SynAnno: Interactive Guided Proofreading of Synaptic Annotations

Leander Lauenburg[1,2], Jakob Troidl[2], Adam Gohain[1], Zudi Lin[2], Hanspeter Pfister[2], Donglai Wei[1]

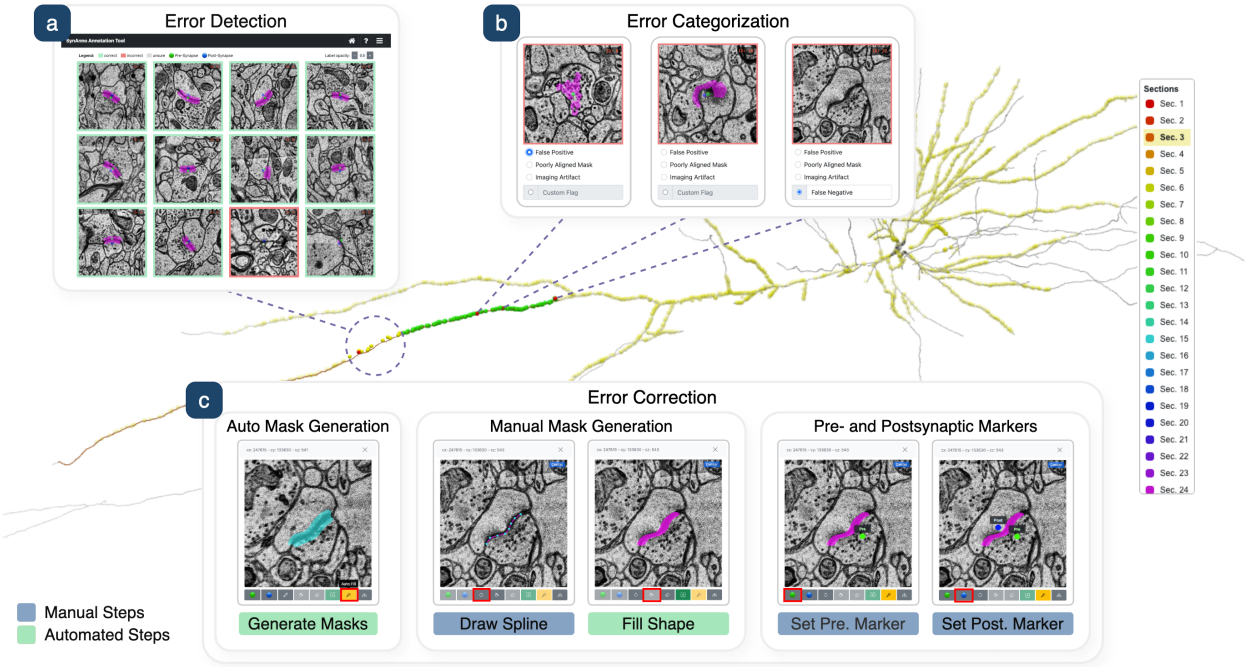

Fig. 1: **Proofreading Workflow.** SynAnno decomposes the workflow into three views: (a) Error Detection: Users review synapse masks in a grid layout, guided by an interactive 3D neuron viewer. (b) Error Categorization: Identified errors are validated and classified. (c) Error Correction: Users fix masks using manual or automatic tools and set pre-/postsynaptic markers to define information flow.

Abstract— Connectomics, a subfield of neuroscience, aims to map and analyze synapse-level wiring diagrams of the nervous system. While recent advances in deep learning have accelerated automated neuron and synapse segmentation, reconstructing accurate connectomes still demands extensive human proofreading to correct segmentation errors. We present SynAnno, an interactive tool designed to streamline and enhance the proofreading of synaptic annotations in large-scale connectomics datasets. SynAnno integrates into existing neuroscience workflows by enabling guided, neuron-centric proofreading. To address the challenges posed by the complex spatial branching of neurons, it introduces a structured workflow with an optimized traversal path and a 3D mini-map for tracking progress. In addition, SynAnno incorporates fine-tuned machine learning models to assist with error detection and correction, reducing the manual burden and increasing proofreading efficiency. We evaluate SynAnno through a user and case study involving seven neuroscience experts. Results show that SynAnno significantly accelerates synapse proofreading while reducing cognitive load and annotation errors through structured guidance and visualization support. The source code and interactive demo are available at: https://github.com/PytorchConnectomics/SynAnno.

Index Terms—Connectomics, Synaptic Annotations, Neuron-Centric, Proofreading Workflow

1 Introduction

Recent advances in connectomics have enabled the large-scale reconstruction of neuronal circuits from high-resolution imaging data across diverse model organisms, including Drosophila [9,27], mouse [38], and even humans [35]. Deep learning–based methods have significantly accelerated this process by automating neuron segmentation and synapse detection. However, despite their high accuracy, these automated approaches still produce errors that necessitate extensive human proofreading [19]. While substantial progress has been made in developing tools and workflows for proofreading neuron segmentation [10, 11], comparatively less attention has been given to the validation and correction of synaptic annotations. This gap poses a critical challenge, as errors in synapse labeling can misrepresent circuit connectivity and lead to flawed functional interpretations. As a result, high-quality neuron-centric proofreading of synaptic annotations remains essential for producing reliable and interpretable connectomic data.

Neuron-centric proofreading of synaptic annotations is inherently challenging due to the complex branching of neuronal arbors and the sheer volume of synaptic connections in large-scale datasets. For instance, the H01 connectome [35] reconstructs one cubic millimeter of human brain tissue at nanoscale resolution, containing over 57 thousand cells and 150 million synapses—yet only a small subset of its neurons has been fully proofread. Traditional proofreading methods typically rely on manual inspection via 3D visualization tools, a process that is

- [1] *Department of Computer Science, Boston College*
- [2] *School of Engineering & Applied Sciences, Harvard University*

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxxx/TVCG.201x.xxxxxxx

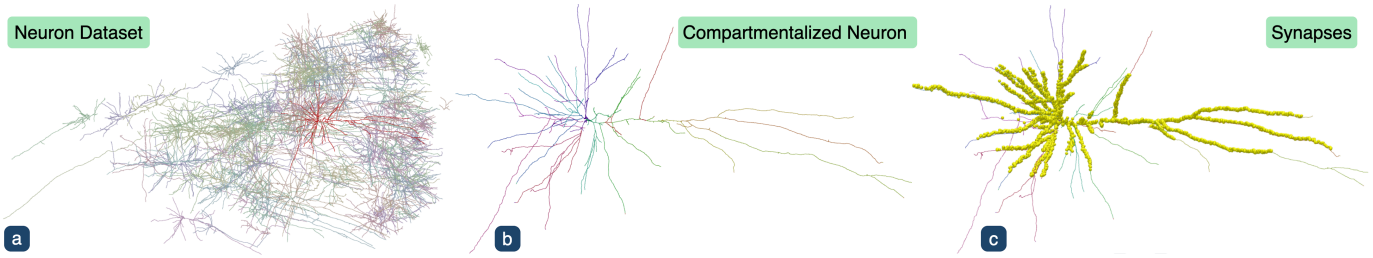

Fig. 2: **Neuron-Centric Synapse Proofreading.** (a) The user selects a specific neuron from the dataset, (b) whose skeleton is compartmentalized and rendered. (c) The goal is to proofread all synapses associated with the selected neuron (loaded as yellow dots).

time-consuming, cognitively taxing, and susceptible to inconsistencies. These challenges underscore the need for interactive tools that can streamline synapse proofreading while preserving both accuracy and scalability.

We present SynAnno, an interactive tool to facilitate guided proofreading of synaptic annotations in large-scale connectome datasets. SynAnno integrates structured workflows with machine learning-assisted error correction, enabling neuroscientists to review and refine AI-generated synapse masks systematically. The system provides a neuron-centric approach to proofreading, allowing users to traverse synapses associated with each neuron compartment in a structured manner (Fig. 2). To enhance efficiency, SynAnno introduces depth-first search (DFS)-based pathfinding for systematic traversal, an abstraction pyramid for multi-scale visualization, and an interactive error-labeling interface. Additionally, machine learning models assist in identifying and correcting synapse mask errors, reducing the overall proofreading burden. We evaluate SynAnno in user- and case studies with 7 domain experts. We find that our structured proofreading workflow, combined with machine learning-assisted corrections, lead to significant improvements in proofreading speed and accuracy. We further discuss the usability insights gained from neuroscience domain experts and highlight opportunities for future improvements. In summary, our contributions include: **(1)** We contribute a guided synapse proofreading workflow. This structured approach allows the effective review of synaptic annotations and incorporates neuron-centric navigation strategies that suggest an optimal proofreading trajectory for each neuron. **(2)** We propose a hierarchical visual interface for error identification and correction. The user first quickly screens groups of synaptic annotations for errors and then further refines the annotations if incorrectly labeled synapses in a separate view. **(3)** We provide an interactive machine learning-assisted error correction approach. SynAnno integrates a deep learning model for interactive synapse mask refinement, improving proofreading efficiency. **(4)** We empirically evaluate SynAnno in a user study with 7 neuroscience and proofreading experts, assessing the effectiveness of SynAnno in real-world proofreading tasks.

