Article

Patterns

A FAIR, open-source virtual reality platform for dendritic spine analysis

Highlights

- We created a dendritic spine morphology analysis tool compatible with DANDI Archives
- Accessible interface features semi-automated segmentation and DataJoint analytics
- Segmentation accuracy exceeded a DIADEM Challenge-winning algorithm
- VR-SASE recreated results from a study of dendritic spines in the injured spinal cord

Authors

Marike L. Reimer, Sierra D. Kauer, Curtis A. Benson, ..., Lakshmi Bangalore, Stephen G. Waxman, Andrew M. Tan

Correspondence

marike.reimer@yale.edu (M.L.R.), andrew.tan@yale.edu (A.M.T.)

In brief

VR-SASE, an open-source virtual reality tool for visualizing and analyzing dendritic spines, integrates with DANDI Archives to ensure compliance with the NIH data sharing mandate. Our platform surpassed the accuracy of gold-standard algorithms, automating calculations of dendritic spine density, length, volume, and surface area. A VR-SASE analysis recreated the results from a study of dendritic spines in the injured spinal cord. VR-SASE enhances neuroanatomical research, facilitating new discoveries in neuroscience.

Reimer et al., 2024, Patterns *5*, 101041
September 13, 2024 Published by Elsevier Inc.
https://doi.org/10.1016/j.patter.2024.101041

CellPress

Patterns

CellPress
OPEN ACCESS

Article

A FAIR, open-source virtual reality platform for dendritic spine analysis

Marike L. Reimer,[1,2,*] Sierra D. Kauer,[1,2] Curtis A. Benson,[1,2] Jared F. King,[1,2] Siraj Patwa,[1,2] Sarah Feng,[1,2] Maile A. Estacion,[1,2] Lakshmi Bangalore,[1,2] Stephen G. Waxman,[1,2] and Andrew M. Tan[1,2,3,*]

[1]Department of Neurology and Center for Neuroscience and Regeneration Research, Yale University School of Medicine, New Haven, CT, USA

[2]Rehabilitation Research Center, US Department of Veterans Affairs, West Haven, CT, USA

[3]Lead contact

*Correspondence: marike.reimer@yale.edu (M.L.R.), andrew.tan@yale.edu (A.M.T.)

https://doi.org/10.1016/j.patter.2024.101041

THE BIGGER PICTURE To know how the nervous system works, it is fundamental to understand the anatomy of neurons. In particular, studying dendritic spines is essential to understand how neurons interact. Dendritic spines are small protrusions along neurons that assist with signal transmission between brain cells. Virtual reality (or VR) imaging tools have allowed detailed study of dendritic spines by enabling 3D visualization of these structures. "VR-SASE" is an open-source VR structural analysis software ecosystem designed for the intuitive analysis of dendritic spines. Enabling accurate and automatic analysis pipelines to study dendritic spines is important to further our understanding of conditions such as spinal cord injuries, which affect dendritic spines' morphological properties.

SUMMARY

Neuroanatomy is fundamental to understanding the nervous system, particularly dendritic spines, which are vital for synaptic transmission and change in response to injury or disease. Advancements in imaging have allowed for detailed three-dimensional (3D) visualization of these structures. However, existing tools for analyzing dendritic spine morphology are limited. To address this, we developed an open-source virtual reality (VR) structural analysis software ecosystem (coined "VR-SASE") that offers a powerful, intuitive approach for analyzing dendritic spines. Our validation process confirmed the method's superior accuracy, outperforming recognized gold-standard neural reconstruction techniques. Importantly, the VR-SASE workflow automatically calculates key morphological metrics, such as dendritic spine length, volume, and surface area, and reliably replicates established datasets from published dendritic spine studies. By integrating the Neurodata Without Borders (NWB) data standard, VR-SASE datasets can be preserved/distributed through DANDI Archives, satisfying the NIH data sharing mandate.

INTRODUCTION

In the centuries following Ramon y Cajal's discovery of dendritic spines, a plethora of methodologies emerged to study them. Dendritic spines are microscopic structures that serve as morphological sites of synaptic contact between neurons in the brain and spinal cord. Importantly, dendritic spine morphology directly influences how electrical inputs through synapses are received, transduced, and ultimately processed by the postsynaptic neuron.[1] As such, dendritic spine structure plays a critical role in supporting synaptic and circuit functions, making them a vital visual proxy for understanding the workings of the nervous system.[2] In pathology, abnormal dendritic spine structures (termed "dendritic spine dysgenesis") are observed in multiple neurological and psychiatric disorders including autism, schizophrenia, Huntington's disease, chronic pain, and Alzheimer's disease.[3,4]

Despite decades of intensive research and development efforts and recent innovations,[5–9] no single solution has demonstrated superiority. By offering a standardized endpoint for dendritic spine studies, DANDI Archives offer a unique opportunity to unify these diverse methodologies. Addressing a critical need to comply with the NIH data sharing mandate, VR-SASE (virtual reality structural analysis software ecosystem) can harmonize these disparate practices in Blender, an open-source three-dimensional (3D) modeling platform. In this study, we utilized VR, Neurodata Without Borders (NWB), Blender, and DataJoint technology to develop a DANDI-compatible dendritic spine morphology analysis platform.[10]

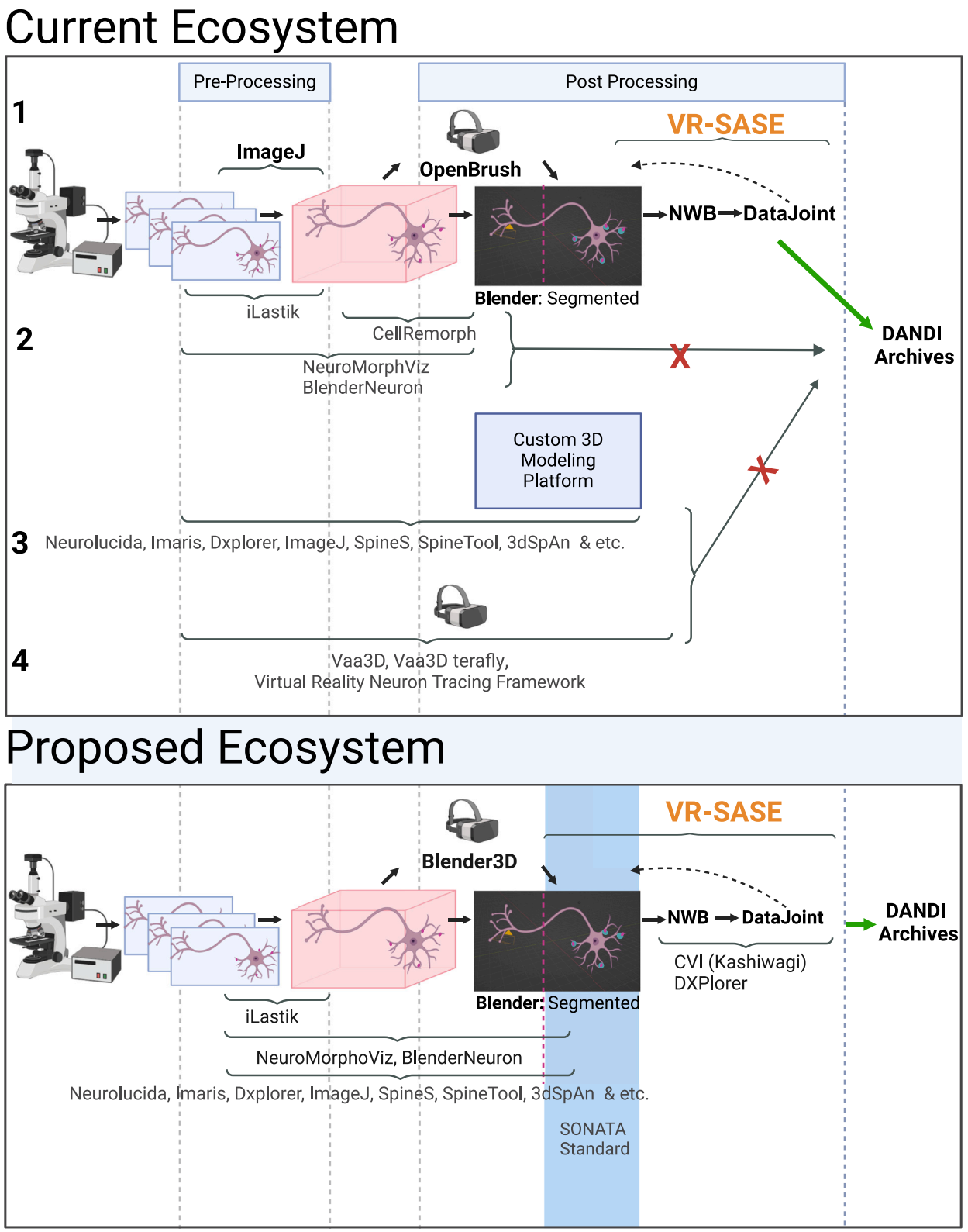

Figure 1. Current and proposed ecosystems

Current ecosystem: the VR-SASE workflow is shown on line 1 with bold text. We used ImageJ for pre-processing, setting image thresholds, and model creation. Post-processing began with segmentation in the Open Brush VR illustration application, where discs were placed to "slice off" dendritic spines. The remainder of segmentation was completed in Blender and automated and analyzed with the VR-SASE add-on. BlenderNeuron, iLastik, and NeuroMorphoViz, shown on line 2,

(legend continued on next page)

Given the significance of dendritic spine morphology in understanding neural circuit function and pathology, it is crucial to unify these diverse methodologies. Here, we developed a VR-SASE, a dendritic spine morphology analysis platform compatible with DANDI Archives and, therefore, the only dendritic spine analysis tool that creates FAIR-compliant data. FAIR data are findable, accessible, interoperable, and reusable and a requirement for researchers who receive funding from the NIH. Data archived/shared on DANDI Archives satisfy the criteria for FAIR compliance.[11] VR-SASE features an accessible user interface (UI), standardized data, automated data segmentation/extraction, advanced analytics, and low cost.

Our workflow from raw imaging data to DANDI Archives is shown in Figure 1, line 1, with bolded text in context with prominent dendritic spine and neuromorphology analysis tools. We used ImageJ for pre-processing, e.g., thresholding and model creation. iLastik,[12] CellRemorph,[13] NeuroMorphViz,[14] and BlenderNeuron,[15] on line 2, may provide superior pre-processing workflows, and we encourage their use; however they do not support DANDI Archives. Likewise, the prominent dendritic spine analysis platforms (line 3) are not DANDI compatible. Neurophysiology platforms incorporating VR are also shown on line 4; however, they support neither dendritic spine analyses nor DANDI Archives.

Our post-processing began in the Open Brush VR environment, where we placed discs on 3D neural models to "slice off" the dendritic spines. The VR-SASE add-on for Blender, an open-source 3D modeling program, automates the remainder of segmentation, performs analyses, and facilitates data management and integration, ultimately creating NWB files, collectively termed the dandiset, when deposited on DANDI Archives.[16] We propose a Blender-centric architecture to leverage the strengths of each platform (Figure 1, proposed ecosystem). Elements common to all studies are shown with bold text.

