Deep learning-based synaptic event detection

Philipp S. O'Neill[1,2], Martín Baccino-Calace[1], Peter Rupprecht[2,3], Rainer W. Friedrich[4,5], Martin Müller[1,2,6], and Igor Delvendahl[1,2,*]

[1]Department of Molecular Life Sciences, University of Zurich (UZH), 8057 Zurich, Switzerland
[2]Neuroscience Center Zurich, 8057 Zurich, Switzerland
[3]Brain Research Institute, University of Zurich, 8057 Zurich, Switzerland
[4]Friedrich Miescher Institute for Biomedical Research, 4058 Basel, Switzerland
[5]Faculty of Natural Sciences, University of Basel, 4003 Basel, Switzerland
[6]University Research Priority Program (URPP), Adaptive Brain Circuits in Development and Learning (AdaBD), University of Zurich, 8057 Zurich, Switzerland
[*]Corresponding author: igor.delvendahl@uzh.ch

ABSTRACT

Quantitative information about synaptic transmission is key to our understanding of neural function. Spontaneous synaptic events carry important information about synaptic efficacy and plasticity. However, due to their stochastic nature and low signal-to-noise ratio, reliable and consistent detection of these events in neurophysiological data remains highly challenging. Here, we present miniML, a novel method for the accurate detection of spontaneous synaptic events based on deep learning. Using simulated ground-truth data, we demonstrate that miniML outperforms commonly used methods in terms of precision and recall over different signal-to-noise conditions. The event detection method generalizes easily to diverse synaptic preparations and different types of data. miniML provides a powerful and easy-to-use deep learning framework for automated, standardized and precise analysis of synaptic events in any cell, thus opening new avenues for in-depth investigations into the synaptic basis of neural function and dysfunction.

Keywords: Synaptic transmission, Machine Learning, Data analysis, Neurons, Electrophysiology, Imaging

INTRODUCTION

Synaptic communication serves as the fundamental basis for a wide spectrum of brain functions, from computation and sensory integration to learning and memory. Synaptic transmission either arises from spontaneous or action potential-evoked fusion of neurotransmitter-filled synaptic vesicles[1] resulting in an electrical response in the postsynaptic cell. Such synaptic events can be recorded using electrophysiological or imaging techniques and are an essential feature of all neural circuits.

Spontaneous fusion of single vesicles ('miniature events') may be caused by random fluctuations in the release machinery or intracellular Ca^{2+} concentration[2], and play an important role in synaptic development and stability[1–4]. Measurements of amplitude, kinetics, and timing of these events provide essential information about the function of individual synapses and neural circuits and are thus key to our understanding of fundamental processes, such as synaptic plasticity or synaptic computation that support neural function[5,6]. For example, amplitude changes of miniature events are a proxy of neurotransmitter receptor modulation, which is thought to be the predominant mechanism driving activity-dependent, long-term alterations in synaptic strength[7,8] and homeostatic synaptic plasticity[9,10]. Synaptic events may also result from presynaptic action potentials in neural networks. Thus, the quantification of synaptic events is paramount for studying synaptic function and for understanding neural coding.

However, the detection and quantification of synaptic events remains a major challenge. Synaptic events are often small in size, usually close to, or even below, the noise level typically encountered in electrophysiological or fluorescence recording, resulting in a low signal-to-noise ratio (SNR). The stochastic nature of their occurrence further complicates their reliable detection and evaluation.

