Video Article

High-throughput Image Analysis of Tumor Spheroids: A User-friendly Software Application to Measure the Size of Spheroids Automatically and Accurately

Wenjin Chen[2,3], Chung Wong[1,3], Evan Vosburgh[1,3], Arnold J. Levine[3,4], David J. Foran[2,3], Eugenia Y. Xu[1,3]

[1]Raymond and Beverly Sackler Foundation, New Jersey

[2]Histopathology and Imaging Shared Resource, Rutgers University

[3]Rutgers Cancer Institute of New Jersey, Rutgers University

[4]School of Natural Sciences, Institute for Advanced Study, New Jersey

Correspondence to: Eugenia Y. Xu at xueu@rutgers.edu

Abstract

The increasing number of applications of three-dimensional (3D) tumor spheroids as an *in vitro* model for drug discovery requires their adaptation to large-scale screening formats in every step of a drug screen, including large-scale image analysis. Currently there is no ready-to-use and free image analysis software to meet this large-scale format. Most existing methods involve manually drawing the length and width of the imaged 3D spheroids, which is a tedious and time-consuming process. This study presents a high-throughput image analysis software application – SpheroidSizer, which measures the major and minor axial length of the imaged 3D tumor spheroids automatically and accurately; calculates the volume of each individual 3D tumor spheroid; then outputs the results in two different forms in spreadsheets for easy manipulations in the subsequent data analysis. The main advantage of this software is its powerful image analysis application that is adapted for large numbers of images. It provides high-throughput computation and quality-control workflow. The estimated time to process 1,000 images is about 15 min on a minimally configured laptop, or around 1 min on a multi-core performance workstation. The graphical user interface (GUI) is also designed for easy quality control, and users can manually override the computer results. The key method used in this software is adapted from the active contour algorithm, also known as Snakes, which is especially suitable for images with uneven illumination and noisy background that often plagues automated imaging processing in high-throughput screens. The complimentary "Manual Initialize" and "Hand Draw" tools provide the flexibility to SpheroidSizer in dealing with various types of spheroids and diverse quality images. This high-throughput image analysis software remarkably reduces labor and speeds up the analysis process. Implementing this software is beneficial for 3D tumor spheroids to become a routine *in vitro* model for drug screens in industry and academia.

Video Link

The video component of this article can be found at http://www.jove.com/video/51639/

Introduction

Three-dimensional (3D) tumor spheroids are "spherically symmetric aggregates of tumor cells analogous to tissues, with no artificial substrate for cell attachment"[1-3]. The cytology and morphology of the tumor spheroids better mimics the *in vivo* tumor tissue organization and microenvironments than monolayer two-dimensional (2D) cells. 3D tumor spheroids have become a practical *in vitro* model for high-throughput screenings of anti-cancer therapeutic drugs or examining the efficacy of candidate drugs before *in vivo* animal or clinical testing[4]. Clinically, the efficacy of any anti-cancer drug treatment is evaluated based on reduced tumor growth. Analogously, spheroid volume can be used as a measure of the efficacy for *in vitro* cancer drug studies. Spheroid volume ($V = 0.5 * Length * Width^2$) is determined based on the major and minor axial length (more commonly known as length and width) of the spheroids[6,7]. Most researchers have to manually draw the length and width on each spheroid, often using the software offered by microscopy companies and sold together with the imaging instruments. This technique becomes problematic when high-throughput drug screens are performed and more than hundreds of images are produced. Some recent studies reported the use of open source image analysis software toolboxes such as CellProfiler[8-10] and ImageJ[11] to develop rudimentary segmentation routines/macros which involved illumination correction and simple thresholding. These routines often need to be re-adjusted for different batches of images according to the illumination condition and the image contrast change; therefore, these software packages cannot meet the robustness requirement of the high-throughput image analysis. Friedrich and collaborators (2009) used proprietary software to measure the spheroid's volume semi-automatically[5]. The method described in Monazzam and his colleagues' paper[10] was a semi-automated method to measure spheroid's size only for a small numbers of images. Therefore, there exists a clear need for robust, flexible, automated and ready-to-use image analysis tools for the 3D tumor spheroids.

In this study, we describe SpheroidSizer – a MATLAB-based and open-source software application to measure the size of tumor spheroids automatically and accurately. SpheroidSizer is designed to process many different batches of images of 3D spheroids in the same session. Utilizing the active contour algorithm[12-14], SpheroidSizer can tolerate image contrast change, robustly ignore gradual change in background illumination and recognize spheroids in the image. It can also tolerate many usual artifacts, *e.g.*, debris, originated from the specimen. The workflow is designed so that users can perform quality control during or after the computation. Manual overwrite of the analysis result can easily be performed on the spot. Taking advantage of the parallel computing toolbox, the analysis speed can be further boosted by coordinating multiple computing cores to work on the computation simultaneously on a user's computer. Moreover, SpheroidSizer outputs the results in two different forms to allow easy interfacing with downstream analysis tools.

Protocol

1. 3D tumor spheroids formation, drug treatments and image collection are performed as described in our previous paper[15].

2. Software Installation

1. Install licensed MATLAB software onto the computer used for image analysis. The following toolboxes from MATLAB are also required to be installed - Signal Processing toolbox, Image Processing toolbox, and Parallel Computing toolbox* (*needed for parallel computing mode only).
   NOTE: Many universities purchase and maintain group licenses so that the software is free to be downloaded and used by affiliated scientists.
2. Install SpheroidSizer program from SpheroidSizer.zip file (http://pleiad.rwjms.rutgers.edu/CBII/downloads/SpheroidSizer.zip):
   1. Save the zip file in your local file system.
   2. Unzip the SpheroidSizer file.
   3. Save the files in a designated directory/folder, which will be subsequently referred to as the "Installation Directory".
      NOTE: SpheroidSizer has been extensively tested on a Windows 7 operating system. It is expected to work on alternative operating systems with minimal adjustments (not tested).