We assessed the utility of VR-SASE using three approaches: first, we compared VR-SASE results with those obtained using conventional analyses to study dendritic spines in spinal cord injury (SCI) and replicated the original key findings. Second, we benchmarked VR-SASE against the DIADEM (digital reconstruction of axonal and dendritic morphology) challenge.[17] VR-SASE outperformed the DIADEM model in terms of accuracy and may provide better utility than existing methods for analyzing dendritic spines. Although expert segmentation is more efficient, our usability study demonstrated accessibility for new users. By bridging the gap between DANDI Archives and Blender, VR-SASE enabled us to create a FAIR-compliant dataset of dendritic spine morphometries. This promotes FAIR compliance for all 3D dendritic spine morphology analysis platforms/methodologies, fostering collaboration across disciplines and community-led development.

RESULTS

Neurolucida, ImageJ, and VR-SASE comparison highlights advances in visualization, accuracy, and precision

Figure 2 provides a visual comparison of conventional dendritic spine analysis platforms, Neurolucida and ImageJ, and VR-SASE. A representative neuron traced in Neurolucida is depicted in Figure 2A, and an enlargement shows individual spines marked with thin red lines in Figure 2B. Figure 2C shows the same spines visualized in VR-SASE. Figure 2D shows the leftmost spine from Figure 2B in ImageJ, maximally enlarged and processed to enhance visibility. Red lines, placed by four experts, capture the length of the spine, demonstrating the inherent variability of manual measurements. In contrast, the same spine is shown from the VR-SASE workflow (Figure 2E), with greatly enhanced resolution. The spine tip is clearly marked by an orange sphere (indicated by an arrow), minimizing ambiguity in length measurements. Overall, VR-SASE offers superior visualization with increased accuracy and precision.

To demonstrate the usability of VR-SASE, we taught users how to interact with the Open Brush UI. After successful completion of our VR training protocol (see experimental procedures), videos were filmed showing trainees segmenting a representative dendrite section, as can be seen in Figures 2F and 2G. The dendrite shown in Figure 2F was segmented in Open Brush by an expert, and a trainee segmented the dendrite shown in Figure 2G. Differences in detected segments between expert versus trainee are marked with orange asterisks, and the spine indicated in Figure 2G is below our size threshold and would be removed from the dataset by our DataJoint filtering criteria, enhancing accuracy and reducing variability between researchers. The duration of the recorded segmentation shows the effect of training on VR-SASE's segmentation time. While trainees can rapidly acquire VR segmentation skills, VR-SASE post-processing is more efficient with experience. "Grabbing" the "slicer" disc with a VR controller was the greatest challenge for trainees. The increased VR-SASE segmentation time arises from imperfectly sliced dendritic spines, requiring manual corrections.

SCI increased abnormal dendritic spine density in the proximal region

To validate VR-SASE, we analyzed a subset of spinal cord neuronal data[18] presented by Dr. Kauer et al. at the 2023 Society for Neuroscience poster presentation ("Pak1 inhibition with romidepsin attenuates H-reflex excitability after SCI") and at the 2022 Paralyzed Veterans of America in Dallas, Texas, which is currently in press at *Journal of Neurophysiology*.[19]

In this study, blinded investigators performed a Sholl analysis with Neurolucida to examine changes in the distribution of dendritic spines in three regions around the neuron's soma: proximal (0–30 μm), medial (30–60 μm), and distal (60–90 μm). Dendritic

are promising alternatives to ImageJ pre-processing. Numerous dendritic spine analysis platforms, shown on line 3, have custom 3D modeling platforms; however, none are compatible with DANDI Archives. Line 4 shows neural analysis platforms that utilize VR; however, these platforms are not designed for dendritic spine analyses. Proposed ecosystem: Blender and the SONOTA data standard have the potential to unite these diverse workflows. If dendritic spine analysis platforms made their segmented models available to/compatible with Blender, then the VR-SASE Blender add-on would bridge the gap to DANDI Archives, promoting FAIR data throughout neurophysiology.

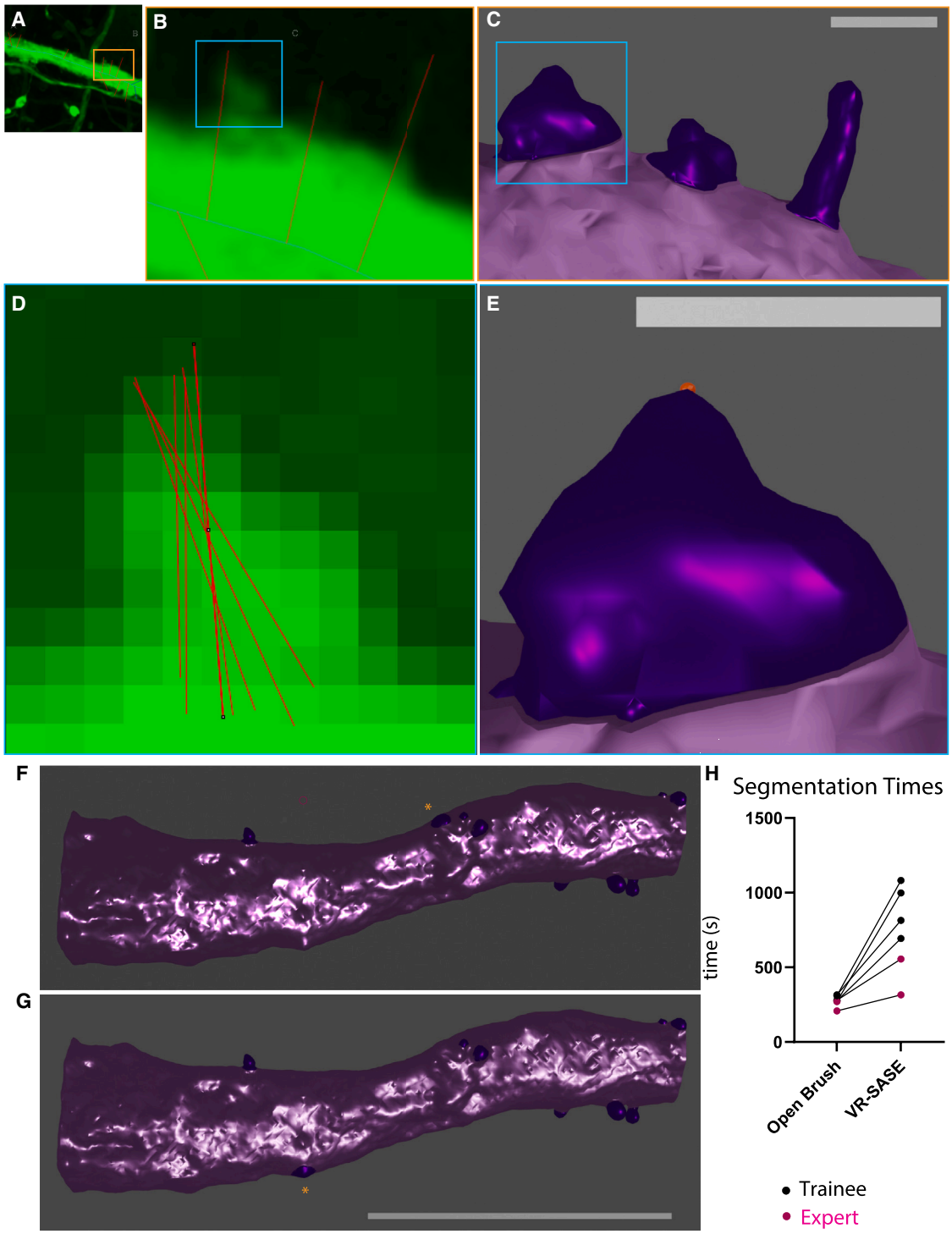

Figure 2. Neurolucida, ImageJ, and VR-SASE comparison highlights advances in visualization, accuracy, and precision

(A) A neuron traced in Neurolucida.

(B) An enlargement showing spines as an overlay marked with thin red lines denoting spines.

(C) The spines in (B) in VR-SASE with much greater clarity and detail.

(D) The leftmost spine maximally enlarged. Four expert analysts placed red lines denoting length, indicating the high degree of variability inherent in manual measurements.

(E) An enlargement of the leftmost spine shown in (B) and (D). The spine tip (arrow) is marked with a sphere to aid in visualization, highlighting how VR-SASE decreases ambiguity in length measurements. Scale bar, 1 μm.

(legend continued on next page)

spines were traced on sequential images spanning each dendrite using Neurolucida's "stack scrolling" interface. This method requires switching between images to mark the endpoints of the spine.

Using Neurolucida, the SCI group had a higher proximal total spine density (unpaired t test: sham [M = 0.1513] and injured [M = 0.3734]; $p = 0.0058$, t = 3.041, df = 23, SEM = 0.07303; Figure 3A), thin spine density (unpaired t test: sham [M = 0.1321] and injured [M = 0.2617]; $p = 0.0322$, t = 2.280, df = 23, SEM = 0.05681; Figure 3B), and mushroom spine density (Mann-Whitney test: sham [M = 0, $n = 9$] and injured [0.06399, $n = 16$]; $p = 0.0356$, U = 35.50 ; Figure 3C).

DataJoint's powerful querying/filtering tools enhance VR-SASE by excluding dendritic spines that do not meet published morphological parameters. Upper and lower bounds for dendritic spine volume were established as 0.01 μm for small thin spines to 0.8 μm[20,21] for large mushroom spines. 3 μm is established as the maximum length for dendritic spines on motor neurons, so we used this criterion.[22,23] Peters and Kaiserman-Abramof established a 0.5 μm minimum length for cortical spines[24]; however, Harris found dendritic spines as short as 0.2 μm in the hippocampal region.[20,21] There is currently no consensus regarding the minimum length of dendritic spines on motor neurons, so we used Harris's 0.2 μm due to the abundance of very small protrusions.

The resulting data subset[16] contained dendritic spines whose volume was between 0.01 and 0.8 μm^3 and whose length ranged between 0.2 and 3 μm. DataJoint refined our pool of 858 potential dendritic spines to 636 dendritic spines. 94 potential spines were excluded for failing to meet minimum size criteria, and 24 were excluded for exceeding it (supplemental information).