Several techniques have been developed for synaptic event detection. These can be broadly divided into three general approaches: (i) Finite-difference approaches use crossings of a predefined threshold to identify events in the data, typically using either raw data[11], baseline-normalized data[12], or their derivatives[13]. (ii) Template-based methods use a predefined template to detect events and generate a matched filter via a scaling factor[14,15], deconvolution[16] or an optimal filtering approach[17,18]. (iii) Bayesian inference can also be used to detect synaptic events[19]. These techniques have significantly advanced the automated analysis of synaptic events. However, they also have several limitations, such as the typically strong dependence of detection performance on a threshold setting or the common need for visual inspection of the results by an experienced investigator to avoid false positives. In addition, the user has to specify several detection and evaluation parameters that have to be adjusted according to the data—sometimes on a per-recording basis—, which complicates the reproducibility of the results obtained. The widespread use of synaptic event recordings and the difficulty in obtaining results that are reliable across investigators and laboratories highlight the need for an automated, easy-to-use, and reproducible method for synaptic event detection.

Advances in recent years have enabled the use of deep learning[20] for scientific data analysis, with diverse use cases and high efficacy[21]. Convolutional neural networks (CNNs) are particularly effective for image classification and can also be used for one-dimensional data[22,23]. CNNs have been successfully applied in neuroscience to segment brain regions[24], detect synaptic vesicles in electron microscopy images[25], identify spikes in Ca^{2+} imaging data[26], and localize neurons in brain slices[27], or neurons with fluorescence signals in time-series data[28,29]. Machine learning, especially deep learning, can thus significantly improve biological data analysis and contribute to a better understanding of neural function.

Here, we present a novel method for detecting and analyzing spontaneous synaptic events in neurophysiological data using a deep neural network and supervised learning. The method is called *miniML* because its main application is the detection of miniature synaptic events. The miniML model is an end-to-end classifier trained on a dataset of annotated synaptic events. When applied to time-series data, miniML provides high performance event detection with negligible false-positive rates. The method is fast, virtually parameter-free, threshold-independent, and generalizable, allowing for efficient and unbiased analysis of synaptic events in different data types. miniML outperforms classical event detection methods in terms of precision and recall, while also exhibiting high reproducibility. We anticipate that our deep learning-based approach to synaptic event analysis will improve data quality and reproducibility in various fields of neuroscience and lead to new insights into synaptic function, plasticity and computation.

RESULTS

Model training

To improve the detection of stochastic synaptic events in inherently noisy electrophysiology data, we developed a novel deep learning-based approach. We designed a recurrent artificial neural network architecture consisting of convolutional layers (Convolutional Neural Network, CNN), long short-term memory (LSTM), and fully connected dense layers[30]. This CNN-LSTM model was trained to classify short segments of univariate electrophysiological time-series data as either positive or negative for a synaptic event using supervised learning. The trained classifier can then be applied to unseen time-series data to localize events (Fig. 1a). The miniML model takes a section of an electrophysiological recording as input and outputs a label for that section of data (Fig. 1b).

The CNN-LSTM network was trained using voltage-clamp recordings of cerebellar mossy fiber to granule cell (MF–GC) miniature excitatory postsynaptic currents (mEPSCs) from a previous publication[31]. We extracted a large number of synaptic events and corresponding event-free sections from the recordings. All samples were visually inspected and labeled to generate the training dataset. We applied data augmentation techniques to include examples of typical false positives in the training data (Methods). In total, the training data comprised ~30,000 samples. We empirically fine-tuned the model's hyperparameters prior to running full training. The data were split into training and validation sets (.75/.25). Across training epochs, loss decreased and accuracy increased, stabilizing after ~30 epochs (Fig. 1c). The model with the highest validation accuracy was selected for further use, achieving 98.4% (SD 0.09, fivefold cross-validation). A receiver operating characteristic (ROC) curve demonstrated excellent classification performance for miniML, with an area under the curve close to 1 (Fig. 1d). Thus, the trained deep learning model faithfully discriminates the training data according to the labels (Fig. 1e, Supplementary Fig. 1).Deep learning

