RESPAN: an accurate, unbiased and automated pipeline for analysis of dendritic morphology and dendritic spine mapping

Sergio B. Garcia[1,2], Alexa P. Schlotter[1,2], Daniela Pereira[3], Franck Polleux[1,2], Luke A. Hammond[1,2,#] *

[1] Department of Neuroscience, Columbia University, New York, NY, USA
[2] Mortimer B. Zuckerman Mind Brain Behavior Institute, Columbia University, New York, NY, USA
[3] Champalimaud Research, Champalimaud Centre for the Unknown, Lisbon 1400-038, Portugal

[#] Present address: Department of Neurology, The Ohio State University, Wexner Medical School, Columbus, OH, USA

*Corresponding author

Luke.Hammond@osumc.edu

Abstract

Accurate and unbiased reconstructions of neuronal morphology, including quantification of dendritic spine morphology and distribution, are widely used in neuroscience but remain a major roadblock for large-scale analysis. Traditionally, spine analysis has required labor-intensive manual annotation, which is prone to human error and impractical for large 3D datasets. Previous automated tools for reconstructing neuronal morphology and quantitative dendritic spine analysis face challenges in generating accurate results and, following close inspection, often require extensive manual correction. While recent tools leveraging deep learning approaches have substantially increased accuracy, they lack functionality and useful outputs, necessitating additional tools to perform a complete analysis and limiting their utility. In this paper, we describe Restoration Enhanced SPine And Neuron (RESPAN) analysis, a new comprehensive pipeline developed as an open-source, easily deployable solution that harnesses recent advances in deep learning and GPU processing. Our approach demonstrates high accuracy and robustness, validated extensively across a range of imaging modalities for automated dendrite and spine mapping. It also offers extensive visual and tabulated data outputs, including detailed morphological and spatial metrics, dendritic spine classification, and 3D renderings. Additionally, RESPAN includes tools for validating results, ensuring scientific rigor and reproducibility.

Key Points

- RESPAN is a new automated pipeline for restoration and segmentation of several morphological features of neurons including soma, dendritic shaft, and dendritic spines from various in light microscopy imaging datasets.
- The RESPAN processing pipeline includes data preparation, image restoration, image segmentation, and subsequent visualization and plotting of the resulting data, providing a comprehensive, automated solution for mapping dendrites and dendritic spines.
- By providing clear guidance and an easy to use GUI for using RESPAN and additional Python-based tools, including nnU-Net and SelfNet, we ensure state-of-the-art automated neuron analysis is broadly accessible for a range of diverse datasets and applications.

Introduction

The dendritic morphology and input connectivity of neurons in the central nervous system (CNS) are among the cardinal features defining a neuronal subtype and play critical roles in specifying the functional properties of individual neurons [2,3]. For example, in the mammalian neocortex, long-range projecting pyramidal neurons (PNs) receive most (>90%) of their excitatory synaptic inputs onto small dendritic protrusions termed, by Ramón Santiago y Cajal, dendritic spines ([4]; reviewed in [5]). For example, quantification of dendritic spine density and distribution in cortical pyramidal neurons (PNs) has been used in thousands of studies over the half century as an anatomical surrogate for the measure of the number of excitatory inputs received by individual PNs subtypes either during development or in adults, both during physiological and pathological contexts, including neurodevelopmental disorders or neurodegeneration [6-8]. Furthermore, the size of the spine head is linearly proportional to the size of the postsynaptic density and the number of postsynaptic AMPA glutamate receptors and therefore has been used to estimate excitatory synaptic maturation during development or synaptic weights in adult neurons [9].

Dendritic spine imaging has been transformed by (1) the emergence of bright fluorescent proteins and the improvement of expression methods such as in utero electroporation (IUE), viral methods or transgenic mouse models allowing sparse, cell-type specific expression of various fluorescent cell fillers or postsynaptic fluorescent proteins [10-12] and (2) by improvement in the spatial and temporal resolutions of single-photon and multiphoton optical microscopy for *in vitro*, *ex vivo* or *in vivo* imaging at submicron resolution [13,14]. However, quantifying

dendritic morphology, dendritic spine density, distribution, and spine head size requires cumbersome, most often manual counts and reconstructions, which is labor intensive, biased by human observation, and therefore constitutes a significant bottleneck for large-scale, unbiased, studies of neuronal connectivity.

Recently, several approaches taking advantage of breakthroughs in deep learning (DL) applied to automated image segmentation have been used to improve the accuracy and throughput of dendritic spine mapping [1,15-20]. However, these approaches suffer from several critical limitations, including insufficient generalizability across diverse imaging conditions, lack of integrated validation mechanisms, and the need for substantial manual correction post-segmentation. Furthermore, many existing tools, such as free tools (DeepD3, SpineTool), or the commercially available platform Imaris (RRID:SCR_007370), either require significant manual intervention, lack compatibility with widely used image analysis software, are either cost-prohibitive or not open-source.

In the present study, we describe a new automated computational pipeline called RESPAN (Restoration Enhanced SPine And Neuron) designed to address these limitations comprehensively. RESPAN is a robust, accessible, and fully automated solution that leverages state-of-the-art deep learning models for image restoration, axial resolution enhancement, and precise segmentation of dendritic spines and neurons. This pipeline offers extensive visual and tabulated data outputs, including detailed morphological and spatial metrics, dendritic spine classification, and 3D renderings. Additionally, RESPAN provides tools for validating results against ground-truth data, ensuring scientific rigor and reproducibility. Unlike other previously developed tools with similar goals, RESPAN operates through a single graphical user interface (GUI), working out of the box without requiring extensive knowledge in coding. All steps of the analysis are performed within the RESPAN environment, eliminating the need to switch between different software tools. Our approach demonstrates high accuracy and robustness, validated extensively across a range of imaging modalities, making state-of-the-art automated neuron analysis broadly accessible for diverse research applications.

Results

Development of the protocol

To address the need for a comprehensive, accurate, unbiased, and user-friendly solution for automated 3D dendritic spine analysis, we developed RESPAN. RESPAN (**Figure 1**) is a high-throughput pipeline that facilitates automated reconstruction and analysis of dendritic spines, employing deep learning approaches for segmentation and restoration of signal-to-noise (SNR) and axial resolution. Integrating these approaches substantially improves spine detection accuracy and increases sensitivity to variations in spine morphology and distribution. To ensure the greatest utility, we combine these capabilities with detailed morphological and spatial metrics alongside a dendritic spine classification network and neuron tracing, providing a complete and comprehensive analysis.

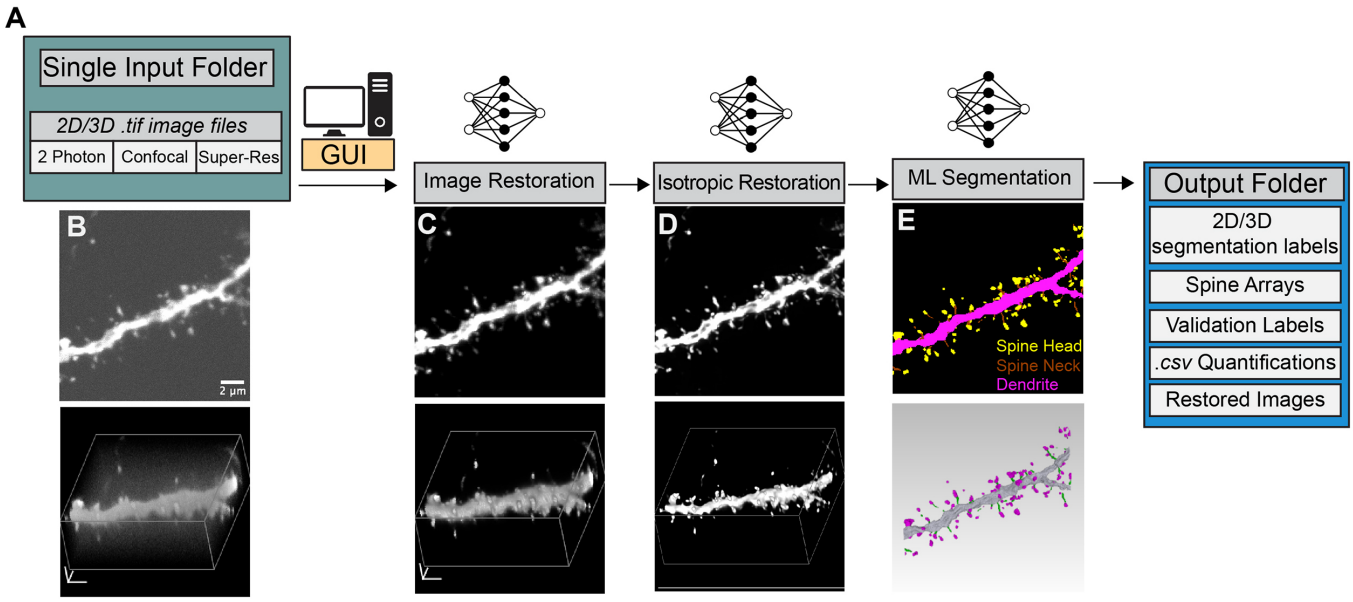

Figure 1. Overview of the RESPAN pipeline for automated dendritic spine mapping. (**A**) Schematic representation of the RESPAN pipeline workflow, beginning with the input of 2D/3D .tif image files and progressing through multiple stages including image restoration, isotropic restoration, segmentation, and quantitative output generation. The pipeline accommodates images from a variety of microscopy techniques including 2-photon, single photon confocal, and super-resolution microscopy. (**B**) Upper panel: Example of a raw fluorescence image acquired with a spinning disk confocal microscope, shown as a maximum intensity projection (MIP), highlighting reduced resolution in the Z-axis axis due to imaging limitations. Lower panel: 3D view of the same raw image. (**C**) Upper panel: Post-restoration image demonstrating the enhanced signal-to-noise ratio achieved using a 3D CARE restoration model, essential for accurate spine detection. Lower panel: 3D view of the post-restoration image. (**D**) Upper panel: Further processing of the CARE restored image using SelfNet model for isotropic restoration, which reduces anisotropy and improves axial resolution, thus improving morphological measurement accuracy. Lower panel: 3D view of the isotropically restored image. (**E**) Upper panel: Final segmentation output from the nnU-Net segmentation model, showing detailed and precise segmentation labels for dendritic spines, necks, and soma. Lower panel: 3D view of the final segmentation output. Scale bars: 2 µm.