2 Related Work

Connectomics and Synapse Analysis. Connectomics seeks to map the complete neural wiring of organisms by identifying neurons and their synaptic connections at synapse-level resolution. A key challenge in connectomics is analyzing the vast number of synapses within dense neural circuits, requiring both structural and functional insights. Synapse analysis encompasses multiple research objectives, including morphology [12, 28], connectivity mapping [9], synaptic classification [13], and synapse density analysis [21], each providing insights into neural plasticity and computational properties [26]. Despite advances in automated synapse detection methods [7, 29], challenges remain in ensuring error detection accuracy, identifying false negatives, and correcting erroneous synapse masks and information flow (Fig. 3). Addressing these issues is critical for generating high-fidelity neural circuits supporting research on cognition and neurological disorders.

Interactive Proofreading for Connectomics. Interactive proofreading tools are vital for correcting errors in automated neuron and synapse segmentations. NeuroBlocks [1] introduced a modular interface for visualizing and editing neural circuits, enabling experts to interactively refine segmentations. FlyWire [10] and NeuTu [42] expanded on this by supporting large-scale, collaborative proofreading of dense connectomic datasets. To improve user efficiency, Haehn et al. [16, 17] proposed guided workflows that direct attention to likely segmentation errors while minimizing cognitive load. Later work addressed the scalability of these systems for increasingly large and complex datasets [15]. In parallel, tools like VAST [2], VICE [14], and Raveler [8] have combined manual and semi-automatic correction with machine learning support, allowing users to iteratively refine annotations. Our work builds on these foundations by introducing a structured, neuron-centric proofreading workflow tailored to synaptic annotations, supporting guided traversal, error categorization, and ML-assisted correction.

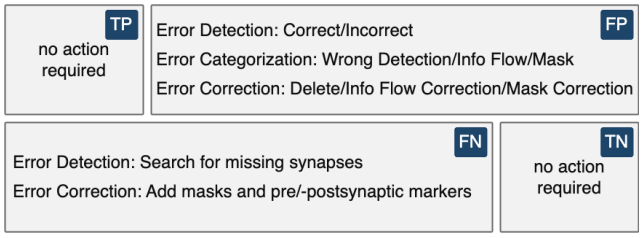

Fig. 3: **Synaptic Annotation Error Taxonomy.** False positives may arise from incorrect detection, information flow assignment, or segmentation masks, each requiring a different correction strategy. False negatives are more challenging, often requiring exhaustive review across all neuron compartments, making them particularly time-consuming to identify.

Synaptic Annotation and Focused Proofreading. Synapse annotation at scale remains a major challenge in connectomics. Plaza et al. [32] introduced scalable methods for synapse labeling, while focused proofreading [30] prioritized regions with a high likelihood of annotation errors to improve efficiency. Lin et al. [22] proposed an active learning framework that identifies informative synapse instances for manual review, thereby improving model performance with minimal human effort. Additionally, neuPrint [31] provides an accessible platform for querying and analyzing synaptic connectivity. Together, these efforts emphasize the need for efficient, targeted proofreading strategies to reduce the manual burden of validating automated annotations.

Automatic Segmentation Error Correction. Connectomics has seen a range of approaches for addressing segmentation errors, particularly split and merge mistakes in neuron reconstructions. Zung et al. [43] trained multiscale 3D convolutional networks to detect and correct such errors, while graph-based models [25] represent neurons as annotated graphs to support structured proofreading. Shape completion techniques, including point cloud models [3] and VAEs [37], learn shape priors to recover fragmented segments. Building on these efforts, our method incorporates a 3D U-Net–based synapse segmentation refinement directly into the proofreading workflow, reducing reliance on manual correction and improving efficiency.

Visualization for Connectomics. Effective visualization is critical for analyzing large-scale connectomics datasets, enabling researchers to navigate and interpret complex neuronal structures [6, 39, 40]. Traditional tools like Neuroglancer [24] provide interactive 3D environments

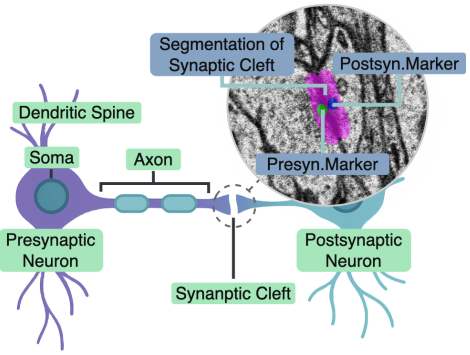

Fig. 4: **Illustration of a Synaptic Structure.** A presynaptic neuron (purple) connects to a postsynaptic neuron (blue) via an axon terminating in a synaptic cleft. The postsynaptic neuron features a dendritic spine that receives the signal. The inset shows an electron microscopy (EM) image with segmented synaptic cleft (magenta), presynaptic marker (green), and postsynaptic marker (blue).

that facilitate the exploration of dense neural reconstructions, yet they can lead to occlusion and cognitive overload. Abstraction-based methods, such as NeuroLines [5], simplify connectivity representations using intuitive metaphors, while ConnectomeExplorer [4] integrates query-based analysis with multi-scale visualization to support comprehensive data exploration. Our approach extends these visualization techniques by incorporating a hierarchical proofreading interface that allows seamless transitions between 2D, quasi-3D, and full 3D views, optimizing the balance between overview and detailed inspection.

3 Neuroscience Fundamentals

Neurons and Synapses. Neurons are the fundamental building blocks of the nervous system and are responsible for transmitting electrical and chemical signals across the brain. Each neuron consists of a cell body (soma), from which dendrites extend, which receive signals, and an axon, which transmits signals to other neurons (Fig. 4). Neurons communicate via synapses, specialized junctions where chemical or electrical signals are transmitted between cells. At the synapse, the presynaptic neuron releases neurotransmitters from vesicles into the synaptic cleft, where they bind to receptors on the postsynaptic neuron. This process allows information to flow directionally through neural circuits. Synaptic polarity, which determines whether a synapse is excitatory or inhibitory, is crucial in shaping neuronal computations. *Inhibitory synapses* decrease the likelihood of an action potential, while *excitatory synapses* increase the likelihood of an action potential.

Information Flow. The spatial arrangement of synapses is critical for understanding how information flows within neural networks. Synapses are often clustered in specific regions of a neuron, forming functional units known as synaptic boutons on axons and dendritic spines on dendrites. The placement of these synaptic connections determines signal integration and network dynamics. Notably, patterns of information flow can vary across species—for example, connections are often one-to-one in mammals, whereas in Drosophila, a single synapse may transmit to multiple postsynaptic targets.

Connectomics Data. Studying neural circuits at the level of individual synapses relies on high-resolution imaging datasets such as the H01 human brain data [35], which captures over 183 million synapses in a cubic millimeter of tissue. Connectome reconstructions provide unprecedented insights into how neurons are wired together to perform computations. However, despite advances in automated segmentation, errors in synaptic annotations persist. These errors can cause misrepresenting connectivity patterns, necessitating extensive proofreading.

4 Goal & Task Analysis

The development of SynAnno was guided by close collaboration with neuroscience experts who emphasized the need for a more structured, scalable, and efficient approach to proofreading synaptic annotations and information flow. While recent advances in automated segmentation and classification have significantly accelerated the reconstruction of large-scale connectomic datasets, the task of verifying and correcting AI-generated synapse masks remains largely manual and cognitively demanding. Neuroscientists require a proofreading workflow that not only systematically guides them through the dataset but also preserves a clear overview of neuron morphology and proofreading progress. At the same time, they need precise control over corrections and the ability to incorporate machine learning-assisted suggestions without compromising accuracy or interpretability. Following a design study methodology, we conducted semi-structured interviews with experts in synaptic annotation and connectomics from institutions working on Drosophila, mouse, and human cortical datasets. These engagements were complemented by feedback gathered at connectomics workshops and conferences, providing insight into domain-specific challenges and common workflow bottlenecks. This formative process led to the articulation of core domain goals, actionable design requirements, and concrete tasks, which informed the development of SynAnno.

4.1 Domain Goals

The primary goal of proofreading synaptic annotations in large-scale connectomics datasets is to ensure that reconstructed neural circuits are functionally meaningful, structurally accurate, and biologically interpretable. Given the scale and complexity of current EM datasets, even small annotation errors can propagate into large-scale misinterpretations of connectivity and circuit function. In collaboration with neuroscience experts, we identified five core scientific objectives that define the priorities and challenges of synapse proofreading:

G1 – Ensure Synaptic Validity. Automatically predicted synapses may include false positives due to imaging artifacts, ambiguous morphology, or segmentation errors. A critical goal is to validate that each annotated synapse represents a true biological connection by confirming the presence of structural indicators (e.g., vesicles, clefts, and membrane specializations). Removing invalid predictions helps prevent spurious connections in downstream connectivity analyses.

G2 – Increase Synapse Recall. False negatives—real synapses missed by automated models—are common, especially for small or morphologically atypical contacts. Recovering these missing instances is essential for maintaining the completeness of the reconstructed connectome. Since undetected synapses often occur in complex or low-contrast regions, targeted tools and workflows are needed to assist in their discovery.

G3 – Guarantee Functional Accuracy. Correctly assigning the directionality and polarity of synapses is key for interpreting how information flows through a neural circuit. Misannotations—such as incorrect pre/post-synaptic assignments—can alter the inferred function of a pathway. Proofreading must ensure that synapse annotations reflect biologically plausible connectivity, based on both structure and context within the neuron.