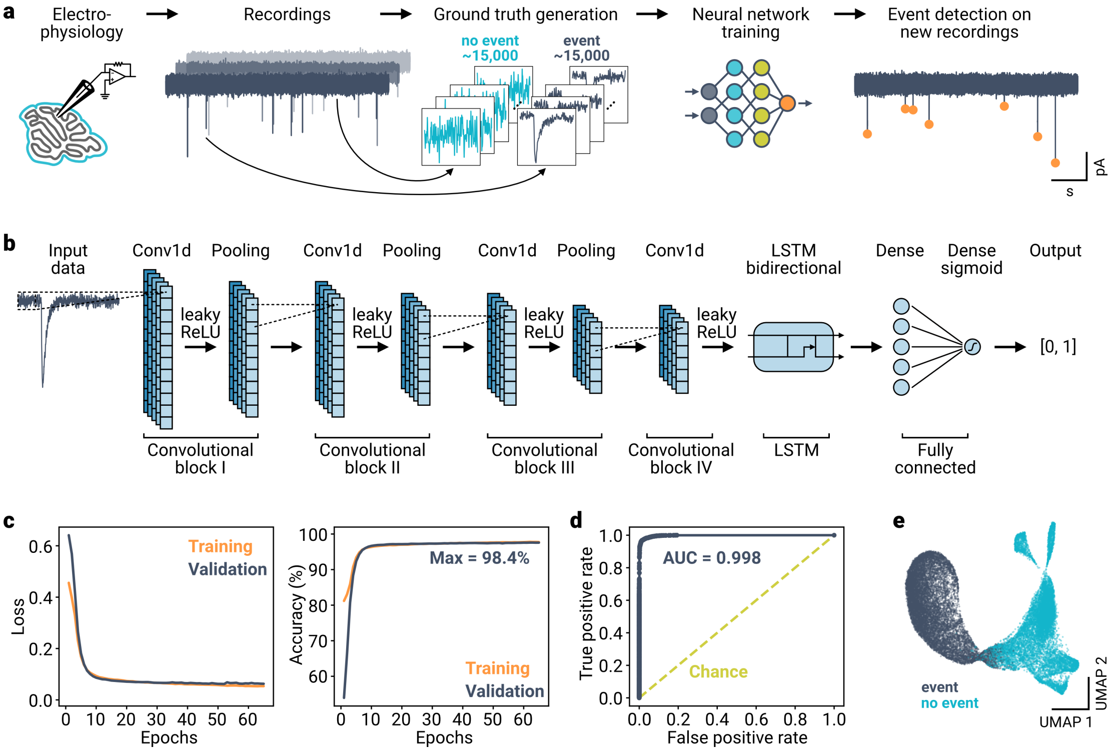


Fig. 1. High performance classification of synaptic events using a deep neural network.
a, Overview of the analysis workflow. Data segments from electrophysiological recordings are extracted and labeled for training an artificial neural network. The resulting model can then be applied to detect events in novel time-series data. **b**, Schematic of the deep learning model design. Data is input to a convolutional network consisting of blocks of 1D convolutional, ReLU, and average pooling layers. The output of the convolutional layers is processed by a bidirectional LSTM block, followed by two fully connected layers. The final output is a label in the interval [0, 1]. **c**, Loss (binary crossentropy) and accuracy of the model over training epochs for training and validation data. **d**, Receiver operating characteristic of the best performing model. Area under the curve (AUC) is indicated; dashed line indicates performance of a random classifier. **e**, UMAP representation of the training data as input to the final layer of the model, indicating linear separability of the two event classes after model training. Also see Supplementary Fig. 1.

typically requires large datasets for training[32]. To investigate how miniML's classification performance depended on the dataset size, we systematically increased the number of training samples. As expected, the accuracy increased with larger datasets. However, the performance gain was marginal (<0.2%) when exceeding 5,000 samples (Supplementary Fig. 2), indicating that relatively small datasets suffice for effective model training[33]. Furthermore, transfer learning with a pretrained model can further reduce the training data size (see below). These results demonstrate the efficacy of deep learning in accurately classifying segments of neurophysiological data containing spontaneous synaptic events.

