

An image analyser program written in MATLAB to detect and analyse calcium sparks

László Zsolt Szabó^{ab}, Beatrix Dienes^b, László Csernoch^b

^aDepartment of Electrical Engineering, Sapientia Hungarian University of Transylvania
e-mail:lszabo@ms.sapientia.ro

^bDepartment of Physiology, University of Debrecen, Debrecen, Hungary
e-mail:{dienes,csl}@phys.dote.hu

Abstract

In muscle cells calcium sparks and embers are elementary intracellular events of excitation-contraction coupling. Study of these events is achieved through confocal microscopy based on fluorescent dyes. The large collection of images in an experiment need automated procedures to detect, analyse and store characteristic data of these elementary events. We developed a computer program written in MATLAB to solve these problems for striated muscle cell experiments. The program is capable of analysing a large number of line scan (time-space) images, to detect the morphology and to compute the characteristic spatial and temporal parameters of the events. The detection of the events on an image is automatic, permitting also interactive user intervention. The parameters of calcium sparks are stored in a relational database allowing the retrieval of data for statistical analysis of large image sets.

Keywords: calcium sparks, confocal microscopy, biological image analysis, MATLAB

MSC: 92C30, 92C05

1. Introduction

In cell biology calcium is known as the most universal signal carrier. Excitation-contraction coupling (ECC) in muscle cells is triggered by the opening of internal calcium stores and finally leads to the contraction of the muscles. The random opening of the calcium release channels (ryanodine receptors, RyR) produce elementary calcium release events, also called calcium sparks. The study of these events is an ongoing study over the last decade [1, 7] primary on frog and rat

muscle cells. At the moment not all the steps of ECC are completely clear. Any tool which could give further insight into the functioning of RyR would make the picture more detailed. The current research tools in our department are TPEN, a calcium chelator, and Maurocalcine, a scorpion toxin. These drugs have been shown to modify the gating of RyR. In recent studies [2], we explored their effects on elementary calcium release events in frog skeletal muscle fibers.

Calcium sparks have been studied by laser scanning confocal microscopy used in the line scan mode. Line scan images provide a two dimensional image, recorded in space and time, with an optical resolution under one micrometer, and a scanning frequency near to one millisecond. The released calcium (sparks) is measured with calcium sensitive fluorescent dyes. Because of the nature of the sampling process in confocal images (scanning on a well located spatial line), and the variety of release event profiles, statistical methods must be used and several hundreds of detected sparks are needed for calculations. Laser scanning microscopes can produce large quantity of images; therefore procedures have been developed for the detection of sparks in images and spark parameter calculations [3, 4, 5, 8]. The main goal of these methods has been to achieve a fully automatic detection of these events. However, due to the large variety in their morphology, emphasis has been also placed on interactive procedures [5].

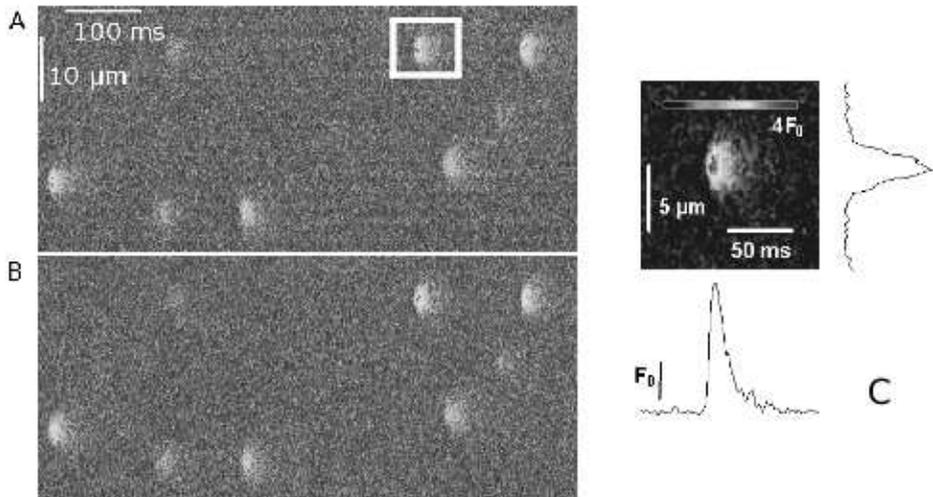


Figure 1: Line scan image, (A) raw image, (B) normalized image F/F_0 , (C) the selected, white box in A, spark with spatial spread and time course