3. Preparation for the Image Analysis by SpheroidSizer

1. Determine the image scale/resolution of the imaging system (absolute scale of the image in microns per pixel (μm/pix)).
   NOTE: If the size of each pixel on the camera chip is known, the image scale can be computed as *objective magnification x pixel size* (μm/pix). This value can be obtained from the imaging software equipped with the microscope as embedded metadata or with help from the imaging system vendor. This value will be required in Step 4.6.
2. Convert any proprietary image file formats to the accepted file formats - TIFF, JPEG, and other common image file formats.
3. Name image files and arrange directories (**Figure 5A**).
   NOTE: The software relies on the proper layout of directory structure and filenames to reformat the results into plate format:
   1. Name the image files in the following format: [plate name]_[row][column].[extension] or [plate name][space][row][column].[extension]. [row] follows the alphabetical order and [column] follows the numerical order.
      NOTE: Freely available automatic batch renaming software can be found to aid users in this step.
   2. Arrange the directory/folder based on the experiment in the following way: Each experiment should have one directory. Under each experiment's directory, there should be subdirectories for each time point. Under each time point subdirectory, there should be all the images from all plates.
      NOTE: In order for the analysis results to be optimally sorted in the formatted results, we suggest keeping the same number of digits for each identifier by filling 0's on the left, *e.g.*, time points are named as 000H, 072H, and 144H.

4. Image Analysis of the Spheroids by SpheroidSizer

1. Open MATLAB, then open the "Command Window", type cd '[Installation Directory]' and press [RETURN].
2. Type "SpheroidSizer1_0" in the "Command Window" and press [RETURN] to launch the SpheroidSizer program.
3. Click the "Browse" button in the SpheroidSizer1.0 window to select the experiment directory that contains all the images.
4. Select "Include Subfolders" toggle beneath the "Folder" text field to process multiple nested image folders under the designated directory.
   NOTE: If "Include Subfolders" toggle is not selected, only the images directly under the directory are processed, and all of the subfolders are ignored.
5. Select the "On-the-fly display" option to display each segmented image on top of its source image for quality control as the computation is being executed.
   NOTE: Computation speed is faster if "On-the-fly display" option is not chosen.
6. Specify "Resolution" (image scale/resolution in μm/pix) of the analyzed images in the box, for the program to correctly convert the spheroid's measurements from pixel to μm.
   NOTE: All the images in the same folder or analyzed together should be taken under the same microscope with the same objective so that the image scale/resolution remains constant for each experiment.
7. (Optional) users can follow Step 5 for Advanced User-defined Settings.
8. Click "Compute" to start the computation.

NOTE: The software conducts an automatic filename check before proceeding to the computation. If a dialog box shows up indicating - "Error exists in filename", click "Exit and show list of errors" and fix the errors in listed filenames (See Step 3.3). Then, click "Compute" to start the computation again.

9. Click "Pause" button to pause the computation; and computation can be resumed by clicking the same button showing "Continue".
NOTE: The "Results Table" displays the "Folder", "File", "Volume" (in mm^3), "Length" (in μm), "Width" (in μm), and "Valid" (check box) for all of the analyzed spheroids (**Figure 5C**). Volume is calculated based on the measured major axis (length) and minor axis (width) (V=0.5 * Length * Width^2). The "Valid" check box is an option for the user to choose if the analysis of the image is valid or invalid after quality control, see Step 6.

5. Advanced User-defined Settings

1. Click "Advanced" button in the SpheroidSizer1.0 window to bring up the Advanced Configurations window in order to adjust user-defined settings (**Figure 5B**).
2. Enter the filenames of interest in the "Format Output" and "List Output" boxes in the Advanced Configurations window under "Input".
3. Enter a number from "2-10" in the "Reduce" box. This is a coefficient for the software to reduce the image size in computation in order to improve the computation speed. The larger the number is, the faster the computation speed. The default "Reduce" is set to 10.
4. Enter the to-be-processed image file extension in the "Include type" box.
5. Enter the image file extensions or endings that are not going to be processed by the program in the "Exclude type" box as follows: "_crude.jpg".
6. Choose "None" for "Special Color" to process 8 bit and 16 bit color images properly; choose "12 bit" for "Special Color" to process 12-bit color images properly
7. Check "Use Parallel Computing" if the computer used for image analysis is equipped with multiple CPUs and/or multi-core CPUs. If this is true, then go to Step 5.7.1; if not, then skip Steps 5.7.1 and 5.7.2.
NOTE: An error will occur if the computer being used does not support the chosen configuration.
   1. Check the "Use Parallel Computing" option in the Advanced Configurations window.
   NOTE: Only use the parallel computing mode when 4 or more cores are available for the computer.
   2. Enter a number from 4-12 in the "#Workers" (computing cores) box.
   NOTE: This number has to be equal to or less than the number of computing cores in user's computer. A maximum of 12 is imposed by the MATLAB parallel computing toolbox supports the maximum of 12 cores. When parallel computing is being executed, a small dialogue box displays asking the user to wait for the parallel computation to finish; the computation cannot be paused, and nor is the "On-the-fly display" feature being executed in parallel computing mode.