G4 – Improve Synapse Mask Quality. Segmentation masks must precisely delineate the extent of each synapse to enable accurate quantification and partner identification. Common issues include merged synapses, fragmented masks, or misaligned boundaries. Improving mask quality through manual or ML-assisted refinement enhances the anatomical fidelity of the dataset and supports more reliable downstream analysis.

G5 – Support Reproducibility and Interpretability. Annotations should be consistent and transparent to facilitate collaboration, validation, and cross-dataset comparisons. Clear workflows, structured annotation processes, and visual traceability ensure that results can be reproduced and interpreted by other researchers. Supporting reproducibility also enables integration with computational pipelines for simulation, analysis, and comparative studies.

4.2 Design Requirements

To meet the domain goals, we translated key challenges into actionable design requirements, addressing scale, structural complexity, and

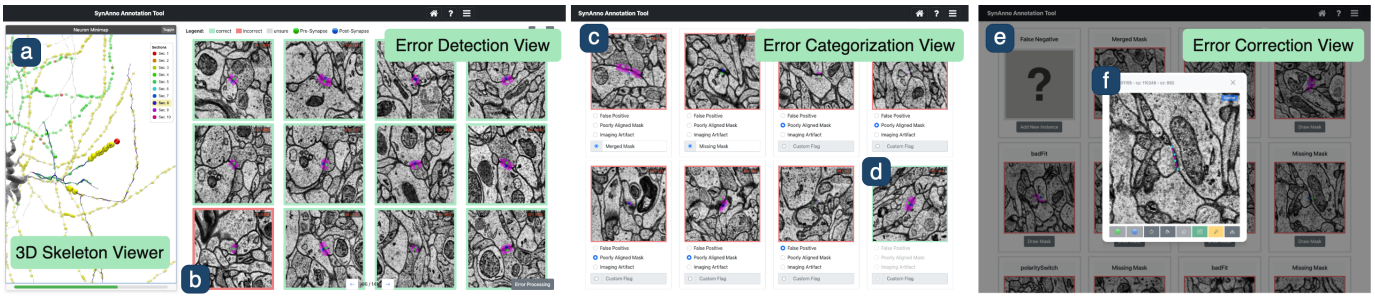

Fig. 5: **SyAnno's Three Main Views**. In the *Error Detection* view (left), users are guided along neuron sections or by the uncertainty-based ordering of synapses within a bounding box. (a) Interactive neuron rendering supports orientation and progress tracking. (b) False positive synapses can be labeled as incorrect. In the *Error Categorization* view (center), users revisit instances previously marked as "incorrect" or "unsure" and assign specific error labels, such as (c) "Merged Mask" or (d) reverting the label to "correct." In the *Error Correction* view (right), users can manually adjust or auto-generate masks, reassign pre- and postsynaptic markers, and (e,f) add false negatives.

cognitive load in proofreading large connectomics datasets.

R1 – Orientation and Partitioning. Unintuitive partitioning of neural structures can lead to a loss of spatial and structural context during large-scale proofreading, impairing annotators' awareness of their current location, connectivity to previously reviewed regions, and the neuron's overall morphology. This weakens their mental model and increases the risk of overlooked sections. The system should therefore provide intuitive visual cues that maintain spatial continuity and clearly associate individual synapses with the broader neuronal structure.

R2 - Process Tracking. Poor process tracking can result in redundant work, overlooked areas, and inefficiencies—especially during collaboration or after task interruptions. The system should clearly show review status—completed, current, and remaining—to support effective progress tracking, maintain contextual awareness, and simplify task resumption.

R3 - Pathfinding. Determining a coherent traversal path through neuronal structures is difficult due to their extensive size and complex connectivity. Without structured navigation, proofreading risks fragmented coverage and difficulties in task delegation. The system should provide a biologically meaningful, deterministic, and recoverable traversal strategy aligned with neuronal topology to enable systematic and reliable task management.

R4 - Visual Hierarchy. Synapse evaluation complexity varies significantly, e.g., due to orientation in anisotropic volumes, synapse size, and data origin. Many synapses can be rapidly assessed, while others require detailed inspection. The system should implement a visual abstraction hierarchy, defaulting to simplified instance views with easy access to high-resolution details, supporting efficiency and accuracy.

R5 - Task Separation. Proofreading involves a sequence of distinct subtasks: detecting potential errors, diagnosing their cause, and applying corrections. Traditional workflows often intermingle these tasks, forcing annotators to switch context frequently. The system should adopt a clear separation-of-concerns approach, enabling annotators to focus on one task at a time and supporting a systematic, efficient, and cognitively manageable workflow.

4.3 Tasks

Based on the domain goals and design requirements outlined above, we derived the following key tasks that SynAnno must support:

T1 – Classify predicted synapses as correct, incorrect, or unsure. Users need to rapidly triage and evaluate large sets of predicted synapse masks. (G1, G5; R1, R4, R5)

T2 – Label incorrect or uncertain synapses with specific error types. Users must be able to assign detailed error labels to support downstream correction and collaborative workflows. (G1, G3, G4, G5; R2, R5)

T3 – Remove predicted synapses that are biologically invalid. Users should have the ability to identify and delete synaptic annotations that do not correspond to real connections. (G1, G2; R2, R3)

T4 – Correct the information flow direction for valid synapses. Enable users to reassign pre- and post-synaptic labels to ensure accurate representation of information flow. (G3; R4, R5)

T5 – Refine or redraw synapse masks of erroneous synapses. For poorly segmented instances or false negatives, users need the ability to manually adjust or redraw masks to ensure anatomical accuracy. (G4; R3, R4, R5)

T6 – Discover, add, and annotate missing synapses to the dataset. Users must be able to identify false negatives and create new annotations, including synapse masks and information flow directions. (G2, G3, G4; R1, R2, R3)

5 Neuron-centric Proofreading Workflows

Overview. Users start by browsing a connectome data set via a Neuroglancer [23] instance to inspect and select a neuron of interest for proofreading. Upon selecting a particular neuron, SynAnno precomputes synapse metadata, downloads the neuron skeleton, and partitions it into anatomical compartments by generating a depth-first traversal path originating from the soma. Finally, SynAnno maps predicted synapses to their respective locations. SynAnno displays the 3D neuron structure and related synapses in an embedded 3D skeleton viewer [41] (Fig. 5a). Each compartment is color-coded and accessible via a compartment legend (Fig. 1, right). Users initiate proofreading by following the traversal order or directly selecting specific compartments. Upon selection, the interface focuses on the corresponding compartment and loads the first page of synapses associated with it (Fig. 5, left). Next, users can engage in both the *synapse mask correction workflow* and the *false negative correction workflow*.

5.1 Synapse Mask Correction Workflow

The synapse mask correction workflow is illustrated in Fig. 6. It begins in the *Error Detection* view (Fig. 5, left), where users review synapse masks and information flow markers, labeling erroneous instances as *incorrect* or *unsure* (Fig. 6a). If confirmation is needed, users may inspect the instance in greater context using one of three options: an enlarged instance modal view, quasi-3D functionality that enables scrolling through adjacent slices, or a linked Neuroglancer view that directly centers on the synapse mask (Fig. 6b). These optional views assist in resolving ambiguous cases. Selected instances can be further categorized in the *Error Categorization* view (Fig. 5, center), where users assign predefined error labels or define custom categories (Fig. 6c). Subsequently, users may download the labeled data or proceed to the *Error Correction* view (Fig. 5, right) to correct the synapse mask and adjust the information flow markers (Fig. 6d).

5.2 False Negative Correction Workflow

False negative synaptic annotations are systematically corrected within the compartment-based proofreading process. Once a particular neuron section has been fully reviewed for false positives in the *Error Detection* view, users are prompted to inspect the respective neuron section for missing synapses. Exploring neuron sections is supported via

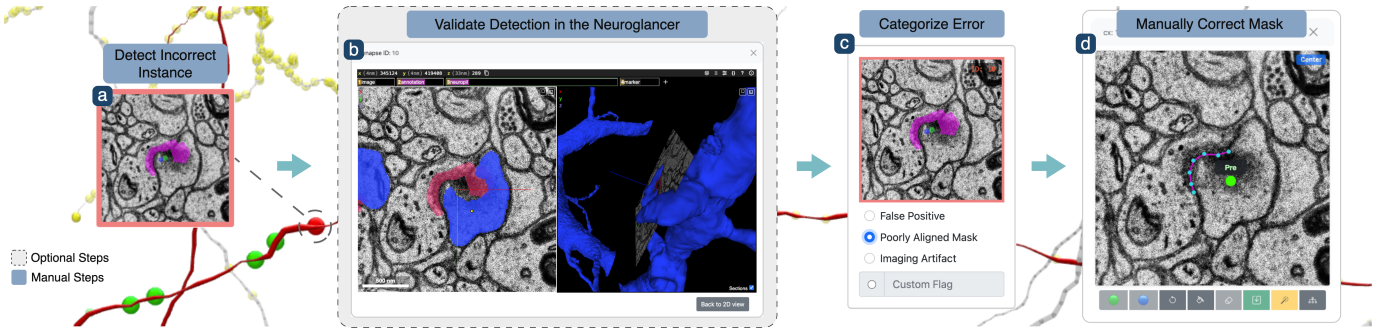

Fig. 6: **Synapse Mask Correction Workflow**. (a) The user reviews a neuron section in the *Error Detection* view, guided by the neuron rendering, and identifies a synapse instance they believe is incorrect, marking it accordingly. (b) Optionally, they inspect the instance in Neuroglancer to confirm their suspicion. (c) After completing the review, they assign the label "Poorly Aligned Mask" in the *Error Categorization* view. (d) Finally, they correct the mask and information flow direction markers in the *Error Correction* view.