Detection and spark parameter calculation represent only one aspect of the procedures applied to the images, other image processing and data storage procedures are also needed during the analysis process (like region selection of images to perform other calculations or just to handle them for other purposes, like image

processing for documents). Therefore we have developed a program in MATLAB (The MathWorks Inc.), partly to have a fully functional Graphical User Interface (GUI) based program to carry out spark detection, on the other hand to have a set of MATLAB based functions used for rapid prototyping and calculations when working with confocal images. As input our program works with images provided by the Zeiss LSM 510 META microscope, but other image formats may be easily adapted. Output of experimental data is saved in files (as spreadsheet program inputs) and is stored in a relational database.

2. Experimental methods

For Ca^{2+} spark measurements, single fibers from the semitendinosus muscle of the frog (*Rana esculenta*) were mechanically dissected and mounted into a chamber with glass bottom as described earlier [2]. The fibers were loaded with 0.1 mM fluo-3 to record the calcium release events. Fibers were imaged using an LSM 510 META laser scanning confocal microscope (Zeiss, Oberkochen, Germany) as described earlier [2]. Line scan images ($[t, x]$) were taken at 1.54 ms/line and 512 pixels/line (pixel size 0.142 μm) with a 63x water immersion objective (1.2 n.a., Zeiss). Fluo-3 was excited with an Argon ion laser (at 488 nm, 1.5 mW), emitted light was collected through a band-pass filter and digitized at 12 bit. To avoid photo-damage of the muscle fiber, the same position was not imaged twice. A typical line scan image is presented in Figure 1 A.

3. Computational methods

3.1. Used software

For program development and calculations MATLAB R12 (The MathWorks Inc.) was used. The graphical interface for interactive procedures was developed with the MATLAB GUI. For image processing the MATLAB Image Processing Toolbox and functions from the DIPimage Toolbox (Quantitative Imaging Group, Delft University of Technology), and for storage of calculations a PostgreSQL relational database were used. Our decision to use MATLAB has been influenced by the powerful mathematical and image processing capability of the MATLAB environment, and the ability of rapid prototyping of interactive, GUI driven applications.

3.2. Program components

As input the program works with image datasets produced by a Zeiss LSM 510 META laser scanning microscope. Though it may handle individual images, the analysis is performed on sets of images obtained from one experiment (number of images between one and few hundreds). The analysis program is based on a MATLAB GUI application. The main modules permit: 1. selection of a dataset

of images for analysis; 2. analysis of the whole dataset in batch mode; analysis of individual images with user interaction in a separate window; 3. a module to handle, process and export images for various other tasks than event detection; 4. module to save experimental data in files and in a relational database.

3.3. Implementation of the detection algorithm

The algorithm is based on the original algorithm by Cheng et al. [3], with modifications. The program can handle images up to 1024 pixels in spatial and 8192 in temporal directions. The $F(t, x)$ image ($t = 1, N; x = 1, M$) is loaded from the original Zeiss LSM image with pixel values coded on 12 bits (integers, see Figure 1A). Then the detection algorithm is run, and finally the image is normalized to the baseline fluorescence value (see Figure 1 B) in each spatial x position (columns of the image). The original image is filtered with a 3x3 median and optionally with a smoothing filter. The pixels above $mean + 2 \cdot \sigma$ are excluded from the image (the mean is calculated for each column). The exclusion is repeated until candidate pixels for exclusion exist. From the excluded pixels a binary mask is formed, in which connected regions are searched. Small connected areas are dropped, resulting in $B(t, x)$ binary mask. Before the normalization, a correction for photobleaching is applied ($P(t) = mean(F(t, x)), B(t, x) = 0; k(t) = P(1)/P(t); F_c(t, x) = k(t) \cdot F(t, x)$). Then the original image (with pixels above threshold excluded) is normalized to the mean of the baseline fluorescence for each column ($F_0(x) = mean(F(t, x)), B(t, x) = 0; F_n(t, x) = F_c(t, x)/F_0(x), t = 1, N$). The user of the program can choose between two decision criteria in accepting the connected regions as spark positions. First one is a decision based just on the size of the connected areas (criteria are configurable by the user). The second is the decision from the original algorithm: the values $mean2$ and $\sigma2$ are calculated on the normalized image without the excluded connected regions. The regions are accepted as spark locations, if pixel values above $mean2 + CRI \cdot \sigma2$ exist in the region. The criterion CRI is a number above 2, for best detections CRI between 3.5 and 3.8 is used [3]. The binary image with the location of the sparks may be modified by this decision, finally the normalized image is recalculated and the new exclusion mask is used. The normalized image, the binary mask and threshold values are stored for later use.