6. Quality Control

1. Click the corresponding cell in the "Results Table" to confirm the accurate boundary contour of a spheroid in the analyzed images,
NOTE: The original and the quality control images will show up on the right side for review. The user can examine all the images sequentially using the down-arrow on the keyboard.
2. Refine the boundary of the spheroid on the selected image using the following two tools, if needed:
   1. Click the "Manual Initialize" button to display the original image. Then click-and-hold the mouse right outside the spheroid and drag the ellipse tool to cover the spheroid on the original image.
   NOTE: The active-contour algorithm initiates using the user-submitted contour and executes to converge on the desired spheroid outline. The "Results Table" will be automatically updated with the new results. The "Manual Initialize" tool allows the user to manually provide initialization for active-contour.
   2. Click the "Hand Draw" button to display the original image. Then use the mouse or a touch enabled screen to precisely draw the boundary of the spheroid.
   NOTE: This outline is directly measured to generate major and minor axes, which are updated in the "Results Table". The "Hand Draw" tool is used only when the "Manual Initialize" tool fails to converge on the desired boundary of the spheroid.
3. Uncheck the checkbox in the "Valid" column in the corresponding row of the "Results Table", when an image does not contain any valid spheroid upon inspection. An "Invalid" label appears at the top-left corner of the quality-control image. If "Valid" is unchecked, the values of all the measurements are empty for the spheroid in the exported formatted and output results files.
NOTE: the following keyboard shortcuts are available to use in the "Result Table": "down arrow" for next image; "v" for valid/invalid; "m" for "Manual Initialize" tool and "h" for "Hand Draw" tool.

7. Saving and Exporting Data

1. Click the "Export Study" button in the SpheroidSizer1.0 window to export the intermediate state of the analysis, if the user needs to exit the software before finishing the project. Specify the name and directory of the file to be saved.
2. Click "Import Study" button to bring back the above intermediate state result from "Export Study" and continue to work on it.
NOTE: The intermediate state files are in a native MATLAB format (.mat) and are not directly readable by any other software programs. A safety feature built into the software makes automatic exports of the open project in case the program exits unintentionally. When needed, the user can find this file, whose name starts with "~tmp" that contains the corresponding time stamp in the [Installation Directory].
3. Click "Format Results" in the SpheroidSizer1.0 window to save the results.
NOTE: Two forms of results are saved in the experiment's directory. The exported filenames can be configured in the Advanced Configurations window (see Protocol Step 5.2). The format output file is a tab-delineated table that organizes the volume value into the

original plate format in the order of an ascending plate number for each time point; and all the time points are organized in an ascending order (**Figure 5D**). The list output file is a tab-delineated table that contains all of the measurements in the form of ordered lists (**Figure 5E**).

Representative Results

SpheroidSizer is designed to produce automated detection, delineation and measurement of the 3D spheroids, with remarkably reduced labor and acutely increased efficiency for large amounts of images. **Figure 1A** shows the workflow of SpheroidSizer. The core computation steps include automated initialization, active contour algorithm and contour quantification. After automated computation, the quality-control feature uses a combination of "Manual Initialize" and "Hand Draw" tools to salvage any imperfect segmentation. **Figure 1B** illustrates the detailed automated active contour algorithm. The initialization step (0th iteration) utilizes basic image processing steps to generate approximate size and location of the spheroid and generate a spherical initiation contour with an estimated size. The initiation contour feeds into the active contour algorithm. In turn it iterates to adjust according to the local image gradient and shape curvature. The active contour algorithm finishes when the contour stabilizes (converges), *i.e.* 477 iterations for this image, or when the pre-defined maximum number of iterations is executed. In this example, the initialization contour is purposely enlarged to better showcase the algorithm. In reality, the initialization is usually very close to the actual boundary and much fewer iterations are needed for the algorithm to converge. Subsequently, the algorithm takes morphometric measurements of the detected spheroid boundary. The major and minor axes of the spheroid are measured using MATLAB image processing toolbox (**Figure 1C**). The major axis is defined as the line segment connecting a single pair of the farthest points on the contour, which is referred to length (L). The minor axis is defined as the longest line perpendicular to the major axis, which is referred to width (W). In this case, the values of L and W are very close since the spheroid is spherical. The volume of the spheroid is calculated as $V = 0.5 *L * W^2$.

One of the features of SpheroidSizer is its automated detection of the boundary of the spheroids even on images with uneven or noisy background utilizing the active contour algorithm (**Figures 2B-D**). Computational processing of bright field images is often plagued by uneven background, which misleads adaptive thresholding-based methods to produce unwanted thresholding results. The issue is especially apparent when multi-well plates are used and walls of the wells may create shadowing effects on the images. However, because the active contour algorithm is not sensitive to the gradual shading change in background, it is able to identify spheroid segmentation in these bright field images with proper initialization. **Figure 2** shows a few examples of images with uneven or noisy background, like uneven illumination (**Figure 2B**), debris (**Figure 2C**) or necrotic core (**Figure 2D**). With automated active contour algorithm, SpheroidSizer delineates these spheroids accurately in all these images as shown in the red contour at the lower panel of each figure.