Neuroglancer, which displays the neuron skeleton and the microscopy imaging context (Fig. 7a). After identifying a candidate synapse, users place a marker at the suspected location. SynAnno then automatically derives a bounding box, which users may adjust before confirming the selection (Fig. 7b). Upon confirmation, the instance is cropped and added to the tile view (Fig. 7c). Each newly added instance is automatically labeled as a false negative (Fig. 7d), eliminating the need for manual categorization. The proofreading process concludes in the *Error Correction* view (Fig. 5, right). SynAnno supports semi-automated segmentation using a 3D U-Net (Sec. 7), with optional manual refinement if the automated result is unsatisfactory (Fig. 7e). Additionally, users can place the presynaptic and postsynaptic markers to indicate the direction of information flow (Fig. 7f), completing the annotation.

In addition to the neuron-centric proofreading workflow, SynAnno supports a *volume-centric proofreading* strategy in which users define a bounding box to load and review synapses within a specific region, supporting exploratory review or targeted validation.

6 User Interface & Interactions

SynAnno provides an interactive interface to perform fine-grained synaptic proofreading while maintaining spatial awareness of complex neuronal morphology. The interface supports efficient error detection, clear task separation, and contextualized navigation through large neural datasets. It is organized into three coordinated views: *Error Detection*, *Error Categorization*, and *Error Correction* (Fig. 5). These views integrate a 2D tile-based instance view (Fig. 5b), a 3D slice viewer, a full 3D Neuroglancer integration (Fig. 8), and an interactive 3D skeleton viewer (Fig. 5a), to build a visual hierarchy (R4) and preserve spatial orientation (R1).

6.1 Error Detection View

The Error Detection view (Fig. 5a-b) addresses the need for scalable triage of predicted synapses (G1, G5) by displaying a grid of instances from the selected neuron compartment or bounding box. Users review each instance and classify it as *correct*, *incorrect*, or *unsure*, enabling rapid error detection (T1). To minimize manual burden, unlabeled tiles default to the *correct* state when progressing to the next page—streamlining labeling in high-quality regions (R5).

3D Skeleton View. To support spatial orientation and reasoning, SynAnno features a fully interactive 3D skeleton viewer (Fig. 5a). It enables users to rotate, zoom, and pan the currently selected neuron to explore its global morphology and detailed local structure. Synapses are visualized as spheres mapped onto the skeleton, with their error labels (*correct*, *incorrect*, or *unsure*) encoded through color, size, and opacity. These spatially contextualized visual cues provide real-time feedback, supporting functional accuracy (G3) and structural completeness (G2).

Branch Management. To enable consistent traversal of complex neuron morphologies, SynAnno partitions neurons into multiple compartments, structuring them in a biologically meaningful depth-first order beginning at the soma (Section 8). This ordering defines a deterministic path, which is visually communicated through compartment-specific colors on the neuron and a legend (R3). Users are prompted to follow this path in the *Error Detection* view. Still, they can also override it through direct interaction with the legend, which enables navigation between sections (Fig. 1, right). Inactive compartments are grayed out, balancing local focus with global context (R1, R2).

Visual Synchronization. The 3D skeleton view is tightly integrated with the *Error Detection* view, forming a dynamic, bidirectional connection between spatial and semantic information. Updates made during proofreading are reflected immediately—for example, through a change in the color of a newly annotated synapse—reinforcing a coherent mental model of progress (G5; R2). Synapses currently under review are visually emphasized—enlarged and fully opaque—while others are rendered semi-transparent or downscaled, depending on their review state. This coordination ensures that users remain oriented and supports structured annotation coverage across views.

Progress Tracking. While visual synchronization and branch management provide direct feedback on progress, the viewer also features a global progress bar at the bottom (Fig. 5a). This bar offers an at-a-glance summary of proofreading completion, helping users monitor coverage, revisit specific branches, and avoid redundant effort—thereby promoting transparency and sustaining motivation.

Tile-Based Instance Inspection. Synapse instances are rendered as 2D tiles (Fig. 5b), each centered on a representative slice (R4). Users can scroll through adjacent slices directly in tile to inspect local structures, allowing fast yet informed validation. If further structural context is needed, the full 3D view opens at the exact spatial synapse instance location, supporting multi-scale exploration and reducing ambiguity during evaluation (G1, G3).

6.2 Error Categorization View

In the *Error Categorization* view (Fig. 5c-d), users can assign predefined error types or define custom labels (T2), directly supporting G1, G3, and G4. This structured stage enforces task separation (R5), enabling users to focus solely on error categorization, and helps maintain annotation transparency and reproducibility (G5). The two-phase workflow of error detection followed by categorization provides a built-in validation loop and enables direct side-by-side comparison of supposedly incorrect instances.

False Negative Discovery. False negatives—i.e., missing synapses—are identified through interactive exploration in Neuroglancer (Fig. 8), which allows users to flag regions where annotations may be absent (G2). When such a region is marked, SynAnno extracts the local volume and creates a corresponding tile view for synaptic annotation. Users can then place directional markers and generate a new synapse instance (T6). This integrated pipeline directly supports tasks involving the discovery and reconstruction of false negatives, promoting structural completeness (G2) and functional accuracy (G3), while maintaining spatial context (R1, R3).

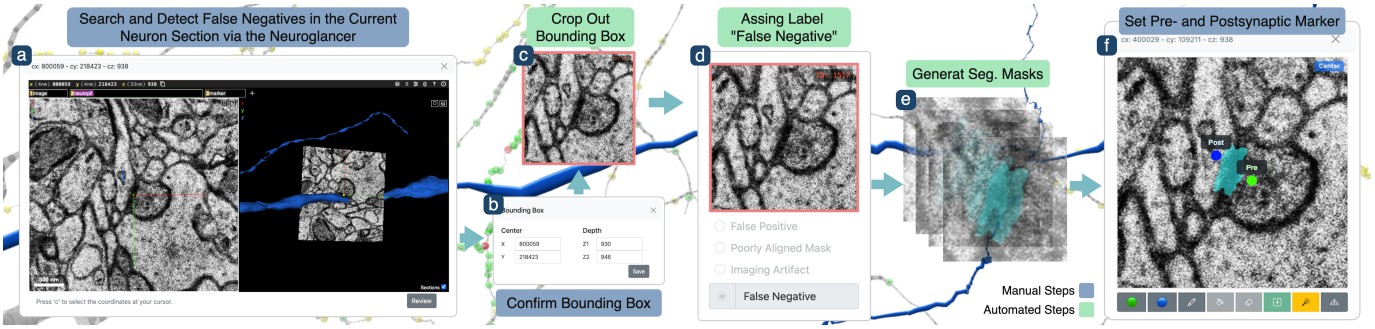

Fig. 7: **False Negative Correction Workflow**. (a) The user navigates along a neuron branch to search for false negatives (FNs) and places a marker at the center of any identified instance. (b) After confirming the bounding box, (c) SynAnno automatically crops the region and (d) labels it as a false negative. (e) In the *Error Correction* view, SynAnno can auto-generate depth-wise synapse masks. (f) The user then sets the markers indicating the direction of the information flow or optionally guides the segmentation process by providing manually drawn masks.

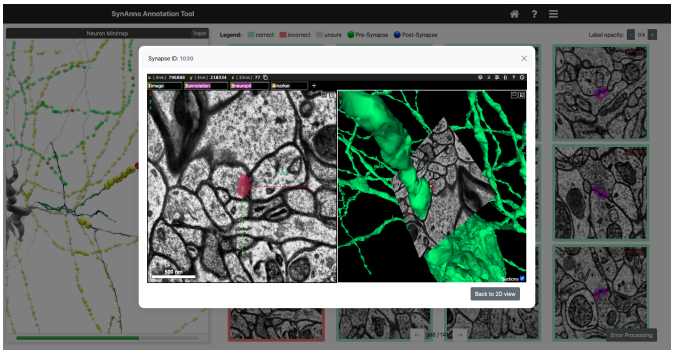

Fig. 8: **Neuroglancer Integration**. In all three main views—*Error Detection*, *Error Categorization*, and *Error Correction* (Fig. 5, left, center, and right, respectively)—users can launch a Neuroglancer view, centered on the currently viewed instance. In the *Error Detection* and *Error Correction* views, this enables users to search for and identify false negatives. User study participants highly valued this integration.

6.3 Error Correction View

The *Error Correction* view (Fig. 5e-f) supports the correction of erroneous or incomplete instances identified during the error categorization flow. This view addresses the need for anatomical fidelity and functional accuracy (G3, G4) by enabling users to revise synapse masks and reassign directional markers (T4).