3.4. Calculation of spark profile data

In order to obtain statistics and to classify the detected sparks, the spatial and time profile of the spark is analyzed by the program (see Figure 1 C), and characteristic data is extracted. For calculations we used the method provided by Hollingworth et al. [6] with small modifications, based on the $(\Delta F/F_0)(t, x) = (F_n(t, x) - F_0(x))/F_0(x), t = 1, N$ image. For the time profile of the spark the average of three lines are selected from the center of the spark, and this line is used as characteristic time course of the spark. The line was extended with 6 pixels before and 20 pixels after the area of the spark. If the pixel with the maximal

intensity from the region of the spark is not part of these lines, the position of the three lines is shifted half distance towards the pixel with maximal intensity value from the spark area. Having t_1 as the first pixel on the central time line above the threshold value (used by location of the region of the spark), and t_2 as the pixel with maximal intensity, two exponentials (see [6])

$$\begin{aligned}(\Delta F/F_0)(t) &= A_0 & t < t_1 \\(\Delta F/F_0)(t) &= A_0 + A_1 \cdot (1 - \exp(\frac{-(t - t_1)}{\tau_{on}})) & t_1 < t < t_2 \\(\Delta F/F_0)(t) &= A_0 + (A_1 \cdot (1 - \exp(\frac{-(t_2 - t_1)}{t_{on}})) - A_2) \cdot \exp(\frac{-(t - t_2)}{\tau_{off}}) \\ &+ A_2 & t > t_2\end{aligned}$$

were least squares fitted to the time course, the first for the rising phase of the spark (between t_1 and t_2), and the second for the failing phase, between t_2 and the offset of the spark. If fitting succeeds, characteristic data for the time course of the spark is obtained: the rise time of the spark ($t_2 - t_1$), the time to peak (from 10% over threshold up to peak) and the full time at half maximum (FTHM) of the spark: the distance between the rising and the falling exponential at half of the amplitude of the time line (where the exponentials meet). For the spatial profile of the spark a Gaussian:

$$(\Delta F/F_0)(x) = A_3 + A_4 \cdot \exp(\frac{-(x - x_0)^2}{2 \cdot var})$$

was fitted to the average of three spatial lines (beginning from the line intersecting t_2). If fitting succeeds, the full width at half maximum value is calculated:

$$FWHM = \sqrt{8 \cdot \ln(2) \cdot var}$$

The amplitude of the curve is considered the amplitude of the spark.

3.5. Interactive procedures

Despite the fact that biological image analysis tends towards fully automated procedures user intervention is still a common requirement. Biological image data contain a large variety of image patterns, some of them are not detected or detection algorithms provide false positive detections. Our interface consists of a main window and several others, started from the main to accomplish various tasks, including: inspecting, adding and deleting events, processing and exporting image data, launching calculations. Sparks may be visualized individually (see Figure 2) by a mouse click through their spatial and time profile and by three dimensional representation (space, time, amplitude). In order to analyse, datasets obtained during an experiment should be selected from the GUI. It is possible to analyse a large image step by step from the GUI, inspecting the result after each image, or to run an analysis script in batch mode, and inspect the results later.