Moving freely in a VR space, VR-SASE enabled users to visualize and segment neuroanatomy from any perspective. Using our pipeline, we analyzed images from the Kauer et al. dataset.[19] Our spine density findings (Figures 3D–3F) are largely in line with their results. We focused our analysis on the proximal region and, likewise, found an increase in total spine density (Mann-Whitney test: sham [M = 0.09019, $n = 9$] and injured [M = 0.3318, $n = 20$]; $p = 0.0017$, U = 26; Figure 3D). We also observed the same trend for thin spine density (Mann-Whitney test: sham [M = 0.09019, $n = 9$] and injured [0.09019, $n = 9$]; $p = 0.0043$, U = 31; Figure 3E). We found mushroom spine density that approached significance (Mann-Whitney test: sham [M = 0, $n = 9$] and injured [M = 0, $n = 20$]; $p = 0.0536$, U = 54; Figure 3F). These findings are in line with previous results showing that SCI induces an abnormal increase in dendritic spine density.[2,25–27]

SCI induces abnormal dendritic spine morphology

Changes in the size and shape of dendritic spines can alter how signals are transmitted and how efficiently synapses function, impacting the physiology of neurons.[2] Following precedent set by our group,[19,25] we measured spine length manually using ImageJ, tabulating values in Excel. These findings showed that SCI led to an increase in the total length of dendritic spines, compared to the sham, driven largely by an increase in thin spine length (unpaired t test: sham [M = 0.4020] and injured [M = 0.5647]; $p < 0.0001$, t = 4.893, df = 21, SEM = 0.3325; Figure 3G). This is in agreement with our published work[25] and is a general trend in neurons after SCI.[27,28] We did not find that the length of dendritic spines increased following SCI (Mann-Whitney test: sham [M = 0.4283, $n = 9$] and injured [M = 0.4206, $n = 20$]; $p = 0.5315$, U = 76; Figure 3H).

We next sought to leverage the additional measures provided by VR-SASE to quantify how contusion injury induces abnormal morphological changes in dendritic spines. We did not find any significant differences between groups for thin spine surface area (Figure 3I). However, we found that thin spine volumes were smaller in the injured animals (Mann-Whitney test: sham [M = 0.06853, $n = 94$] and injured [M = 0.04969, $n = 526$]; $p = 0.0131$, U = 20,759; Figure 3J). We also found a significant difference between the volume-to-surface area ratio between injured and uninjured animals. The injured group had a significantly lower ratio (Mann-Whitney test: sham [M = 0.08784, $n = 94$] and injured [M = 0.07661, $n = 526$]; $p = 0.0012$, U = 19,562; Figure 3K). We removed all spines less than 0.1 μm^3 and found that this effect was eliminated (data not shown). By restricting our analysis to spines with volume less than 0.1 μm^3, we confirmed that this effect was due to the tiny spines (Mann-Whitney test: sham [M = 0.07719, $n = 61$] and injured [M = 0.06928, $n = 409$]; $p = 0.0160$, U = 10,096; Figure 3L).

The time required to generate these analyses was substantially decreased when compared to the same analyses carried out in Neurolucida and ImageJ. Positioning the disc slicers on VR neurons took an average of 17 min per neuron (supplemental information), representing a substantial decrease in researcher effort. By augmenting length and density measures with surface area and volume, VR-SASE enhances our understanding of how structural changes in dendritic spines contribute to pathological states. By providing these measures efficiently, VR-SASE accelerates and expands our research capabilities.

VR-SASE validated with the DIADEM gold standard

The DIADEM challenge was a widely lauded effort consisting of elite research groups pitting automated neural reconstruction algorithms against each other, generating neural models long considered to be the gold standard.[29] To assess the accuracy of our VR segmentation, we reconstructed a drosophila olfactory neuron, OP-09, from the DIADEM final round.[18] We then compared our reconstructed neuron (Figure 4B) with the gold-standard reconstruction (Figure 4C) by super-imposing them on the maximum projection of OP-09 (Figure 4A). Figure 4D shows our reconstruction of OP-09 on top of the maximum projection. Deviations from the true morphology are visible as green around the edges. The gold-standard reconstruction (Figure 4E) has 58% more green visible, indicating that our tracing is more accurate (supplemental information). Our segmentation of OP-09 is shown in Figure 4F. Figure 4G is an enlargement showing the placement of the teal slicer discs. VR-SASE used the teal discs to slice the complex neuron into its component pieces. Each section was assigned a random color, highlighting segmentation accuracy, as well as Blender's utility in visualizing information.

(F) A segmented dendrite created by a trainee from our usability study. Scale bar, 10 μm.
(G) The same dendrite as in (F) from our VR-SASE expert.
(H) While users can quickly acquire basic Open Brush skills, segmenting with VR-SASE workflows takes longer when users are less experienced.

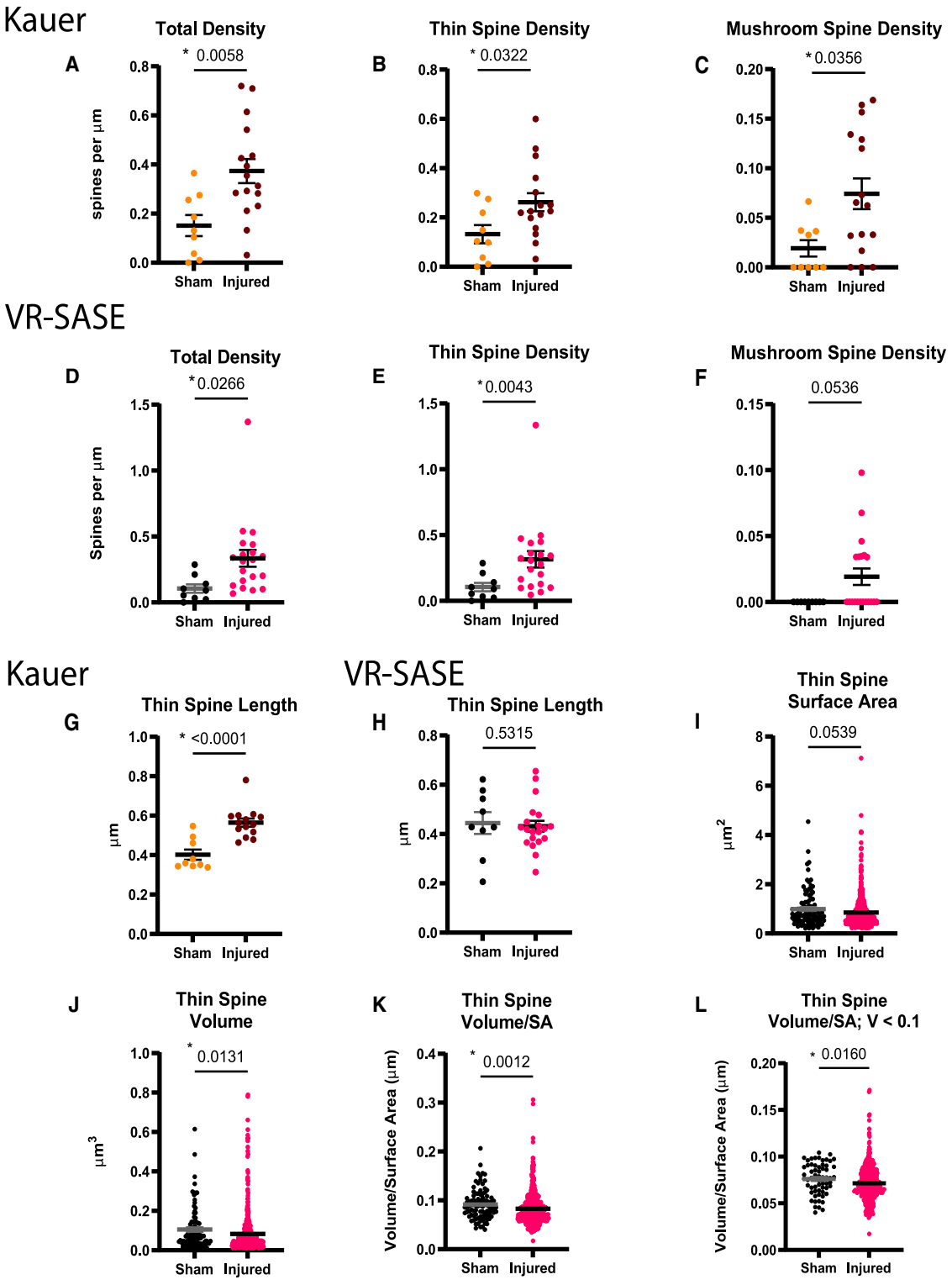

Figure 3. SCI increases proximal dendritic spine density and abnormal morphology

(A–C) VR-SASE recreated the Neurolucida-traced findings of Kauer et al.[19] Their analysis showed that compared to sham, the SCI group had a significantly higher proximal total spine density (A), thin spine density (B), and mushroom spine density (C).

(D–F) We likewise found in the SCI group significantly higher proximal total spine density (D) and thin spine density (E), while mushroom spine density approached significance, as shown in (F).

(legend continued on next page)

We examined the model in Blender using its native 3D-Printing add-on and found the volume to be 1,216.1861 μm^3 and the surface area to be 3,061.1194 μm^2. The time to position the teal discs slicers was just under 15 min. The benchmark for manual reconstructions of simple neurons is approximately 20 min, although complex neurons, such as OP-09, take longer.[17]

By reconstructing OP-09, a gold-standard neuron with VR-SASE, we demonstrated that the accuracy of VR-SASE surpassed the gold standard by 58% (supplemental information). Additionally, VR-SASE provided additional meta-information about the neuron in its ability to generate insights with volume and surface area data.

Advanced analytics with VR-SASE

To advance the goal of uniting strengths of diverse dendritic spine analysis methodologies, we recreated those pioneered by Pchitskaya et al.[7] and Kashiwagi et al.[30] Pchitskaya et al. employed a method to aid in dendritic spine classification by creating a cord length distribution histogram (CLDH).[7] This technique measures the internal cords of dendritic spines, creating a histogram that can be used to distinguish between dendritic spine clusters. Figure 5A shows a dendritic spine. In Figure 5B, the surface of the spine has been removed to show its internal cords. Pchitskaya et al. generated their cords randomly, which could lead to under-sampled regions. To ensure complete coverage, we created cords at every vertex in the model. We excluded nearest-neighbor vertices and sampled one in every ten of their possible connections, generating a rich collection of cords. Figure 5C depicts the CLDH for the dendritic spine. This underscores how data standardization enabled VR-SASE to employ analysis techniques.

Next, we sought to expand VR-SASE's capabilities by calculating the convex volume index (CVI) for a dendritic spine. Figure 5D shows the dendritic spine from Figure 5A with its head surrounded by a convex hull (gold mesh), recreating the method used by Kashiwagi et al.[30] Figure 5E shows this method used to quantify the concavity of the entire dendritic spine. Applying this measure of dendritic spine concavity to DataJoint queries, as we did with dendritic spine length and volume, can foster innovative dendritic spine clustering techniques.