Marker and Segmentation Correction. SynAnno's error correction interface (Fig. 5f) enables precise editing of pre- and postsynaptic markers (Fig. 1c, right), which can be interactively placed, adjusted, or removed to ensure the correct direction of information flow (T4; G3). Synapse mask correction (T5) is implemented using a hybrid approach that balances automation and manual control (R4, R5). Corrections can be applied via three mechanisms: *(1)* fully automated using a 3D U-Net, *(2)* semi-automated by guiding the model with user-provided masks on selected slices, or *(3)* fully manual through spline drawing on all individual slices. The manual interface allows users to define masks by placing control points that form filled splines in the style of H01 synapse masks (Fig. 1c, center). An integrated eraser tool supports localized refinement. This strategy accommodates a range of user preferences and preserves anatomical precision (G4; R4, R5). The *Error Correction* view also supports adding and segmenting new synapses (Fig. 5e) that were missed during automated prediction (T6).

7 Machine Learning Guided Error Correction

The primary challenge in synapse segmentation lies in accurately identifying each instance and generating an initial 2D mask. Once this first 2D mask was created, annotators must spatially extend the mask to cover the 3D structure of the synapses. This repetitive, slice-by-slice manual annotation compromises efficiency and consistency, particularly when scaling to large datasets. To alleviate this burden, we implement a 3D U-Net [33, 44]-based model designed to fully automate the segmentation or volumetrically extend masks across all slices of an instance. By leveraging deep learning, our approach significantly reduces manual intervention, enabling annotators to focus on higher-level corrections while ensuring consistent volumetric segmentation.

Model Integration. Our 3D U-Net supports users in creating synapse masks for false negatives and instances labeled as *incorrect* in SynAnno's *Error Detection* view and categorized accordingly in SynAnno's error *Error Categorization* view (Fig. 2a–d). Users can employ the model for fully automated instance segmentation or integrate it into an interactive refinement workflow. In this workflow, the user iteratively guides and enhances the model's output. Rather than relying solely on automated predictions, users can intervene by manually drawing masks on selected slices within the *Error Correction* view (Fig. 2e–f). Once a mask is provided, the model propagates the mask depthwise, using the user-defined mask(s) as a seed. The resulting synapse mask can then be reviewed across slices, with manual and auto-generated masks visually distinguished using different colors for clarity (Fig. 1c, left vs. center). If the output is unsatisfactory, users can refine it by selecting additional key slices, drawing new masks, and re-triggering the synapse segmentation. The model incorporates these corrections, iteratively updating its predictions based on user input. This adaptive workflow allows for flexible intervention, ranging from minimal corrections—where the model generates an accurate mask with little to no user input—to fully manual painting if required.

Model Training and Dataset Generation. Our method employs a deep 3D U-Net that takes in a dual-channel 3D volume, one channel for the raw EM image and one for user-defined seed mask, to produce a refined single-channel synapse segmentation. Due to memory constraints, the model was trained on a small subset of the H01 dataset, comprising 300 instances for training and 50 for cross-validation. We applied data augmentation, such as random flips and rotations, to increase training diversity. Despite the small training set, the model demonstrated robust performance across diverse scenarios. To emulate realistic user behavior, each instance undergoes multiple segmentation scenarios. One or more seed segmentation layers are retained, while most target slices are hidden from the model. The number of hidden slices is drawn from a normal distribution, introducing variability in the training data. Testing revealed that including instances with zero mask is essential to ensure reliable mask generation in the absence of any seed input and to prevent the model from overcommitting to seed masks.

The model employs a five-level encoder-decoder architecture (32–256 feature channels) and is trained using a weighted binary cross-entropy loss between predicted masks and ground truth masks. Optimization is performed with the Adam optimizer [20], and model selection is based on validation loss checkpointing. Training proceeds for up to 400 epochs with early stopping (patience of 25) and an initial learning rate of 5×10^{-5}, reduced on plateau by a factor of 0.5. All

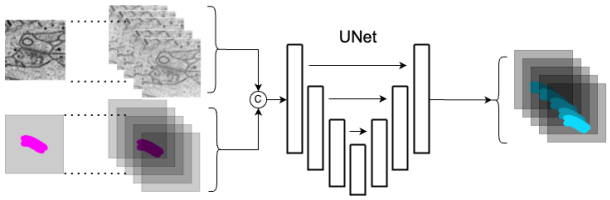

Fig. 9: **U-Net Model for Synapse Mask Prediction.** The electron microscopy (EM) image volume is paired with user-defined seed masks (left, magenta) to form a two-channel input. The corrected masks (right, blue) are auto-generated depthwise around the user's original mask.

experiments were conducted on a SLURM-managed compute node with 1×NVIDIA A10 GPU, 4 CPU cores, and 128 GB of RAM.

8 Data & Implementation

SynAnno relies on five core data sources: (1) raw EM images, (2) synapse masks, (3) neuron masks, (4) neuron skeletons, and (5) metadata such as voxel size. These sources collectively provide the structural context needed for guided proofreading of synaptic connectivity. Neuron segmentation supports the selection and isolation of individual neurons for focused analysis. Skeletons provide a simplified geometric representation for rendering, compartmentalization, and pathfinding, enabling users to navigate and interpret neuronal morphology. Metadata enables systematic synapse-to-skeleton mapping to support circuit-level analysis and allows flexible filtering and processing of synaptic data during proofreading.

Neuron Skeletons. We obtain the neuron skeletons via H01's cloud storage using the `CloudVolume` [36] library. Spatial coordinates were normalized to nanometer units, after which we resolved structural inconsistencies and restored missing parent-child relationships. To facilitate downstream partitioning, we pruned peripheral branches that did not significantly contribute to the neuron's structural or functional interpretation. Upon user selection, SynAnno retrieves the corresponding neuron skeleton and partitions it into structurally coherent segments. (Fig. 2b). Each neuron is modeled as an undirected graph, rooted at the soma when available; otherwise, a central node is estimated using the PageRank algorithm. A depth-first search (DFS) traversal imposes a hierarchical node ordering based on traversal indices. Branch points—nodes with degree ≥ 3—serve as natural mask boundaries. To avoid over—fragmentation, small segments are merged with adjacent ones based on traversal continuity and connectivity constraints. The resulting segments are then ordered to reflect biologically meaningful hierarchies, enabling efficient, compartment-wise proofreading. Skeleton processing, partitioning, and pathfinding were implemented using the `Navis` [34] and `NetworkX` [18] libraries. For the usability studies, case studies, and the interactive demo of SynAnno available at the time of writing, the materialization metadata was limited to synapses associated with a curated subset of 104 fully proofread neurons.

Synapse Data. Synapse masks and metadata were retrieved from the H01 cloud storage and the associated connections database. To establish spatial correspondence between neuronal morphology and synaptic connectivity, synapse coordinates were extracted and converted from voxel space to nanometers, forming a spatial point cloud. The materialization table was filtered to retain only synapses associated with the selected neurons. A KDTree was constructed from the skeleton node coordinates to enable efficient nearest-neighbor queries. Each synapse was mapped to its closest skeleton node, and its corresponding structural section and traversal index were recorded. This mapping yielded a dataset of synapses embedded in their neuronal context.

System. SynAnno is implemented using Python's Flask framework and vanilla JavaScript. The open-source code and a demo are available: `https://github.com/PytorchConnectomics/SynAnno`.

9 Evaluation

To assess the usability, effectiveness, and real-world applicability of *SynAnno*, we conducted a qualitative user study with seven experienced connectome proofreaders, a post-session survey, and a set of in-depth case studies targeting critical annotation and proofreading tasks. We aimed to gather user sentiment and detailed task-specific feedback to inform ongoing system development. The evaluation was designed to address three core questions: **(1)** How well does *SynAnno* support the cognitive workflows of expert annotators? **(2)** Which aspects of the interface and interaction design facilitate or hinder efficient proofreading? **(3)** What improvements could enhance user experience and task performance? We first present the results of the user study and survey, followed by detailed case studies and a summary of updates implemented in direct response to user feedback.

9.1 User Study

We conducted seven semi-structured interviews with domain experts to evaluate the usability and effectiveness of SynAnno. Each session lasted approximately one hour. Following a brief introduction to SynAnno, participants were invited to independently steer the interface and test the workflows based on their expectations and annotation practices. A short follow-up survey was distributed for participants to complete independently after the session.

Survey Results. The survey included eight questions rated on a five-point Likert scale (Strongly Disagree, Disagree, Neutral, Agree, Strongly Agree), covering key aspects such as workflow support, navigation, segmentation, and labeling.