Unlike conventional methods that necessitate time-consuming manual annotation, RESPAN provides a streamlined process that can be executed by users with no programming experience through an intuitive graphical user interface (GUI). Furthermore, to ensure ease of deployment, RESPAN is made available as a standalone application in addition to Python code. This eliminates barriers posed by creating and managing Python environments and troubleshooting package installations.

RESPAN incorporates state-of-the-art deep learning models for image restoration (**Figure 1B-C**), enhancing signal-to-noise of fluorescent imaging and spatial resolution critical for measuring dendritic and spine morphology prior to analysis [21-23]. These steps provide significant advantages for experiments with limited signal-to-noise ratio (SNR), such as dynamic imaging of dendrites using 2-photon imaging *in vivo*. These restored images undergo further restoration to reduce anisotropy, improving axial resolution and, thus, the accuracy of morphological measurements. Finally, the images are segmented using an adaptive and self-configuring nnU-Net architecture [21], trained to detect soma, dendrite branches and dendritic spines.

Prior to segmentation, RESPAN allows for optional image restoration (**Figure 1B-C**) to improve signal and contrast. This stage is essential for accurate and robust spine detection in challenging imaging conditions due to limited light exposure to sensitive samples or demanding temporal resolution requirements, such as *in vivo* 2-photon imaging of spines. To achieve this, RESPAN employs a 3D CSBDeep restoration model [23], trained using paired low and high SNR image volumes. These image sets were acquired in a variety of samples to best reflect

the diversity of imaging conditions and biological features, including soma, dendritic branches, and spines, thus maximizing the generalizability of the trained model. For spinning disk confocal microscopy, we used a set of 6 paired image volumes with dimensions 1500 x 1500 x 130, with a pixel size of 65nm and a Z-step size of 150nm. Prior to training the model, the high SNR images were deconvolved using a Richardson-Lucy algorithm with 40 iterations. For 2-photon microscopy, we used 200 paired image volumes, with dimensions ~ 300 x 600 x 12, with a pixel size of 102nm, and a Z-step size of 1µm. Training was performed in Python using image augmentation (rotation and flipping) using a patch sizes between 512 x 512 x 64 and 64 x 64 x 8 for 80 epochs using parameters previously described [23].

Next, an optional axial restoration approach (**Figure 1D**) is used to reduce anisotropy and improve the accuracy of spine morphology measurements. Imaging methods, such as confocal microscopy, often suffer from significantly lower axial compared to lateral resolution, greatly benefit from this approach. RESPAN enables this by incorporating Self-Net, a two-stage unsupervised neural network approach. The model provided with RESPAN was trained in Python as previously described [22], using 6 image volumes with dimensions 1500 x 1500 x 120, with volumes representing a diversity of biological features and imaging conditions [22].

For image segmentation (**Figure 1E**), RESPAN utilizes a 3D full-resolution nnU-Net model [21]. For the models provided with RESPAN, we used image volumes acquired using a Zeiss Airyscan confocal microscope, equipped with a 63x 1.4 NA oil PlanApochromat objective, a Yokogawa W1 spinning disk confocal microscope, equipped with a 100x 1.35NA silicone PlanApochromat objective and a Prime-BSI sCMOS camera, and a Bergamo 2-photon confocal microscope, equipped with an Olympus 25x 1.0 NA SCALEVIEW-A2 Plan objective. To maximize the utility and generalizability of RESPAN, we resampled all image datasets to an optimal lateral sampling of 65nm/voxel and an axial sampling size of 150nm/voxel. This allowed us to pool images from different modalities to create a more generalizable model. Users can take advantage of this by automatically scaling their input data to match that of the model within RESPAN, increasing the likelihood of achieving immediately usable results and making it possible to rapidly improve on these outputs to train experiment-specific models at different resolutions, if required. In some cases, resampling raw data to this high-resolution generalizable model may be inefficient, for this reason, additional models tailored to specific modalities are also provided. Model training utilized a total of 47 image volumes representing a total of 2489 spines accompanied by ground truth labels generated by a consensus of two independent experts.

To further increase generalization and accuracy, training data was expanded using augmentation, including random rotations, flips, the addition of shot noise, and Gaussian blurring (**Figure 2A**), with 3D full-resolution models being trained for 1000 epochs with an initial learning rate of 0.01 and parameters previously described [21]. Following training, the output accuracy of the model compared to ground truth labels in validation volumes was evaluated. The model achieved a pseudo-Dice score of 0.89 for spines and 0.96 for dendrites, with 1 indicating perfect overlap. Models and their corresponding training plans are provided.

Following restoration and segmentation, spines are comprehensively analyzed. This includes intensity and

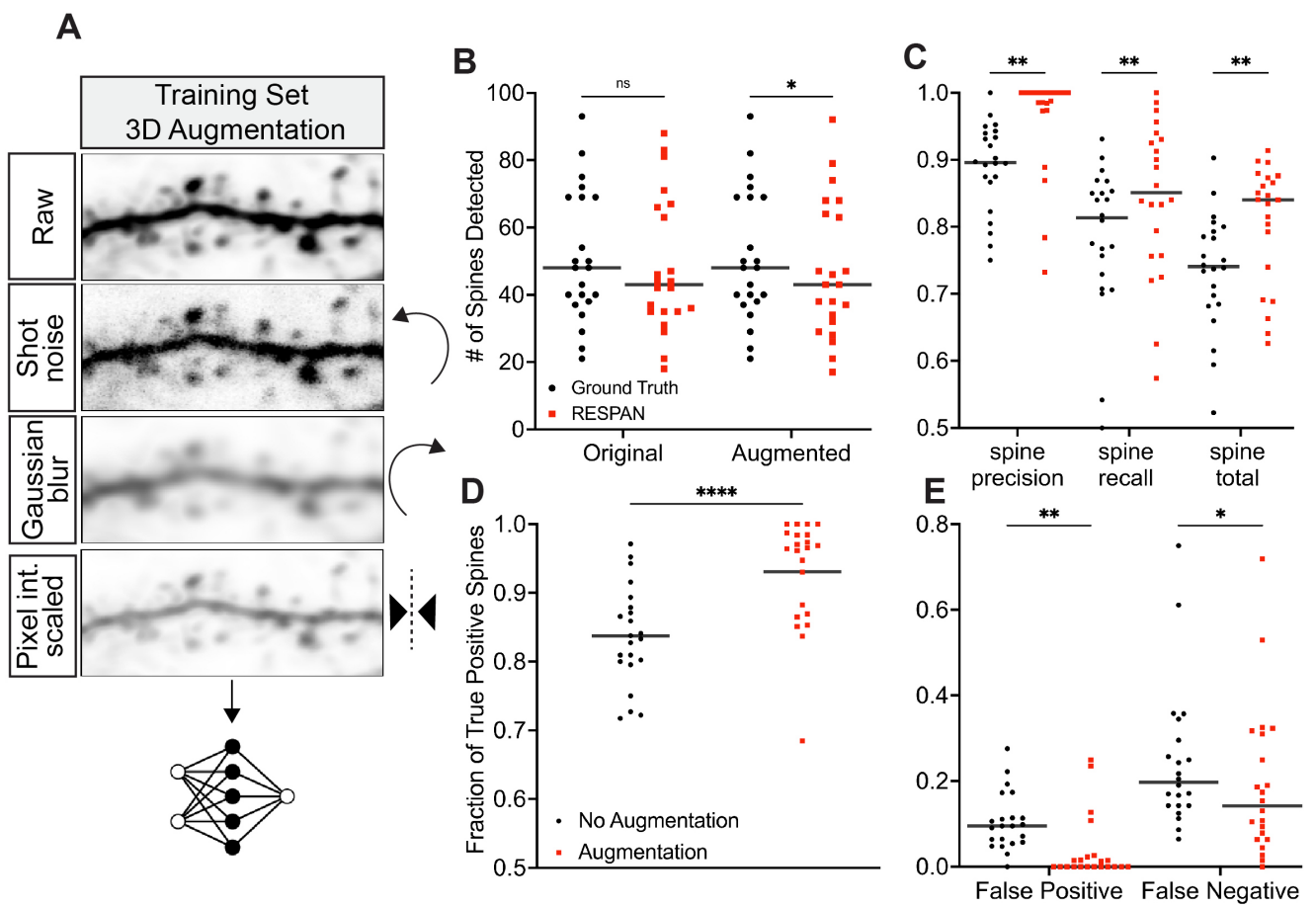

Figure 2. Enhancement of Spine Detection Performance Through Data Augmentation. (**A**) Data augmentation techniques applied during model training, illustrated through representative images: original raw image and subsequent transformations including shot noise, Gaussian blur, and pixel intensity scaling, aimed at enhancing model robustness and generalizability; random rotations and flips were performed to the data through the 3D stack to avoid overfitting during model training. (**B**) Comparison of the number of spines detected in original and augmented datasets by ground truth (black) and RESPAN (red), analyzed using two-way ANOVA, indicating no significant difference (ns) to significant differences (*p<0.05, ****p<0.0001) in detection counts. (**C**) Scatter plots comparing spine detection metrics—precision, recall, and total performance—between original and augmented datasets, analyzed using two-way ANOVA with statistical significance indicated (**p<0.01). (**D**) Bar graph showing the fraction of true positive spines detected, comparing models trained with no augmentation (black) and with augmentation (red), analyzed using the Wilcoxon matched-pairs signed rank test. The asterisks indicate a statistically significant increase in the detection of true positives (****p<0.0001). (**E**) Scatter plot illustrating the reduction in false positives and false negatives in augmented datasets, analyzed using two-way ANOVA, with statistical tests indicating significant differences in error rates (**p<0.01, *p<0.05).

morphology measurements for all channels, including spine width, length, volume, sphericity, solidity, aspect ratio, extent, and roughness. Spine location, the identification label of its parent dendrite, and the local spine density are also reported, along with the distance to soma (if present). A neural network trained to classify each spine using an extracted 3D volume centered on its location further enhances the utility of these measurements.