DISCUSSION

Dendritic spine morphology dynamically changes in pathological conditions. Researchers need effective tools to study these changes; however, it is imperative for all dendritic spine analysis platforms to create FAIR data. To meet these needs, studies conducted using VR-SASE can be preserved and shared via the free online repository DANDI Archives.[16] We found that VR-SASE enhanced visualization, providing accessible controls, semi-automated segmentation, and sophisticated DataJoint analyses. The VR-SASE Blender add-on automated key morphological measures, including dendritic spine length, volume, and surface area. These parameters play crucial roles in understanding electrical interactions,[31,32] synaptic strength,[33] and calcium dynamics.[34,35] Automated collection of morphological data empowers VR-SASE users to interpret their anatomical data in less time with less subjective error risk.[36,37] Integrating NWB and DataJoint enhances adherence to FAIR principles,[11] improves data interoperability, streamlines data organization, ensures dataset completeness, and minimizes human errors.[36]

We applied VR-SASE to investigate dendritic spine changes following SCI, demonstrating its accuracy and utility by recreating Kauer et al.'s findings[19] that SCI causes abnormal dendritic spine density. We extended their findings by showing that SCI causes abnormally decreased thin spine volumes with significantly smaller volume-to-surface area ratios.[34,35,38,39] In this vein, while our 0.2 μm minimum length criteria include fine structural resolution, these may not fall within identity criteria for dendritic spines in other tissue types or analytical approaches.[24] Nonetheless, our finding of decreased volume-to-surface area ratio in tiny spines of injured mice demonstrates VR-SASE's ability to produce this important metric.

We further validated the accuracy of our VR-SASE method by reconstructing a *Drosophila* olfactory neuron from the DIADEM challenge, finding our results in closer agreement with the gold-standard reconstruction. Our reconstruction depicted the neuroanatomy more accurately, demonstrating the effectiveness of VR-SASE in creating extremely accurate models.

Blender is an optimal platform to integrate diverse modalities, as it is already widely used in academic research. A recent Google Scholar query for "Blender Model" returned more than 200,000 results from diverse fields. The use of VR is making substantial inroads in neuroanatomical studies[40–43] and will continue to evolve. Blender is amenable to machine learning approaches, such as PointNet, which would further expedite analyses.[44] Blender's versatile Boolean modifiers can segment virtually any 3D shape, making VR-SASE an excellent hub for training AI on 3D models throughout biology.

Our proposed Blender-centric architecture would unite the current dendritic spine ecosystem (Figure 1). BlenderNeuron, NeuroMorphoViz, and iLastik pre-processing workflows may expedite pre-processing. NeuroMorphoViz currently integrates with Blender and is developing dendritic spine segmentation tools (https://github.com/BlueBrain/NeuroMorphoVis). Basu et al. developed a machine learning algorithm automating segmentation; however, although this system leverages 3dSpAn's pioneering automated dendritic spine segmentation algorithm, it requires manual corrections via their 2D custom UI.[9] Using 3dSpAn's pioneering automated dendritic spine segmentation algorithm but post-processing 3dSpAn models in Blender/VR-SASE may prove more effective than either methodology alone. Similarly, combining VR-SASE DataJoint queries with the automated dendritic spine population sorting algorithms featured

(G and H) Kauer et al. found that SCI mice had significantly longer thin dendritic spines than sham (G) using ImageJ[19]; however, VR-SASE did not show a significant difference between groups (H).

(I and J) Using the additional metrics provided by VR-SASE, we assessed thin spine surface area (I) and volume (J). No differences were found between groups for surface area; however, injured mice had dendritic spines with less volume.

(K and L) We assessed the volume-to-surface area ratio of thin spines and found that the injured group had decreased values (K). This was not true of spines greater than 0.1 $\mu m3$ but was observed in dendritic spines smaller than 0.1 $\mu m3$ (L).

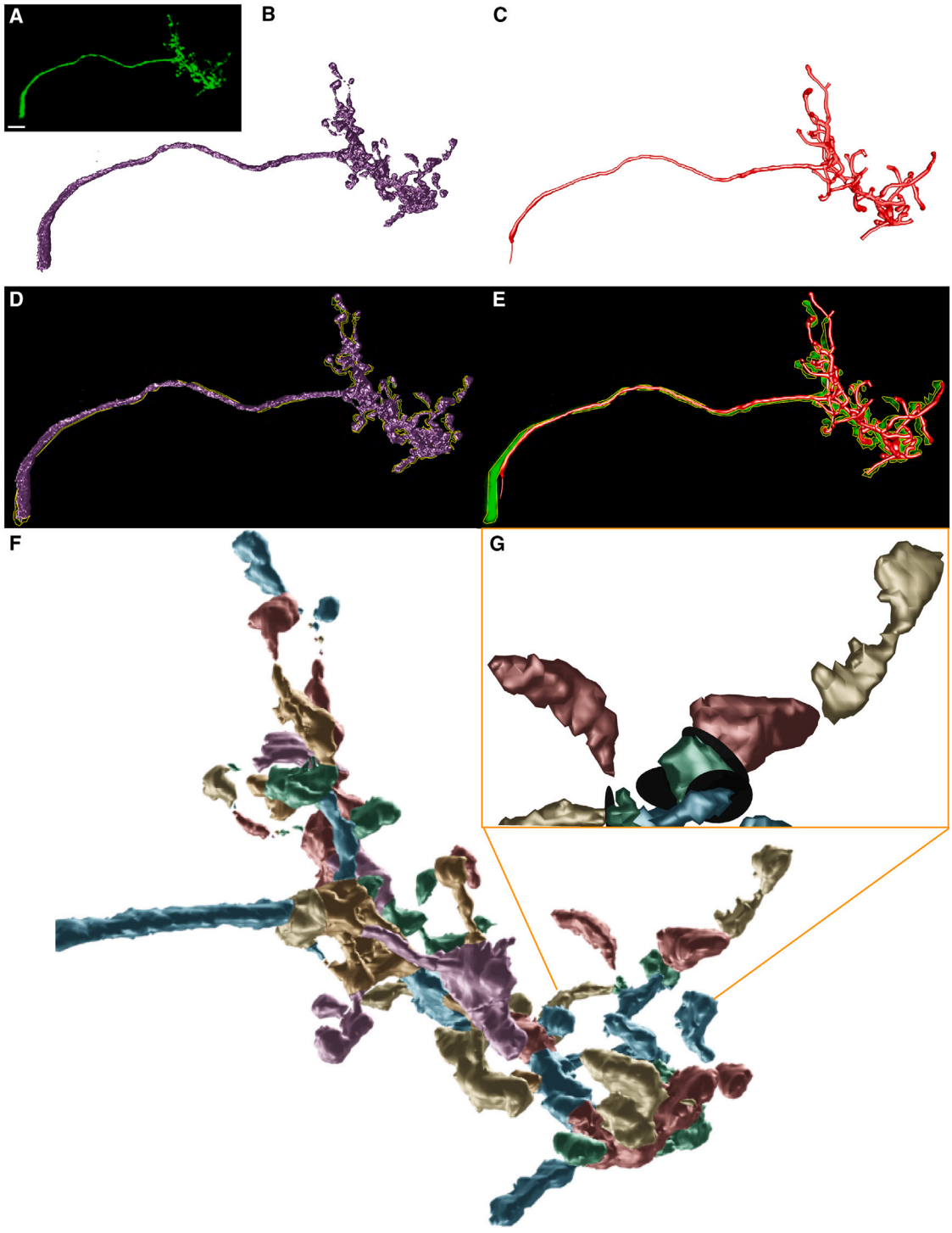

Figure 4. VR-SASE validated with the DIADEM gold standard

(A) OP-09, a neuron from the DIADEM challenge. Scale bar, 10 μm.

(B and C) VR-SASE reconstruction (B) and gold-standard reconstruction (C).

(D and E) We super-imposed the reconstructions, respectively, on the maximum projection (A). We quantified the green regions and found the gold-standard reconstruction had 58% more green visible, indicating that VR-SASE depicted anatomy more accurately.

(F) Results of segmentation indicated with different colors.

(G) Expanded view showing an enlargement demonstrating how black slicer discs were placed to segment the neuron.

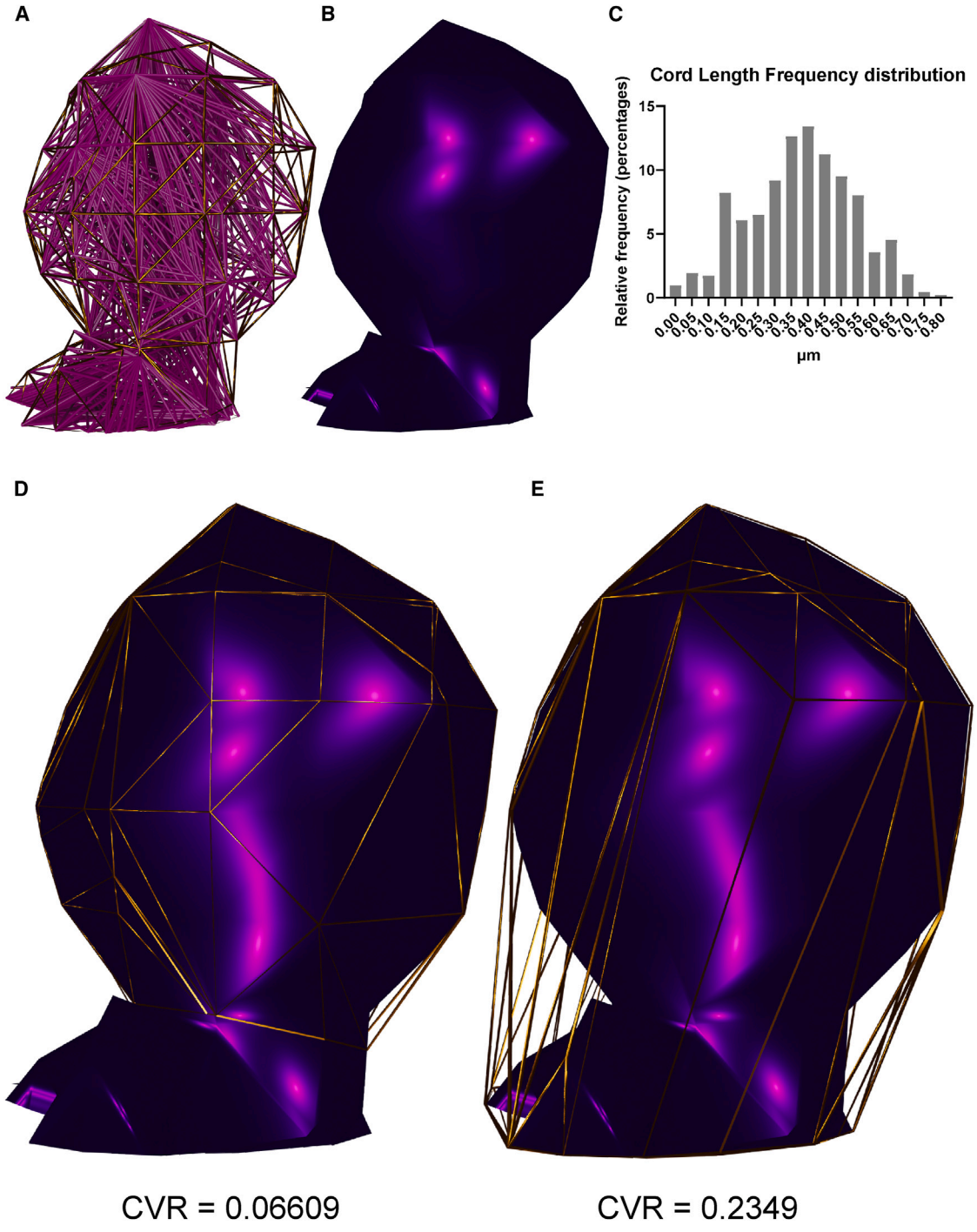

Figure 5. VR-SASE post hoc analyses demonstrate advanced analytic capabilities

(A) A dendritic spine.