Figure 10 summarizes the aggregated survey responses. Each row represents a specific question (Q1–Q8). Neutral responses are centered around the y-axis, with disagreement extending to the left and agreement to the right. Percentages on the right side of each bar indicate the share of users who agreed with each statement. The three highest-rated items—Q1 (workflow support), Q2 (intuitive labeling), and Q3 (dataset navigation)—each received 100% agreement, highlighting these as key strengths of the tool. Statements on general proofreading efficiency and ML-assisted synapse segmentation (Q4–Q5) received 85% agreement. Slightly lower scores were recorded for Q6 and Q7 (both 71%), which addressed logical neuron traversal and progress tracking, respectively. Q8 (false negative handling) received the lowest score (57%). Feedback on Q4 seems to reflect that some participants, particularly those involved in initial synapse segmentation or region-of-interest identification, were less familiar with workflows for correcting existing masks. Concerning Q5, the discussion focused on when synapse masks are necessary. For workflows focused on annotating information flow direction alone, synapse mask correction was seen as less relevant. However, for tasks such as estimating contact areas, the ML-assisted synapse segmentation was considered highly beneficial. Responses to Q5 and Q6 aligned with suggestions for structural improvements, including synapse-weighted partitioning, manual compartment definition, and hierarchical skeleton representations in the legend. Q8 reflects concerns voiced during interviews regarding reliably identifying false negatives. Participants emphasized the need to traverse entire compartments and noted a cognitive disconnect between proofreading and initial annotation tasks. Participants' qualitative feedbacks are summarized below.

Interface and Visualization. A clear point of consensus was the need for larger, resizable tiles in the *Error Detection* view (Fig. 5, left), along with the dynamic adjustment of tile size and slice count based on dataset properties and available screen space. The integration of Neuroglancer (Fig. 8) was widely praised as a powerful feature for context-rich exploration of challenging instances. One participant suggested enabling direct access to Neuroglancer to streamline context switching. Others, in turn, proposed further enhancing the instance module view to eliminate the need to switch to Neuroglancer entirely. Several usability enhancements were suggested to improve the overall interface, such as keyboard shortcut–based navigation and making neuron branches selectable via a clickable skeleton in the minimap. The contextual visual cues provided by skeleton rendering (Fig. 5a) were frequently cited as valuable for supporting orientation.

Workflow and Task Logic. The most universally praised feature of the tool was its structured, section-based review workflow, as illustrated in Figure 6. Participants consistently reported that this approach supported sustained cognitive focus and enabled systematic traversal of

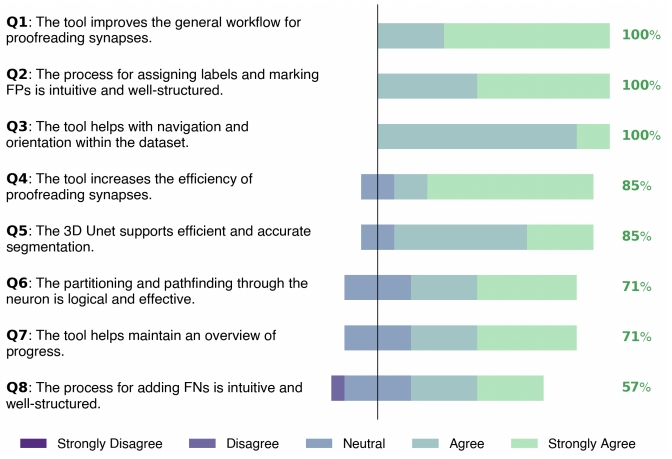

Fig. 10: **User Study Results.** We collected aggregated survey responses from N=7 users. The results show strong satisfaction with core functionalities, including workflow support, intuitive labeling, and navigation (Q1–Q3), each receiving 100% agreement. Moderate agreement was observed for ML-assisted segmentation and proofreading progress tracking (Q4–Q7). The lowest score was given to the intuitiveness of adding false negatives (Q8), indicating a need for improvement.

the neuron. The ability to pause and resume work at the exact location was described as a transformative capability for long-term annotation projects. Participants expressed divergent views on task flow structure. Some preferred immediate labeling after inspecting an instance to avoid the cognitive burden of revisiting items. Others favored separating review from labeling, emphasizing benefits such as self-checking, comparative validation, and more accurate error correction. Progress tracking was consistently highlighted as a key feature. Individual participants requested additional visual aids, including section-level progress bars and markers for reviewed instances in the tile view, to reduce oversight and duplication. Improved partitioning strategies were also discussed. One key recommendation was to balance the number of synapses per section to promote equitable workload distribution. Another suggestion was to allow users to partition neurons manually, offering greater flexibility in managing complex data regions.

Error Handling and Labeling. Participants offered nuanced feedback on error labeling strategies, emphasizing the need for additional predefined labels such as *Missing Mask* and *Merged Mask* to enhance error detection precision and filtering efficiency while reducing reliance on manually created custom labels. A significant request was the ability to define custom error vocabularies on a per-dataset basis to accommodate domain-specific terminology and validation objectives.

One of the most debated topics was including an **"unsure"** label. Several participants regarded this option as essential for managing ambiguous cases, enabling deferred judgment or later comparison in the *Error Categorization* view. They expressed concern that forcing a binary choice between **"correct"** and **"incorrect"** could introduce decision bias. Conversely, others worried that the **"unsure"** label could reduce accountability, encouraging annotators to overuse it to avoid making difficult decisions. The prevailing recommendation was to make the availability of this label configurable, allowing individual labs to tailor its use according to their review policies. The *Error Categorization* view was widely praised for supporting iterative proofreading. However, alternative modes were also proposed, such as a swipe-based interface, which would present one instance at a time. This could be useful when individual instances are largely uncorrelated.

False Negative Detection. False negative (FN) detection emerged as one of the most challenging and contested aspects of the annotation workflow. Participants consistently emphasized that accurately identifying FNs requires systematic, branch-by-branch traversal of the entire neuron. This makes the process not only cognitively demanding but also difficult to integrate seamlessly into standard instance-based review flows. Although the idea of embedding FN detection directly into the main review interface was met with appreciation, it became clear that this approach is quite different from the synapse mask review. Attempting to combine both so closely led to the loss of focus, increased error risk, and a breakdown in the logical flow of work. Therefore, some participants advocated for explicitly separating FN detection into its dedicated workflow phase, allowing users to adopt a different mindset and strategy for this task. This separation would also enable optimizations such as tailored tools for scanning long branches, toggling neuron skeleton visibility, and navigating via section or projection views.

Synapse Masks Correction and Tooling. Users appreciated the ability to directly correct synapse masks within SynAnno, highlighting the marker correction feature—used to indicate information flow direction—for its speed and efficiency. The option to place markers on arbitrary, independent slices was well-received for its flexibility. The U-Net autocomplete feature was a major strength, reducing redundancy and fatigue during review. However, users expressed a need for more manual control, suggesting features like a brush tool for mask painting, adjustable auto-segmentation, and finer control over mask radius.

9.2 Case Study

To evaluate the real-world applicability of SynAnno, we conducted four focused case studies **(C1 - C4)**, each involving a single expert user.

C1 – Detecting Erroneous Synapses. Here, the user reviewed pre-existing synapse masks using the *Error Detection* View (Fig. 5b), labeling each instance as *correct*, *incorrect*, or *unsure*. They noted that the grid-based layout enabled fast and structured progression through the neuron, unlike their usual workflow of manually navigating 3D volumes using spreadsheets. However, the small, fixed tile size was seen as a bottleneck, as it required frequent context switching. Thus, we made the number of tiles per page configurable to accommodate different screen sizes and user preferences. While the user praised the structured flow and the ability to progress quickly through the neuron, they may have overused the unsure" label to delegate complex cases for supposed later review, reinforcing earlier discussions around annotation confidence and reviewer accountability. The user also requested a feature to track which instances had already been reviewed on a page. At the time of testing, 24 instances were shown per page without any additional identifiers; we have since added visible instance IDs in the top-right corner of each tile to address this need.

C2 – Identifying False Positive Synapses. Here, the user was asked to review the synapse masks of five neuron compartments using the *Error Categorization* View (Fig. 5c) to identify false positive synaptic annotations. The task was completed with surprising speed and confidence, although the user relied heavily on Neuroglancer. Compared to a previous dataset of a fly brain, the H01 dataset was perceived as more challenging, with synapses exhibiting less pronounced features. Consequently, the user suggested adding an option to bypass the 3D skeleton view and jump directly into the more detailed 3D Neuroglancer interface, reducing the number of context switches. Overall, the user quickly adapted to the tool and inquired when it would be available in their daily workflow.

C3 – Correcting Synapse Masks. This task involved correcting poorly aligned or incomplete synapse masks using the drawing module and U-Net–assisted auto-completion using the *Error Correction* View (Fig. 5d). The user was asked to correct the masks slice by slice manually. For a second subset, we enabled support for the 3D U-Net, allowing both fully automated and semi-automated synapse segmentation. The U-Net was highly praised for enabling users to select an arbitrary slice as seed inputs. Overall, the tool significantly reduced the repetitiveness of mask drawing. The user observed that drawing a single representative slice could dramatically improve the quality of the synapse mask. However, they requested a tool to make corrections to auto-generated masks, like freeform adjustments.