For studies requiring further information on dendrite and neuron morphology, RESPAN can automatically generate a neuron tracing as a .swc file using Vaa3D's APP2 plugin (23603332) to process the restored and segmented image data. Finally, a summary table reports per-image statistics, including total spine counts, spine density, per-spine measurement averages, dendrite length, and volume. Complementing these statistics, RESPAN generates 2D and 3D spine 'arrays' (**Figure S2**), which are individual cropped dendritic segments corresponding to individual spine detection events and their corresponding masks. These spine arrays enable quick inspection, validation, and subsequent machine-learning classification of individually isolated spines. This capability can also transform large and complex image volumes into readily explorable datasets, facilitating exploration and publishing results in a manner open to interrogation. To demonstrate their usefulness, we provide example spine arrays from a WT neuron dataset (**Figure S2**).

To ensure transparency, RESPAN provides users access to intermediate data, including validation images that overlay analyzed spines with all putative spines and raw data for verification purposes. These outputs allow users to audit the accuracy of spine detection before proceeding with analysis. The final output from RESPAN includes detailed tables and visual representations of spine density, distribution, and other relevant metrics. These results are presented in a format that facilitates easy exploration and interpretation of the data, including density heatmaps, projection overlays, and optional full-resolution 3D datasets.

Estimating accuracy of spine detection by RESPAN against human experts

We employed pixel-level and object-level metrics to evaluate the performance of RESPAN's spine and dendrite detection by comparing segmentation outputs with ground truth validation data excluded from training. For pixel-level metrics, we measured the Intersection over Union (IoU) ($|A \cap B|$ $|A \cup B|$) and the Dice Coefficient (DC) (2 $|A \cap B||A| + |B|$) with A representing RESPAN output pixels, and B representing ground truth pixels respectively. For object-level comparison, we used an IoU metric with a threshold of 50% to quantify detected objects. The resulting true positives (TP), false positives (FP), and false negatives (FP) were used to calculate the proportion of correctly detected objects ($\mathrm{Precision}=\mathrm{TPTP+FP}$) and the proportion of true objects that were correctly detected ($\mathrm{Recall}=\mathrm{TPTP+FN}$). Spine head Precision was 0.93 with a Recall of 0.86, demonstrating highly accurate spine detection.

Evaluation of Multi-User Spine Detection Variability against RESPAN

To further evaluate and quantify the accuracy and reliability of RESPAN in detecting dendritic spines, we compared its performance against manual annotations by multiple users of varying expertise (**Figure 3**). First, we acquired 5 independent fluorescence images of dendritic segments using a spinning disk confocal microscope (see example in **Figure 3A**). All images were subsequently annotated by two experts, with significant experience in microscopy imaging and manual spine detection, to create a ground truth (GT) consensus dataset (**Figure 3B-C**). The ground truth datasets across multiple images were used to assess the spine detection performance of 4 independent human observers with varying degree of expertise (from experienced to naïve users). Second, we asked each user to open the raw fluorescence image and place a single multi-point ROI on what they considered to be the center of mass of a spine head within the volume of the stack. We then compiled all the ROIs placed by each user on the raw fluorescence images and superimposed them onto the GT consensus mask. Using custom Python code, we extracted the number of multi-point ROIs that were detected for a given GT spine mask. We repeated this for all five image stacks across all four users to generate a spine probability map defined as the probability that each spine was detected by all four (probability of 1 in x-axis of panel G of Figure 1), 0.75 if 3 out of 4 users detected spine from GT, and similarly for other probabilities. (**Figure 3D-E**). The spine probability map allows us to visualize the variability and probability of multiple users being able to correctly identify a spine head. We then tabulated the total number of spines detected across all images by each user to generate a correlation matrix (**Figure 3F**). The inter-user correlation matrix depicts the pairwise detection agreement among users, with values ranging from 0 (no agreement) to 1 (perfect agreement). We

noted that the inter-user agreement on the total amount of spines detected for the same image sets remains very low, which would result in statistically different and potentially biased results depending on the user performing the spine annotation (**Figure 3F**).

Next, we plotted the probability of a spine being detected by no users (0), by only one user (0.25), by two users (0.50), by three users (0.75), or by all four users (1.00) (**Figure 3G**). This enabled the calculation of a probability histogram where it became evident that ~10% of spines from GT were not detected by any of the four users (false negative) (**Figure 3G**). In order to compare RESPAN accuracy and performance at spine detection compared to these 4 individual users, we then processed the same raw fluorescence image stacks using the RESPAN pipeline, which automatically segmented the dendritic spines. Using custom written python code, we overlaid the RESPAN segmentation output to our GT consensus mask and used an Intersection over Union (IoU) criteria of 0.50 to decide whether the RESPAN segmentation correctly overlapped with the GT mask per spine. We then plotted the number of spines detected by users at a given probability detected by RESPAN (**Figure 3H**). First, we noticed a stark decrease (> 2 fold) in the fraction of false positive for RESPAN compared to the four users and a reduction in the fraction of spines detected only by some but not all users (probability between 0.25-0.75) (**Figure 2G-H**). Finally, the overall proportion of spines detected by RESPAN that were detected by all four users (probability of 1) compared to GT is slightly increased. This analysis argues that our new RESPAN pipeline performs as well if not better than human observers.

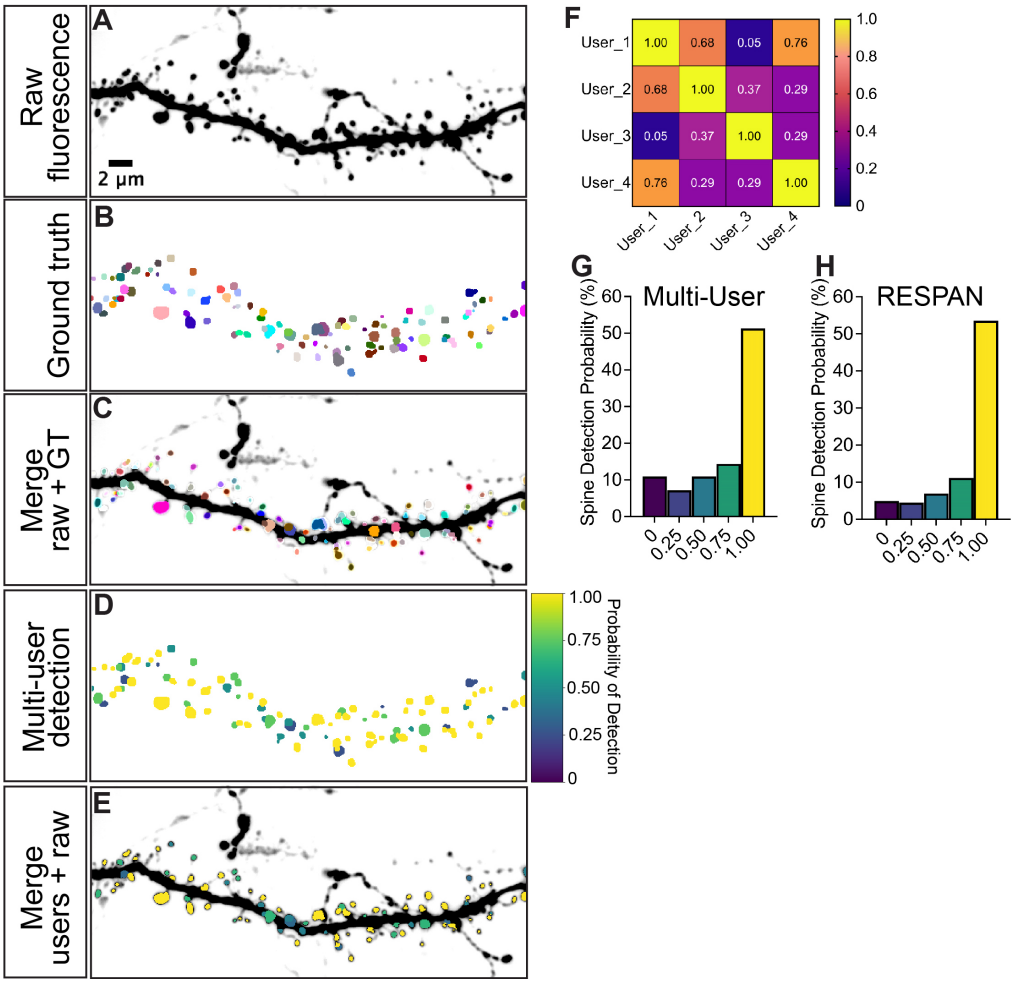

Figure 3. Evaluation of RESPAN performance at spine detection against expert annotation (ground truth) or multiple, non-expert human observers. (**A**) Raw fluorescence image (maximum intensity projection from 3D stack) of a dendritic segment used as input for both manual annotation by multiple experts and automated processing by RESPAN. (**B**) Compilation of ground truth (GT) annotations by experts in the field; colors identify unique spine labels. (**C**) Composite image merging the raw fluorescence with the GT annotations overlaid. (**D**) Image of spine masks showing the probability of detection per spine from four independent users; each spine mask is pseudo-colored to reflect the probability of it being detected with the probability legend on the right. (**E**) Composite image merging the raw fluorescence with the multi-user probability masks overlaid. (**F**) Inter-user correlation matrix showing the pairwise detection agreement among users, with values ranging from 0 (no agreement) to 1 (perfect agreement). (**G**) Histogram of spine detection probabilities by multiple users, indicating the percentage of spines detected at different probability thresholds. (**H**) Histogram of spine detection probabilities by RESPAN, comparing the likelihood of spine detection at various thresholds, illustrating RESPAN's performance relative to manual annotations.

