores formed by Aβ42 oligomers. In this case, CellSpecks identified 830 active sites (Fig. 6). Distributions for mean open times, mean close times, mean P_O, and peak amplitudes for all channels detected are shown in Fig. 6, *A–D*. Notice that the distributions shown in Fig. 6, *A* and *B* are the mean values for all channels (one value per channel), which are different than those in Fig. 5, *A* and *B* representing the open and close times for all events (multiple values per channel). Similarly, the amplitude distribution in Fig. 6 *D* represents

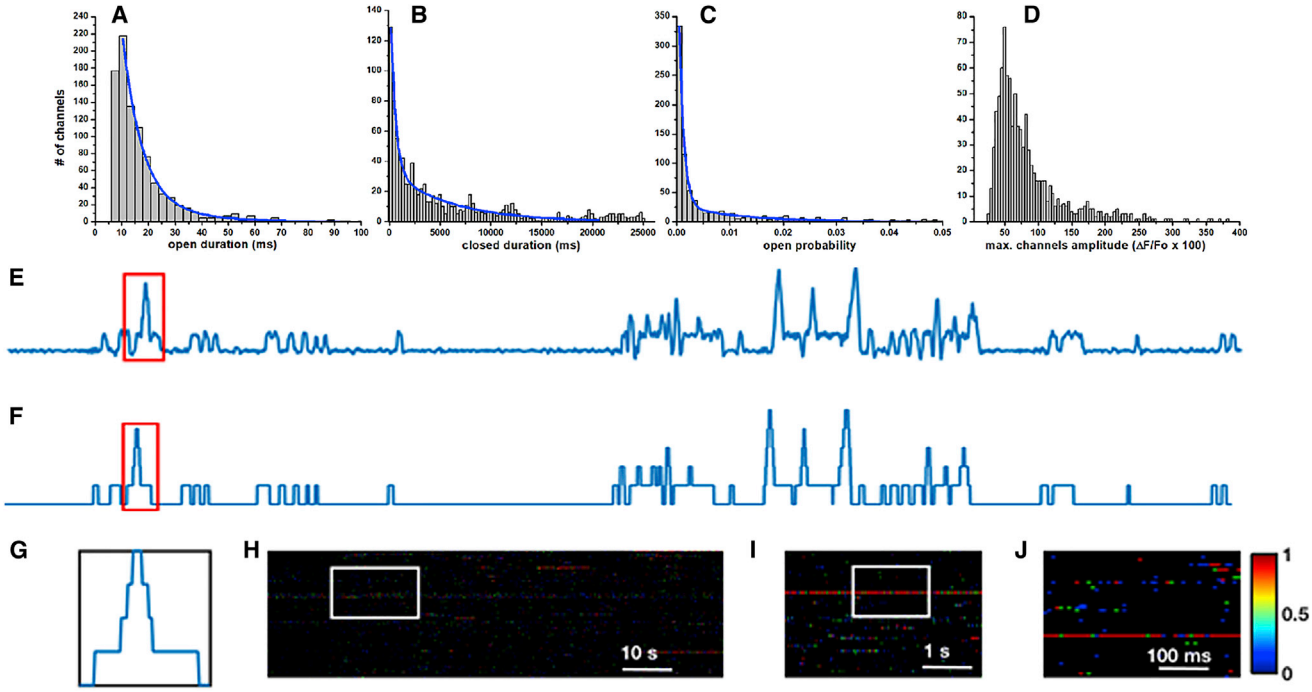

FIGURE 6 Characterizing the activity of Ca^{2+}-permeable ion pores formed by Aβ42 oligomers in a 40 × 40 μm^2 plasma membrane patch of an *X. laevis* oocyte imaged through TIRFM. The stack has 5000 frames recorded at 400 frames/s. In this example, 1 μg/mL of Aβ42 oligomers was applied to the bathing solution, and Ca^{2+} influx to the cytoplasm was enhanced by applying a hyperpolarizing potential of −80 mV. (*A*) Distribution of the channels' mean events duration (channels mean *open* time) for 930 active sites. Double exponential fit (*solid curve*) yields decay constants of 7.2 and 20.1 ms. (*B*) Distribution of the mean closed times (times at which events were not present) fitted by a double exponential decay with constants values of 0.473 and 6034 s, respectively. The plot in (*C*) shows the corresponding distribution of channels P_O fitted by a double exponential decay function (*solid line*), with the curve yielding values of $P_{O1} = 0.0084$ and $P_{O2} = 0.012$. (*D*) The corresponding distribution of maximal amplitude per channel in which variations in amplitude among different channels are evident. (*E*) A sample fluorescence trace representing the Ca^{2+} influx through a single Aβ42 pore. Multiple conductance levels during individual events are indicated in the fluorescence trace and clearly shown in the idealized trace (*F*). (*G*) shows a single event in which the pore opens up to four conductance levels. Channel-chip representation of the gating of all channels detected in the stack (*H*) with progressively zoomed-in versions are shown in (*I*) and (*J*).

the largest amplitude observed for a given channel, whereas that in Fig. 5 *D* represents the amplitudes of all events detected.

Some of these channels have multiple conductance levels. A representative time trace of such a channel is shown in Fig. 6 *E*. An idealized version of the trace using the idealization software TraceSpecks (32) is shown in Fig. 6, *F* and *G*. CellSpecks is also capable of providing "channel chips," an efficient way of plotting the behavior of a large population of ion channels over time (18). The channel-chip representation can also provide a way of viewing selected regions to reveal gating characteristics at a finer resolution. Channel-chip representation for all 830 Aβ42 pores in a 40 × 40 μm plasma membrane patch of *X. laevis* oocyte is shown in Fig. 6 *H*, with zoomed-in versions in Fig. 6, *I* and *J*. As clear from Fig. 6 *C*, majority of these pores have extremely low P_O with mixed mean open durations less than 20 ms (Fig. 6 *A*) separated by close events that are a few thousand milliseconds long (Fig. 6 *B*). The low P_O of Aβ42 pores is also confirmed by the channel-chip representation, showing significantly longer dark stripes (close events) as compared to brighter dots (open events) (Fig. 6, *H–J*).