C4 – Searching and Adding False Negative Synapses. This task combined exploratory traversal with instance creation, including marking the center of false negative (FN) candidates, assigning the markers for the information flow direction, and generating synapse masks. The

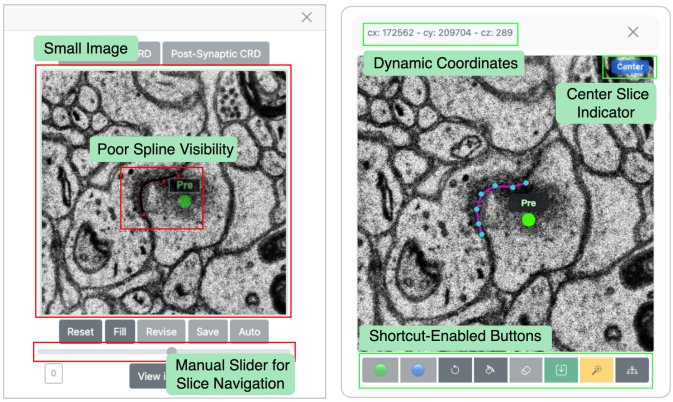

Fig. 11: **Refactored Drawing Module.** Updated drawing module based on user feedback, featuring mouse wheel image slice scrolling, a center slice indicator, dynamic coordinates, keyboard shortcuts, and improved spline highlighting. The figure depicts the initial version to the right and the refactored version to the left.

user navigated to a selected neuron section, identified unannotated synapses, and marked their center points. SynAnno extracted the regions, automatically labeled them as FNs, and enabled rapid annotation. The U-Net auto-completion and marker placement tools were seen as user-friendly and efficient (Fig. 7). Nonetheless, this case study confirmed that false negative detection remains the most cognitively demanding task. Users requested visual aids—such as heatmaps or section-level progress indicators—to help avoid missed regions. The session also sparked a discussion on whether FN detection should be integrated into the proofreading interface or moved to a dedicated view. Overall, FN detection remains the most persistent challenge. That said, the effort to build an end-to-end tool was well received, and the user emphasized that SynAnno should continue to support FN addition.

9.3 Incorporating User Feedback

Based on the feedback gathered from the usability and case studies, we implemented several key updates to SynAnno. One notable enhancement is the revised drawing module, shown in Figure 11. We increased the image size, removed the slider, and introduced mouse wheel-based slice navigation. The slider was initially included to provide precise control over the displayed slice, particularly for accurate seed mask placement. However, we replaced it with wheel-based scrolling to reduce interaction complexity and align the user experience with that of the *Error Detection* and *Error Correction* View modules.

To address orientation concerns, we introduced a tag marking the center slice in the top-left corner. Additionally, slice coordinates are now displayed and dynamically updated in the header during scrolling, helping users maintain spatial awareness. To streamline interactions further, we implemented keyboard shortcuts for essential actions such as closing the window and re-adding pre- or postsynaptic markers. The enlarged image and module layout are designed to decrease the need to open Neuroglancer, while also aligning with user preferences for a more self-contained workflow. In response to user feedback, we also improved the visibility of splines by enhancing their highlighting prior to color filling. Further improvements include the ability to revise annotations directly in the *Error Categorization* view (Fig. 5,d). The number of displayed tiles can now be configured based on dataset complexity, screen size, and user preference. Instance IDs were added to each tile in the *Error Detection* view (Fig. 5, left) to help users better track and distinguish them, showing them directly on the image tiles. Additionally, navigation was improved: users can now jump to specific neuron sections by clicking entries in the metadata table, rather than paging through the interface to reach a particular section.

10 Discussion

User Expertise in Synapse Proofreading. Identifying experts for our user study was a challenge, as large-scale synapse proofreading remains a specialized task within the neuroscience community. While many labs work with connectomics data, relatively few researchers are directly involved in manual proofreading at scale. The availability of high-resolution datasets has grown in recent years, but the process of verifying AI-generated synapse masks remains tedious and time-consuming. By introducing SynAnno, we aim to lower the barrier to entry for connectomics proofreading, making it easier for neuroscientists to efficiently correct errors and improve synapse mask quality. We hope that structured proofreading workflows, such as those implemented in SynAnno, will encourage broader adoption of systematic annotation review across neuroscience labs.

Limitations. While SynAnno improves proofreading efficiency, it has certain constraints. The tool is designed primarily for synaptic annotation and structured proofreading of neuron-specific synapse collections. Currently, it does not support full connectome-wide comparisons of synaptic patterns or large-scale statistical analysis of synapse distributions. Additionally, while our 3D U-Net model aids in synapse mask correction, its performance depends on the quality of the training data. Challenging cases, such as synapses located near dense axonal crossings, may still require substantial manual correction. Another limitation is our instance selection algorithm, which currently does not fully resolve overlapping synapse masks, sometimes requiring users to manually distinguish between adjacent synaptic instances. False negative detection remains a cognitively demanding task, indicating the need for additional visual aids or dedicated interfaces. Addressing these issues will be key to enhancing the tool's usability and effectiveness.

Balancing Automation and User Control. A core design principle of SynAnno is to integrate machine learning into the proofreading workflow while maintaining user oversight. Automated synapse segmentation models can reduce the burden of manual annotation, but full reliance on AI-generated results risks introducing systematic errors. Our approach allows users to interactively validate and correct errors while leveraging AI-assisted guidance. However, this comes with a trade-off: increased automation speeds up proofreading but can reduce user engagement in critical decision-making. Future iterations of SynAnno will aim to refine this balance by providing more adaptive user controls, such as the ability to manually partition neurons and refine synapse segmentation decisions.

11 Conclusion and Future Work

SynAnno introduces an interactive and structured approach to proofreading synaptic annotations, addressing the challenges of large-scale connectomics datasets. By guiding users through a neuron-centric workflow, integrating hierarchical visualization, and leveraging machine learning-assisted error correction, SynAnno enhances the accuracy and efficiency of synaptic annotation. The tool provides a scalable solution for validating AI-generated masks, ensuring that neuronal connectivity reconstructions remain biologically accurate. While designed for synaptic annotation proofreading, its structured workflow and interactive correction mechanisms could be extended to other neuroanatomical annotation tasks that require human validation. Looking ahead, we plan to expand SynAnno 's capabilities to better support evolving connectomics research. A key focus is to both expand the training of our 3D U-Net model and implement an online training strategy that enables dynamic adaptation to challenging segmentation cases by directly incorporating manually corrected instances from the *Error Correction* view. Additionally, we aim to extend SynAnno 's applicability to other neuronal structures, further broadening its impact in neuroscience. Enhancing user-driven neuron partitioning will provide greater flexibility in handling complex neuronal architectures, while improvements to our instance selection algorithm will refine segmentation handling, reducing overlap and ensuring clear, distinct annotations. As neuroscience datasets continue to grow in scale and complexity, SynAnno will evolve to meet the increasing demand for precise, efficient proofreading tools, enabling more accurate and reliable connectome reconstructions.

References

[1] A. K. Ai-Awami, J. Beyer, D. Haehn, N. Kasthuri, J. W. Lichtman, H. Pfister, and M. Hadwiger. Neuroblocks–visual tracking of segmentation and proofreading for large connectomics projects. *IEEE transactions on visualization and computer graphics*, 22(1):738–746, 2015. Publisher: IEEE. 2

[2] D. R. Berger, H. S. Seung, and J. W. Lichtman. VAST (volume annotation and segmentation tool): efficient manual and semi-automatic labeling of large 3D image stacks. *Frontiers in neural circuits*, 12:88, 2018. Publisher: Frontiers Media SA. 2

[3] J. Berman, D. B. Chklovskii, and J. Wu. Bridging the Gap: Point Clouds for Merging Neurons in Connectomics. In *MIDL*, 2022. 2

[4] J. Beyer, A. Al-Awami, N. Kasthuri, J. W. Lichtman, H. Pfister, and M. Hadwiger. ConnectomeExplorer: Query-Guided Visual Analysis of Large Volumetric Neuroscience Data. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 19(12):2868–2877, 2013. doi: 10.1109/TVCG.2013.142 3

[5] J. Beyer, A. Al-Awami, N. Kasthuri, J. W. Lichtman, H. Pfister, and M. Hadwiger. NeuroLines: A Subway Map Metaphor for Visualizing Nanoscale Neuronal Connectivity. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 20(12):2369–2378, 2014. doi: 10.1109/TVCG.2014.2346312 3

[6] J. Beyer, J. Troidl, S. Boorboor, M. Hadwiger, A. Kaufman, and H. Pfister. A survey of visualization and analysis in high-resolution connectomics. In *Computer Graphics Forum*, vol. 41, pp. 573–607. Wiley Online Library, 2022. Issue: 3. 2

[7] J. Buhmann, A. Sheridan, C. Malin-Mayor, P. Schlegel, S. Gerhard, T. Kazimiers, R. Krause, T. M. Nguyen, L. Heinrich, W.-C. A. Lee, and others. Automatic detection of synaptic partners in a whole-brain Drosophila electron microscopy data set. *Nature methods*, 18(7):771–774, 2021. 2

[8] D. B. Chklovskii, S. Vitaladevuni, and L. K. Scheffer. Semi-automated reconstruction of neural circuits using electron microscopy. *Current Opinion in Neurobiology*, 20(5):667–675, 2010. doi: 10.1016/j.conb.2010.08.002 2

[9] S. Dorkenwald, A. Matsliah, A. R. Sterling, and others. Neuronal wiring diagram of an adult brain. *Nature*, 634(8032):124–138, 2024. 1, 2

[10] S. Dorkenwald, C. E. McKellar, T. Macrina, and others. FlyWire: online community for whole-brain connectomics. *Nature methods*, 19(1):119–128, 2022. Publisher: Nature Publishing Group US New York. 1, 2

[11] S. Dorkenwald, C. M. Schneider-Mizell, D. Brittain, A. Halageri, C. Jordan, N. Kemnitz, M. A. Castro, W. Silversmith, J. Maitin-Shephard, J. Troidl, and others. CAVE: Connectome annotation versioning engine. *bioRxiv*, 2023. 1