Biological validation of RESPAN

To provide another validation of RESPAN, we tested whether applying it for biological samples with previously published effects on spine density and morphology would enable detection of quantitative phenotype in a completely unbiased way. We conducted a blind genotype/phenotype analysis using *in vivo* 2-photon microscopy of dendritic segments from WT and SRGAP2A$^{+/-}$ mutant neurons. This analysis leveraged previously published findings from our lab demonstrating that SRGAP2A$^{+/-}$ mutants show increased spine density and spine neck length compared to adult littermate wild-type mice, while spine volume remains unchanged compared to WT in adult layer 2/3 CPNs [24-27]. Neurons were sparsely labeled via *in utero* electroporation (IUE) with a FLEX-mGreenLantern (mGL) and low levels of Cre recombinase in order to obtain sparse expression of mGL in layer 2/3 CPNs as previously described [1] and imaged through a cortical window to capture spine morphology over whole dendritic segments (**Figure 4A**).

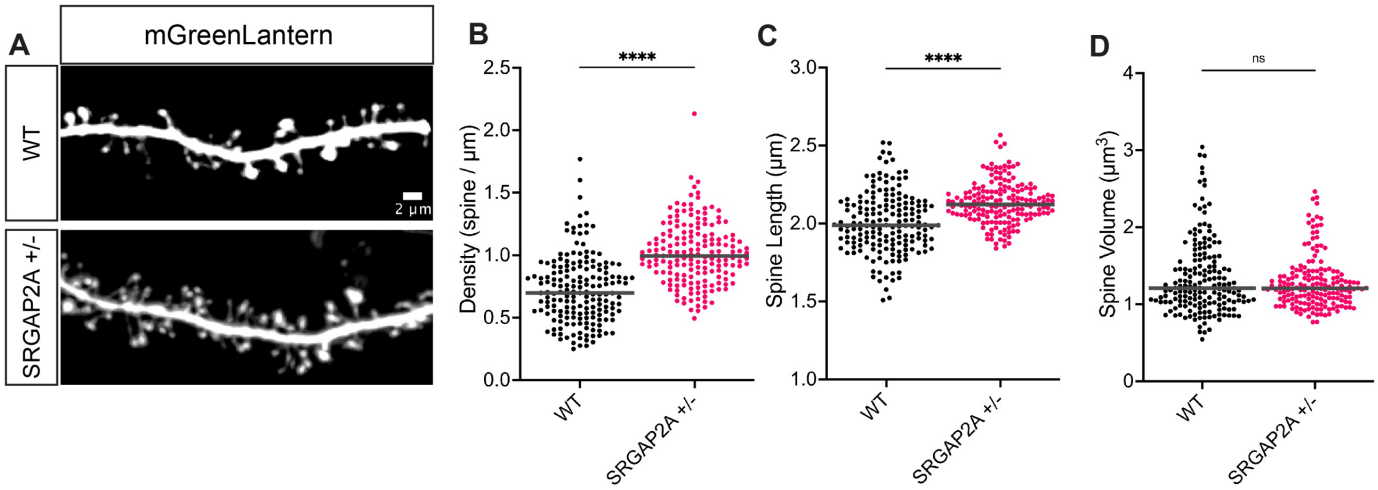

Figure 4. Blind genotype/phenotype validation using RESPAN. (A) *In vivo* 2-photon microscopy images of dendritic segments from layer 2/3 CPNs in wild type (WT) and SRGAP2A$^{+/-}$ mutant mice. Neurons were sparsely labeled through in utero electroporation (IUE) using low amounts of Cre recombinase and a FLEX-mGreenLantern construct, imaged through a cortical window. Scale bar, 2 µm. (**B**) Spine density comparison between WT and SRGAP2A$^{+/-}$ mutants, showing a significant increase in spine density per unit length in mutants. Results analyzed using the non-parametric Kolmogorov-Smirnov test (****p<0.0001). (**C**) Spine length comparison between WT and SRGAP2A$^{+/-}$ mutants, with mutants exhibiting significantly longer spines. Analysis performed using the non-parametric Kolmogorov-Smirnov test (****p<0.0001). (**D**) Comparison of spine volume between WT and SRGAP2A$^{+/-}$ mutants, revealing no significant difference in spine volume, as determined by the non-parametric Kolmogorov-Smirnov test (ns, not significant).

The RESPAN analysis successfully replicated previously characterized phenotypes via an unbiased and blind approach. Quantitative results showed that SRGAP2A$^{+/-}$ mutant mice have a significantly higher spine density compared to WT neurons (**Figure 4B**), confirming the expected increase. Similarly, spines in SRGAP2A$^{+/-}$ mutants were significantly longer than those in WT neurons (**Figure 4C**), consistent with our prior findings. Notably, there was no significant difference in spine volume between WT and SRGAP2A$^{+/-}$ mutants (**Figure 4D**), aligning with the established data that spines mature to the same degree in both genotypes.

This validation confirms that RESPAN can accurately detect and quantify dendritic spine phenotypes in an unbiased and blind manner, providing results that are directly in line with previously published evidence. This demonstrates the RESPAN's robustness and reliability in analyzing dendritic spine morphology in diverse biological contexts.

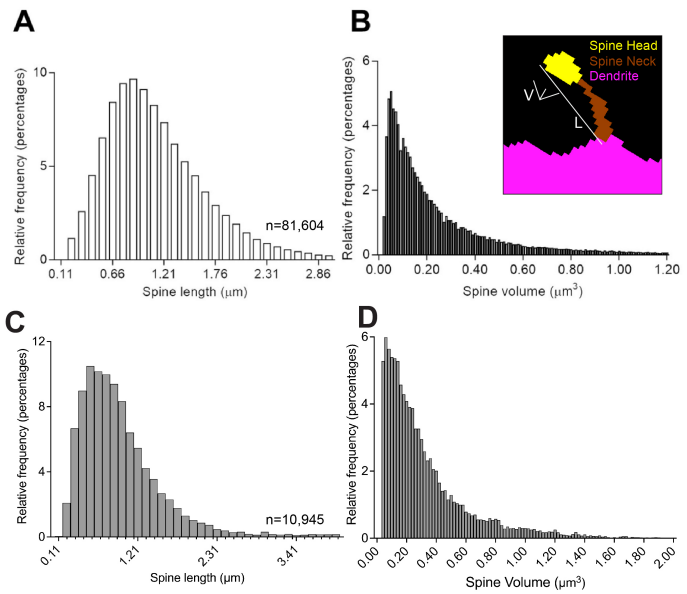

Figure 5. Biological validation of RESPAN performance for unbiased and unsupervised spine length and spine volume quantification. (**A**) Histogram displaying the relative frequency distribution of spine lengths in wild type (WT) neurons, based on a previous analysis of 81,604 spines (see [1]). The x-axis represents spine length in micrometers and the y-axis shows the relative frequency percentage. (**B**) Histogram showing the relative frequency distribution of spine volumes in WT neurons. The x-axis represents spine volume in cubic micrometers and the y-axis shows the relative frequency percentage. The inset illustrates an example of a dendritic spine segmented, color-coded to illustrate the spine head (yellow), spine neck (pink), and dendritic shaft (magenta). (**C**) Histogram depicting the relative frequency distribution of spine lengths in WT neurons, based on analysis of 10,945 spines. This analysis was performed using a RESPAN trained model and the graphs were made using the *.csv* output tables with no further modifications. (**D**) Histogram illustrating the relative frequency distribution of spine volumes in wild-type layer 2/3 CPNs. (D) The graphs were generated from RESPAN analysis of a series of WT neuronal segments acquired using a spinning disk confocal microscope. A spine head volume minimum and maximum threshold of 0.035 and 2 um^3 was used, respectively. Output tables from RESPAN were merged to include all spines within the spine head volume threshold.

To further validate the performance of RESPAN in detection of morphological features of dendritic spines, we analyzed spine length and volume in a larger population of WT neurons. **Figure 5A-B** present data from a previous publication where all excitatory and inhibitory spines on a single neuron were mapped and analyzed, providing a ground truth for spine distribution [1]. This analysis of 81,604 spines previously revealed that spine length follows a normal distribution, while spine volume shows a Poisson (lognormal) distribution with many small spine heads and a smaller population of large spine heads [1].

In our validation, we used RESPAN to analyze 10,945 spines from similar adult layer 2/3 CPNs in wild-type mice imaged using spinning disk confocal microscopy (**Figure 5C-D**). We applied a threshold of 0.035 µm³ and 2 µm³ for spine volume in RESPAN. The results showed that RESPAN was able to replicate the same distribution patterns for spine length and spine volume as previously characterized in the ground truth data. Specifically, RESPAN detected spine length and spine volume which conforms to similar distributions than previously published dataset using Vaa3D [1].