The quality-control feature of SpheroidSizer is key to a high-throughput workflow. The "Manual Initialize" and "Hand Draw" tools are the valuable complimentary tools for this application. Among hundreds or thousands of images, it is unavoidable that the automated algorithm is not able to correctly detect the spheroids in some images. As illustrated in **Figure 3A**, when improper detection of the spheroid is caused due to the initialization step, *i.e.* improper size or location of the initiation contour in the image (top panel), the "Manual Initialize" tool works by allowing the user to properly define the location and size of the spheroid manually (lower panel). It triggers the active contour algorithm to initiate with manually defined contour and execute to converge on the desired outline. For these tough images like the original image in **Figure 3B**, the spheroid is located in a distracting and noisy background. SpheroidSizer is not able to properly identify the spheroid by the automated method (top panel) or by the "Manual Initialize" tool with proper initialization (middle panel). In this case, the "Hand Draw" tool can be used to manually draw the outline of the spheroid as illustrated in the lower panel. The program uses the user-defined boundary to measure the major and minor axes of the spheroid and calculate the volume. All the corrected results are immediately incorporated into the "Results Table" and can be accordingly exported.

To determine the performance of SpheroidSizer in larger data sets, we first compare operation time by analyzing the same set of 288 images using 1) manual measurements with microscope vendor supplied software; 2) SpheroidSizer with a single-core regular laptop; and 3) SpheroidSizer with a multi-core parallel computing performance workstation. The manual measurements follow our typical protocol prior to developing the software: the length and width of each spheroid are drawn by hand and measured using the vendor program (as seen the red lines in the top panel of **Figure 4A**); then the user copies down the values of the measurements. SpheroidSizer processes each image by generating the spheroid boundary (as shown the red outline in the lower panel of **Figure 4A**), measuring the axial major and minor length, and exporting the results in spreadsheets. As seen in **Table 1**, based on the calculation from 288 images, it takes an average of 31.67 sec to measure one spheroid per image manually; while it only takes SpheroidSizer less than 2 sec when running on a single-core regular laptop; and less than 1 sec when running on a 12-core performance workstation. Therefore, image analysis is over 18x faster per image using SpheroidSizer than manual measurements. It dramatically reduces labor when more than thousands of images are analyzed. Next, we compare the variability in measurements of the 24 spheroids shown in **Figure 4A** between manual measurements and SpheroidSizer. The 24 spheroids are measured three times by both methods; and the standard deviation of each individual spheroid is calculated. As seen in **Figure 4B**, the standard deviation from SpheroidSizer (green line and dots) is close to zero except for the three spheroids that are corrected at the quality-control step, which still show smaller standard deviation than those from the manual measurements method. All these indicate that SpheroidSizer performs the image analysis more efficiently and accurately.

We conducted a drug screen using human BON-1 3D tumor spheroids to find out which compounds in combination with an hsp90 inhibitor are the potential candidates for testing the anti-tumor effects *in vivo*. Human BON-1 3D tumor spheroids were grown on agarose-coated 96-well plates as described in the previous paper[15]. Eight different compounds with six serial dilutions plus media and vehicle were screened for their single and combinatorial effects with 10 nM and 20 nM hsp90 inhibitor in duplicates respectively. Two spheroids were used for each concentration of the individual compound or the combined compounds. Four 96-well plates with total 384 spheroids were used. All the spheroids were imaged at 0, 72, 144, 168, and 192 hr. A total of 1,920 images were produced from this experiment. It took SpheroidSizer only 30 min to complete the computational analysis of the 1,920 images with an additional 50 min for the quality control and data export. SpheroidSizer speeds up the image analysis process enormously. **Figure 5A** shows a screen shot of the folder arrangements and filenames for this experiment as an example for Protocol Step 3.3. **Figures 5B-E** shows the screen shots of the image analysis pop-up windows and results using SpheroidSizer as illustrations for Protocol Steps 4, 5, and 7. Taking the volumes of the 3D spheroids from the formatted results table exported from SpheroidSizer, we made graphs - growth of 3D tumor spheroids upon compound treatments versus time of treatments. Two representative graphs from this experiment are shown in **Figure 5F** and **5G**. Figure 5F shows that the combined treatments of hsp90 inhibitor and cladribine (green line)

inhibit the growth of 3D spheroids more than the single treatment of hsp90 inhibitor (purple line) or cladribine (orange line), suggesting that the combined treatments of hsp90 inhibitor and cladribine may have anti-tumor effects *in vivo*. **Figure 5G** shows that the combined treatments of hsp90 inhibitor and adriamycin (green line) does not inhibit the growth of 3D spheroids more than the single treatment of adriamycin (orange line) or hsp90 inhibitor (purple line), suggesting that the combined treatments of hsp90 inhibitor and adriamycin may not have anti-tumor effects *in vivo*. This experiment helped us better select the compounds to test their anti-tumor effects *in vivo* and SpheroidSizer is the key to the speedy experimental data analysis.

Methods for Image Analysis	Total Time on manual masurements or computation (Second/288 images)	Total Time on Quality Control (Second/288 images)	Total Time on Image Analysis (Second/288 images)	Average Time on Image Analysis (Second/image)
Manual Measurements	7306	1814	9120	31.67
SpheroidSizer in a single-core regular laptop	251	240	491	1.70
SpheroidSizer in 12-core performance workstation	17	180	197	0.68