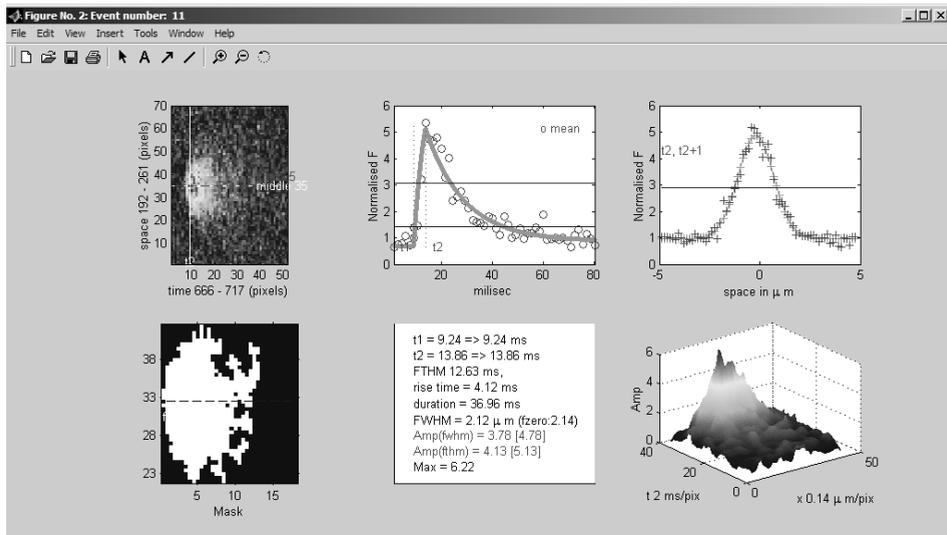


Figure 2: Display of spark data (Graphical user interface)

3.6. Database model

Results of an experiment are stored in a relational database with the calculated elementary event data (characteristic values for sparks: full width at half maximum (FWHM), full time at half maximum (FTHM), rise time, amplitude, location: experiment, image, space and time). The program generates inputs for spreadsheet programs, but also for a relational database. A simple database stores dataset and image description and the linked morphological data for elementary events. SQL query language is used to detect event sets with common properties across experiments.

4. Results

4.1. Testing the efficiency of the algorithm

In order to test the efficiency and the limits of the algorithm we prepared test images with calcium sparks with known amplitudes. These were obtained from a typical, created by averaging areas from the normalized images with similar events (in amplitude, rise time, FWHM, FTHM). The image of the spark was normalized, and placed on the test images scaled to simulate different amplitudes. Random gaussian noise was added. We considered a signal-to-noise ratio (SNR) for the images between 2 and 4 (see [8]). Tests were run for 10 images each with 36 randomly positioned sparks, with SNR=2, 3 and 4. At SNR=3, with a test criterion

of $\text{CRI}=3.6$ the program could detect 80% of the sparks with amplitude $0.6 \Delta F/F$, 100% of detection occurred at amplitudes $> 0.75 \Delta F/F$ (see Figure 3 A. Label of each bin represents the mean value for $\Delta F/F$ amplitudes). Amplitude of sparks found on our experimental images range from 0.2 to $4 \Delta F/F$.

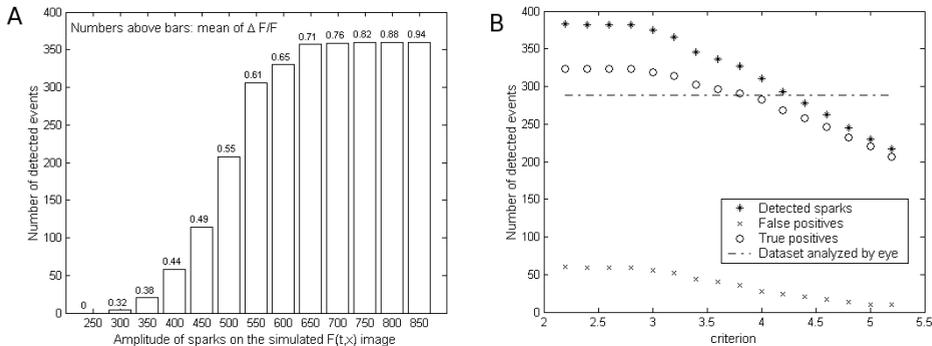


Figure 3: Results on simulated (A) and experimental images (B)

