

SpineTool 2.0: A Deep Learning Platform with an Interactive User Interface for Dendritic Spine Segmentation and Morphological Analysis

Daria Antonova

Peter the Great St. Petersburg Polytechnic University

Sofia Gurenko

Peter the Great St. Petersburg Polytechnic University

Margarita Pegova

Peter the Great St. Petersburg Polytechnic University

Anita Ustinova

Peter the Great St. Petersburg Polytechnic University

Georgiy Shteynberg

Peter the Great St. Petersburg Polytechnic University

Vyacheslav Chukanov

Peter the Great St. Petersburg Polytechnic University

Ekaterina Pchitskaya

`katrin.pchitskaya@gmail.com`

Peter the Great St. Petersburg Polytechnic University

Article

Keywords:

Posted Date: October 21st, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-10996498/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

SpineTool 2.0: A Deep Learning Platform with an Interactive User Interface for Dendritic Spine Segmentation and Morphological Analysis

Daria Antonova^{1,+}, Sofia Gurenko^{1,+}, Margarita Pegova¹, Anita Ustinova¹, Georgiy Shteynberg², Vyacheslav Chukanov^{1,2}, and Ekaterina Pchitskaya^{1,3,*}

¹Laboratory of Biomedical Imaging and Data Analysis, Institute of Biomedical Systems and Biotechnology, Peter the Great St. Petersburg Polytechnic University, 194021, St. Petersburg, Russia

²Higher School of Applied Mathematics and Computational Physics, Peter the Great St. Petersburg Polytechnic University, 29, Polytechnicheskaya, 195251, St. Petersburg, Russia

³Laboratory of Molecular Neurodegeneration, Institute of Biomedical Systems and Biotechnology, Peter the Great St. Petersburg Polytechnic University, 194021, St. Petersburg, Russia

*katrin.pchitskaya@gmail.com

⁺these authors contributed equally to this work

ABSTRACT

Quantitative analysis of dendritic spine morphology is central to understanding synaptic function and its disruption in neurological disease, yet reliable measurement depends on accurate segmentation of spines from fluorescence microscopy images — a task still limited by faint or discontinuous spine necks and by tools that require extensive manual correction. Here we present SpineTool 2.0, an integrated software platform that combines a cascade of deep learning-based segmentation models four three-dimensional U-Net networks performing dendrite segmentation, neck reconstruction, resolution enhancement, and spine segmentation with an interactive, Napari-based user interface that provides three-dimensional visualization together with a comprehensive set of manual correction, editing, and project-management tools. The platform supports both fully automatic (AI-based) and semi-automatic workflows across heterogeneous in vitro, ex vivo, and in vivo confocal and two-photon datasets without dataset-specific retraining. Benchmarked against existing dendritic segmentation tools, SpineTool 2.0 achieves markedly higher accuracy in both dendritic shaft and individual spine segmentation, while substantially reducing the loss of spine necks and the merging of adjacent spines that limit current approaches. Applied to in vitro primary hippocampal neurons from a low-dose amyloid- β model of Alzheimer’s disease, SpineTool 2.0 detected a larger and more complete population of dendritic spines than the original SpineTool and revealed statistically significant morphological changes under A β exposure that had previously gone undetected. These results show that improvements in automatic segmentation accuracy can directly alter the biological conclusions drawn from morphometric analysis, underscoring the need for accurate, correctable segmentation tools in synaptic biology. SpineTool 2.0, together with its training and test datasets, is released as open-source software.

1 INTRODUCTION

Dendritic spines are small membrane protrusions extending from the surface of neuronal dendrites that form the postsynaptic compartment of most excitatory synapses between the dendrite and axon [1]. The synapse is located on the spine head, which communicates with the dendritic shaft through a narrow neck; consequently, spine geometry directly constrains synaptic function — head volume correlates with synaptic strength and receptor content [2], while the neck acts as a diffusional and electrical barrier between the head and the parent dendrite [3]. It is widely known that changes in spine size, shape, and density are linked to learning and memory consolidation, and disruptions of spine morphology are a hallmark of neurodevelopmental and neurodegenerative disease [4]. Nevertheless, although this general link between spine morphology and function is well established, studying it in detail is impossible to resolve without reliable methods for spine morphology investigation, including both their segmentation from images and the subsequent quantification of morphology descriptors. This makes quantitative analysis of dendritic spine morphology one of the central challenges in the field.

Dendritic spine morphology can be studied across a wide range of experimental preparations, including in vitro neuronal cultures, ex vivo brain slices, and in vivo imaging. Depending on the biological model and imaging requirements, spine images are acquired using different microscopy techniques, including confocal, two-photon, and

electron microscopy. Early studies relied primarily on two-dimensional image analysis; however, because dendritic spines are complex three-dimensional structures, 2D projections provide only limited information about their morphology. The development of volumetric microscopy and z-stack imaging enabled three-dimensional analysis of spine volume, neck geometry, and other structural characteristics with substantially higher accuracy [5]. The first, and one of the most widely used, open-source software packages with a user interface for segmenting and classifying dendritic spines into morphology groups was NeuronStudio [6]; parts of its functionality were later incorporated into Neurolucida, which has since been widely used for manual or semi-automatic neuron tracing and morphological analysis, although such approaches remain labor-intensive and require extensive user interaction. Subsequently, tools enabling direct visual segmentation of individual spines were developed in both two and three dimensions, including SpineJ [7] in 2D and 3dSpAn [8] and SpineTool [9] in 3D. Alongside this shift, the field moved away from classifying spines into discrete morphological groups and toward direct morphometric measurements and clustering of spine shapes [4].

The most common approach remains expert visual analysis of spine images; although this can yield reliable results, it is highly labor-intensive and suffers from low reproducibility due to differences in interpretation between raters [10, 11]. To address these limitations, three-dimensional dendritic spine analysis increasingly relies on semi-automatic and automatic segmentation to separate dendritic spines from the dendrite and background in z-stack images. Semi-automatic methods mentioned above require user input, such as selecting a region of interest, adjusting thresholds, or correcting segmentation results [12]. Recent advances in deep learning have opened new possibilities for automatic image segmentation [13], and these methods have increasingly been applied to neuronal images [14]. Automatic methods perform most of these steps with minimal manual intervention, making them more suitable for large datasets; in recent years, tools such as DeepD3 [15] and RESPAN [11] have been developed for this purpose, using deep learning-based segmentation and image restoration to improve the speed, reproducibility, and accuracy of dendritic spine analysis. However, the main challenges for such algorithms remain their accuracy across heterogeneous imaging conditions and their dependence on large annotated training datasets. Moreover, despite the capacity of neural networks to automate segmentation, the possibility of manual correction and annotation remains necessary to refine their results; we address this by developing an integrated software platform with a comprehensive set of tools and a user interface for interactive correction.