Table 1. Comparison of operation time on image analysis between manual measurements and SpheroidSizer when analyzing the same set of 288 images. Please click here to view a larger version of this table.
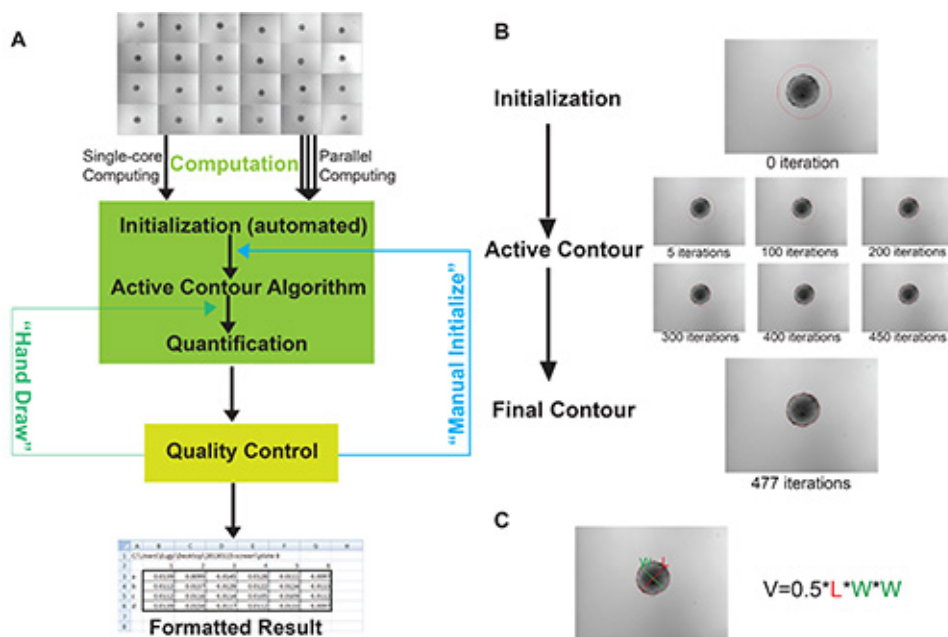