Using a deep-learning classifier to detect synaptic events

Synaptic events are typically recorded as continuous time-series data of either membrane voltage or current. To apply the trained classifier to detect events in arbitrarily long data, we used a sliding window approach (Fig. 2a). The continuous time-series data are divided into sections corresponding to the input shape of the CNN-LSTM classifier, using a given stride (Fig. 2a). The use of strides reduces the number of inferences needed and speeds up computation time while maintaining high detection performance (Supplementary Fig. 3). By reshaping the data into overlapping sections, model inference can be run in batches and parallel processing techniques, including graphics processing unit (GPU) computing, can be used, resulting in analysis times of a few seconds for minute-long recordings (Supplementary Fig. 3). The miniML detection method is thus time-efficient and can be easily integrated into data analysis pipelines.

The model output predicts the label—no event or event—for each time step (i.e., stride) with a numerical value ranging from zero to one (Fig. 1a). These output values can be interpreted as the confidence that the corresponding data segment contains a synaptic event. With this approach, the deep learning model inference outputs a time-series prediction trace that has a slightly lower temporal resolution compared to the original input. Applying a threshold to the prediction trace can then be used to extract data segments with individual events from a recording. While 0.5 is a reasonable default threshold, ROC analysis (Fig. 1d) can optimize the threshold for a given model. Crucially, the exact choice of threshold is not critical to detection performance (see below). To quantify events, the extracted sections are checked for overlapping events, and all detected events are aligned by the steepest slope. Individual event statistics can then be obtained using standard methods. Fig. 2b illustrates the sliding window approach to detect spontaneous events in time-series data, such as a continuous voltage-clamp recording of spontaneous mEPSCs in a cerebellar GC. miniML provided a clear peak for all synaptic events present, without false positives (Fig. 2c, Supplementary Fig. 4), allowing fast and reliable event quantification (Fig. 2d). These data demonstrate that a deep learning classifier can be applied to synaptic event detection in electrophysiological time-series data.