(B) Interior of the spine showing cords created with a post-hoc analysis in VR-SASE.

(C) The CLDH, a distribution of their lengths. Ptichkaya et al. demonstrated that this technique improves classification accuracy.[8] Their proposed method randomly creates cords in the interior of the spine, which introduces inaccuracies by over- and under-sampling regions. We mitigate these errors by creating cords at each vertex in the 3D model of the spine.

(D and E) Our post-hoc analysis for concave volume index, the difference in volume between a spine head and its convex hull, which is a shell around the spine without concave surfaces. This method was pioneered by Kashiwagi et al. to assess dendritic spine head morphology.[30] Given that the strict classification of dendritic spines relies on the presence of an indent (concave surface),[39] this measure has utility in dendritic spine clusterization and could be augmented still further with DataJoint advanced analytics.

by DXPlorer could greatly enhance both. Moreover, DXPlorer provides metrics for dendritic spine necks, morphology critical to pain studies.

Our proposed framework replaces Open Brush, the VR illustration application we used, with Blender 3D, Blender's dedicated VR environment (Figure 1). VR-SASE workflows are compatible with Blender 3D or any other VR application featuring 3D model manipulation (e.g., scale, rotate, move) and import/export. Moreover, Blender3D may enable pre- and post-processing completely within its virtual environment.

We took preliminary steps toward this proposed framework by conducting two post hoc analyses. First, by creating a CLDH with improved cord distribution, we expanded on the methodology proposed by Pchitskaya et al.[7] Second, we calculated the CVI for a dendritic spine head using the method published by Kashiwagi,[30] which we applied the entire dendritic spine. Augmented with DataJoint's analytics, this measure of dendritic spine concavity could refine collections of dendritic spines in the same manner our study assessed only spines satisfying established length and volume parameters. Furthermore, our CVI data also demonstrated that VR-SASE can segment dendritic spine necks, a step toward quantifying this vital compartmental morphology.[32]

Beyond the focus on dendritic spines, VR-SASE has the potential for diverse biomedical research applications. For example, the platform could analyze axonal morphology, neuronal circuitry, or even organ reconstructions through the body. Compatibility with electron microscopy workflows makes VR-SASE a broadly applicable dendritic spine analysis platform and facilitates the interchange of 3D formats via FAIR repositories, such as NeuroML-DB.[15] An inherent strength of VR-SASE lies in its standardized data, enabling integration with existing tools. We implemented advanced DataJoint functions in a Jupyter notebook, so our advanced DataJoint analytics can be recreated from this file. DANDI Hub and Jupyter Lab facilitate Jupyter Notebook integrations with other programming languages, including Julia, R, and MATLAB. This interoperability across diverse programming languages contributes to a richer understanding of dendritic spine dysgenesis.

In conclusion, VR-SASE marks a substantial advancement in neuroanatomical research. We created a dendritic spine morphology dandiset and a pipeline enabling access to the FAIR-compliant DANDI Archives for diverse 3D methodologies.

Limitations

Our study focused on dendritic spine density, volume, and surface area; however, branch order diagrams are a highly utilized tool for studying dendrite morphology.[23] This could be addressed in future work bringing VR-SASE into compliance with the SONOTA data standard. Depicted by the blue bar in Figure 1, all data concerning the relationship between NeuroML-DB dendrite branches and nodes should comply with the SONOTA data standard.[45] Because VR-SASE is a functioning pipeline to DANDI Archives, this work represents significant progress toward uniting these diverse methodologies.

The VR-SASE workflow should be supplemented with additional pre- and post-processing methods. Our 3D model creation protocol, meant to provide the simplest solution, requires greater refinement. Threshold settings impact the presence and connection of dendritic spines. ImageJ's gamma correction enhances the contrast between dendritic spines and the background, which could mitigate these issues.[46] We encourage readers to try the BlenderNeuron[15] and iLastik[12] pre-processing workflows as alternatives to our ImageJ-based method.

Our "catch-all" classification for thin spines contains 25 disconnected spines. This likely includes mushroom spines disconnected due to their faint necks. An additional post-processing review may resolve the small discrepancy in mushroom spine density compared to Kauer et al.[19] We did not find that SCI causes an increase in spine length, as Kauer and others have.[2,4] The length measurement variability within and between the VR-SASE and ImageJ and variations arising from stubby spine morphology are recognized challenges in the field[47,48] (see Figure S3). Moreover, a meta-analysis found an en passant tendril miscategorized as a thin spine in the sham group. Creating a dendritic spine morphometry dandiset (DANDI Archives dataset) was our foremost objective, and we recognize that our pre- and post-processing workflows would benefit from additional reviews and refinement.

To optimize meta-data management, VR-SASE could be extended to be compatible with the Cedar Metadata Workbench.[34] The pandas Python library could expedite analyses, as could automating the Sholl analysis. Manual post-processing is occasionally necessary, prompting our creation of a dedicated "manual segmentation" tool. While researcher involvement remains crucial, Blender's hotkeys accelerate quality control.

The open-source nature of VR-SASE promotes collaboration and community-driven development. Researchers are encouraged to contribute to the platform's improvement by sharing code, proposing features, and reporting issues through the GitHub repository. This collective effort fosters innovation, accelerating the advancement of dendritic spine analysis techniques. By mitigating the analysis bottleneck, VR-SASE plays a pivotal role in reducing the time required for developing new therapies and treatments. In conclusion, VR-SASE fills a critical gap in dendritic spine analysis by integrating VR technology into a standardized, open-source software application, marking a significant step forward in cell biology methods.

EXPERIMENTAL PROCEDURES

Resource availability

Lead contact

Further information and requests for resources and reagents should be directed to and will be fulfilled by the lead contact, Andrew M. Tan (andrew.tan@yale.edu).

Materials availability

This study did not generate new unique reagents.

Data and code availability

- Processed dendritic spine morphology data in NWB format are available at DANDI Archives: https://doi.org/10.48324/dandi.000723/0.240716.1414.[16]
- Images, Blender files, NWB files, and analysis products (Table 1) are available on Data Dryad: https://doi.org/10.5061/dryad.w3r2280z0.
- Blender VR-SASE add-on code and the Jupyter Computational Notebook with DataJoint queries are available in Zenodo: https://doi.org/10.5281/zenodo.10607576.[10]
- All other data supporting this paper will be shared by the lead contact upon request.

Table 1. Resources

Resource	Source	Identifier
ImageJ	https://imagej.net/ij/download.html	RRID: SCR_003070
Blender	https://www.blender.org/download/	RRID: SCR_008606
Neurolucida 360	MBF Bioscience	RRID: SCR_016788
Oculus Rift S	Meta Platforms	–
Oculus Quest 2	Meta Platforms	–
Open Brush	https://openbrush.app/	–
Blender - Python module installer	https://cgfigures.gumroad.com/l/pymodinstall	–
PyNWB	https://pynwb.readthedocs.io/en/stable/	RRID: SCR_017452
DataJoint	https://datajoint.com/	RRID: SCR_014543
Neuroinformatics Morphology Viewer	https://neuroinformatics.nl/HBP/morphology-viewer/#	–

Image analysis workflow

Neuronal images were collected from mice that received either a contusion injury or a sham injury.[19] To identify α-motor neurons, we employed a screening process based on data from previous studies.[25,39] These neurons, located in the ventral horn Rexed lamina IX, met the criteria of having a soma diameter larger than 25 μm and a cell body cross-sectional area greater than 450 μm^2. For further analysis, we included only YFP+ α-motor neurons with visible cell bodies and dendritic branches at least 45μm in length. 3D reconstructions of motor neurons were analyzed to examine dendritic spine density and distribution. Thin, stubby, and mushroom spines on each dendritic branch were identified and marked by placing a disc to slice or segment them apart from the dendrite. Putative mushroom spines were assessed from orthogonal perspectives and classified as mushroom spines if their head was wider than their neck was long, which is an established classification.[4,25,28,39] Ambiguous cases were handled by comparing the virtual cursor with the length of the dendritic spine neck and the width of the dendritic spine head. As with all light microscopy workflows, the resolution of dendritic spine necks can cause mushroom spines to be classified as stubby spines.[48] The stubby spines 0.2 μm in length originally described by Peters and Kaiserman-Abramof Abramov[24] are representative of the morphology we typically observed for stubby spines.

To measure subtle alterations in dendritic spine densities, we pooled all non-mushroom spines into the category "thin."[4,25,28,39,49] Limiting our classification to "thin" and "mushroom" enabled us to measure subtle variations in spine distribution. Dim neurons and those with pronounced YFP puncta were excluded from the study, as they produce digital models with high noise-to-signal ratios, obscuring potential dendritic spines.[50]

ImageJ

We used v.1.535f1 of NIH ImageJ (https://imagej.nih.gov/ij/download.html) to process images and convert them to OBJ format, which can be opened in Open Brush (https://openbrush.app/). Note that ImageJ functionality recently changed, and OBJ files must now be exported through the 3D Viewer Plugin.

To ensure high-fidelity conversion, clear separation between dendritic spines and the background was necessary. To achieve this, ImageJ's autocorrect brightness and contrast function was applied at the center of the image stack. A border of each image was selected and filled using the rectangular selection tool, creating a bar of known dimensions to reconstruct the scale. The image type was set to 8 bit and then saved as a PGM file before "saving as" in OBJ format. The resampling factor was set to 1 to prevent smoothing. Multiple threshold settings were tested for each image stack, and a region was compared with the original image stack to ensure the presence of all dendritic spines in the region.

Segmentation with open brush

Open Brush, a room-scale VR 3D painting platform was originally developed by Google under the name Tilt Brush but was renamed Open Brush after they released it as open source. Open Brush is available through multiple platforms, depending on VR hardware requirements. We used two VR headsets to demonstrate platform independence (see hardware descriptions). We installed Open Brush on the Oculus Quest 2 VR headset, produced by Meta Platforms, using the Meta Store interface. This interface is not available on the Oculus Rift S, so Open Brush software was downloaded through Steam, a digital distribution service.

The OBJ files created by ImageJ were placed into Open Brush's media library, allowing import into the virtual environment, where a blinded researcher scaled them to the size of the room. The neural model was then "pinned" to ensure that the slicer discs remained attached to the model. Slicers are flattened cylinders created in Blender and imported from Open Brush's media library. They were resized and positioned to create a clean interface for when they are subtracted from the 3D mesh of the neuronal reconstruction at a later stage.