A major limitation of existing dendritic spine segmentation algorithms is their limited ability to detect faint or poorly resolved spine necks. In many current approaches, necks are either ignored or not explicitly reconstructed as connectivity-preserving components linking the spine head to the dendritic shaft, leading to the loss of valuable spines from analysis and forcing the selection of compromise segmentation parameters [16]. Only a few semi-automatic methods have specifically addressed the recovery of weakly visible necks, such as 2dSpAn [17], 3dSpAn [8], and SpineJ [7], typically relying on optimal path search between manually or automatically selected connection points. These methods differ in the criteria used to define the optimal path, the strategy for selecting anchor points, and the procedure for segmenting voxels along the detected path; however, their performance strongly depends on parameter selection, and a single set of parameters is often insufficient to achieve robust results across an entire image. Consequently, despite the availability of these specialized tools, existing approaches still require substantial user intervention, including manual correction of segmentation errors and incomplete neck reconstructions. In the present work, we present an improved algorithm for spine neck reconstruction that combines an intensity- and distance-weighted graph search for the connecting path with two alternative volume-reconstruction methods — threshold flood-fill and Morphological Active Contours — offering greater robustness to low signal-to-noise ratio and flexibility across different data types than existing approaches. We then use this algorithm to generate a training dataset and train a dedicated neural network stage that performs automatic neck reconstruction directly from image intensity.

In this work, we present an integrated framework for dendritic spine segmentation and morphology analysis that combines a cascade of deep learning segmentation models with an interactive annotation platform. The proposed cascade separates dendrite segmentation, reconstruction of weakly fluorescing spine necks, image resolution enhancement, and dendritic spine segmentation into dedicated stages. In parallel, this interactive correction capability serves a dual purpose, allowing either rapid generation of high-quality training data or controlled refinement of segmentation results for quantitative biological analysis. Finally, we assess how improvements in segmentation accuracy affect downstream dendritic spine morphometry by comparing statistical outcomes between control neurons and a model of Alzheimer’s disease, emphasizing the importance of both accurate automatic segmentation and interactive quality control.

2 RESULTS

2.1 An integrated platform with user-interface for dendritic microscopy image segmentation and analysis

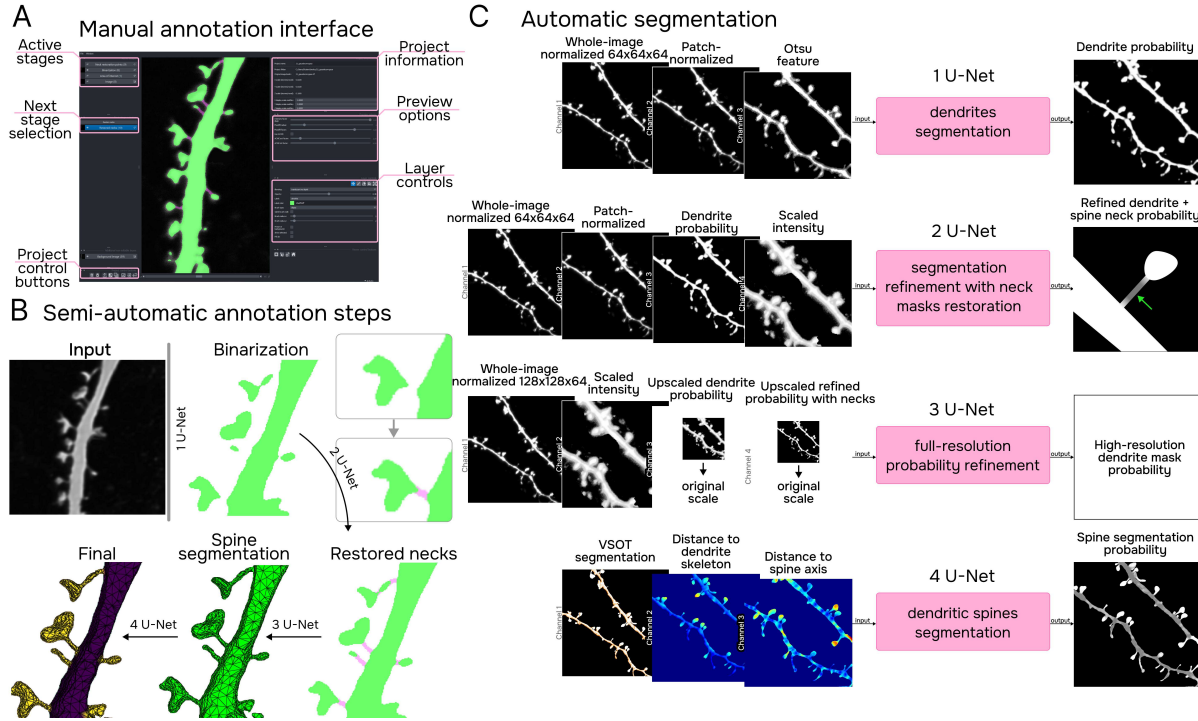


Figure 1. Graphical user interface and neural network pipeline architecture. A - The manual annotation interface features side panels for stage management, preview options, layer controls, and project metadata. B - Stages of semi-automatic segmentation processing: binarization, neck restoration, mesh construction, and the final 3D model with highlighted spines. C - Automatic segmentation consists of four U-Net models: 1 U-Net receives the raw 3D image, normalized frame, and Otsu feature map to calculate dendrite probability; 2 U-Net uses the output of 1 U-Net and an intensity map to refine the dendrite shaft and restore necks; 3 U-Net processes high-resolution data for detailed boundary refinement; 4 U-Net generates the final dendritic spine segmentation probability based on VSOT segmentation and spatial distance maps to the skeleton and spine axis.

An integrated software platform was developed to combine all major stages of dendritic morphology analysis within a unified workflow. Fluorescence microscopy images can be processed entirely within a single application, eliminating the need to transfer data between multiple software packages and resolving compatibility issues between different stages of the image analysis pipeline. Users can choose between AI-based and semi-automatic methods depending on image quality and analysis requirements for a given dataset. Intermediate results can be interactively inspected and corrected at each processing stage, enabling the generation of accurate annotations and training datasets, while the final segmentation results can be exported directly to SpineTool for quantitative morphological analysis according to the methodology described previously [18].

The platform supports heterogeneous fluorescence microscopy datasets acquired under different imaging conditions, spatial resolutions, and experimental settings. A single integrated workflow can be applied to *in vitro*, *ex vivo*, and *in vivo* data without switching between dedicated analysis pipelines or retraining neural network models for individual imaging conditions. These capabilities are achieved, firstly, through a generalized segmentation model trained on heterogeneous datasets, and secondly, by additional semi-automatic processing methods that work across different microscope types, image resolutions, and imaging settings.

Both automatic neural network segmentation and semi-automatic algorithmic segmentation operating modes are controlled through the user interface shown in Figure 1A. Segmentation using semi-automatic methods involves several steps, including the identification of the dendritic shaft, the restoration of disconnected dendritic spine necks, the dendrite mesh building, and the segmentation of dendritic spines. These steps are shown in Figure 1B. Segmentation using an artificial intelligence model consists of four sequential three-dimensional U-Net models. Instead of solving dendritic spine segmentation using a single end-to-end network, the problem was divided into four

consecutive tasks: dendrite segmentation, neck reconstruction, image resolution restoration, and dendritic spine segmentation. The output of each stage is used as an additional input for subsequent models, allowing progressively more accurate reconstruction of dendritic structures. The inputs and outputs of each of the developed models are shown in Figure 1C. The solution architecture is described in detail in the supplementary materials section 3 "Proposed AI segmentation framework".

2.2 Interactive visualization and correction enable efficient human-in-the-loop annotation

The platform provides an integrated environment for interactive visualization of fluorescence microscopy data and segmentation results. Built on the Napari framework, it supports both two-dimensional and three-dimensional visualization of volumetric microscopy images. Image volumes can be inspected from arbitrary viewing directions, navigated through individual image planes, and visualized together with segmentation results. In addition to voxel representations, polygonal meshes can be displayed and edited, allowing the analysis of both fluorescence microscopy data and fully polygonal datasets, such as MICrONS and H01 reconstructions [16]. Different image display settings are available to facilitate visual inspection of fluorescence images.