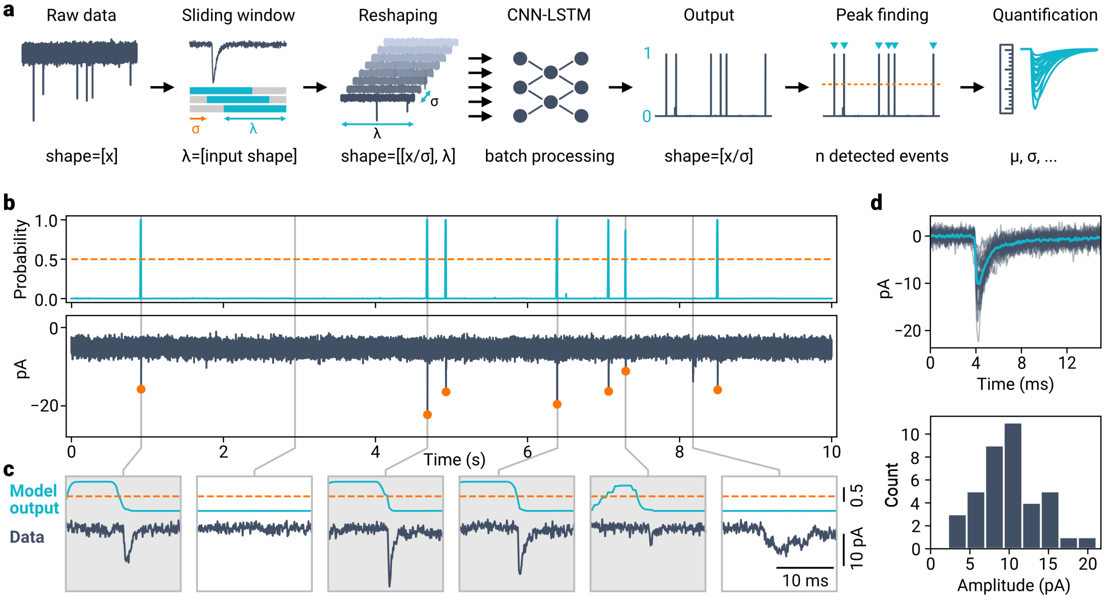


Fig. 2. Application of a deep learning classifier to detect synaptic events in time-series data.
a, Event detection workflow using a deep learning classifier. Time-series data are reshaped with a window size corresponding to length of the training data and a given stride. Peak detection of the model output allows event localization (orange dashed line indicates peak threshold) and subsequent quantification. **b**, Example of event detection in a voltage-clamp recording from a cerebellar granule cell. Top: Prediction trace (top) and corresponding raw data with synaptic events (bottom). Dashed line indicates threshold of 0.5, orange circles indicate event localizations. **c**, Zoom in for different data segments. Detected events are highlighted by gray boxes. **d**, Event analysis for the cell shown in **b** (total recording time, 120 s). Top: Detected events with average (light blue). Bottom: Event amplitude histogram.

Benchmarking of synaptic event detection methods

To benchmark miniML's event detection performance, we compared it with commonly used template-based approaches (template-matching[15] and deconvolution[16]), and a finite-threshold-based method[12]. Some of these—or similar—algorithms are implemented in proprietary software solutions for recording and analyzing of electrophysiological data. We also included a recently developed Bayesian event detection approach[19] and the automated event detection routine of MiniAnalysis software (Synaptosoft Inc.). Although no longer commercially available, MiniAnalysis is still being used in a number of recent publications. To quantify and compare the performance of synaptic event detection methods, we developed a standardized benchmarking pipeline (Fig. 3a). We first performed event-free voltage-clamp recordings from mouse cerebellar GCs (in the presence of blockers of inhibitory and excitatory transmission; see Methods and Supplementary Fig. 4). Events with a two-exponential time course were then generated and superimposed on the raw recording to produce ground-truth data (Figure 3b–c). Event amplitudes were drawn from a log-normal distribution (Supplementary Fig. 5) with varying means to cover a range of SNRs typically observed in recordings of miniature events (2–15 dB, data from n = 170 GC recordings, Fig. 3d). We generated events with kinetics that closely matched the template used for the matched-filtering methods, thus ensuring a conservative comparison of miniML with other methods applied under optimal conditions (i.e., using the exact average event shape as template). To quantify detection performance, we calculated recall (i.e., sensitivity) and precision (proportion of correct identifications). Recall depended on SNR for all methods, with deconvolution and miniML showing the highest values (Fig. 3e). The precision was highest for miniML, which detected no false positives at any SNR, in contrast to all other methods (Fig. 3f). In addition, miniML showed superior results when changing event kinetics, indicating higher robustness to variable event shapes (Fig. 3h and Supplementary Fig. 5), which may be particularly important in neurons with diverse synaptic inputs due to mixed receptor populations[34], or during pharmacological experiments[35].