Figure 1. SpheroidSizer - an open-source software application for measuring the spheroid's size. A) The core workflow of the application. **B)** Illustration of the active contour algorithm at different stages of iteration. Please note that the initialization contour (iteration 0) was purposely enlarged in order to showcase the algorithm. **C)** The major and minor axial length measurements and calculation of the volume by SpheroidSizer. L - major axis: the line segment connecting a single pair of farthest points on the contour (referred to length); W - minor axis: the longest line perpendicular to the major axis (referred to width).
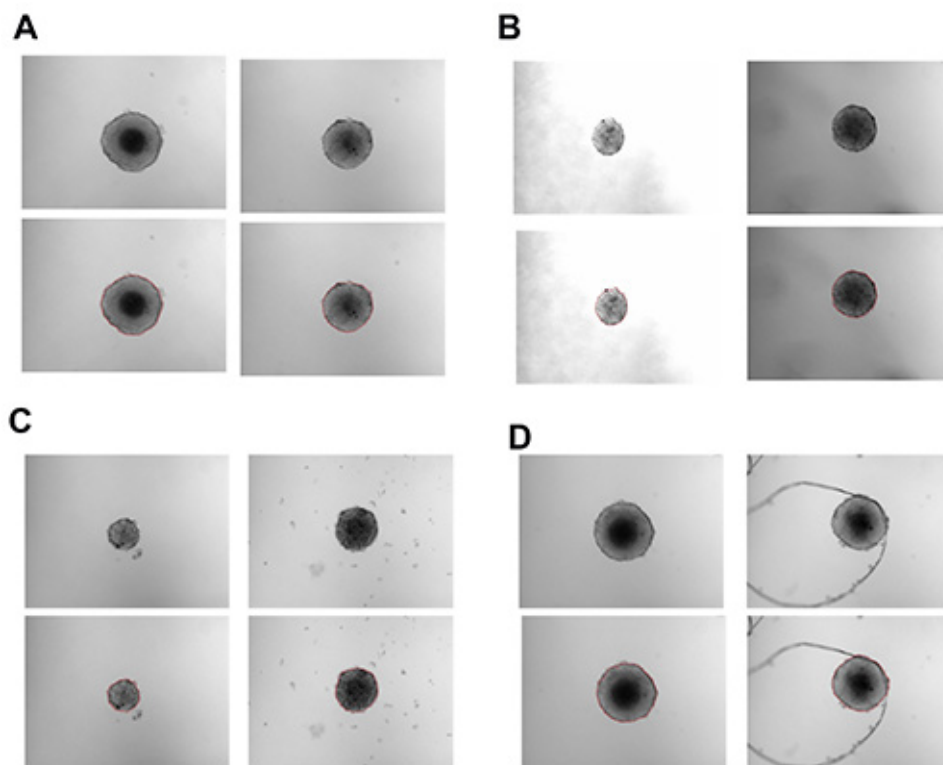
Figure 2. Representative results from the automated segmentation of SpheroidSizer, showing robustness against various image conditions. A) Typical good quality images. **B)** Images with different brightness and contrast. **C)** Images with distracting debris. **D)** Images of spheroids with necrotic core. Images at the top panel of each figure are the source/original images; images at the lower panel of each figure are the quality-control images; and the red outline is the spheroid segmentation drawn by automated computation.
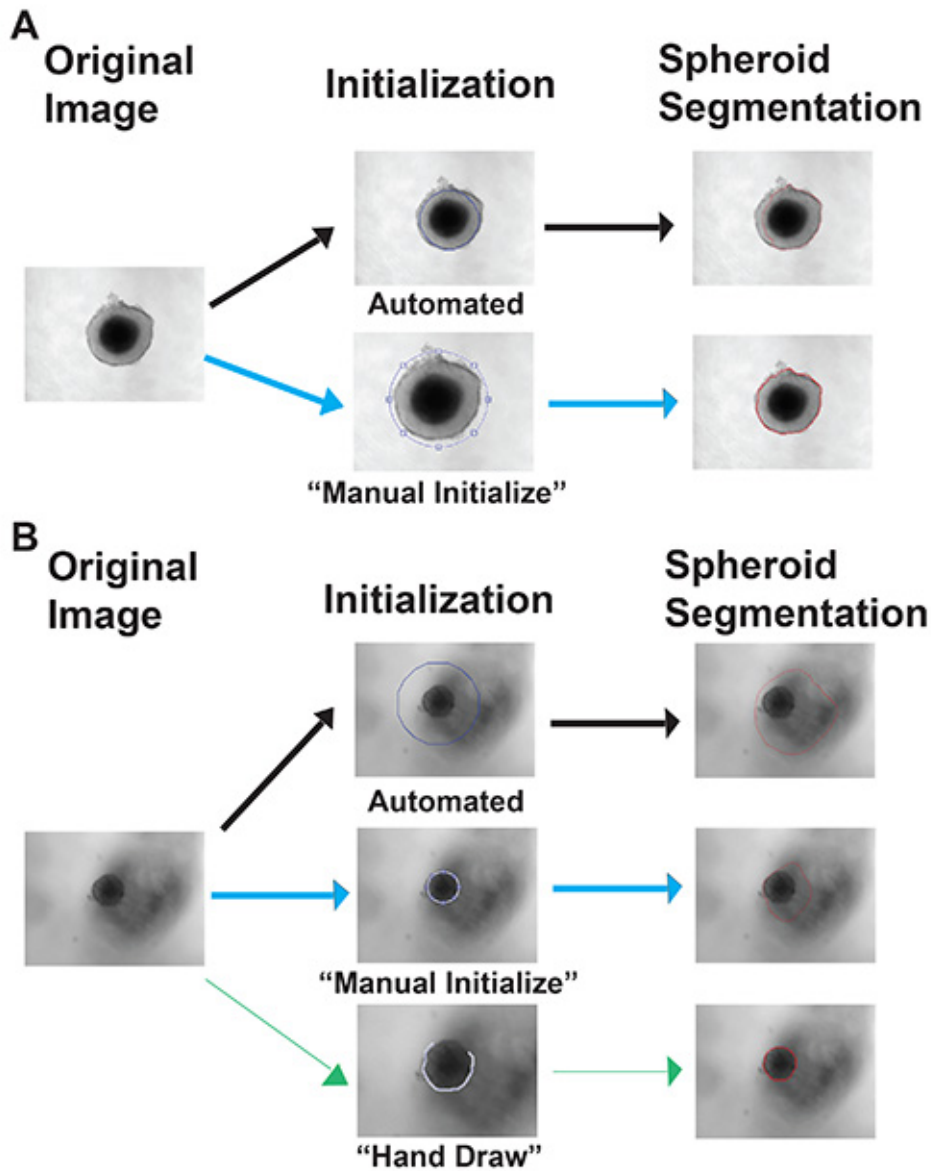
Figure 3. Illustration of the "Manual Initialize" and "Hand Draw" tools. A) The "Manual Initialize" tool allows the drawing of a fitting ellipse shape across the spheroid for initialization, when inaccurate spheroid segmentation occurs after the automated initialization. **B)** The "Hand Draw" tool allows accurate hand drawing of the spheroid boundary, when inaccurate spheroid segmentations occur with both automated and manual initialization. The blue line around the spheroid shows the initialization contour; the red outline is the identified spheroid boundary. Please note that the spheroid in "Manual Initialize" in A) and the spheroid in "Hand Draw" in B) is purposely enlarged to better showcase the tools.
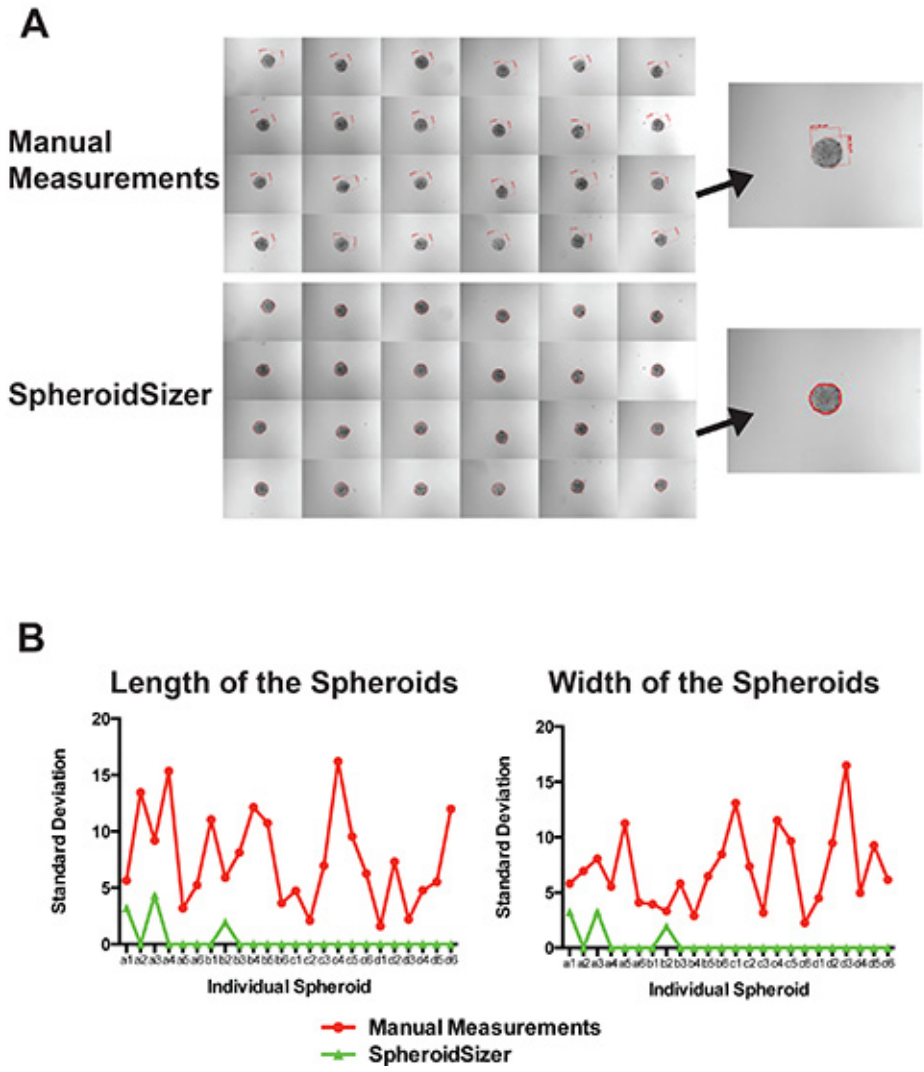
Figure 4. Comparison of image analysis performance between SpheroidSizer and manual measurements when analyzing the same set of 24 images. A) Representative spheroids to show how the length and width of spheroids are determined by manual measurements and SpheroidSizer. Top 24 images contain hand-drawn length/width of each spheroid in red lines using manual measurements; lower 24 images (the same 24 images) contain computer-drawn spheroid boundary in red outline using SpheroidSizer. **B)** Standard deviation of length or width from three measurements on each individual spheroid.