The annotation workflow is organized as a sequence of processing stages that can be executed automatically or semi-automatically depending on the selected analysis strategy (Fig.2A). The analysis begins by selecting one or more regions of interest, allowing independent processing of individual dendritic segments. Dendrite segmentation is then performed using either the semi-automatic or AI-based approach. The resulting segmentation can be immediately inspected and, if necessary, corrected before proceeding to dendritic spine neck reconstruction. In the semi-automatic workflow, spine attachment points are first identified and the missing necks are reconstructed, whereas the AI-based workflow integrates both operations into a single processing stage. After optional refinement, semi-automatic spine segmentation is performed on reconstructed polygonal meshes and converted back into voxel masks, whereas the AI workflow directly predicts voxel-level spine segmentation. The final segmentation can be exported for quantitative morphological analysis in SpineTool. This program's data processing steps are shown in a step-by-step tutorial included in the supplementary materials, provided both for the convenience of new users and to support the reuse, reproducibility, and further development of the software as an open-source resource for the community. (указать где именно).

Interactive correction is available at all stages of the annotation workflow. The platform provides “brush” and “eraser” tools that operate either within a single image plane or simultaneously across multiple planes using ellipsoidal and elliptical-cylindrical volumes, illustrated in Fig. 2B. A three-dimensional flood-fill tool allows for quick correction of connected regions by filling in the surrounding boundaries. These editing tools enable efficient refinement of segmentation results for both dendrites and spines, minimizing manual effort.

To facilitate efficient manual annotation, all editing operations support iterative refinement using standard undo and redo features. In addition, the processing workflow is non-linear: users can return to any previous segmentation stage, modify the algorithm parameters, and continue the analysis from that point without restarting the entire pipeline. This branching workflow allows users to compare different parameter settings, segmentation algorithms, or manual corrections while preserving all intermediate results, which simplifies collaborative annotation and the systematic evaluation of alternative processing strategies. The complete state of the project, including all intermediate processing results, can be saved at any stage, enabling annotation to be resumed after interruption without repeating previous computations.

2.3 A generalized deep learning model SpineTool 2.0. achieves accurate dendritic spine segmentation across heterogeneous microscopy data

The training set combined sources that differed in sample preparation, microscopy method and resolution: confocal images of neuronal cultures and of hippocampal slices, together with two-photon images from public datasets. We trained a single model in one pass, rescaling every image to a common $0.1 \mu\text{m}$ per pixel in XY , with no adjustment made to any individual dataset.

The task itself is split into four sequential stages. Each stage addresses a narrower problem, which is what makes training on the pooled data of both microscopy modalities possible without the per-dataset tuning such data usually demand. The first three stages operate on image intensity; the fourth does not, and applies to voxel models of dendrites regardless of how they were obtained.

On dendritic shaft segmentation stage, test data set consist of 15 image samples, with 3 per cross-validation experiment. The assignment of images to cross-validation subsets is provided in the supplementary materials. SpineTool 2.0 achieved an F1-score of 0.782 ± 0.061 , compared with 0.698 ± 0.083 for RESPAN and 0.420 ± 0.198 for DeepD3. The corresponding IoU values were 0.646, 0.542 and 0.284 (Table 1). RESPAN achieved the highest recall (0.900 ± 0.072) but substantially lower precision (0.582 ± 0.115), indicating systematic over-segmentation of the

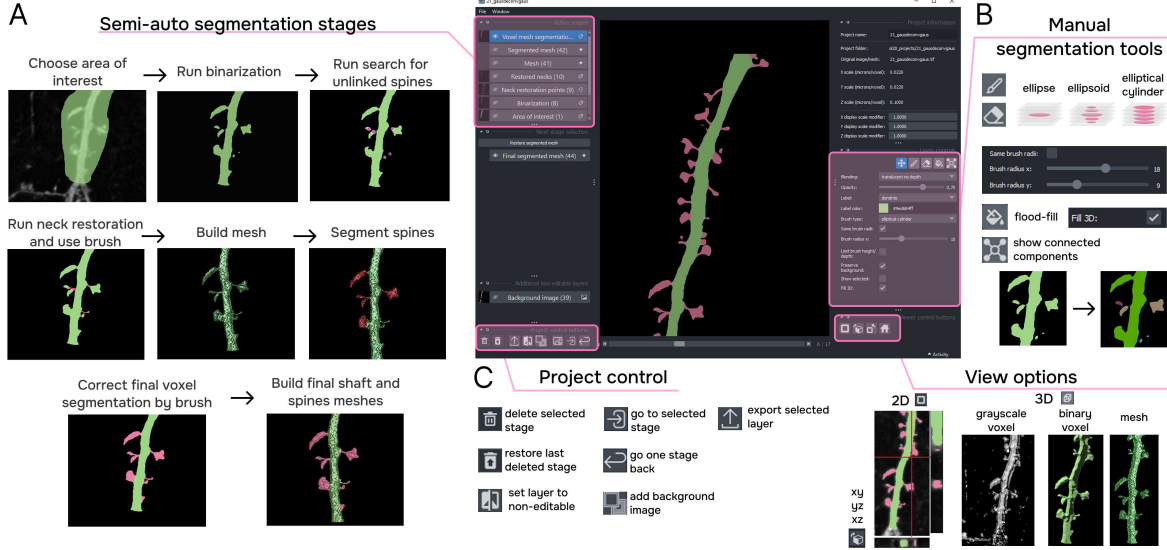


Figure 2. Interactive annotation tool interface and processing pipeline overview. The main user screen apart from the sample preview window has three distinct functional blocks: A - The stages of image processing with a semi-automatic annotation tool, showing the name of each stage along with a preview of the result. The stages the user has completed are displayed in the upper-left pane and support customization of visibility, as well as the ability to switch back to the desired stage; B - Main annotation and visualization tools, including three configurable brush types, a flood-fill tool, contrast adjustment, connected component detection, and 2D/3D visualization with orthogonal (XY, XZ, and YZ) views; C - Project management tools, including workflow navigation, undo/redo functionality, export of segmentation masks for neural network training or external software, and loading of additional image channels (e.g., fluorescence of other cellular structures).

dendritic shaft. DeepD3 showed both lower accuracy (recall 0.433 ± 0.223 , precision 0.452 ± 0.230) and markedly higher variability across image stacks. Figure 3A shows that DeepD3 fails outright on some stacks rather than performing moderately on all of them, reflecting inconsistent performance on heterogeneous datasets.

Spine segmentation stage accuracy was tested on 15 images acquired in our laboratory, and on 6 dendrite segments from VSOT algorithm [19] research, with 4 to 5 images per cross-validation experiment. The assignment of data to cross-validation subsets is provided in the supplementary materials. The fourth stage refines the initial segmentation produced by VSOT [19] using a convolutional neural network, with particular attention to spine attachment regions where shaft and spine boundaries are frequently misclassified. Since both methods operate on binary voxel masks without image intensity information, this comparison isolates the contribution of the proposed refinement network. RESPAN and DeepD3 instead perform end-to-end segmentation directly from the raw stack, Table 2.0 states this difference explicitly. At a matching threshold of $IoU \geq 0.5$, the cascade reached an F1-score of 0.561 ± 0.182 , against 0.031 ± 0.020 for VSOT, 0.066 ± 0.062 for RESPAN and 0.018 ± 0.037 for DeepD3. This accuracy ranking remains the same at the voxel level, but the gap narrows: the Dice coefficient is 0.799 ± 0.058 , compared to 0.564, 0.415, and 0.317. VSOT correctly localizes many spine voxels but frequently fails to produce complete spine objects. RESPAN and DeepD3 fail earlier in the chain, missing almost all of annotated spines outright. SpineTool 2 substantially improves object-level detection, although approximately 30% of spines remain undetected, and precision (0.482 ± 0.200) indicates that false-positive objects are still common.