These findings confirm that RESPAN can consistently reproduce known and well-established biological phenotypes of spine density and morphology, further validating its accuracy and reliability for automated dendritic spine analysis. Therefore, RESPAN performs robustly across different datasets, providing outputs that are directly in line with previously published evidence.

Discussion

Advantages and comparison with other methods

Reproducibility remains a critical challenge facing neuroscience research. Our new approach, RESPAN, addresses this by providing a robust means of quantifying dendritic spines in diverse datasets. Unlike manual or

semi-automated approaches, which involve adjustments, fine-tuning, corrections, and other interactions that introduce intra- and inter-observer variability and bias, RESPAN offers a fully automated and consistent approach. This process relies on restoring image volumes and employing deep learning models, which can be audited and refined for enhanced accuracy.

To our knowledge, RESPAN is the only freely available tool that integrates image restoration, axial resolution restoration, and deep learning segmentation into a single, user-friendly application. This comprehensive approach provides a complete solution for a process that would otherwise require combining a variety of image analysis tools in addition to Fiji and Python, with significant overhead involved in transferring and tracking data between these operations before yielding a result. Additionally, instead of performing image operations on the CPU, RESPAN leverages GPU processing to efficiently handle images in parallel, facilitating the generation of 2D and 3D spine 'arrays'.

Critically, unlike other methods, RESPAN provides a built-in tool for automatically validating the quality of analysis against ground-truth data. Segmentation validation is often performed manually or less rigorously by researchers with limited coding experience. By incorporating this capability, we believe RESPAN can encourage a rigorous approach to automated quantification and increase the repeatability of published results.

RESPAN significantly lowers the barriers to performing advanced image quantification for researchers without coding expertise. This addresses challenges presented by other tools that require Python or MATLAB scripts, including tools that have been incorporated into our approach. Furthermore, unlike other methods that limit analysis outputs to dendritic spine coordinates [17], or require the use of multiple programs not within the same environment [17], our approach provides comprehensive 3D analysis within a single graphical user interface environment. RESPAN can also be readily adapted to new research questions using Fiji, a widely used image analysis software, and the tools within RESPAN (Fiji: doi:10.1038/nmeth.2019).

RESPAN outperforms existing methods in terms of accuracy and ease-of-use. For a detailed comparison of RESPAN with similar tools, please refer to **Table 1**.

Limitations

RESPAN currently only supports input data in TIFF format, requiring conversion of proprietary formats to TIFF prior to use in our software. To overcome this limitation, we provide a Fiji macro with a GUI to facilitate batch conversation to TIFF using OMERO Bio-Formats [28]. For ease of use, we also generate our output data in TIFF, but future releases would benefit from added support for Zarr format.

Users should also be aware of the limitations of applying deep learning models to image data [29]. Restoration models should be trained for the specific biological features, resolution, and imaging modality they are being used for. Otherwise, in addition to performing sub optimally, these models can artefactually generate features that do not exist and degrade features that do exist. Users should also be aware of this characteristic if comparing conditions with appreciable changes in morphology and train models that capture the full diversity of biological features across such conditions.

Expertise needed to implement the protocol

This protocol is aimed at all neuroscientists or biologists and allows researchers with no computational expertise to automatically analyze image volumes and validate their results against ground truth data using a standalone application with a graphical user interface (GUI). We also provide clear guidance in approaches necessary to fine-tune and train models better adapted to the researcher's own image data, where possible, facilitating this in our GUI as well.

RESPAN requires limited specialized expertise and is distributed with trained models that can be readily adapted to most confocal image volumes with sufficient resolution to resolve spines. In cases where new imaging conditions or modalities are being used, new models should be trained, with experts reaching a consensus on manual annotations.

Experimental Design

To utilize RESPAN, image volumes should be acquired using a technique that provides sufficient resolution and optical sectioning to resolve spines, such as confocal or light-sheet microscopy. These datasets must include at least one channel representing the labeled dendrite or neuron. Additional channels can also be included and used for intensity measurements.

Materials and methods

Equipment

- A Windows computer with at least 32 GB of RAM and an NVIDIA GPU is required. For benchmarking, RESPAN was deployed on a workstation running Windows 10 Enterprise 64 bit, equipped with a 16 core AMD Ryzen Theadripper PRO 5955WX, 256 GB DDR4 memory, a 1TB Samsung 860 SSD and a 24 GB NVIDIA GeForce RTX 4090 GPU.

Data

- Example data: Example datasets are available from the GitHub repository (https://github.com/lahammond/RESPAN)

Software

- RESPAN executable (https://github.com/lahammond/RESPAN) and Python 3.9 environment (https://www.anaconda.com/download) installed with pytorch 2.3.0, torchvision, pytorch-cuda=12.1, and nnunet 2.3.1

Equipment setup

Downloading and installing RESPAN

Download the RESPAN application .zip file from the GitHub repository (https://github.com/lahammond/RESPAN) to the Windows computer to be used for analysis. Image restoration and segmentation models, as well as a Fiji macro for file conversion are available on the RESPAN Github page.

Creating Python environment for nnU-Net

To ensure optimal performance we recommend installing nnU-Net in a separate environment using the instructions below:

1. Install Anaconda https://www.anaconda.com/download
2. Download and unzip nnU-Net 2.3.1 from https://github.com/MIC-DKFZ/nnUNet/releases/tag/v2.3.1
3. Run Anaconda Prompt and use the following commands:

```
conda create -n nnunet python=3.9

conda install pytorch torchvision pytorch-cuda=12.1 -c pytorch -c nvidia
```

4. Change directory to the directory where nnU-Net 2.3.1 was unzipped. For example:

```
cd C:\Downloads\nnUNet
```

5. Install nnU-Net:

```
pip install -e .
```

Procedure 1: Running RESPAN from the GUI, without fine-tuning or restoration

Preparing data for RESPAN analysis

Timing 1-5 min

1. Organize data by storing images as TIFF files in subfolders within a single parent folder. Within each subfolder, store related images, such as multiple time points for a single dendrite/neuron or multiple

dendrites/neurons for a single experimental condition. This folder structure facilitates analysis, including spine tracking and comparisons between conditions.

CRITICAL: If your image data is stored in a proprietary format, such as .ND2 or .CZI, please use the provided Fiji macro to batch convert to TIFF format.

RESPAN is designed to batch process multiple datasets, even when the experimental conditions or analysis settings differ. The Analysis_Settings.yml file facilitates this step by storing data processing settings and the locations of restoration models.

2. Place a copy of the provided Analysis_Settings.yml file in each subfolder. This file does not need to be modified to use RESPAN for standard use. Refer to Procedure 3 for advanced use.

Running RESPAN

Timing 2 min of unattended computer time per dataset (depending on dataset size)

3. Start RESPAN. In the GUI that appears, select the folder containing subfolders to be processed and update the other parameters as needed. See Box 1 for further information.
4. Click “Run”. The log window within the GUI will update as the analysis progresses.
5. Results will be automatically saved in subfolders. Refer to Anticipated Results for further information.

Procedure 2: Validating RESPAN outputs

Preparing data for validation

Timing 1-5 min

1. Generate a ground truth dataset, either by manually creating annotations or by correcting annotations generated by RESPAN.

    CRITICAL: If correcting annotations generated by RESPAN to create a ground truth dataset, it is important to remember these annotations and their corresponding images cannot be included in the training set for a new model if they will also be used to validate that model.

Running RESPAN

Timing 2 min of unattended computer time per dataset (depending on dataset size)

2. Start RESPAN. In the GUI that appears, select the “Analysis Validation” tab.
3. Select the directory containing results from RESPAN and the directory containing ground truth annotations for the images processed by RESPAN.
4. Update the Spine and Dendrite Detection parameters to match those used when processing the images with RESPAN.
5. Click “Run”. The log window within the GUI will update as the analysis progresses.
6. An “Analysis_Evaluation.csv” file will be automatically saved within the folder containing the results from RESPAN. Refer to Anticipated Results for further information.

Procedure 3: Running RESPAN from the GUI, with fine-tuning and restoration

Preparing data for RESPAN analysis

Timing 1-5 min

1. Organize data as described in Procedure 1.
2. Place a copy of the provided Analysis_Settings.yml file in each subfolder.

Modifying analysis settings

Timing 2 min

3. Open Analysis_Settings.yml using an editor such as Sublime Text (https://www.sublimetext.com/) to adjust specific settings required in each subfolder.
4. For each channel you wish to preprocess with image restoration, set the corresponding "care_restore" field to True. Then update the "care_model_path" field with the directory containing the restoration model to be used.
5. For each channel you wish to preprocess with axial resolution restoration, set the corresponding "sn_restore" field to True. Then update the "sn_model_path" field with the directory containing the restoration model to be used.

Running RESPAN

Timing 5 min of unattended computer time per dataset (depending on dataset size)

6. Start RESPAN and update the GUI as previously described.

Procedure 4: Training a nnU-Net model with new data

Preparing data for nnU-Net training

Timing 1-3 hours

1. Organize image volumes containing the fluorescence channel used for spine and dendrite segmentation into a single folder. If image restoration will be used in RESPAN prior to segmentation, process these images using the same restoration models prior to adding them to this folder.
2. In a separate folder, organize corresponding image volumes containing label annotations, either created manually or by correcting label outputs from RESPAN. These files should have the same filename as their corresponding fluorescence channel image.

Training the model

Timing 12-24 hours

3. Start RESPAN, and at the top of the GUI, select the nnU-Net Training tab.
4. Within this tab, select the directory containing the fluorescence channel image volumes. Then select the folder containing the label annotation image volumes.
5. Click "Run" to initiate training.
6. When training is complete, a new segmentation model will be created in the directory containing the fluorescence channel image volumes.