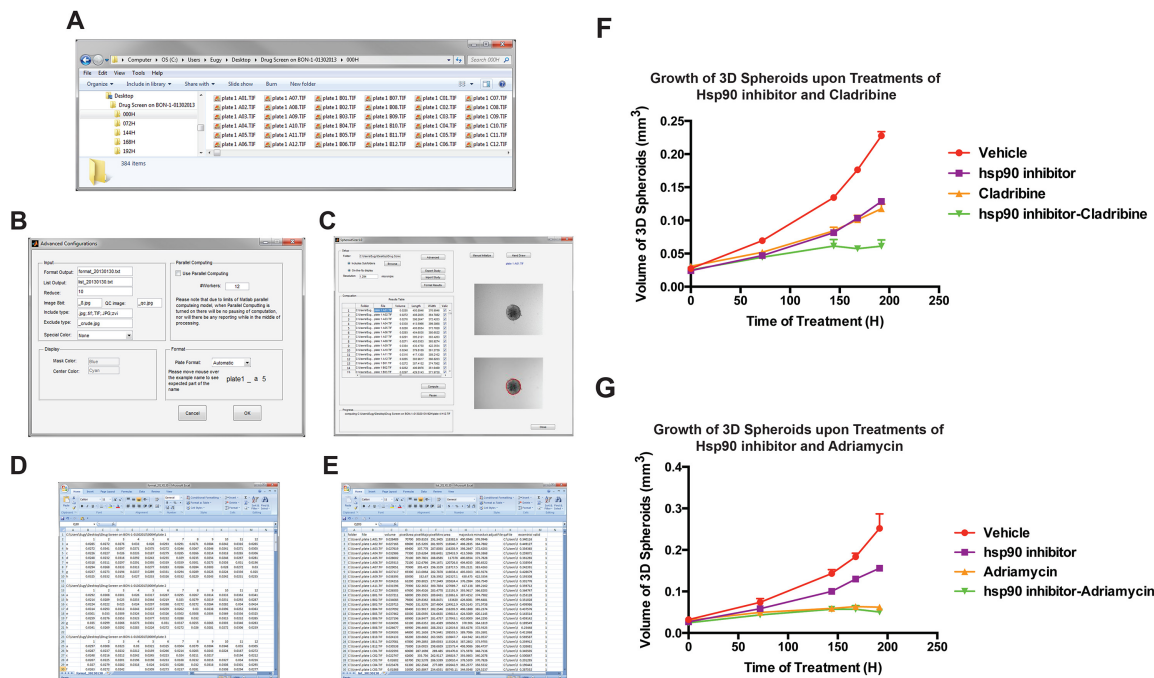July 2014 | 89 | e51639 | Page 8 of 10

Figure 5. A representative example of utilization of SpheroidSizer in drug screen - image analysis on the spheroids' images that were collected from a drug screen using BON-1 3D tumor spheroids. A) A screen shot of the folder arrangements and filenames for this project. **B)** A screen shot of the advanced configurations window in SpheroidSizer. **C)** A screen shot of SpheroidSizer1.0 window with displayed Results Table. **D)** A screen shot of the format output file exported from SpheroidSizer. **E)** A screen shot of the list output file exported from SpheroidSizer. **F)** Growth of 3D tumor spheroids upon the treatments with hsp90 inhibitor and cladribine. **G)** Growth of 3D tumor spheroids upon the treatments with hsp90 inhibitor and adriamycin. Please click here to view a larger version of this figure.