Table 1. Dendrite/background segmentation performance. Values are mean $\pm$ SD over the 15 images (Eq.); bold marks the best score per column. Our results are cross-validated.

Method	F1-score	Precision	Recall	IoU
SpineTool 2.0	0.782 ± 0.061	0.781 ± 0.132	0.829 ± 0.155	0.646 ± 0.083
RESPAN	0.698 ± 0.083	0.582 ± 0.115	0.900 ± 0.072	0.542 ± 0.097
DeepD3	0.420 ± 0.198	0.452 ± 0.230	0.433 ± 0.223	0.284 ± 0.154

Table 2. Spine segmentation. Instance-level F1/Precision/Recall (a spine is counted as detected at $\text{IoU} \geq 0.5$) and voxel-level Dice/IoU. Values are mean $\pm$ SD over the 15 images; bold marks the best score among the 4 .

Method	F1-score	Precision	Recall	Dice _{vox}	IoU _{vox}
SpineTool 2.0*	0.561 $\pm$ 0.182	0.482 $\pm$ 0.200	0.700 $\pm$ 0.133	0.799 $\pm$ 0.058	0.669 $\pm$ 0.080
VSOT*	0.031 $\pm$ 0.020	0.016 $\pm$ 0.011	0.409 $\pm$ 0.171	0.564 $\pm$ 0.088	0.397 $\pm$ 0.088
RESPAN	0.066 $\pm$ 0.062	0.060 $\pm$ 0.052	0.076 $\pm$ 0.080	0.415 $\pm$ 0.078	0.265 $\pm$ 0.065
DeepD3	0.018 $\pm$ 0.037	0.011 $\pm$ 0.025	0.052 $\pm$ 0.077	0.317 $\pm$ 0.180	0.201 $\pm$ 0.126

* The spine segmentation stage of SpineTool 2.0 and VSOT was evaluated on the ground-truth binary foreground (as in the cascade design), whereas RESPAN and DeepD3 segmented spines on their own binarization output. The two starred methods received an equivalent input: , so they are directly comparable with each other. The spine comparison is therefore not fully symmetric across all four methods and should be read together with Table 1.

Three failure modes recur across the comparison maps (Figure 3A) and shaft-spine separations (Figure 3B). The first is loss of dendritic spine necks, which disconnects spine heads from the dendritic shaft and prevents subsequent morphological analysis. The second is inaccurate placement of the shaft-spine boundary. This affects VSOT and the cascade, both of which receive a binary foreground in which necks are already present and must decide only where a spine ends and the shaft begins. Arrows in Figure 3B mark representative cases: spine bases extended into the shaft, or attachment regions truncated. The third is merging of adjacent spines into a single connected component, producing one false positive together with multiple false negatives during object-level evaluation. Underlying these is a difference in how the two end-to-end methods err at the voxel level. RESPAN primarily over-segmented dendritic shafts, whereas DeepD3 more frequently misplaced shaft-spine boundaries

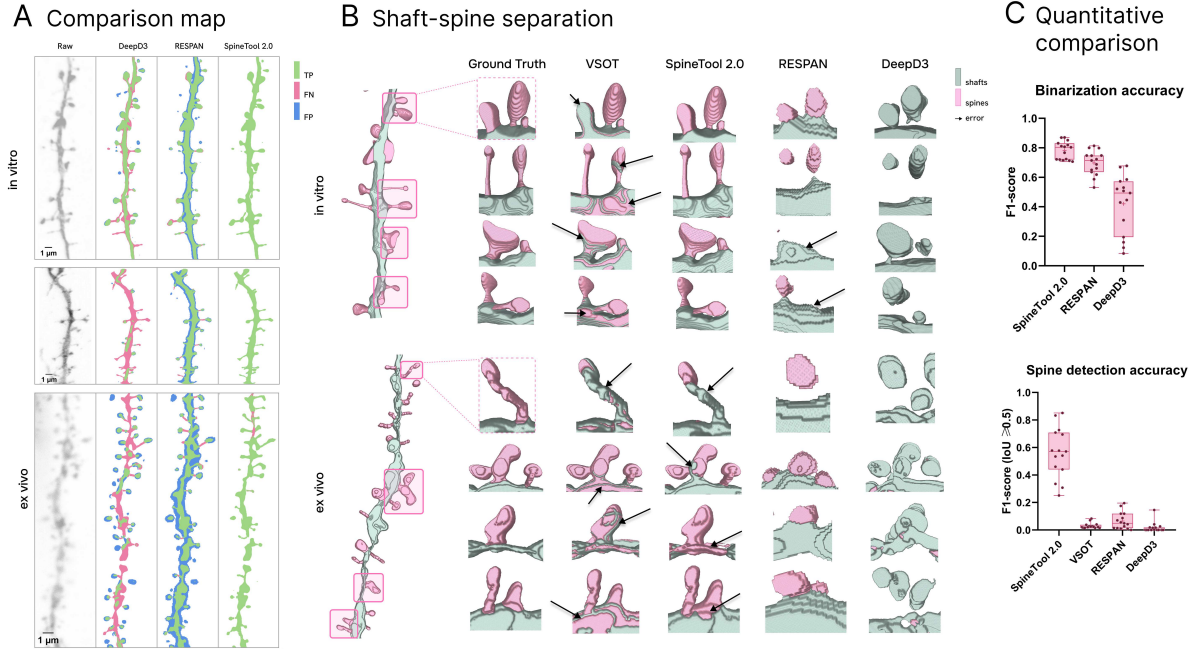


Figure 3. Spine Tool 2.0 achieves high accuracy dendritic spines segmentation results among other AI-based models and algorithms. A - The accuracy maps of dendrite surface segmentation on in vitro and ex vivo test samples. Green indicates correctly segmented dendritic regions, blue and red indicate false-positive and false-negative dendritic regions, respectively. B - A visual representation of the segmentation of a dendrite on a stem and spines on the mentioned in vitro and ex vivo samples, with a closer look at representative sections of the dendrite and segmentation errors indicated by arrows. C - Boxplots showing the dendrite mask segmentation accuracy and the per-object ($\text{IoU} > 0.5$) accuracy of dendrite spine segmentation.

2.4 Accurate segmentation directly influences quantitative analysis of dendritic spine morphology and its interpretation in Alzheimer’s disease modeling hippocampal neurons

The primary hippocampal neuron cultures morphology dataset [20] was processed using SpineTool and SpineTool 2.0. For SpineTool, specially cropped image fragments were used, whereas full-frame images were fed into SpineTool 2.0, as the new software can successfully handle branching dendritic regions. The comparison was carried out on identical sets of 3D confocal images of hippocampal neurons under control conditions and following A β exposure. A total of 66 dendritic segments were obtained (9 more than with the first version) along with 1,045 dendritic spines (520 more than with the original SpineTool).