For tracing procedures, the main light source within the VR environment was positioned between the user and the neuron. This control over lighting provided clear visibility, good contrast resolution on the digitized neuronal tissue model, and consistency between models. Dendritic spines were segmented on one side of the dendrite, and then the user moved to the other side of the neuron, adjusting the light to shine directly on the neuron (as above to ensure clear visualization of the neuronal tissue model).

Segmentation with Blender add-on

We developed a custom add-on for Blender, an open-source 3D modeling suite, using their application programming interface for the Python programming language (see Figure S1). We used the CGFigures add-on to install the PyNWB and DataJoint libraries to enable NWB file creation and database integration, respectively. Writing images to NWB files requires the Pillow library, which we also installed in this manner.

Open Brush allowed positioning slicer discs on a neuronal model, which can be exported as FBX files. These were imported into Blender. The models and discs were then scaled to their original dimensions and aligned with Blender's coordinate system. Slicers give their names to the dendritic spines they segment. This standardizes naming conventions without restricting the scope of possible analyses.

To ensure data integrity, the original OBJ file of the neuronal model was re-imported and the model from the FBX file discarded, retaining only the correctly positioned slicer discs from VR segmentation.

Segmentation with the VR-SASE Blender add-on begins with copying all the slicers and joining them into a single mesh with Blender's "join" functionality. The merged slicers are then applied to the neural model as the object of a "Boolean difference" modifier operation. This removes the portions of the neural model connecting the dendritic spines to the dendrite. The "separate meshes" button on the VR-SASE segmentation tools panel creates individual meshes of each spine.

The separated dendritic spines must have their origin updated through Blender's graphical UI (GUI). Then, the slicer discs must be placed in a collection. To automatically segment the dendritic spines, they were selected along with the collection containing the slicers.

The "segment solid spines" button placed each spine into its own collection, naming both meshes after the slicer. VR-SASE created meshes containing the base and tip of the spine, taking its name, and prepending endpoints. To improve visualization and assist with quality control, an empty mesh was placed at the spine, distinguishing them from manually segmented dendritic spines.

To obtain surface area metrics, a copy of the original dendrite was created and made hollow with the "solidify" modifier. Its dendritic spines were removed as before and selected as before. Next, the "segment hollow spines" button was pressed, adding the surfaces to the collection with their corresponding solid spine, prepending "surface" to their name.

Manual segmentation

We created a tool to address cases when the VR-SASE Blender add-on was unable to segment dendritic spines. To use manual segmentation, Blender's mode was changed to edit mode and the spine tip was selected by clicking its vertex or face. Pressing the manual segmentation button places the dendritic spine mesh into a collection containing its endpoints.

If neither automatic nor manual segmentation completes successfully, then Blender's native functions suffice to complete the process. In these cases, the dendritic spine was duplicated, "surface_" was prepended to its name, and the faces of its base were removed manually. When endpoints were not positioned correctly, vertices were placed manually, marking the dendritic spine base and tip, and then joined into a single mesh whose name starts with "endpoints_". These strings were used to apply different logic to different object categories when the NWB file was created.

NWB file generation

After all dendritic spines were segmented, data were written to an NWB file using the "write NWB file" button. The VR-SASE add-on iterates through all the collections in Blender's "scene collection." For each collection, our add-on calculated the dendritic spine length based on its endpoints, and Blender's Bmesh programming module determined values for volume, surface area, and center of mass. These parameters along with the meta-data from the study were written to an NWB file using the PyNWB library. From the PyNWB library, we used OpticalChannel and ImageSegmentation packages to save the NWB-specified meta-data. The physical parameters generated by the VR-SASE Blender add-on were added as columns to plane segmentations, a structure provided by NWB that united data for each dendritic spine.

DataJoint integration

DataJoint is a scientific database framework streamlined for research applications (https://github.com/datajoint). We employed it for the following purposes: (1) streamlining data integration, management, and analysis, (2) providing metrics for the Sholl analysis by calculating the distance between each spine and the starting point of the dendrite, (3) refining dendritic spine classification by restricting morphological parameters to published criteria, and (4) ensuring that each dendritic spine has values for length, volume, and surface area.

The following DataJoint query identifies dendritic spines based on their values of volume and length: mouse*session*dendrite*(image_segmentation & 'volume $\geq$ "0.01‴ & 'volume $\leq$ "0.8"' &'length $\geq$ "0.2"' & 'length $\leq$ "3"') *distance_to_soma.

The VR-SASE Blender add-on applied DataJoint functionality by establishing a connection with a remote database using values for host, user_id, and password provided via its GUI. This connection allows VR-SASE to use DataJoint completeness's criteria to ensure data segmentation completion criteria, e.g., each spine possesses values for length, surface area, volume, and center of mass.

We designed a customized database schema using DataJoint's schema definition language, where tables represented experimental entities such as animals, sessions, dendrite morphology, and dendritic spine parameters. The dendrite morphological data needed for the Sholl analysis are not part of the NWB data model. We tabulated these data in a CSV file, which we integrated with the VR-SASE Blender add-on. DataJoint ensured that these data were appropriately matched with spine morphology and NWB meta-data. The final table in our pipeline is a computed table, a class of DataJoint table with extra integrity protection, which calculated the distance between each spine and the starting point of its dendrite, supplying the parameter necessary for the Sholl analysis.

This data subset was exported as a CSV and further analyzed in Excel with pivot tables.

Finally, DataJoint applied quality control checks to the dendritic spine segmentation. Because DataJoint ensures that data are complete, it identified dendritic spines that were missing values for length, surface area, or volume.

DataJoint integrated diverse data sources, ensured that spine morphology data were complete, refined our classification of dendritic spines, and enabled a Sholl analysis by calculating the distance between dendritic spines and the start of their dendrites. VR-SASE enables researchers to deposit their analysis results directly into a database, promoting research best practices.[36]

Sholl analysis

To investigate the distribution of dendritic spines relative to the cell body, we employed a Sholl analysis, as described previously[25,27,39,51,52] (see Figure S2). We calculated the average dendritic spine density within the proximal region, 0–30 μm from the starting point of the dendrite. We utilized Blender's native 3D modeling features to create a vertex, marking the starting point of the dendrite. To facilitate data binning, concentric circles were placed around this point, incremented by 30 μm. We used Blender's built-in measurement tool to find the length of dendrites encompassed within each circle and tabulated in a CSV file along with the coordinates of the dendrite starting point.

Using a computed table, a table class with extra integrity protection, DataJoint calculated the distance between the center of mass of each spine and the starting point of their dendrite using the distance function from the Python math library.

$$\text{distance to soma} = \text{math.dist}(\text{spine center of mass}, \text{soma contact point})$$

To integrate the measured dendrite length data with the spine morphology data and enable further analysis, the Blender VR-SASE add-on linked the CSV file containing the dendrite starting point coordinates and dendrite length measurements to the DataJoint Pipeline, where it was associated with the corresponding spine morphology data and NWB meta-data. The parameter "distance to soma" identified dendritic spines within the circular bins. Density was calculated as follows:

$$\text{spine density} = \text{number of dendritic spines in bin}/\text{dendrite length}.$$

By combining the functionalities of Blender and DataJoint, our pipeline for data integration and analysis, we established a comprehensive workflow for quantifying dendritic spine density and conducting Sholl analysis. The integration of these tools facilitated accurate and efficient assessment of dendritic spine distribution and provided valuable metrics for our study.

Quantification and statistical analysis

We analyzed tissue from the following groups: SCI and vehicle treatment ($n = 5$) and sham injured ($n = 3$). This resulted in 20 dendrites in the SCI group and 9 in the sham group. This translated to 858 total dendritic spines: 164 dendritic spines in the sham group and 694 in the SCI group.

We conducted statistical analyses using appropriate tests and evaluated the results based on a significance level of 0.05. We used two-tailed analyses and chose either parametric or non-parametric tests based on the nature of the data. We exported the dataset from the DataJoint pipeline and transferred these data to Prism to perform unpaired t tests or Mann-Whitney tests as appropriate. All graphs are plotted as mean ± SEM.

Figure generation

Figure 1 was created in BioRender.

Figure 2: Neurolucida, ImageJ, and VR-SASE demonstrate advances in visualization, accuracy, and precision when using a virtual reality platform. We generated images for Figures 2A and 2B by taking screenshots within the Neurolucida environment (https://www.mbfbioscience.com/products/neurolucida-360/). We processed them with the automated brightness/contrast tool in ImageJ to enhance visibility. Figure 2D depicts the maximum size of a dendritic spine in ImageJ. Compression artifacts are large for such a small image, so we used a screenshot to capture a visual representation. Figures 2E and 2F were created using Blender's rendering engine. A virtual

camera and lights were placed and positioned around the model in Blender's 3D viewport, ensuring appropriate visibility of features. The scale was indicated by creating a scale bar in the same plane as the neural tissue.

For Figure 3, we analyzed a subset of the Kauer[19] image data using ImageJ to create 3D models, which we initially segmented in Open Brush and then exported to Blender to finish segmentation. Blender created NWB files for each dendrite, which are in the linked DANDIset. A DataJoint database pipeline refined the dataset based on morphological parameters and calculated the distance between each spine and the start of its dendrite. The filtered DataJoint pipeline was exported as a CSV, which served as the data source for pivot tables in V1.5_DiscDataCompilation. Graphs and statistics were created in Prizm.

For Figure 4, we reconstructed a neuron from the DIADEM challenge. We used ImageJ, Open Brush, and Blender as above and created a 2D image of our reconstruction with Blender's rendering tool (OP-09_VR-SASE.png; Figure 4B). The HBP Neuron Morphology Viewer transformed the DIADEM reconstruction into 2D format (OP-09_GoldStandard.png; Figure 4C). Both images were registered on the maximum projection of the original neuron in Photoshop (OP-09_Both_aligned.psd). The maximum projection (MAX_OP-09.png; Figure 4A) was enlarged to the size of the gold standard, and our reconstruction was scaled down. Each reconstruction super-imposed over the original tissue was exported from Photoshop, and regions with the maximum projection visible were traced and measured (OP-09_VR-SASE-green_ROI.tif and OP-09_GoldStandardgreen_ROI; Figures 4D and 4E). Figures 4F and 4G are renderings of our segmentation results, randomly colored by ChatGPT. The Blender files contain scripts to adjust colors, and their outputs are named OP-09_Rainbow and OP-09_RainbowCloseup.

For Figure 5, the post hoc analyses were created in ChatGPT and executed in Blender's scripting environment prior to rendering the images. Within the Blender file, L712_Dendrite1aMushroomConvexHull.blend, the script "ConvexHull" created a convex shell around the dendritic spine and its head. Meshes in the collection, MushroomSpineConvexHull, were examined through Blender's UI, and volumes corresponding to the spine, head, and convex hulls are included within the script. To assist with visualization, the hulls were also converted to wireframes.