[12] S. Dorkenwald, N. L. Turner, T. Macrina, K. Lee, R. Lu, J. Wu, A. L. Bodor, A. A. Bleckert, D. Brittain, N. Kemnitz, and others. Binary and analog variation of synapses between cortical pyramidal neurons. *Elife*, 11:e76120, 2022. 2

[13] N. Eckstein, A. S. Bates, A. Champion, M. Du, Y. Yin, P. Schlegel, A. K.-Y. Lu, T. Rymer, S. Finley-May, T. Paterson, and others. Neurotransmitter classification from electron microscopy images at synaptic sites in Drosophila melanogaster. *Cell*, 187(10):2574–2594, 2024. 2

[14] F. Gonda, X. Wang, J. Beyer, M. Hadwiger, J. W. Lichtman, and H. Pfister. VICE: Visual Identification and Correction of Neural Circuit Errors. In *IEEE Visualization Conference (VIS)*, pp. 447–458, 2021. doi: 10.1111/cgf.14320 2

[15] D. Haehn, J. Hoffer, B. Matejek, A. Suissa-Peleg, A. K. Al-Awami, L. Kamentsky, F. Gonda, E. Meng, W. Zhang, R. Schalek, and others. Scalable interactive visualization for connectomics. In *Informatics*, vol. 4, p. 29. MDPI, 2017. Issue: 3. 2

[16] D. Haehn, V. Kaynig, J. Tompkin, J. W. Lichtman, and H. Pfister. Guided proofreading of automatic segmentations for connectomics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 9319–9328, 2018. 2

[17] D. Haehn, S. Knowles-Barley, M. Roberts, J. Beyer, N. Kasthuri, J. W. Lichtman, and H. Pfister. Design and evaluation of interactive proofreading tools for connectomics. *IEEE transactions on visualization and computer graphics*, 20(12):2466–2475, 2014. Publisher: IEEE. 2

[18] A. A. Hagberg, D. A. Schult, and P. J. Swart. Exploring Network Structure, Dynamics, and Function using NetworkX. In G. Varoquaux, T. Vaught, and J. Millman, eds., *Proceedings of the 7th Python in Science Conference*, pp. 11 – 15. Pasadena, CA USA, 2008. 7

[19] G. Jefferis, L. Collinson, C. Bosch, M. Costa, and P. Schlegel. Scaling up Connectomics: the road to a whole mouse brain connectome. *Wellcome*, 2023. 1

[20] D. P. Kingma and J. Ba. Adam: A Method for Stochastic Optimization, Jan. 2017. arXiv:1412.6980 [cs]. doi: 10.48550/arXiv.1412.6980 6

[21] A. Lin, R. Yang, S. Dorkenwald, A. Matsliah, A. R. Sterling, P. Schlegel, S.-c. Yu, C. E. McKellar, M. Costa, K. Eichler, and others. Network statistics of the whole-brain connectome of Drosophila. *Nature*, 634(8032):153–165, 2024. 2

[22] Z. Lin, D. Wei, W.-D. Jang, S. Zhou, X. Chen, X. Wang, R. Schalek, D. Berger, B. Metejek, L. Kamentsky, and others. Two-Stream Active Query Suggestion for Large-Scale Object Detection in Connectomics. In *European Conference Computer Vision (ECCV)*, 2020. 2

[23] J. Maitin-Shepard, A. Baden, W. Silversmith, E. Perlman, F. Collman, T. Blakely, J. Funke, C. Jordan, B. Falk, N. Kemnitz, Tingzhao, C. Roat, M. Castro, Sridhar Jagannathan, Moenigin, J. Clements, A. Hoag, B. Katz, D. Parsons, Jingpeng Wu, L. Kamentsky, P. Chervakov, P. Hubbard, S. Berg, J. Hoffer, Akhilesh Halageri, C. Machacek, K. Mader, L. Roeder, and P. H. Li. google/neuroglancer:, Oct. 2021. doi: 10.5281/ZENODO.5573293 4

[24] J. Maitin-Shepard and others. Neuroglancer: Web-Based Visualization of Large-Scale 3D Electron Microscopy Data. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2017. 2

[25] B. Matejek, D. Haehn, H. Zhu, D. Wei, T. Parag, and H. Pfister. Biologically-constrained graphs for global connectomics reconstruction. In *CVPR*, 2019. 2

[26] Y. Mehta, D. Tyulmankov, A. Rajagopalan, G. Turner, J. Fitzgerald, and J. Funke. Model based inference of synaptic plasticity rules. *Advances in Neural Information Processing Systems*, 37:48519–48540, 2024. 2

[27] A. Nern, F. Loesche, S.-y. Takemura, and others. Connectome-driven neural inventory of a complete visual system. *Nature*, Mar. 2025. doi: 10.1038/s41586-025-08746-0 1

[28] N. Ofer, D. R. Berger, N. Kasthuri, J. W. Lichtman, and R. Yuste. Ultrastructural analysis of dendritic spine necks reveals a continuum of spine morphologies. *Developmental neurobiology*, 81(5):746–757, 2021. 2

[29] T. Parag, D. Berger, L. Kamentsky, B. Staffler, D. Wei, M. Helmstaedter, J. W. Lichtman, and H. Pfister. Detecting synapse location and connectivity by signed proximity estimation and pruning with deep nets. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pp. 0–0, 2018. 2

[30] S. M. Plaza. Focused proofreading to reconstruct neural connectomes from EM images at scale. In *International Workshop on Deep Learning in Medical Image Analysis*, pp. 249–258. Springer, 2016. 2

[31] S. M. Plaza, J. Clements, T. Dolafi, L. Umayam, N. N. Neubarth, L. K. Scheffer, and S. Berg. neuPrint: An open access tool for EM connectomics. *Frontiers in Neuroinformatics*, 16:896292, July 2022. doi: 10.3389/fninf.2022.896292 2

[32] S. M. Plaza, T. Parag, G. B. Huang, D. J. Olbris, M. A. Saunders, and P. K. Rivlin. Annotating synapses in large EM datasets. *arXiv preprint arXiv:1409.1801*, 2014. 2

[33] O. Ronneberger, P. Fischer, and T. Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. In N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, eds., *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, vol. 9351, pp. 234–241. Springer International Publishing, Cham, 2015. Series Title: Lecture Notes in Computer Science. doi: 10.1007/978-3-319-24574-4_28 6

[34] P. Schlegel, C. Barnes, Sridhar Jagannathan, B. Pedigo, and R. Court. navis-org/navis: Version 1.1.0, Nov. 2021. doi: 10.5281/ZENODO.5710143 7

[35] A. Shapson-Coe, M. Januszewski, D. R. Berger, A. Pope, Y. Wu, T. Blakely, R. L. Schalek, P. H. Li, S. Wang, J. Maitin-Shepard, and others. A petavoxel fragment of human cerebral cortex reconstructed at nanoscale resolution. *Science*, 384(6696):eadk4858, 2024. Publisher: American Association for the Advancement of Science. 1, 3

[36] W. Silversmith, A. Zlateski, J. A. Bae, I. Tartavull, N. Kemnitz, J. Wu, and H. S. Seung. Igneous: Distributed dense 3D segmentation meshing, neuron skeletonization, and hierarchical downsampling. *Frontiers in Neural Circuits*, 16:977700, Nov. 2022. doi: 10.3389/fncir.2022.977700 7

[37] D. Stutz and A. Geiger. Learning 3d shape completion from laser scan data with weak supervision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1955–1964, 2018. 2

[38] The MICrONS Consortium, J. A. Bae, M. Baptiste, and others. Functional connectomics spanning multiple areas of mouse visual cortex, July 2021.

doi: 10.1101/2021.07.28.454025 1

[39] J. Troidl, C. Cali, E. Gröller, H. Pfister, M. Hadwiger, and J. Beyer. Barrio: customizable spatial neighborhood analysis and comparison for nanoscale brain structures. In *Computer Graphics Forum*, vol. 41, pp. 183–194. Wiley Online Library, 2022. Issue: 3. 2

[40] J. Troidl, S. Warchol, J. Choi, J. Matelsky, N. Dhanyasi, X. Wang, B. Wester, D. Wei, J. W. Lichtman, H. Pfister, and others. Vimo-Visual analysis of neuronal connectivity motifs. *IEEE transactions on visualization and computer graphics*, 30(1):748–758, 2023. Publisher: IEEE. 2

[41] C. Weaver, C. Bruns, and M. Helvensteijn. SharkViewer 1.1, Aug. 2014. doi: 10.5281/ZENODO.11290 4

[42] T. Zhao, D. J. Olbris, Y. Yu, and S. M. Plaza. NeuTu: software for collaborative, large-scale, segmentation-based connectome reconstruction. *Frontiers in Neural Circuits*, 12:101, 2018. Publisher: Frontiers Media SA. 2

[43] J. Zung, I. Tartavull, K. Lee, and H. S. Seung. An error detection and correction framework for connectomics. *Advances in neural information processing systems*, 30, 2017. 2

[44] Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger. 3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation. *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pp. 424–432, 2016. doi: 10.1007/978-3-319-46723-8_49 6