4.2. Usage of the program on experimental images

We present the results of a test run on a set of 20 images from an experiment (study of the effect of maurocalcine), containing 288 sparks detected by eye. In Figure 3 B we present the results of test runs using CRI values between 2.2 and 5.2. The number of true positiv detections (TP, sparks for which parameters could be determined by the program) and the number of false positives (FP, where fitting algorithms did not succeed) both decrease with larger CRI value, the value of TP-FP has maximal values for CRI values between 3.4 and 3.6. For criterion $\text{CRI}=3.6$, 336 events were detected by the program (with 40 false positives). Revised by an experimenter, 92% of the TP detections were accepted, 8% were modified manually (merged events or bad parameter calculation).

5. Discussion, conclusion

Detection algorithms used in most spark analyzer programs are based on the threshold method provided by Cheng et al. [3]. Modified versions of the algorithm were developed [4, 8], in order to assure a good detection rate on images recorded in various studies (smooth or skeletal muscle cells of rat or frogs). To handle user intervention and easy analysis parameter modification, graphical user interfaces have been provided for some of these programs [5]. In a solution using the MATLAB environment we have developed a set of functions for MATLAB for confocal image analysis. The MATLAB functions may be linked to process calculation chains

which are usable from the GUI, due to the rapid user interface building capabilities of the MATLAB environment. Detection criteria based only on the size of the events results in a lower detection rate. The threshold based detection strategy using CRI2 may be configured to achieve a high detection rate, near to 95% when all adjustable detection parameters are properly set and signal-to-noise ratio of the image is between 2 and 4.

References

- [1] KLEIN, M. G., SCHNEIDER, M. F., Ca²⁺ sparks in skeletal muscle (review). *Progress, Biophysics and Molecular Biology*, 92, (2006), 308–332.
- [2] CSERNOCH, L., FODOR, J., FEHÉR, M., DIENES, B., DELI, T., SZENTESI, P., SZABÓ, L., MARTY, I., Suppressed SR calcium release and modified elementary calcium release events in Triadin-overexpressing cultured myotubes, *Biophysical Journal*, 92, (2007), Suppl., 79a.
- [3] CHENG, H., SONG, L. S., SHIROKOVA, N., GONZALEZ, A., LAKATTA, E. G., RIOS, E., STERN, M. D., Amplitude Distribution of Calcium Sparks in Confocal Images: Theory and Studies with an Automatic Detection Method, *Biophysical Journal* Volume 76 February (1999), 606–617.
- [4] SONG, L. S. , STERN, M. D., LAKATTA, E. G. , CHENG, H., Partial depletion of sarcoplasmic reticulum calcium does not prevent calcium sparks in rat ventricular myocytes. *Journal of Physiology* (1997), 505.3, pp. 665–675.
- [5] SEBILLE, S., CANTEREAU, A., VANDEBROUCK, C., BALGHI, H., CONSTANTIN, B., Calcium sparks in muscle cells: interactive procedures for automatic detection and measurements on line-scan confocal images series, *Computer Methods and Programs in Biomedicine* Volume 77, (2005/1).
- [6] HOLLINGWORTH, S., PEET, J., CHANDLER, W. K., BAYLOR, Calcium Sparks in Intact Skeletal Muscle Fibers of the Frog, *J. Gen. Physiology* Volume 118 December (2001), 653–678.
- [7] CSERNOCH, L., Sparks and embers of skeletal muscle: the exciting events of contractile activation, *European Journal of Physiology*, DOI 10.1007/s00424-007-0244-0.
- [8] PICT, E., ZIMA, A., BLATTER, L. A., BERS, D. M., SparkMaster - Automated Calcium Spark Analysis with ImageJ, *Article in Press. American Journal of Physiology, Cell Physiology* , (March 21, 2007), doi:10.1152/ajpcell.00586.2006.

László Zsolt Szabó

PO Box 4, Op.9, Tirgu Mures, R-540485

Beatrix Dienes, László Csernoch

PO Box 22, Debrecen, Hungary, H-4012