The CLDH requires multiple steps to recreate. Within the L712_023-01-31_16.11.24_Dendrite1aMushroomCord.blend, there are 3 scripts: "AddCylinders," "ColorCylinders," and "ExportCylinderNames." The first script creates a CSV containing cord names and lengths. Cords connecting external points of the dendritic spine mesh were removed manually, and the remaining cord names using "ExportCylinderNames" were compared to the original list in Excel, which deleted unique values from the original list, matching the remaining cords with their length (MushroomCords.csv). The cord lengths were graphed in Prizm (MushroomCords.pzfx) (Figure 5C). Figures 5A, 5B, 5D, and 5E were created using Blender's rendering engine as described above.

Diadem validation

To generate the image for validating the accuracy of VR-SASE, we obtained raw images and the gold-standard SWC file for OP-09 from the olfactory projection fiber dataset from the DIADEM challenge: https://diadem.janelia.org/olfactory_projection_fibers_readme.html.

We used the Neuroinformatics Morphology Viewer to convert the SWC file into a PNG file: https://neuroinformatics.nl/HBP/morphology-viewer/#. We then created a maximum projection of OP-09 using ImageJ's Z project function to create a flattened representation of the stack of raw images, which we saved in PNG format.

We transferred both the gold-standard reconstruction and maximum projection images to Photoshop, creating a layer for each. We then scaled up the maximum projection so that it aligned with the gold-standard reconstruction. Our reconstruction, rendered in Blender as described above, was added as an additional layer in Photoshop, scaled down to the size of the gold standard, and aligned with the maximum projection. Both superimposed reconstructions were flattened and exported as flattened TIFF files with 3,000 × 1,500 pixels for analysis in ImageJ, where they were scaled to 168.78 × 84.39 μm. Regions where the maximum projection was visible represent modeling inaccuracies and were quantified by tracing regions of interest with ImageJ's polygon selection tool, enabling precise quantification.

Post hoc analyses

CLDH

CLDH facilitates partitioning morphologically similar dendritic spines into clusters, further refining analyses of dendritic spine morphology.[8] A CLDH is a probability distribution of the lengths of internal cords connecting vertices of a dendritic spine mesh. To perform a post hoc CLDH analysis, we used the Text Editor (see Figure S2), Blender's native Python scripting environment, to create a set of internal cords for a representative dendritic spine. For each vertex, we created cords for 1 in every 10 of their possible connections, excluding vertices within 0.2 μm. This created a rich but manageable set of cords. Our post hoc analysis also created a CSV file containing the name and length of each cord. Cords extending over concave surfaces represent unrealistic geometries and were removed manually. The lengths of the remaining cords are the basis for our CLDH.

CVI

CVI is based on the difference in volume between a dendritic spine and its convex hull. The convex hull is the smallest possible "shell" enclosing all vertices of the dendritic spine head mesh without having any concave surfaces.[30] To perform a post hoc CVI calculation, we used a Python script executed in the Text Editor to create a convex hull surrounding the dendritic spine head. We obtained the volume of the convex hull from the 3D Printing add-on and compared it to the volume of the dendritic spine.

$$\mathrm{CVI} = (\text{hull volume} - \text{spine volume})/\text{spine volume}$$

We next extended the methodology proposed by Kashiwagi et al. to calculate the ratio for the entire dendritic spin.

User training procedure

Prior to VR-SASE training, we used a commercially available VR application, Beat Saber, to establish a manual dexterity baseline and habituate trainees to the VR environment. To develop facility with the UI, trainees with experience analyzing dendritic spines played Country Rounds on its easy setting, in practice mode, until successful completion. After this introductory VR experience, we taught trainees to load and interact with models in VR. We conducted free-form practice and demonstration until trainees demonstrated/felt comfortable with the required skills (see user documentation in the supplemental information). In supervised, and recorded, training sessions, trainees placed discs on a model neuron, receiving feedback from the trainer. Segmented neurons were exported from Open Brush, and the resulting FBX files were imported into Blender. To demonstrate the effects of skill level on post-processing time, we recorded VR-SASE's automated and manual workflows for all dendrite sections.

Hardware

Segmentation was conducted on multiple platforms, including the following:

- Alienware Desktop PC with Intel Core i7-10700k CPU @ 3.80 GHz with 16.0 GB installed RAM running Windows 11 Home Operating System 22H2 with the Oculus Rift S VR headset, from Meta Platforms.
- Alienware Laptop PC with 12th Gen Intel Core i9-12900HK, 2,500 MHz, 14 cores, logical processors 32.0 GB installed RAM running Windows 11 Home Operating System tethered to an Oculus Quest 2 VR headset, from Meta Platforms.

Blender segmentation was carried out on these machines, as well as computers with the following specifications:

- Intel(R) Core(TM) i7-1065G7 CPU @ 1.30 GHz with 32.0 GB installed RAM running Windows 11 Home Operating System 22H2.
- 16-inch MacBook Pro Apple M2 Max with 12-core CPU, 38-core GPU, 16-core Neural Engine 96 GB unified memory MacOS 13 Ventura.

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at https://doi.org/10.1016/j.patter.2024.101041.

ACKNOWLEDGMENTS

The work is funded by grants from the Paralyzed Veterans of America (PVA), VA Rehabilitation Research & Development Service (RR&D), and The Nancy Taylor Foundation for Chronic Diseases. The Center for Neuroscience and Regeneration Research is a collaboration between the PVA and Yale University.

AUTHOR CONTRIBUTIONS

A.M.T. conceived, funded, and supervised this research. M.L.R. designed, developed, and utilized VR-SASE and analyzed its dataset. S.D.K. designed and conducted the SCI experiment. C.A.B. provided expert guidance on dendritic spine classification and analysis platforms. S.P., S.F., and M.A.E. developed the prototype and documented the methods A technical review was provided by J.F.K. L.B and S.G.W. provided editing and review.

DECLARATION OF INTERESTS

The authors declare no competing interests.

DECLARATION OF GENERATIVE AI AND AI-ASSISTED TECHNOLOGIES IN THE WRITING PROCESS

We employed ChatGPT-3 during programming for prototyping and troubleshooting, during early stages of manuscript drafting, and to create the color scheme used to display segmentation results from the DIADEM Challenge. After using ChatGPT-3, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

Received: February 13, 2024
Revised: May 13, 2024
Accepted: July 16, 2024
Published: August 12, 2024

REFERENCES

1. Tan, A.M., and Waxman, S.G. (2015). Dendritic spine dysgenesis in neuropathic pain. Neurosci. Lett. *601*, 54–60. https://doi.org/10.1016/j.neulet.2014.11.024.
2. Tan, A.M., Choi, J.S., Waxman, S.G., and Hains, B.C. (2009). Dendritic spine remodeling after spinal cord injury alters neuronal signal processing. J. Neurophysiol. *102*, 2396–2409. https://doi.org/10.1152/jn.00095.2009.
3. Kim, Y., Noh, Y.W., Kim, K., and Kim, E. (2021). Hyperactive ACC-MDT Pathway Suppresses Prepulse Inhibition in Mice. Schizophr. Bull. *47*, 31–43. https://doi.org/10.1093/schbul/sbaa090.
4. Benson, C.A., Fenrich, K.K., Olson, K.L., Patwa, S., Bangalore, L., Waxman, S.G., and Tan, A.M. (2020). Dendritic Spine Dynamics after Peripheral Nerve Injury: An Intravital Structural Study. J. Neurosci. *40*, 4297–4308. https://doi.org/10.1523/jneurosci.2858-19.2020.
5. Argunşah, A.Ö., Erdil, E., Ghani, M.U., Ramiro-Cortés, Y., Hobbiss, A.F., Karayannis, T., Çetin, M., Israely, I., and Ünay, D. (2022). An interactive time series image analysis software for dendritic spines. Sci. Rep. *12*, 12405. https://doi.org/10.1038/s41598-022-16137-y.
6. Adams, M.M., and Hicks, A.L. (2005). Spasticity after spinal cord injury. Spinal Cord *43*, 577–586. https://doi.org/10.1038/sj.sc.3101757.
7. Ekaterina, P., Peter, V., Smirnova, D., Vyacheslav, C., and Ilya, B. (2023). SpineTool is an open-source software for analysis of morphology of dendritic spines. Sci. Rep. *13*, 10561. https://doi.org/10.1038/s41598-023-37406-4.
8. Pchitskaya, E., and Bezprozvanny, I. (2020). Dendritic Spines Shape Analysis-Classification or Clusterization? Perspective. Front. Synaptic Neurosci. *12*, 31. https://doi.org/10.3389/fnsyn.2020.00031.
9. Basu, S., Saha, P.K., Roszkowska, M., Magnowska, M., Baczynska, E., Das, N., Plewczynski, D., and Wlodarczyk, J. (2018). Author Correction: Quantitative 3-D morphometric analysis of individual dendritic spines. Sci. Rep. *8*, 17142. https://doi.org/10.1038/s41598-018-35164-2.
10. Reimer, M.L. (2024). A FAIR, Open-Source Virtual Reality Platform for Dendritic Spine Analysis - VR-SASE Addon and DataJoint Code. Zenodo. https://doi.org/10.5281/zenodo.10607576.
11. Rübel, O., Tritt, A., Ly, R., Dichter, B.K., Ghosh, S., Niu, L., Baker, P., Soltesz, I., Ng, L., Svoboda, K., et al. (2022). The Neurodata Without Borders ecosystem for neurophysiological data science. Elife *11*, e78362. https://doi.org/10.7554/eLife.78362.
12. Berg, S., Kutra, D., Kroeger, T., Straehle, C.N., Kausler, B.X., Haubold, C., Schiegg, M., Ales, J., Beier, T., Rudy, M., et al. (2019). ilastik: interactive machine learning for (bio)image analysis. Nat. Methods *16*, 1226–1232. https://doi.org/10.1038/s41592-019-0582-9.
13. Keto, L., and Manninen, T. (2023). CellRemorph: A Toolkit for Transforming, Selecting, and Slicing 3D Cell Structures on the Road to Morphologically Detailed Astrocyte Simulations. Neuroinformatics *21*, 483–500. https://doi.org/10.1007/s12021-023-09627-5.
14. Abdellah, M., Hernando, J., Eilemann, S., Lapere, S., Antille, N., Markram, H., and Schürmann, F. (2018). NeuroMorphoVis: a collaborative framework for analysis and visualization of neuronal morphology skeletons reconstructed from microscopy stacks. Bioinformatics *34*, i574–i582. https://doi.org/10.1093/bioinformatics/bty231.
15. Birgiolas, J., Haynes, V., Gleeson, P., Gerkin, R.C., Dietrich, S.W., and Crook, S. (2023). NeuroML-DB: Sharing and characterizing data-driven neuroscience models described in NeuroML. PLoS Comput. Biol. *19*, e1010941.
16. Reimer, M.L. (2024). VR-SASE Virtual Reality Dendritic Spine Analysis (DANDI archive). https://doi.org/10.48324/dandi.000723/0.240716.1414.
17. Brown, K.M., Barrionuevo, G., Canty, A.J., De Paola, V., Hirsch, J.A., Jefferis, G.S.X.E., Lu, J., Snippe, M., Sugihara, I., and Ascoli, G.A. (2011). The DIADEM data sets: representative light microscopy images of neuronal morphology to advance automation of digital reconstructions. Neuroinformatics *9*, 143–157. https://doi.org/10.1007/s12021-010-9095-5.
18. Reimer, M.L.e.a. (2024). A FAIR, Open-Source Virtual Reality Platform for Dendritic Spine Analysis. Dryad. https://doi.org/10.5061/dryad.w3r2280z0.
19. Kauer, S.D., Benson, C.A., Carrara, J.M., Tarafder, A.A., Ibrahim, Y.H., Estacion, M.A., Waxman, S.G., and Tan, A.M. (2024). In Press: PAK1 inhibition with Romidepsin attenuates H-reflex hyperexcitability after spinal cord injury. J. Physiol.
20. Harris, K.M., and Kater, S.B. (1994). Dendritic spines: cellular specializations imparting both stability and flexibility to synaptic function. Annu. Rev. Neurosci. *17*, 341–371. https://doi.org/10.1146/annurev.ne.17.030194.002013.
21. Harris, K.M. (1999). Structure, development, and plasticity of dendritic spines. Curr. Opin. Neurobiol. *9*, 343–348. https://doi.org/10.1016/s0959-4388(99)80050-6.
22. Fogarty, M.J., Mu, E.W.H., Lavidis, N.A., Noakes, P.G., and Bellingham, M.C. (2020). Size-Dependent Vulnerability of Lumbar Motor Neuron Dendritic Degeneration in SOD1(G93A) Mice. Anat. Rec. *303*, 1455–1471. https://doi.org/10.1002/ar.24255.
23. Fogarty, M.J., Kanjhan, R., Bellingham, M.C., and Noakes, P.G. (2016). Glycinergic Neurotransmission: A Potent Regulator of Embryonic Motor Neuron Dendritic Morphology and Synaptic Plasticity. J. Neurosci. *36*, 80–87. https://doi.org/10.1523/jneurosci.1576-15.2016.
24. Peters, A., and Kaiserman-Abramof, I.R. (1970). The small pyramidal neuron of the rat cerebral cortex. The perikaryon, dendrites and spines. Am. J. Anat. *127*, 321–355. https://doi.org/10.1002/aja.1001270402.
25. Benson, C.A., Olson, K.L., Patwa, S., Reimer, M.L., Bangalore, L., Hill, M., Waxman, S.G., and Tan, A.M. (2021). Conditional RAC1 knockout in motor neurons restores H-reflex rate-dependent depression after spinal cord injury. Sci. Rep. *11*, 7838. https://doi.org/10.1038/s41598-021-87476-5.
26. Tan, A.M., and Waxman, S.G. (2012). Spinal cord injury, dendritic spine remodeling, and spinal memory mechanisms. Exp. Neurol. *235*, 142–151. https://doi.org/10.1016/j.expneurol.2011.08.026.