DISCUSSION

We developed a method, CellSpecks, that allows rapid and accurate detection, localization, and analysis of thousands of functional Ca^{2+}-permeable ion channels imaged by TIRFM in intact cells. Although the program is currently being staged for use in experiments primarily for the detection and analysis of functional Ca^{2+}-permeable channels, we believe CellSpecks is equally applicable to the detection and analysis of other channels and molecules studied through fluorescence microscopy (39). Nevertheless, the development of the software will be ongoing, and new features and capabilities will be incorporated as the need arises. Furthermore, although many find the speed of Java to be limiting in such calculation-heavy programs, Java is approaching the speed of C++ for many applications. Improvements to the Java Virtual Machine and compilers will continue the trend toward C++ and Java speed parity

(40). We currently do not see any issue with the performance or platform specificity; however, if needed for speed, interface, or other technical reasons (such as experimentation with CUDA or OpenCL), porting parts or the entirety of the program to another language will be considered in the future.

CellSpecks' limitations are mostly the result of being built to operate on a particular type of input data. The program only works with signals that deviate from the baseline in a positive direction. However, this can be easily fixed in the source code (available from the authors upon request) if needed by the experiment. The one known confounding scenario for CellSpecks' detection algorithm is in differentiating two large nearby events whose fluorescence overlaps both spatially and temporally. Although for ion channels with low P_O, encountering such a scenario is rare, in cases where the P_O is high, such situations may arise. The solution to this problem will likely come as an event filter that splits into discrete events any event with two or more distinct centroid peaks. Resolving this issue will be coupled with incorporating a feature for tracking the movement of channels (or fluorescence molecules) in the future.

Although the specificities discussed above are limitations to the flexibility of the program, they are in many ways what makes this program useful. Using a line-scan detection software like SparkMaster would be ineffective for analyzing 2D image stacks (21). Similarly, 2D image stack detection software such as GMimPro, developed for Windows operating system, does a very good job at tracking single molecules but is inefficient when a comprehensive statistical analysis for thousands of channels in a single stack is sought because of the significant postprocessing required to extract these statistics (20). Furthermore, the detection of multiple conductance levels and channels with extremely small P_O is beyond the scope of GMimPro because it excludes real events when short-lived false events are excluded.