Procedure 5: Training a CARE-3D model for image restoration

Preparing data for CARE-3D training

Timing 30 min

1. Organize image volumes acquired using low SNR parameters into a single folder.
2. In a separate folder, organize corresponding image volumes acquired using high SNR parameters.

Training the model

Timing 3-5 hours

3. Start RESPAN, and at the top of the GUI, select the CARE Training tab.

4. Within this tab, select the directory containing the low SNR image volumes. Then, select the folder containing the high SNR image volumes.

5. Click “Run” to initiate training.

6. When training is complete, a new CARE restoration model will be created in the directory containing the low SNR image volumes.

Procedure 6: Training a SelfNet model for axial restoration

Preparing data for SelfNet training

Timing 10 min

1. Organize image volumes containing the fluorescence channel used for spine and dendrite segmentation into a single folder. If image restoration will be used in RESPAN prior axial resolution restoration, process these images using the same image restoration model prior to adding them to this folder

Training the new model

Timing 0.25 - 2 hours

2. Start RESPAN, and at the top of the GUI, select the SelfNet Training tab.

3. Within this tab, select the directory containing the image volumes organized in step 1.

4. Click “Run” to initiate training.

When training is complete, a new SelfNet restoration model will be created in the directory containing the image volumes.

Timing

An image volume of 100MB is expected to take 1 min to process in RESPAN.

Anticipated Results

By following the RESPAN protocols, it is possible to segment and quantify the dendritic shaft and spines in an automatic and unbiased manner. The final RESPAN outputs include validation images, 3D renderings of dendrites and spines, a .swc tracing file of the dendrite or neuron, tabular results for each image, and a summary table for each condition or time series. Intermediary and 3D full-resolution validation data can also be saved, if desired.

Intermediary data for validation

RESPAN will create a folder \Validation_Data, which contains image data used for validating the performance of the analysis and allowing faster reprocessing of datasets if spine filtering settings in the GUI need to be refined. Within this folder, the \Restored_Images subfolder contains image volumes following restoration. The \Segmentation_Labels subfolder contains the initial segmentation results with image volumes consisting of voxels labeled for dendrites, spines, and soma. If the provided models don’t perform accurately, the annotations

in these output images can be corrected and then used to train a new segmentation model. The \Validation_MIPs subfolder contains multi-channel Maximum Intensity Projection (MIP) images used to audit RESPAN's performance. The channels in these images are 1) input channel used for spine analysis, 2) all putative spines, 3) quantified spines with unqiue IDs matching tabular data, 4) quantified dendrites with unique ID matching tabular data, 5) dendrite/neuron skeleton, 6) distance from dendrite. Finally, if "Save 3D validation data" is enabled in the GUI, the \Validation_Vols subfolder will contain multi-channel full-resolution 3D volumes with the same set of six channels.

Tabular data

The \Tables folder will contain tabular data in .csv format. Each input volume will generate an individual results file with the filename "Detected_spines_{filename}.csv" and contents matching those described in Table 2. A summary of results from all files in the input folder is also generated with the filename "Detected_spines_summary.csv" and contents matching those described in Table 3.

Neuron tracing

If "Generate SWC file for dendrite/neuron" is enabled in the RESPAN GUI, the \SWC_files folder will contain a .swc tracing file for each input volume. These can be viewed and analyzed further in neuron analysis tools such as Vaa3D [20] or Simple Neurite Tracer [30].

Spine arrays

RESPAN extracts a 3D image volume around each detected spine and saves these as arrays for further analysis and visualization. The \Spine_Arrays folder will contain four spine array images for each input volume with the following channels: 1) input channel used for spine analysis, followed by other channels in the image, 2) spine mask with unique ID matching tabular data, 4) dendrite mask. "Masked_Spines_MIPs_{filename}.tif" and "Masked_Spines_Slices_{filename}.tif" contain signal only within the detected spine as a MIP of the 3D volume, or a single slice centered on the detected spine, respectively. "Spine_MIPs_{filename}.tif" and "Spine_slices_{filename}.tif" also include signal outside the spine as a MIP or slice, respectively. If "Save 3D spine arrays" is selected, a "Spine_vols_{filename}.tif" image will be generated containing 3D volumes centered around each detected spine.

Analysis Evaluation

When performing analysis validation from the RESPAN GUI, an "Analysis_Evaluation.csv" file will be generated with contents matching those described in **Table 4**.

Box 1 | RESPAN Parameters

This Box describes the parameters needed in the RESPAN GUI and additional considerations.

Image Data and Model Selection

- **Select data directory:** the location of the parent folder, which contains subfolders, each with an analysis_settings.yml file and TIF files to be processed.
- **Select segmentation model directory:** the location of the nnU-Net model directory provided with this software or one trained on your data.

Spine and Dendrite Detection

- **Channel containing dendrite/neuron signal:** specify the channel best representing the dendrite/neuron. If processing a single channel dataset, leave as 1.
- **Minimum dendrite volume (µm3):** putative dendrites or neurons detected smaller than this volume and their associated spines will be excluded from the analysis.
- **Spine volume filter (µm3):** specify the minimum and maximum volume of spines to be analyzed. Putative spines detected outside this range will be excluded from the analysis.
- **Spine distance filter (µm):** maximum distance spines should extend from the dendrite. Spines located further than this distance will be excluded from the analysis.

Image Data and Model Resolution

- **Image voxel size XY (µm):** lateral sampling used for acquiring data.
- **Image voxel size XY (µm):** axial sampling or step-size used for acquiring data.
- **Model voxel size XY (µm):** lateral sampling in data used for training segmentation model.
- **Model voxel size XY (µm):** axial sampling in data used for training segmentation model.
- **Use voxel sizes in analysis_settings.yml:** If enabled, overrides the above scaling parameters in favour of those specified in the settings file. Useful when processing datasets acquired using multiple imaging conditions or requiring different restoration models.

Temporal Analysis and Spine Tracking

- **Spine tracking:** If enabled, allows subsequent tracking and temporal analysis of spines.
- **Histogram matching:** Used to match signal intensity across image volumes being used for spine tracking.
- **Registration method:** Method used for registering image volumes for spine tracking.

Additional Options

- **Save 3D validation data:** Saves complete 3D datasets including labeled spines, which is useful for 3D validation, but significantly increases data storage requirements.
- **Save intermediate data:** Saves all intermediate data for troubleshooting challenging datasets.
- **Use image restoration:** If enabled, utilizes models specified in the analysis_settings.yml file to restore signal-to-noise using CARE.
- **Use axial restoration:** If enabled, utilizes models specified in the analysis_settings.yml file to restore signal-to-noise using SelfNet.
- **Generate SWC file for dendrite/neuron:** If enabled, generates an SWC file for the dendrite/neuron using Vaa3D. Requires Vaa3D to be installed and path to be correct in the analysis_settings.yml file.

Box 2 | Analysis Settings

This Box describes the parameters in the Analysis_Settings.yml file that can be modified for the advanced use of RESPAN.

Parameters

- input_resXY: The input resolution in the X and Y axis.
- input_resZ: The input resolution in the Z axis.
- model_resXY: The resolution in the X and Y axis of the data used to train the model.
- model_resZ: The resolution in the Z axis of the data used to train the model.

Analysis:

- **tiles_for_prediction:** The number of tiles used when processing images. This can be adjusted to modify performance.
- **roi_volume_size:** The size of volume extracted for each detected spine (in μm)
- **GPU_block_size:** Parameters used to adjust the image volume used for GPU processing. This can be modified to improve performance depending on GPU specifications.

nnUnet

- **type:** The type of nnU-Net model being used for segmentation.
- **conda_path:** The directory where Anaconda is installed.
- **env:** The environment where nnU-Net is installed.

Vaa3D:

- **path:** The directory where Vaa3D is installed.

Channe1-4:

- **care_restore:** The variable used to enable image restoration for this channel (True/False).
- **care_model_path:** The directory containing the model for image restoration for this channel.
- **sn_restore:** The variable used to enable axial restoration for this channel (True/False).
- **sn_model_path:** The directory containing the model for axial restoration for this channel.

Table 1 | Software Comparison

Feature/Metric	SpinePipe	SpineTool	DeepD3	3DSpan	DeepSpineTool	Imaris
Compatible input formats	.tif	*.tif*	.tif	*.tif, .hdr, .img*	*.tif*	.ims
Inputs required for neural network training	Input image and corresponding label image (.tif)		Input image corresponding spine annotation (.tif) traced dendrite (*.swc)*			
Image outputs	spine, dendrite, and soma masks overlaid with raw image data (.tif) spine arrays in 2D and 3D (.tif) neuron/dendrite tracting (.swc)	binarized image (.tif) spine meshes (*.off)*	spine ROIs (*.rois)* spine masks (*.tif)* spine prediction maps (*.prediction)*	segmented sub-volumes (*.hdr/.img)*	segmentation (.tif)	Segmentation (.tif)
Tabular outputs	detected spine counts and metrics per image and summary table (.csv)	spine and dendrite classification (*.json)*		segmentation analysis (.csv)		Detected spine counts and metrics (.csv)
Installation requirements	Windows PC with an NVIDIA GPU , Python 3.9, Anaconda	Windows PC, 1GB RAM, 6GB disk space, Anaconda	Anaconda Python 3.7 environment with tensorflow and deepd3 package	Windows PC	Windows PC, Python 3.6, CUDA 10.1, 8GB GPU, 8GB RAM	Windows PC or MacOS
Acquisition modality used for training models	2-photon, spinning disk confocal, Airyscan laser scanning confocal	Confocal	2-photon	2-photon	Not specified	NA
User Experience	GUI for single image and batch processing	Jupyter Notebook	GUI for single image analysis, scripting required for batch analysis	GUI	GUI	GUI
Open Source Software	Yes	Yes	Yes	Yes	No	No