27. Tan, A.M., Stamboulian, S., Chang, Y.W., Zhao, P., Hains, A.B., Waxman, S.G., and Hains, B.C. (2008). Neuropathic pain memory is maintained by Rac1-regulated dendritic spine remodeling after spinal cord injury. J. Neurosci. *28*, 13173–13183. https://doi.org/10.1523/jneurosci.3142-08.2008.

28. Bandaru, S.P., Liu, S., Waxman, S.G., and Tan, A.M. (2015). Dendritic spine dysgenesis contributes to hyperreflexia after spinal cord injury. J. Neurophysiol. *113*, 1598–1615. https://doi.org/10.1152/jn.00566.2014.

29. Manubens-Gil, L., Zhou, Z., Chen, H., Ramanathan, A., Liu, X., Liu, Y., Bria, A., Gillette, T., Ruan, Z., Yang, J., et al. (2023). BigNeuron: a resource to benchmark and predict performance of algorithms for automated tracing of neurons in light microscopy datasets. Nat. Methods *20*, 824–835. https://doi.org/10.1038/s41592-023-01848-5.

30. Kashiwagi, Y., Higashi, T., Obashi, K., Sato, Y., Komiyama, N.H., Grant, S.G.N., and Okabe, S. (2019). Computational geometry analysis of dendritic spines by structured illumination microscopy. Nat. Commun. *10*, 1285. https://doi.org/10.1038/s41467-019-09337-0.

31. Miller, J.P., Rall, W., and Rinzel, J. (1985). Synaptic amplification by active membrane in dendritic spines. Brain Res. *325*, 325–330. https://doi.org/10.1016/0006-8993(85)90333-6.

32. Segev, I., and Rall, W. (1988). Computational study of an excitable dendritic spine. J. Neurophysiol. *60*, 499–523. https://doi.org/10.1152/jn.1988.60.2.499.

33. Arellano, J.I., Benavides-Piccione, R., Defelipe, J., and Yuste, R. (2007). Ultrastructure of dendritic spines: correlation between synaptic and spine morphologies. Front. Neurosci. *1*, 131–143. https://doi.org/10.3389/neuro.01.1.1.010.2007.

34. Bell, M.K., Holst, M.V., Lee, C.T., and Rangamani, P. (2022). Dendritic spine morphology regulates calcium-dependent synaptic weight change. J. Gen. Physiol. *154*, e202112980. https://doi.org/10.1085/jgp.202112980.

35. Li, Y., Gorassini, M.A., and Bennett, D.J. (2004). Role of Persistent Sodium and Calcium Currents in Motoneuron Firing and Spasticity in Chronic Spinal Rats. J. Neurophysiol. *91*, 767–783. https://doi.org/10.1152/jn.00788.2003.

36. Reimer, M.L., Bangalore, L., Waxman, S.G., and Tan, A.M. (2021). Core principles for the implementation of the neurodata without borders data standard. J. Neurosci. Methods *348*, 108972. https://doi.org/10.1016/j.jneumeth.2020.108972.

37. International Brain Laboratory, Bonacchi, N., Chapuis, G.A., Churchland, A.K., DeWitt, E.E.J., Faulkner, M., Harris, K.D., Huntenburg, J.M., Hunter, M., Laranjeira, I.C., et al. (2023). A modular architecture for organizing, processing and sharing neurophysiology data. Nat. Methods *20*, 403–407. https://doi.org/10.1038/s41592-022-01742-6.

38. Ishii, K., Nagaoka, A., Kishida, Y., Okazaki, H., Yagishita, S., Ucar, H., Takahashi, N., Saito, N., and Kasai, H. (2018). em>In Vivo Volume Dynamics of Dendritic Spines in the Neocortex of Wild-Type and Fmr1 KO Mice. eneuro *5*. ENEURO.0282-0218.2018. https://doi.org/10.1523/eneuro.0282-18.2018.

39. Patwa, S., Benson, C.A., Dyer, L., Olson, K.L., Bangalore, L., Hill, M., Waxman, S.G., and Tan, A.M. (2019). Spinal cord motor neuron plasticity accompanies second-degree burn injury and chronic pain. Physiol. Rep. *7*, e14288. https://doi.org/10.14814/phy2.14288.

40. Usher, W., Klacansky, P., Federer, F., Bremer, P.T., Knoll, A., Yarch, J., Angelucci, A., and Pascucci, V. (2018). A Virtual Reality Visualization Tool for Neuron Tracing. IEEE Trans. Vis. Comput. Graph. *24*, 994–1003. https://doi.org/10.1109/tvcg.2017.2744079.

41. McDonald, T., Usher, W., Morrical, N., Gyulassy, A., Petruzza, S., Federer, F., Angelucci, A., and Pascucci, V. (2021). Improving the Usability of Virtual Reality Neuron Tracing with Topological Elements. IEEE Trans. Vis. Comput. Graph. *27*, 744–754. https://doi.org/10.1109/tvcg.2020.3030363.

42. Bria, A., Iannello, G., and Peng, H. (2015). An Open-Source VAA3D Plugin for Real-Time 3D Visualization of Terabyte-Sized Volumetric Images, pp. 520–523.

43. Wu, Y.C., Maymon, C., Paden, J., and Liu, W. (2023). Launching Your VR Neuroscience Laboratory. Curr. Top. Behav. Neurosci. *65*, 25–46. https://doi.org/10.1007/7854_2023_420.

44. Qi, C.R., and Hao Su, K.M.L.J.G. (2017). PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. held in Hawaii Convention Center. https://doi.org/10.48550/arXiv.1612.00593.

45. Dai, K., Hernando, J., Billeh, Y.N., Gratiy, S.L., Planas, J., Davison, A.P., Dura-Bernal, S., Gleeson, P., Devresse, A., Dichter, B.K., et al. (2020). The SONATA data format for efficient description of large-scale network models. PLoS Comput. Biol. *16*, e1007696. https://doi.org/10.1371/journal.pcbi.1007696.

46. Bai, W., Zhou, X., Ji, L., Cheng, J., and Wong, S.T.C. (2007). Automatic dendritic spine analysis in two-photon laser scanning microscopy images. Cytometry A. *71*, 818–826. https://doi.org/10.1002/cyto.a.20431.

47. Choi, J., Lee, S.E., Lee, Y., Cho, E., Chang, S., and Jeong, W.K. (2023). DXplorer: A Unified Visualization Framework for Interactive Dendritic Spine Analysis Using 3D Morphological Features. IEEE Trans. Vis. Comput. Graph. *29*, 1424–1437. https://doi.org/10.1109/tvcg.2021.3116656.

48. Tønnesen, J., Katona, G., Rózsa, B., and Nägerl, U.V. (2014). Spine neck plasticity regulates compartmentalization of synapses. Nat. Neurosci. *17*, 678–685. https://doi.org/10.1038/nn.3682.

49. Cao, X.C., Pappalardo, L.W., Waxman, S.G., and Tan, A.M. (2017). Dendritic spine dysgenesis in superficial dorsal horn sensory neurons after spinal cord injury. Mol. Pain *13*, 1744806916688016. https://doi.org/10.1177/1744806916688016.

50. Treves, S., Vukcevic, M., Griesser, J., Armstrong, C.F., Zhu, M.X., and Zorzato, F. (2010). Agonist-activated Ca2+ influx occurs at stable plasma membrane and endoplasmic reticulum junctions. J. Cell Sci. *123*, 4170–4181. https://doi.org/10.1242/jcs.068387.

51. Tan, A.M., Samad, O.A., Liu, S., Bandaru, S., Zhao, P., and Waxman, S.G. (2013). Burn injury-induced mechanical allodynia is maintained by Rac1-regulated dendritic spine dysgenesis. Exp. Neurol. *248*, 509–519. https://doi.org/10.1016/j.expneurol.2013.07.017.

52. Tan, A.M., Chang, Y.W., Zhao, P., Hains, B.C., and Waxman, S.G. (2011). Rac1-regulated dendritic spine remodeling contributes to neuropathic pain after peripheral nerve injury. Exp. Neurol. *232*, 222–233. https://doi.org/10.1016/j.expneurol.2011.08.028.