CellSpecks is also advantageous over FLIKA (23) when it comes to the detection of individual channels. CellSpecks focuses on detecting the activity of many simultaneously active single channels in a cell, whereas FLIKA is written for analysis of Ca^{2+} puffs resulting from the concentrated gating of clustered IP_3R channels. Apart from differences in the temporal and spatial properties of single-channel events and Ca^{2+} puffs these programs were written to detect, there are a few notable differences between the two detection paradigms. CellSpecks computes noise threshold pixel by pixel, thus eliminating the need for the user-specified baseline level. FLIKA, on the other hand, has several thresholds (subtraction of baseline intensity, width of Gaussian filter, frequency thresholds for temporal bandpass filtering, etc.) that the user must employ before initializing puff detection. Selection of meaningful thresholds can be a trial and error process. Because the choice of thresholds is user-dependent, it introduces some arbitrariness in predetection processing of the stack and can be time consuming. FLIKA, on the other hand, is better equipped to deal with changing baselines.

Developing flexible tools for biological discovery drives the need for a very specific set of features. This lends itself to actively developing a piece of software simultaneously to meet these needs. In this case, we had two primary requirements for any single piece of software. The first requirement is the ability to accurately detect and localize every channel in the image field. This could not be done manually or by existing packages, especially when the channels have extremely low P_O as mentioned above. The second requirement—the ability to export and measure any number of behavioral characteristics—is only solved by being able to design and extract novel measurements from the detection data within the program, a feature not available in any multiframe analysis software that meets our detection requirements. An additional advantage to home-built software is the ability to easily incorporate statistical routines into the exported material, including parameter correlation and the ability to export extremely large sets of data in virtually any format. CellSpecks allows us to meet both of these requirements and with a public release provides a tool for researchers solving similar problems.

Although manual analysis was clearly missing some small or infrequent events that became apparent on a second close inspection, the sheer number of these legitimate channels, for example, as depicted in Fig. 4, expounds the importance of automatic detection and analysis. Without these channels, an analysis of Ca^{2+} flow would be incomplete at best. With this data, we can confidently analyze the population P_O and maximal amplitudes. Parameters such as P_O—the distribution of which is shown, for example, in Fig. 5 *C*—prove meaningful when compared to the same channel population after the addition of, for instance, an nAChR antagonist. Other parameters—such as the mean open time, mean closed time, maximal amplitude distributions for each channel (e.g., Fig. 6, *A*, *B*, and *D*), and lifetimes and amplitudes of all events detected in the recording (e.g., Fig. 6, *A*, *B*, and *D*) that CellSpecks stores—can be used in much the same manner to infer meaningful relationships. Our ability to determine temporal relationships from the channel-chip representation (for example, Fig. 6 *H*) and spatial relationships from channel location maps (for example, Fig. 2, *left panel*) hold endless potential as a beneficiary of CellSpecks' ability to retain and reuse detection information in memory. Importantly, all this information can be accessed with a few clicks without selecting or adjusting any parameters in the software.

Though the need for yet another data or image analysis program is not the first conclusion when confronted with a new type of experimental data, the scarcity of signal-detection software capable of accepting video data sets and accounting for variable experimental parameters such as spot size, duration and intensity drove the desire to have a flexible framework within which current and future

experimenters could design and execute sophisticated algorithms and statistical analysis. CellSpecks, the program developed for the purpose of automating localized cell membrane Ca^{2+} event detection, is in the short term an effective tool but more importantly a good basis for future development.

SUPPORTING MATERIAL

Supporting Materials and Methods, one video, and one data file are available at http://www.biophysj.org/biophysj/supplemental/S0006-3495(18)31163-9.