Resolution Specification Option	Yes	Yes	No	Yes	No	No
Batch Analysis Option	Yes	No	Yes	No	No	Yes
Train Model on User Data	Yes	No	Yes	No	No	No
Spine Count Output	Yes	No	Yes	Yes	No	Yes
Validation Option	Yes	No	Yes	No	No	No
ImageJ Compatible Output	Yes	Yes	Yes	No	Yes	Yes

Table 2 | Detected_spines (per image)

Column	Title	Description
1	label	Unique identifier assigned to each spine
2	X	X-coordinate of the spine, indicating its horizontal position within the image
3	Y	Y-coordinate of the spine, indicating its vertical position within the image
4	Z	Z-coordinate of the spine, indicating its position along the depth axis in the image stack
5	spine_area_um2	Cross-sectional area of the spine measured in μm^2
6	Spine_vol_um3	Volume of the spine volume measured in μm^3
7	Spine_length_um	Length of the spine measured in µm
8	Dist_to_dendrite_um	Distance from the spine to the nearest point on the parent dendrite measured in µm
9	Dist_to_soma_um	Distance from the spine to the nearest point on the parent soma measured in µm
10	C1_mean_int	Mean intensity of the first channel (C1) measured within spine
11	C1_max_int	Maximum intensity of first channel (C1) measured within spine
12	C1_int_density	Integrated density, or sum intensity (mean * volume) of first channel (C1) measured within spine
13+		Mean intensity, max intensity, and integrated density for additional channels present in the image volume

Table 3 | Detected spines summary (per series)

Column	Title	Contents
1	Filename	Input image filename
2	res_XY	Input image resolution in the X and Y axis
3	res_Z	Input image resolution in the Z-axis
4	dendrite_length_um	Length of the dendrite in µm
5	dendrite_vol_um3	Volume of the dendrite in μm^3
6	total_spines	Total number of dendritic spines quantified
7	spines_per_um	Density of spines per µm of dendrite length
8	spines_per_um3	Density of spines per μm^3 of dendrite volume
9	avg_spine_area_um2	Average area of spines, in μm^2
10	avg_spine_vol_um3	Average volume of spines in μm^3
11	avg_spine_length_um	Average length of spines in µm from the dendritic surface
12	avg_dist_to_dendrite_um	Average distance of spines in µm to dendritic surface
13	avg_dist_to_soma_um	Average distance of spines in µm to soma surface
14	avg_C1_mean_int	Average mean intensity of the first channel (C1) measured within spines
15	avg_C1_max_int	Average maximum intensity of the first channel (C1) measured within spines
16	avg_C1_int_density	Average integrated density, or sum intensity (mean * volume) of the first channel (C1) measured within spines
17+		Average mean intensity, max intensity, and integrated density for additional channels present in the image volume

Table 4 | Analysis Evaluation

Column	Title	Contents
1	Filename	Input image filename
2	res_XY	Input image resolution in the X and Y axis
3	res_Z	Input image resolution in the Z-axis
4	gt_total_spines	Total spines detected in the ground truth image
5	gt_total_spines_filtered	Total spines analyzed after filtering in the ground truth image
6	output_total_spines	Total spines detected by RESPAN
7	output_total_spines_filtered	Total spines analyzed after filtering by RESPAN
8	TruePos_IoU50	Number of true positive spines detected
9	FalsePos_IoU50	Number of false positive spines detected
10	FalseNeg_IoU50	Number of false negatives
11	Spine_precision_IoU50	The precision of the spine detection
12	Spine_recall_IoU50	The recall of the spine detection
13	gt_total_spine_vol	Total spine volume in the ground truth image
14	output_total_spine_vol	Total spine volume detected by RESPAN
15	total_spine_iou	The Intersection over Union (IoU) of spine voxels in the ground truth image and detected by RESPAN
16	total_spine_vol_difference	The difference in total spine volume between the ground truth image and detected by RESPAN
17	gt_dendrite_length	The total length of the dendrite(s) in the ground truth image
18	output_dendrite_length	The total length of the dendrite(s) detected by RESPAN
19	dendrite_length_difference	The difference between dendrite lengths between the ground truth image and those detected by RESPAN
20	gt_dendrite_vol	The total volume of the dendrite(s) in the ground truth image
21	output_dendrite_vol	The total volume of the dendrite(s) detected by RESPAN
22	total_dendrite_iou	The Intersection over Union (IoU) of dendrite voxels in the ground truth image and detected by RESPAN
23	dendrite_vol_difference	The difference between dendrite volume in the ground truth image and those detected by RESPAN

Author contributions statements

Conceptualization: SBG, FP, LH; Methodology: SBG, LH; Investigation: SBG, APS, DP, LH; Formal analysis: SBG, APS, DP, LH; Critical Reagents: SBG, LH; Funding acquisition: FP, LH; Project administration: SBG, FP, LH; Supervision: SBG, FP, LH; Writing – original draft: SBG, FP, LH.

Acknowledgments

We thank Aleksandra Recupero and Emiliano Zamponi from the Polleux lab for helping with image annotation as well as Polleux lab members for extensive discussions. We thank Qiaolian Lu for her help with mouse colony management. Imaging was performed in the Zuckerman Institute's Cellular Imaging platform. This work was also supported by a grant from the National Institute of Health (NIH, R35 NS127232) and a NOMIS Foundation Research Project.

Competing interests

The authors declare no competing interests.

References cited

1 Iascone, D. M. *et al.* Whole-Neuron Synaptic Mapping Reveals Spatially Precise Excitatory/Inhibitory Balance Limiting Dendritic and Somatic Spiking. *Neuron* **106**, 566-578 e568 (2020). https://doi.org:10.1016/j.neuron.2020.02.015

2 *Dendrites*. (Oxford University Press, 2016).

3 Lefebvre, J. L., Sanes, J. R. & Kay, J. N. Development of dendritic form and function. *Annu Rev Cell Dev Biol* **31**, 741-777 (2015). https://doi.org:10.1146/annurev-cellbio-100913-013020

4 Cajal, S. R. Estructura de los centros nerviosos de las aves. *Rev. Trim. Histol. Norm. Patol.* **1**, 1-10 (1888).

5 Garcia-Lopez, P., Garcia-Marin, V. & Freire, M. The discovery of dendritic spines by Cajal in 1888 and its relevance in the present neuroscience. *Prog Neurobiol* **83**, 110-130 (2007). https://doi.org:10.1016/j.pneurobio.2007.06.002

6 Fiala, J. C., Spacek, J. & Harris, K. M. Dendritic spine pathology: cause or consequence of neurological disorders? *Brain Res Brain Res Rev* **39**, 29-54 (2002). https://doi.org:10.1016/s0165-0173(02)00158-3

7 Herms, J. & Dorostkar, M. M. Dendritic Spine Pathology in Neurodegenerative Diseases. *Annu Rev Pathol* **11**, 221-250 (2016). https://doi.org:10.1146/annurev-pathol-012615-044216

8 Yuste, R. & Bonhoeffer, T. Genesis of dendritic spines: insights from ultrastructural and imaging studies. *Nat Rev Neurosci* **5**, 24-34 (2004). https://doi.org:10.1038/nrn1300

9 Matsuzaki, M. *et al.* Dendritic spine geometry is critical for AMPA receptor expression in hippocampal CA1 pyramidal neurons. *Nat Neurosci* **4**, 1086-1092 (2001). https://doi.org:10.1038/nn736

10 Feng, G. *et al.* Imaging neuronal subsets in transgenic mice expressing multiple spectral variants of GFP. *Neuron* **28**, 41-51 (2000). https://doi.org:10.1016/s0896-6273(00)00084-2

11 Saito, T. & Nakatsuji, N. Efficient gene transfer into the embryonic mouse brain using in vivo electroporation. *Dev Biol* **240**, 237-246 (2001). https://doi.org:10.1006/dbio.2001.0439

12 Tabata, H. & Nakajima, K. Efficient in utero gene transfer system to the developing mouse brain using electroporation: visualization of neuronal migration in the developing cortex. *Neuroscience* **103**, 865-872 (2001). https://doi.org:10.1016/s0306-4522(01)00016-1

13 Denk, W. *et al.* Anatomical and functional imaging of neurons using 2-photon laser scanning microscopy. *J Neurosci Methods* **54**, 151-162 (1994). https://doi.org:10.1016/0165-0270(94)90189-9

14 Fuhrmann, M. *et al.* Super-Resolution Microscopy Opens New Doors to Life at the Nanoscale. *J Neurosci* **42**, 8488-8497 (2022). https://doi.org:10.1523/JNEUROSCI.1125-22.2022