Discussion

This study presents a fast, flexible, effective and automated program - SpheroidSizer for accurate determination of the size of 3D tumor spheroids. SpheroidSizer is easy to use and requires minimal user input. The most critical steps for accurate, smooth and successful running of SpheroidSizer include: that the spheroids are imaged in the center of the field without touching the edge of the well; all the files to be analyzed together as one project should be imaged under the same microscope with the same objective; all the files to be analyzed are correctly named and arranged as indicated in the protocol; and correct user-defined settings are entered before computation.

The advantages of SpheroidSizer include its ability to tolerate gradual background change in the image as well as to generate smooth contours which correspond to the general spherical shapes of the spheroids utilizing the active contour algorithm. Performance of active contour can be compromised in two situations: poor initialization, or presence of other local edges distracting from the desired contour. Specifically in our tested cases, the second situation sometimes happen when the necrotic core of a large spheroid attracts the active contour resulting in smaller contour being reported. It is worth noting that other automated threshold-based methods also suffer in this situation unless the threshold is specifically set by hand. The software therefore puts forward effort to help users detect and remedy compromised segmentation by providing easy quality-control features. If segmentation error happens from poor initialization, uses can use "Manual Initialize" tool to override the automated initialization. When image quality is too poor for active contour, users can easily "Hand Draw" the contour that feeds into quantification. Existing software such as CellProfiler can be adapted for this application in a semi-automated fashion. The workflow can be cumbersome when large amounts of images with different imaging conditions are presented or when a subset of images needs more human intervention to measure correctly. SpheroidSizer provides an all-in-one suite for computation and quality control to manage high-throughput image analysis workflow.

SpheroidSizer is currently limited to detecting one spheroid per image and only measures the axial length of the spheroid. The program can be extended to support further quantification needed by researchers such as quantification on spheroids with necrotic core, detecting multiple spheroids in one image or monitoring the shape of the spheroids. Furthermore, the program can be modified to detect and measure the size of the excised tumors from animal or human, which will certainly be beneficial for researchers when conducting *in vivo* pre-clinical or clinical research. Post-processing of the detected spheroids can also be investigated aiming at reducing human effort needed for quality control and further improving throughput. SpheroidSizer is a generalized image analysis application for 3D tumor spheroids that are produced from any cell types, and therefore can be used by a wide cancer research community.

Disclosures

No conflicts of interest declared.

References

1. Hamilton, G. Multicellular spheroids as an *in vitro* tumor model. *Cancer Lett.* **131**, 29-34, doi:S0304-3835(98)00198-0 [pii] (1998).
2. Sutherland, R. M., McCredie, J. A., & Inch, W. R. Growth of multicell spheroids in tissue culture as a model of nodular carcinomas. *J Natl Cancer Inst.* **46**, 113-120 (1971).
3. Inch, W. R., McCredie, J. A., & Sutherland, R. M. Growth of nodular carcinomas in rodents compared with multi-cell spheroids in tissue culture. *Growth.* **34**, 271-282 (1970).
4. Hirschhaeuser, F. *et al.* Multicellular tumor spheroids: an underestimated tool is catching up again. *J Biotechnol.* **148**, 3-15, doi:10.1016/j.jbiotec.2010.01.012. S0168-1656(10)00039-8 [pii] (2010).
5. Friedrich, J., Seidel, C., Ebner, R., & Kunz-Schughart, L. A. Spheroid-based drug screen: considerations and practical approach. *Nat Protoc.* **4**, 309-324, doi:10.1038/nprot.2008.226. nprot.2008.226 [pii] (2009).
6. Yuhas, J. M., Li, A. P., Martinez, A. O., & Ladman, A. J. A simplified method for production and growth of multicellular tumor spheroids. *Cancer Res.* **37**, 3639-3643 (1977).
7. Ayers, G. D. *et al.* Volume of preclinical xenograft tumors is more accurately assessed by ultrasound imaging than manual caliper measurements. *J Ultrasound Med.* **29**, 891-901, doi:29/6/891 [pii] (2010).
8. Carpenter, A. E. *et al.* CellProfiler: image analysis software for identifying and quantifying cell phenotypes. *Genome Biol.* **7**, R100, doi:gb-2006-7-10-r100 [pii] (2006).
9. Kamentsky, L. *et al.* Improved structure, function and compatibility for CellProfiler: modular high-throughput image analysis software. *Bioinformatics.* **27**, 1179-1180, doi:10.1093/bioinformatics/btr095 [pii] (2011).
10. Monazzam, A. *et al.* A new, fast and semi-automated size determination method (SASDM) for studying multicellular tumor spheroids. *Cancer Cell Int.* **5**, 32, doi:10.1186/1475-2867-5-32 (2005).
11. Schneider, C. A., Rasband, W. S., & Eliceiri, K. W. NIH Image to ImageJ: 25 years of image analysis. *Nat Methods.* **9**, 671-675 (2012).
12. Chan, V. L. Active contours without edges. *IEEE Trans Image Process.* **10**, 266-277, doi:10.1109/83.902291 (2001).
13. Bernard, O., Friboulet, D., Thevenaz, P., & Unser, M. Variational B-spline level-set: a linear filtering approach for fast deformable model evolution. *IEEE Trans Image Process.* **18**, 1179-1191, doi:10.1109/TIP.2009.2017343 (2009).
14. Kass M., W., A. and Terzopoulos, D. Snakes: Active contour models. *International Journal of Computer Vision.* **1**, 321-331 (1987).
15. Wong, C., Vosburgh, E., Levine, A. J., Cong, L., Xu, E. Y. Human neuroendocrine tumor cell lines as a three-dimensional model for the study of human neuroendocrine tumor therapy. *J Vis Exp.* (66), doi:10.3791/4218 (2012).