AUTHOR CONTRIBUTIONS

S.I.S. performed research, contributed analytic tools, analyzed data, and wrote the manuscript. M.S. performed research, contributed analytic tools, analyzed data, and wrote the manuscript. D.S. performed research, contributed analytic tools, analyzed data, and wrote the manuscript. I.P. designed research and wrote the manuscript. G.U. designed research, contributed analytic tools, analyzed data, and wrote the manuscript. A.D. designed research, contributed analytic tools, analyzed data, and wrote the manuscript.

ACKNOWLEDGMENTS

This works was supported by the National Institutes of Health through grants R01 AG053988 (to A.D. and G.U.) and R01 GM065830 (to I.P.).

REFERENCES

1. Bootman, M. D., T. J. Collins, …, P. Lipp. 2001. Calcium signalling—an overview. *Semin. Cell Dev. Biol.* 12:3–10.
2. Berridge, M. J. 2012. Calcium signalling remodelling and disease. *Biochem. Soc. Trans.* 40:297–309.
3. Berridge, M. J., M. D. Bootman, and P. Lipp. 1998. Calcium–a life and death signal. *Nature.* 395:645–648.
4. Bezprozvanny, I. 2009. Calcium signaling and neurodegenerative diseases. *Trends Mol. Med.* 15:89–100.
5. Demuro, A., I. Parker, and G. E. Stutzmann. 2010. Calcium signaling and amyloid toxicity in Alzheimer disease. *J. Biol. Chem.* 285:12463–12468.
6. Toglia, P., K. H. Cheung, …, G. Ullah. 2016. Impaired mitochondrial function due to familial Alzheimer's disease-causing presenilins mutants via Ca^{2+} disruptions. *Cell Calcium.* 59:240–250.
7. Toglia, P., and G. Ullah. 2016. The gain-of-function enhancement of IP_3-receptor channel gating by familial Alzheimer's disease-linked presenilin mutants increases the open probability of mitochondrial permeability transition pore. *Cell Calcium.* 60:13–24.
8. Berridge, M. J., M. D. Bootman, and H. L. Roderick. 2003. Calcium signalling: dynamics, homeostasis and remodelling. *Nat. Rev. Mol. Cell Biol.* 4:517–529.
9. Clapham, D. E. 2007. Calcium signaling. *Cell.* 131:1047–1058.
10. Clapham, D. E. 1995. Calcium signaling. *Cell.* 80:259–268.
11. Demuro, A., and I. Parker. 2003. Optical single-channel recording: imaging Ca2+ flux through individual N-type voltage-gated channels expressed in Xenopus oocytes. *Cell Calcium.* 34:499–509.
12. Demuro, A., and I. Parker. 2004. Imaging single-channel calcium microdomains by total internal reflection microscopy. *Biol. Res.* 37:675–679.
13. Demuro, A., and I. Parker. 2006. Imaging single-channel calcium microdomains. *Cell Calcium.* 40:413–422.
14. Demuro, A., and I. Parker. 2005. Optical single-channel recording: imaging Ca^{2+} flux through individual ion channels with high temporal and spatial resolution. *J. Biomed. Opt.* 10:11002.
15. Navedo, M. F., and L. F. Santana. 2013. CaV1.2 sparklets in heart and vascular smooth muscle. *J. Mol. Cell. Cardiol.* 58:67–76.
16. Ullah, G., A. Demuro, …, J. E. Pearson. 2015. Analyzing and modeling the kinetics of amyloid beta pores associated with Alzheimer's disease pathology. *PLoS One.* 10:e0137357.
17. Smith, I. F., and I. Parker. 2009. Imaging the quantal substructure of single IP_3R channel activity during Ca^{2+} puffs in intact mammalian cells. *Proc. Natl. Acad. Sci. USA.* 106:6404–6409.
18. Demuro, A., M. Smith, and I. Parker. 2011. Single-channel Ca^{2+} imaging implicates Aβ1–42 amyloid pores in Alzheimer's disease pathology. *J. Cell Biol.* 195:515–524.
19. Demuro, A., and I. Parker. 2005. "Optical patch-clamping": single-channel recording by imaging Ca^{2+} flux through individual muscle acetylcholine receptor channels. *J. Gen. Physiol.* 126:179–192.