When calculating morphometric metrics, SpineTool detected only minimal or non-significant changes in spine length (Length, $p = 0.5129$), whereas SpineTool 2.0 revealed a highly significant increase in key volumetric and linear parameters under A β exposure (Volume, $p < 0.0001$; Length, $p < 0.0001$). Spinetool 2 is able to detect long dendritic spines in the AB group, and corrects the spine length in the control group, reducing both the maximum and average spine lengths. Additionally, SpineTool2 effectively eliminated spurious fluctuations in the convex hull ratio ($p = 0.3879$), providing smoother 3D meshes and clearer boundary delineation. In terms of spine morphology, this means that the spines detected by the SpineTool 2.0 are close to a rounded shape. And based on the change in volume statistics, we may conclude the spines are large and probably belong to the mushroom-shaped spine group. Under A β treatment, an increase in the open angle ($p = 0.0091$) was noted, reflecting greater spine morphological diversity as well as an overall shift in the average value. While the previous analysis showed an artificial increase in the length-to-volume ratio (LVR), SpineTool 2.0 correctly captured a significant decrease ($p < 0.0001$), reflecting a proportional enlargement of spine heads relative to neck elongation. Together, these results confirm that SpineTool 2.0 offers substantially higher sensitivity, neck reconstruction accuracy, and overall reliability when analyzing synaptic architecture.

3 DISCUSSION

Most existing studies in the dendritic spines analysis field focus on either segmentation algorithms or morphometric analysis tools. In practice, however, dendritic spine analysis requires a considerably longer workflow, including segmentation, manual correction, spine extraction, morphometric analysis, and statistical evaluation. The proposed platform integrates these stages into a single software environment while maintaining seamless compatibility with SpineTool, eliminating the need to transfer data between independent software packages.

One of the main objectives of this work was to develop a segmentation framework that generalizes across heterogeneous fluorescence microscopy datasets rather than optimizing performance for a single acquisition protocol. Unlike many existing approaches, the proposed model was trained using images acquired under different microscopy modalities, resolutions, and experimental conditions, allowing the same network to be applied without retraining for individual datasets. Existing segmentation models achieved competitive performance on some image stacks but failed completely on others, leading to substantially larger variability across datasets. For practical high-throughput analysis, consistent performance across different imaging conditions is often more important than mean accuracy on a subset of images, since unstable behaviour requires manual inspection of every processed dataset, which defeats the purpose of automation.

The four-stage architecture proved particularly important for robust segmentation. By decomposing the overall task into simpler subtasks, each model could be trained using relatively small annotated datasets, reducing the amount of manual annotation required for network development. Dedicated reconstruction of dendritic spine necks addresses one of the major limitations of existing methods, where detached spine heads are frequently lost from subsequent analysis [16]. To our knowledge, machine learning has not previously been applied to automatic reconstruction of neck masks. Necks are the first structure to disappear as signal-to-noise falls, and the consequences of this were demonstrated in the accuracy analysis results of existing methods: spine heads survive as isolated bodies with no connection to the shaft. Once the neck is lost, the spine head becomes disconnected from the dendritic shaft and is frequently removed during subsequent filtering and therefore lost for subsequent analysis. In the proposed framework, neck reconstruction is addressed explicitly using a dedicated neural network stage, allowing disconnected spine heads to be recovered before spine segmentation. Refinement of the initial VSOT segmentation further demonstrates that learning-based post-processing can substantially improve object-level segmentation even when an initial voxel representation is already available.

Although automatic segmentation substantially reduces manual effort, complete automation remains difficult because fluorescence microscopy datasets differ considerably in image quality, staining protocols, and acquisition parameters. Instead of replacing the expert, the proposed platform follows a human-in-the-loop paradigm in which automated algorithms generate an initial segmentation that can be reviewed and corrected before subsequent

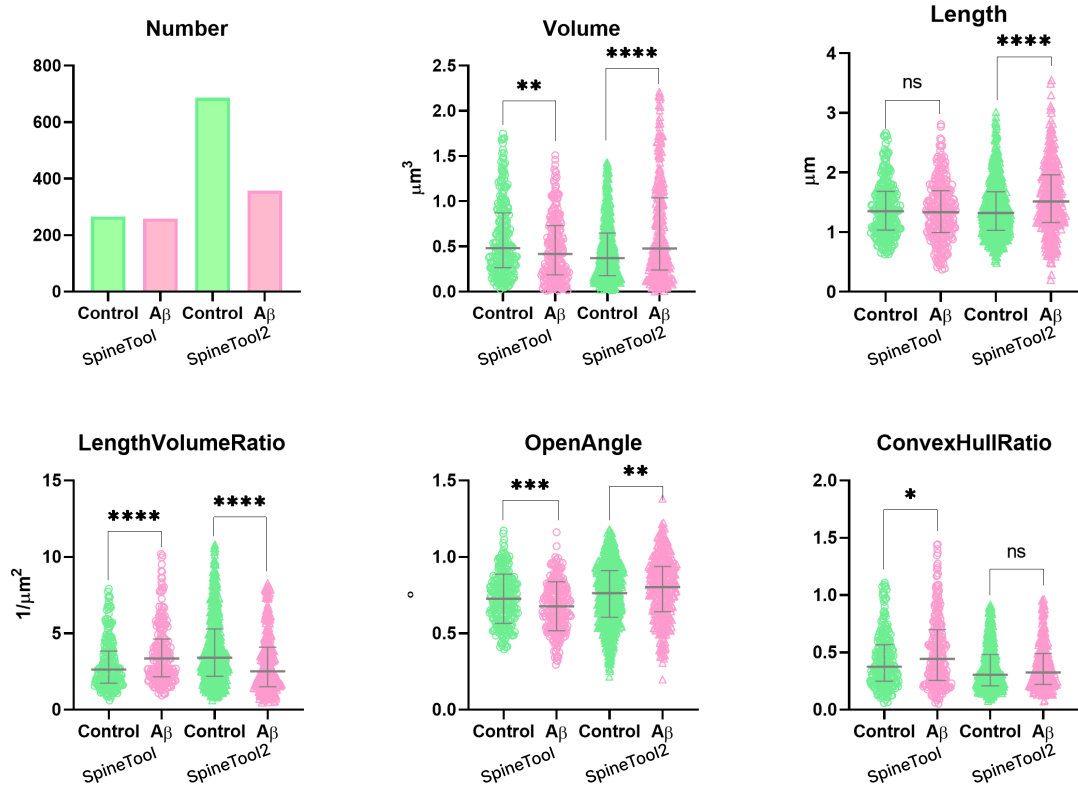


Figure 4. Comparative morphometric analysis of the primary hippocampal neuron cultures morphology dataset [20] processing in SpineTool and SpineTool2. Quantitative comparison between Control and pathological $A\beta$ groups, including the total number of spines detected by SpineTool and SpineTool2. For Volume (SpineTool) and OpenAngle (SpineTool) metrics, data are presented as mean $\pm$ SEM (Welch’s t-test). For Length, LengthVolumeRatio, ConvexHullRatio, Volume (SpineTool2), and OpenAngle (SpineTool2) metrics, data are presented as median with interquartile range [Q1; Q3] (Mann–Whitney test). * $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$, **** $p < 0.0001$, ns = not significant.

processing stages. Intermediate corrections prevent segmentation errors from propagating through the pipeline, improve the quality of the final reconstruction, and simultaneously provide high-quality annotations that can be incorporated into future training datasets. The possibility of combining AI-based and semi-automatic methods within the same workflow further increases flexibility, allowing users to select the most appropriate strategy for a particular dataset.