15 Argunsah, A. O. *et al.* An interactive time series image analysis software for dendritic spines. *Sci Rep* **12**, 12405 (2022). https://doi.org:10.1038/s41598-022-16137-y
16 Das, N. *et al.* 3dSpAn: An interactive software for 3D segmentation and analysis of dendritic spines. *Neuroinformatics* **20**, 679-698 (2022). https://doi.org:10.1007/s12021-021-09549-0
17 Fernholz, M. H. P., Guggiana Nilo, D. A., Bonhoeffer, T. & Kist, A. M. DeepD3, an open framework for automated quantification of dendritic spines. *PLoS Comput Biol* **20**, e1011774 (2024). https://doi.org:10.1371/journal.pcbi.1011774
18 Pchitskaya, E., Vasiliev, P., Smirnova, D., Chukanov, V. & Bezprozvanny, I. SpineTool is an open-source software for analysis of morphology of dendritic spines. *Sci Rep* **13**, 10561 (2023). https://doi.org:10.1038/s41598-023-37406-4
19 Peng, H., Bria, A., Zhou, Z., Iannello, G. & Long, F. Extensible visualization and analysis for multidimensional images using Vaa3D. *Nat Protoc* **9**, 193-208 (2014). https://doi.org:10.1038/nprot.2014.011
20 Peng, H., Ruan, Z., Long, F., Simpson, J. H. & Myers, E. W. V3D enables real-time 3D visualization and quantitative analysis of large-scale biological image data sets. *Nat Biotechnol* **28**, 348-353 (2010). https://doi.org:10.1038/nbt.1612
21 Isensee, F., Jaeger, P. F., Kohl, S. A. A., Petersen, J. & Maier-Hein, K. H. nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation. *Nat Methods* **18**, 203-211 (2021). https://doi.org:10.1038/s41592-020-01008-z
22 Ning, K. *et al.* Deep self-learning enables fast, high-fidelity isotropic resolution restoration for volumetric fluorescence microscopy. *Light Sci Appl* **12**, 204 (2023). https://doi.org:10.1038/s41377-023-01230-2
23 Weigert, M. *et al.* Content-aware image restoration: pushing the limits of fluorescence microscopy. *Nat Methods* **15**, 1090-1097 (2018). https://doi.org:10.1038/s41592-018-0216-7
24 Charrier, C. *et al.* Inhibition of SRGAP2 function by its human-specific paralogs induces neoteny during spine maturation. *Cell* **149**, 923-935 (2012). https://doi.org:10.1016/j.cell.2012.03.034
25 Fossati, M. *et al.* SRGAP2 and Its Human-Specific Paralog Co-Regulate the Development of Excitatory and Inhibitory Synapses. *Neuron* **91**, 356-369 (2016). https://doi.org:10.1016/j.neuron.2016.06.013
26 Schmidt, E. R. E., Kupferman, J. V., Stackmann, M. & Polleux, F. The human-specific paralogs SRGAP2B and SRGAP2C differentially modulate SRGAP2A-dependent synaptic development. *Sci Rep* **9**, 18692 (2019). https://doi.org:10.1038/s41598-019-54887-4
27 Schmidt, E. R. E. *et al.* A human-specific modifier of cortical connectivity and circuit function. *Nature* **599**, 640-644 (2021). https://doi.org:10.1038/s41586-021-04039-4
28 Linkert, M. *et al.* Metadata matters: access to image data in the real world. *J Cell Biol* **189**, 777-782 (2010). https://doi.org:10.1083/jcb.201004104
29 Laine, R. F., Arganda-Carreras, I., Henriques, R. & Jacquemet, G. Avoiding a replication crisis in deep-learning-based bioimage analysis. *Nat Methods* **18**, 1136-1144 (2021). https://doi.org:10.1038/s41592-021-01284-3
30 Longair, M. H., Baker, D. A. & Armstrong, J. D. Simple Neurite Tracer: open source software for reconstruction, visualization and analysis of neuronal processes. *Bioinformatics* **27**, 2453-2454 (2011). https://doi.org:10.1093/bioinformatics/btr390

Supplementary figures.
Bernal et al.
Title: RESPAN: an accurate, unbiased and automated pipeline for analysis of dendritic morphology and dendritic spine mapping

Figures S1-2

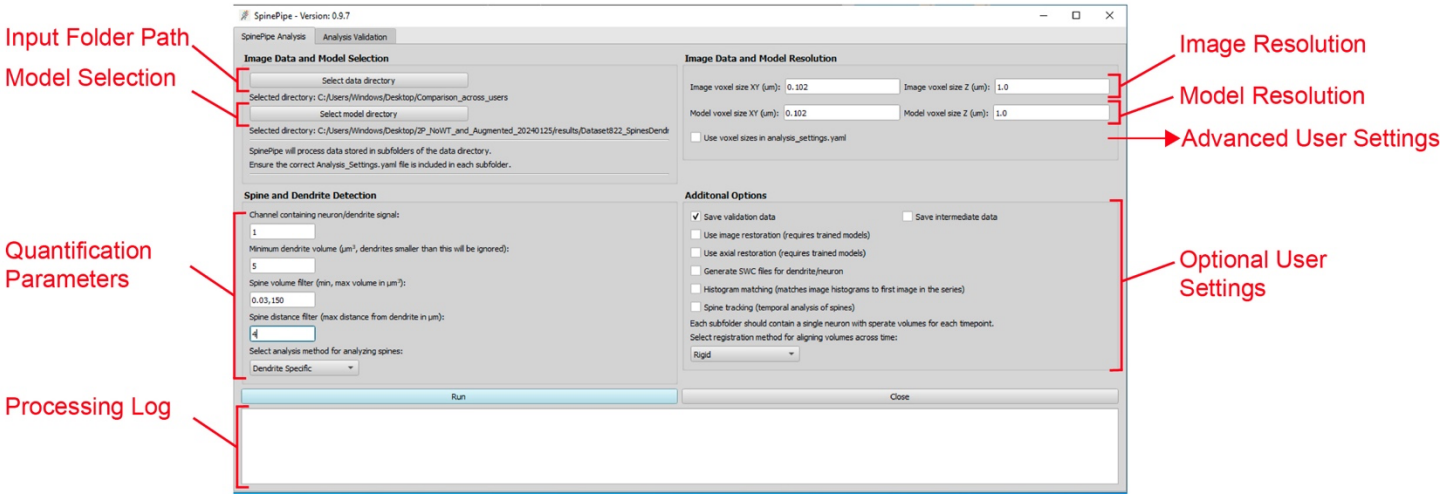

Figure S1. Graphical User Interface of the RESPAN Pipeline.
This figure illustrates the user-friendly graphical user interface (GUI) of the RESPAN pipeline, designed to simplify the process of analyzing dendritic spine data. Key components of the GUI are labeled and include:

- **Input Folder Path:** Allows users to specify the directory containing the 2D/3D .tif image files to be processed.
- **Model Selection:** Allows users to specify the directory containing the appropriate segmentation model.
- **Quantification Parameters:** Users can adjust parameters that define the criteria for spine analysis, such as minimum dendrite volume and spine volume filter.
- **Processing Log:** Displays a real-time log of the processing steps, providing feedback and progress updates.
- **Image Resolution and Model Resolution:** Settings to input the resolution parameters for the images being analyzed and the segmentation model being used.
- **Advanced User Settings:** Offers access to additional configuration options inside the Analysis_Settings.yml file for more precise control over the analysis.
- **Optional User Settings:** Includes extra settings that can be toggled to accommodate specific analysis needs, such as saving intermediate data or using image restoration features. Specific details for each optional setting can be found in the protocol section of the manuscript.

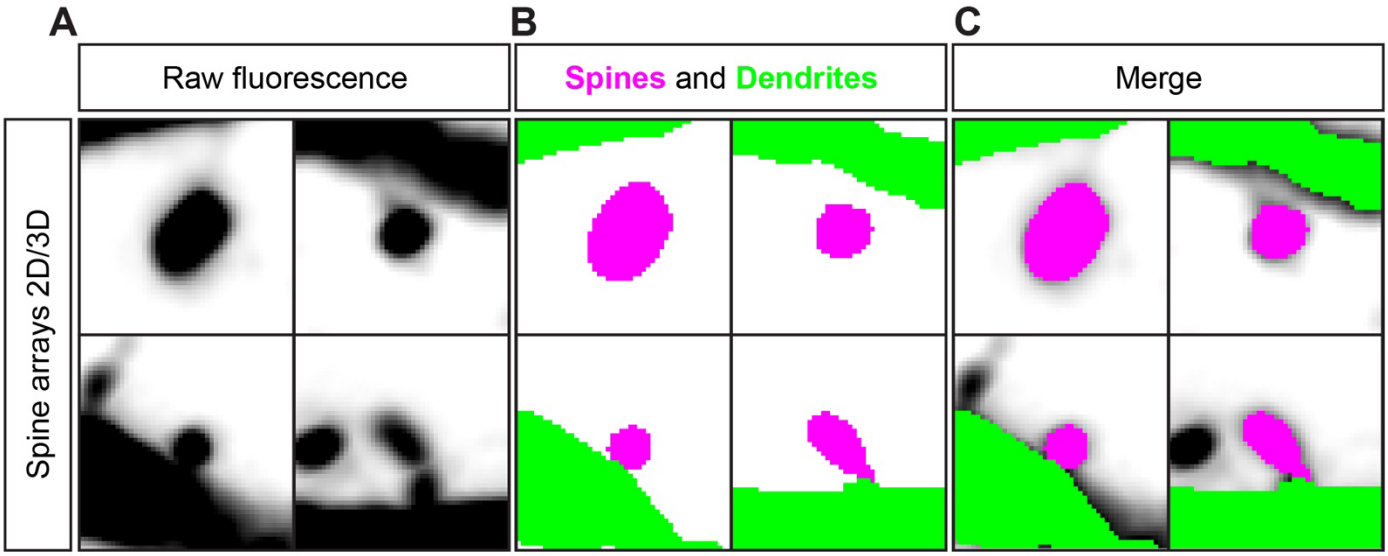

Figure S2. Visualization of Spine Arrays Generated by RESPAN.
This figure showcases the 2D/3D spine arrays that are automatically generated as outputs from the RESPAN pipeline. These allow for segmentation validation and easy visual exploration of all detected spines. The panels illustrate:(**A**) Raw fluorescence image cropped field of views where a spine was detected; (**B**) Isolated spine heads highlighted within the same field, demonstrating the ability of RESPAN to accurately identify and extract individual spine components; (**C**) Segmented dendrites corresponding to the raw fluorescence image, color-coded for clarity and enhanced visualization.