20. Mashanov, G. I., and J. E. Molloy. 2007. Automatic detection of single fluorophores in live cells. *Biophys. J.* 92:2199–2211.
21. Picht, E., A. V. Zima, …, D. M. Bers. 2007. SparkMaster: automated calcium spark analysis with ImageJ. *Am. J. Physiol. Cell Physiol.* 293:C1073–C1081.
22. Steele, E. M., and D. S. Steele. 2014. Automated detection and analysis of Ca^{2+} sparks in x-y image stacks using a thresholding algorithm implemented within the open-source image analysis platform ImageJ. *Biophys. J.* 106:566–576.
23. Ellefsen, K. L., B. Settle, …, I. F. Smith. 2014. An algorithm for automated detection, localization and measurement of local calcium signals from camera-based imaging. *Cell Calcium.* 56:147–156.
24. Cheng, H., L. S. Song, …, M. D. Stern. 1999. Amplitude distribution of calcium sparks in confocal images: theory and studies with an automatic detection method. *Biophys. J.* 76:606–617.
25. Kong, C. H., C. Soeller, and M. B. Cannell. 2008. Increasing sensitivity of Ca^{2+} spark detection in noisy images by application of a matched-filter object detection algorithm. *Biophys. J.* 95:6016–6024.
26. v Wegner, F., M. Both, and R. H. Fink. 2006. Automated detection of elementary calcium release events using the á trous wavelet transform. *Biophys. J.* 90:2151–2163.
27. Bányász, T., Y. Chen-Izu, …, L. T. Izu. 2007. A new approach to the detection and statistical classification of Ca^{2+} sparks. *Biophys. J.* 92:4458–4465.
28. Bray, M. A., N. A. Geisse, and K. K. Parker. 2007. Multidimensional detection and analysis of Ca^{2+} sparks in cardiac myocytes. *Biophys. J.* 92:4433–4443.
29. Francis, M., X. Qian, …, M. S. Taylor. 2012. Automated region of interest analysis of dynamic Ca^{2+} signals in image sequences. *Am. J. Physiol. Cell Physiol.* 303:C236–C243.
30. Shkryl, V. M., L. A. Blatter, and E. Ríos. 2012. Properties of Ca^{2+} sparks revealed by four-dimensional confocal imaging of cardiac muscle. *J. Gen. Physiol.* 139:189–207.
31. Gnanasambandam, R., M. S. Nielsen, …, J. K. Dreyer. 2017. Unsupervised idealization of ion channel recordings by minimum description length: application to human PIEZO1-channels. *Front. Neuroinform.* 11:31.
32. Shah, S. I., A. Demuro, …, G. Ullah. 2018. TraceSpecks: a software for automated idealization of noisy patch-clamp and imaging data. *Biophys. J.* 115:9–21.
33. Bruno, W. J., G. Ullah, …, J. E. Pearson. 2013. Automated maximum likelihood separation of signal from baseline in noisy quantal data. *Biophys. J.* 105:68–79.
34. Dargan, S. L., A. Demuro, and I. Parker. 2006. Imaging Ca^{2+} signals in Xenopus oocytes. Xenopus Protocols. Springer, pp. 103–119.

35. Lock, J. T., I. Parker, and I. F. Smith. 2015. A comparison of fluorescent Ca^{2+} indicators for imaging local Ca^{2+} signals in cultured cells. *Cell Calcium.* 58:638–648.

36. Johnson, J. B. 1925. The Schottky effect in low frequency circuits. *Phys. Rev.* 26:71–85.

37. Timmer, J., and M. Koenig. 1995. On generating power law noise. *Astron. Astrophys.* 300:707–710.

38. Demuro, A., E. Palma, …, R. Miledi. 2001. Inhibition of nicotinic acetylcholine receptors by bicuculline. *Neuropharmacology.* 41:854–861.

39. Reck-Peterson, S. L., N. D. Derr, and N. Stuurman. 2010. Imaging single molecules using total internal reflection fluorescence microscopy (TIRFM). *Cold Spring Harb. Protoc.* 2010:pdb.top73.

40. Lewis, J., and U. Neumann. 2009. Performance of Java versus C++. http://scribblethink.org/Computer/javaCbenchmark.html.