The neural network segmentation pipeline allowed us to segment a higher number of dendritic spines in the experiment comparing dendrite morphology in normal conditions and in a model of Alzheimer’s disease. Analysis of morphological changes between normal and pathological conditions revealed that the new segmentation method detected more large and long spines in the $A\beta$ exposure group. The change in the segmentation algorithm affected the p-values for differences between the normal and pathological groups. This highlights the need of developing, evaluating, and applying the most accurate algorithms for segmenting dendritic spines to provide a more reliable basis for drawing conclusions about the morphology of dendrites.

The complete platform is distributed as open-source software, providing a reproducible environment for segmentation, annotation, and quantitative analysis. The training and testing datasets used in the study are also publicly available, providing a foundation for further development of deep learning approaches for dendritic spine analysis. The assembly of well-annotated datasets, curated and expanded by the community, is essential for further progress in this direction: the availability of such data is a substantially greater bottleneck for the application of AI-based methods than the adaptation of new network architectures or training strategies. To further adapt the tool to diverse data types, we plan to incorporate a fine-tuning workflow based on the manual corrections saved during

annotation, allowing the models to be progressively adapted and their performance improved over time.

Although the proposed framework substantially improves object-level segmentation, approximately 30% of dendritic spines remain undetected and false-positive detections are still common. Most remaining errors arise from weak fluorescence, incomplete neck reconstruction, and merging of neighboring spines. Additional validation on larger and more diverse microscopy datasets will be required to further assess model generalization.

Future work will focus on expanding annotated training datasets, improving adaptation to new imaging conditions, and investigating more advanced neural network architectures. The platform will also be extended with additional post-processing and morphometric analysis methods, providing a complete end-to-end environment for dendritic spine analysis.

4 MATERIALS AND METHODS

4.1 Animals

Albino inbred mice (FVB/NJ) were obtained from the Jackson Laboratory (Jackson Laboratory, Bar Harbor, ME, USA, strain #001800) and used as a source of brain tissue for experiments with primary hippocampal cultures. Transgenic Tg(Thy1-EGFP)MJrs/J mice were used for experiments with fixed brain slices for morphological studies of dendrites and dendritic spines. These mice were established and maintained in a vivarium with 4 to 5 mice per cage and a 12-hour light/dark cycle in the animal facility. Food and water were available ad libitum.

FVB/NJ mouse pups at postnatal days 0–1 (P0–P1) used for primary hippocampal culture preparation were anesthetized by hypothermia on a bed of crushed ice for approximately 10 min. Following anesthesia, the pups were euthanized by rapid decapitation using sharp surgical scissors, and the brains were immediately removed for hippocampal dissection. For fixed brain slice preparation, adult Tg(Thy1-EGFP)MJrs/J mice were deeply anesthetized by intraperitoneal administration of urethane (250 mg/mL in 0.9% NaCl; Sigma, Cat. No. U2500-250G) at a dose of 2 g/kg body weight, as described previously [18]. The depth of anesthesia was confirmed by the absence of responses to external stimuli before initiation of the terminal procedure. Under deep anesthesia, mice underwent transcardial perfusion with PBS followed by 1.5% paraformaldehyde (PFA) in PBS. Following perfusion, the animals were euthanized by decapitation using surgical scissors. The brains were removed for subsequent fixation and preparation of hippocampal sections. All methods involving animals were performed in accordance with the relevant institutional guidelines and regulations.

4.2 Datasets

The proposed segmentation framework was trained and evaluated using both publicly available datasets and manually annotated confocal microscopy images. For training binarization model of AI segmentation pipeline, an open-source dataset of 30 two-photon images of organotypic hippocampal culture slices, collected in DeepD3 research [15], were used. The dataset contains manual annotations of 1,872 dendritic spines and is one of the largest publicly available datasets for dendritic spine segmentation in fluorescence microscopy. Six voxelized dendrites (one apical and five basal) reconstructed from serial electron microscopy and introduced with the VSOT segmentation algorithm [19] were also used for spines segmentation model of proposed AI segmentation tool. The dataset provides 344 dendritic spine meshes.

To develop a segmentation model capable of generalizing across different imaging scales and protocols, an additional confocal microscopy dataset was collected. The dataset includes both ex vivo and in vitro fluorescence microscopy images. The ex vivo subset includes six confocal image stacks of mouse hippocampal dendrites with 12 manually annotated dendritic segments and 506 dendritic spines. The in vitro subset includes six confocal image stacks of primary hippocampal neuron cultures, including image stacks of neurons exposed to low amyloid toxicity. In total, this subset contains 8 manually annotated dendritic segments and 199 dendritic spines. Additional four in vitro image stacks were taken from the dataset used in our previous work [20] and segmented manually. A detailed description of the manual annotation procedure is provided in the supplementary methods.

To evaluate the proposed framework in a downstream biological analysis, we used an in vitro confocal microscopy dataset of primary hippocampal cultures under control conditions and low amyloid toxicity, previously described in [20]. The four image stacks used for model development were excluded from this analysis. The dataset was processed using both the semi-automatic SpineTool software [9] by protocol [18] and the proposed AI-based automatic segmentation tool. As a result, the control group contained 53 dendritic segments with 550 dendritic spines. The amyloid-treated group, representing an in vitro model of Alzheimer’s disease, contained 23 dendritic segments with 360 dendritic spines.

The collected manually annotated ex vivo and in vitro fluorescence microscopy datasets were also used for quantitative evaluation of segmentation accuracy. Segmentation results produced by DeepD3 [15], RESPAN [11],

and VSOT [19] were compared with those obtained using the proposed AI method.

A detailed description of the segmentation procedures is provided in the Supplementary Methods, including the automatic, semi-automatic, and manual annotation workflows. In total, four independent datasets were used in the study, covering two-photon, confocal, and serial electron microscopy, including both ex vivo and in vitro. A summary of all datasets is provided in Table 3.

Table 3. Summary of datasets used in study.

Dataset	Microscopy	Sample	Images	Dendrites	Spines	Purpose
DeepD3 [15]	Two-photon fluorescence	Organotypic hippocampal slice cultures	30	–	1,872	Binarization model training
VSOT [19]	Serial electron microscopy	Voxelized dendrites (1 apical, 5 basal)	6	6	344	Spines segmentation model training
Confocal ex vivo (this study)	Confocal fluorescence	Mouse hippocampus	6	12	506	Four models of integrated solution trainings
Confocal in vitro [20] (new annotations)	Confocal fluorescence	Primary hippocampal neuron cultures	9	18	435	Four models of integrated solution trainings
Morphology dataset [20]	Confocal fluorescence	Primary hippocampal neuron cultures	–	53 / 23 *	550 / 360 *	Biological validation (control vs. amyloid toxicity)

* Two sets of data quantities are listed, separated by a slash: the amount of data previously extracted by spinetool and the amount of data obtained by the new spinetool2.0, respectively.

4.3 Neck restoration

The semi-automatic neck reconstruction algorithm takes as input a binary segmentation mask of the dendrite with segmented dendritic spine heads and the corresponding fluorescence image.

Firstly, connected components are identified on the binary mask. The largest component is then considered the dendrite shaft, while the remaining components are filtered out based on a volume threshold set by the user. The components that pass the filtering are then identified as spine heads, which need to be attached to the trunk. To determine the attachment points to the dendrite trunk, 3D thinning is performed [21]. Next, a centerline is extracted from the skeleton by iteratively selecting, among the neighboring skeleton vertices, the vertex whose direction has the smallest cosine distance to the direction toward the distal end of the skeleton. The geometric center of the head becomes the reference point on the spine, and the nearest point on the centerline becomes the reference point on the dendritic shaft. Spine heads located farther than a predefined distance from the centerline are discarded.

The reconstruction of the neck is divided into two stages: finding a path between points and reconstructing the volume of the neck. The search for the best route between attachment points of the spine to the shaft is performed by solving a shortest-path problem on a graph. An undirected weighted graph is constructed from the image, in which each voxel corresponds to a graph vertex connected to its 26 neighboring voxels. The weight $w_{i,j}$ of each edge is computed based on both the Euclidean distance between adjacent voxels and their fluorescence intensities.

$$w_{i,j} = \left(1 - \frac{br_i + br_j}{2}\right) * f_s + \text{dist}(p_i, p_j), \quad (1)$$

where br_i and br_j denote the normalized fluorescence intensities of voxels v_i and v_j , respectively ($0 \leq br \leq 1$); p_i and p_j are the spatial coordinates of the corresponding voxels; $\text{dist}(p_i, p_j)$ is the Euclidean distance between them; and f_s is an intensity weighting factor that controls the contribution of voxel intensity to the edge weight.

The pathway obtained in the previous step represents a one-voxel-wide skeleton of the dendritic spine neck. Its thickness is then reconstructed using one of two implemented methods: Threshold Floodfill or Morphological ACWE. For the Threshold Floodfill method, each voxel neighboring the neck skeleton is included in the reconstructed neck

mask if it satisfies the intensity criterion

$$br_i \geq f_f * br_0 \quad (2)$$

where br_i and br_0 denote the normalized fluorescence intensities of voxels v_i and v_0 , respectively, and f_s is the flood-fill threshold factor.

As an alternative neck reconstruction method, we implemented a modified Morphological Active Contours Without Edges (Morphological ACWE) algorithm [22]. The method evolves the neck boundary by minimizing the Chan Vese energy functional

$$\begin{aligned} F(c_1, c_2, S) = & \lambda_1 \int_{\text{inside}(S)} |I(\mathbf{x}) - c_1|^2 d\mathbf{x} \\ & + \lambda_2 \int_{\text{outside}(S)} |I(\mathbf{x}) - c_2|^2 d\mathbf{x} \\ & + \mu \text{Area}(S) + \nu \text{Volume}(\text{inside}(S)), \end{aligned} \quad (3)$$

where $I(\mathbf{x})$ denotes the image intensity, S is the evolving neck surface, c_1 and c_2 are the mean intensities inside and outside the surface, respectively, and λ_1 , λ_2 , μ , and ν are weighting parameters.

The surface evolution is implemented using the morphological approximation of the ACWE level-set formulation proposed by Márquez-Neila et al. [22]. Starting from the one-voxel-wide neck skeleton obtained in the previous step, the contour is iteratively evolved until convergence, yielding the reconstructed neck volume.

4.4 Neural network models of integrated segmentation tool

The proposed AI segmentation framework is based on three-dimensional U-Net model [13]. Unlike conventional end-to-end approaches, the framework consists of multiple specialized networks, each designed to solve a specific task: dendrite segmentation, neck reconstruction, image resolution restoration, and dendritic spine segmentation. The models are applied sequentially, with the output of earlier stages serving as additional input channels for subsequent networks. This modular design allows each model to be optimized for its specific task while preserving information extracted in previous stages. A detailed description of the data preprocessing pipeline and tensor generation procedure is provided in the Supplementary Methods.

All models share the same five-level encoder-decoder 3D U-Net architecture [23]. Depending on the stage, the networks receive three or four input channels and predict one or two output channels. The models were trained using task-specific loss functions, including mean squared error, Dice-based losses, topology-aware skeleton loss, and masked binary cross-entropy. The Adam optimizer [24] with $lr = 0.005$ was used for training all models.

Training and inference were performed on overlapping three-dimensional image patches. The complete network configurations, training hyperparameters, input and output representations, and loss function definitions for each stage are summarized in Supplementary Table S1. A detailed description of the data preprocessing, tensor generation, model training, and inference procedures is provided in the Supplementary Methods.

Since the predictions are performed within a sliding window, model 4 — which segments dendritic spines on the dendritic shaft — can receive tiles containing dendritic spines only as input. For tiles where preliminary rough segmentation shows that only a portion of a spine — not attached to a dendrite — is present, a special post-processing procedure has been developed; its algorithm is also described in the supplementary materials.

4.5 Morphology statistics

The extracted dendritic spines were filtered in a manner similar to how the dataset was processed during the analysis by the SpineTool [9]. SpineTool was used to obtain spine metrics. In total, 11 classical scalar metrics were obtained — length, volume, surface area, convex hull volume, convex hull ratio, average distance, coefficient of variance in distance, open angle, length to volume ratio, length to surface area ratio.

Statistical analysis of metric distributions was performed using GraphPad Prism software. Outliers were identified and removed using the ROUT method [25] implemented in GraphPad Prism software, with the aggressiveness parameter set to $Q = 1\%$. The Shapiro-Wilk test was applied to evaluate the normality of the data. For normally distributed datasets, continuous variables were compared using a Student's t-test in the case of homogeneous variances, or Welch's t-test if variances were unequal. For non-normally distributed data, statistical significance was assessed using the Mann-Whitney U test. In the plots shown in Figure 4, normally distributed data are presented as mean $\pm$ SD, whereas non-normally distributed data are expressed as median with interquartile range.

402 Acknowledgements

403 We thank Asya Bolshakova for lab management and reagents supply, Olga Vlasova for organizational support, Ilya
404 Bezprozvanny for scientific advising and access to the experimental equipment, Grigoriy Chevykalov for developing
405 software prototypes.

406 Author contributions statement

407 V.C. and E.P. contributed to conceptualization; D.A., S.G., and U.A. contributed to methodology; D.A., M.P., S.G.
408 and G.S. contributed to software development; M.P. S.G. and G.S. performed validation and investigation; S.G.
409 performed formal analysis; E.P. provided resources; D.A., M.P., S.G., and U.A. performed data curation, contributed
410 to writing the original draft and visualization; D.A., M.P., S.G., and E.P. contributed to review and editing; V.C.,
411 D.A. and E.P. provided supervision; A.U., D.A., V.C., and E.P. contributed to project administration. All authors
412 reviewed and approved the manuscript.

413 Conflict of interest

414 None declared.

415 Data and code availability

416 The source code for the neural network models, training procedures, and the SpineTool2 segmentation platform is
417 publicly available on GitHub at <https://github.com/Biomed-imaging-lab/SpineTool2>.

418 Funding

419 The research was funded by the Ministry of Science and Higher Education of the Russian Federation (FSEG-2024-
420 0025). D.A. is a recipient of a fellowship from the Brain Program of the IDEAS Research Centre.

421 Ethics approval

422 All animal procedures were approved by the Bioethics Committee of the Peter the Great St Petersburg Polytechnic
423 University at St Petersburg, Russia and followed the principles of European convention (Strasbourg, 1986) and the
424 Declaration of International medical association about humane treatment of animals (Helsinki, 1996). All methods
425 were carried out in accordance with relevant guidelines and regulations. The study was carried out in compliance
426 with the ARRIVE guidelines.

427 References

- 428 1. Nimchinsky, E. A., Sabatini, B. L. & Svoboda, K. Structure and function of dendritic spines, DOI: [10.1146/](https://doi.org/10.1146/annurev.physiol.64.081501.160008)
429 [annurev.physiol.64.081501.160008](https://doi.org/10.1146/annurev.physiol.64.081501.160008) (2002).
- 430 2. Matsuzaki, M. et al. Dendritic spine geometry is critical for ampa receptor expression in hippocampal cal
431 pyramidal neurons. Nat. Neurosci. 4, 1086–1092, DOI: [10.1038/nm736](https://doi.org/10.1038/nm736) (2001).
- 432 3. Bourne, J. N. & Harris, K. M. Balancing structure and function at hippocampal dendritic spines, DOI:
433 [10.1146/annurev.neuro.31.060407.125646](https://doi.org/10.1146/annurev.neuro.31.060407.125646) (2008).
- 434 4. Pchitskaya, E. & Bezprozvanny, I. Dendritic spines shape analysis—classification or clusterization? perspective,
435 DOI: [10.3389/fnsyn.2020.00031](https://doi.org/10.3389/fnsyn.2020.00031) (2020).
- 436 5. Kashiwagi, Y. et al. Computational geometry analysis of dendritic spines by structured illumination microscopy.
437 Nat. Commun. 10, DOI: [10.1038/s41467-019-09337-0](https://doi.org/10.1038/s41467-019-09337-0) (2019).
- 438 6. Rodriguez, A., Ehlenberger, D. B., Dickstein, D. L., Hof, P. R. & Wearne, S. L. Automated three-dimensional
439 detection and shape classification of dendritic spines from fluorescence microscopy images. PLoS ONE 3, DOI:
440 [10.1371/journal.pone.0001997](https://doi.org/10.1371/journal.pone.0001997) (2008).
- 441 7. Levet, F., Tønnesen, J., Nägerl, U. V. & Sibarita, J.-B. Spinej: A software tool for quantitative analysis of
442 nanoscale spine morphology. Methods 174, 49–55, DOI: <https://doi.org/10.1016/j.ymeth.2020.01.020> (2020).
- 443 8. Das, N. et al. 3dspan: An interactive software for 3d segmentation and analysis of dendritic spines. Neuroinfor-
444 matics 20, 679–698, DOI: [10.1007/s12021-021-09549-0](https://doi.org/10.1007/s12021-021-09549-0) (2022).

9. Pchitskaya, E., Vasiliev, P., Smirnova, D., Chukanov, V. & Bezprozvanny, I. Spinetool is an open-source software for analysis of morphology of dendritic spines. *Sci. Reports* 13, 10561, DOI: [10.1038/s41598-023-37406-4](https://doi.org/10.1038/s41598-023-37406-4) (2023).
10. Bello-Medina, P. C., Flores, G., Quirarte, G. L., McGaugh, J. L. & Alcalá, R. A. P. Mushroom spine dynamics in medium spiny neurons of dorsal striatum associated with memory of moderate and intense training. *Proc. Natl. Acad. Sci.* 113, E6516–E6525, DOI: [10.1073/pnas.1613680113](https://doi.org/10.1073/pnas.1613680113) (2016). <https://www.pnas.org/doi/pdf/10.1073/pnas.1613680113>.
11. Garcia, S. B. et al. Respan: A deep learning pipeline for accurate and automated restoration, segmentation, and quantification of dendritic spines. *bioRxiv* 2024.06.06.597812, DOI: [10.1101/2024.06.06.597812](https://doi.org/10.1101/2024.06.06.597812) (2025).
12. Smirnov, M. S., Garrett, T. R. & Yasuda, R. An open-source tool for analysis and automatic identification of dendritic spines using machine learning. *PLOS ONE* 13, e0199589– (2018).
13. Ronneberger, O., Fischer, P. & Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Navab, N., Hornegger, J., Wells, W. M. & Frangi, A. F. (eds.) *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, 234–241 (Springer International Publishing, 2015).
14. Moen, E. et al. Deep learning for cellular image analysis, DOI: [10.1038/s41592-019-0403-1](https://doi.org/10.1038/s41592-019-0403-1) (2019).
15. Fernholz, M. H. P., Nilo, D. A. G., Bonhoeffer, T. & Kist, A. M. Deepd3, an open framework for automated quantification of dendritic spines. *PLOS Comput. Biol.* 20, e1011774– (2024).
16. Ofer, N., Berger, D. R., Kasthuri, N., Lichtman, J. W. & Yuste, R. Ultrastructural analysis of dendritic spine necks reveals a continuum of spine morphologies. *Dev. Neurobiol.* 81, 746–757, DOI: [10.1002/dneu.22829](https://doi.org/10.1002/dneu.22829) (2021).
17. Basu, S. et al. 2dspan: Semiautomated 2-d segmentation, classification and analysis of hippocampal dendritic spine plasticity. *Bioinformatics* 32, 2490–2498, DOI: [10.1093/bioinformatics/btw172](https://doi.org/10.1093/bioinformatics/btw172) (2016).
18. Ustinova, A. et al. Generate and analyze three-dimensional dendritic spine morphology datasets with spinetool software. *Curr. Protoc.* 4, e70061, DOI: <https://doi.org/10.1002/cpz1.70061> (2024).
19. Lyu, B., Wang, J., Risher, W. C. & Yu, G. Vsot: volume-surface optimization for accurate ultrastructure analysis of dendritic spines. *Bioinformatics* 41, btaf220, DOI: [10.1093/bioinformatics/btaf220](https://doi.org/10.1093/bioinformatics/btaf220) (2025).
20. Smirnova, D., Ustinova, A., Chukanov, V. & Pchitskaya, E. 3d dendritic spines shape descriptors for efficient classification and morphology analysis in control and alzheimer’s disease modeling neurons. *Bioinformatics* 42, btag025, DOI: [10.1093/bioinformatics/btag025](https://doi.org/10.1093/bioinformatics/btag025) (2026). <https://academic.oup.com/bioinformatics/article-pdf/42/2/btag025/66483564/btag025.pdf>.
21. Lee, T.-C., Kashyap, R. L. & Chu, C.-N. Building skeleton models via 3-d medial surface/axis thinning algorithms. *CVGIP: Graph. Model. Image Process.* 56, 462–478, DOI: [10.1006/cgip.1994.1042](https://doi.org/10.1006/cgip.1994.1042) (1994).
22. Márquez-Neila, P., Baumela, L. & Alvarez, L. A morphological approach to curvature-based evolution of curves and surfaces. *IEEE TPAMI* (2014).
23. Özgün Çiçek, Abdulkadir, A., Lienkamp, S. S., Brox, T. & Ronneberger, O. 3d u-net: Learning dense volumetric segmentation from sparse annotation. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9901 LNCS, 424–432, DOI: [10.1007/978-3-319-46723-8_49](https://doi.org/10.1007/978-3-319-46723-8_49) (Springer Verlag, 2016).
24. Kingma, D. & Ba, J. Adam: A method for stochastic optimization. *Int. Conf. on Learn. Represent.* (2014).
25. Motulsky, H. J. & Brown, R. E. Detecting outliers when fitting data with nonlinear regression - a new method based on robust nonlinear regression and the false discovery rate. *BMC Bioinforma.* 7, DOI: [10.1186/1471-2105-7-123](https://doi.org/10.1186/1471-2105-7-123) (2006).

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- [SpineTool20suppl.pdf](#)