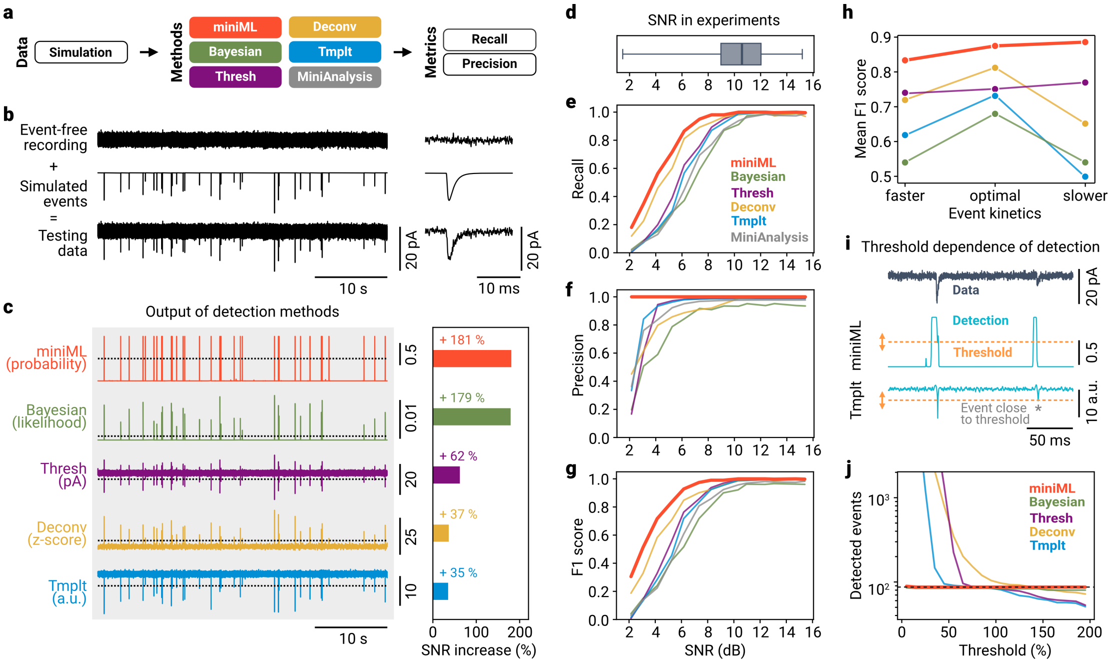


Fig. 3. Deep-learning-based event detection is superior to previous methods.
a, Scheme of event detection benchmarking. Six methods are compared using precision and recall metrics. **b**, Event-free recordings were superimposed with generated events to create ground-truth data. Depicted example data have a signal-to-noise ratio (SNR) of 9 dB. **c**, Left: Output of the detection methods for data in **b**. Right: Improvement in SNR relative to the data. Note that MiniAnalysis is omitted from this analysis because the software does not provide output trace data. **d**, SNR from mEPSC recordings at MF–GC synapses (n = 170, whiskers cover full range of data). **e–g**, Recall, precision, and F1 score versus SNR for the six different methods. Data are averages of three independent runs for each SNR. **h**, Average F1 score versus event kinetics. Detection methods relying on an event template (template-matching, deconvolution and Bayesian) are not very robust to changes in event kinetics. **i**, Evaluating the threshold dependence of detection methods. Asterisk marks event close to detection threshold. **j**, Number of detected events vs. threshold (in % of default threshold value, range 5–195) for different methods. Dashed line indicates true event number.

Conventional methods for detecting events in time-series data typically produce a detection trace with a shape identical to the input data and values in an arbitrary range (Fig. 3c). In contrast, machine learning-based miniML generates output in the interval [0, 1], which can be interpreted as the confidence of event occurrence. The output of event detection methods (i.e., the detection trace) significantly increases the SNR with respect to the original data. In our benchmark scenario, the Bayesian method and miniML provided the greatest discrimination from background noise (Fig. 3c). Notably, miniML's detection trace peaks did not depend on event amplitudes, setting it apart from other detection methods (Supplementary Fig. 5), where the output may contain explicit or implicit event amplitude information. To locate the actual event positions, a threshold must be applied to the detection trace, which can drastically affect the results, as peaks in the detection trace may depend on the event amplitude. For template-based methods, recommendations are provided on threshold selection[15,16], but users may need to adjust this parameter according to their specific data. The choice of the threshold strongly influences detection performance, as even small changes can lead to marked increases in false positives or false negatives. To investigate the threshold dependence of different methods, we systematically varied the threshold and analyzed the number of detected events and the F1 score. Notably, miniML's detection performance remained consistent over a wide range (5–195%, Fig. 3i–j and Supplementary Fig. 5), with false positives only occurring at the lower threshold limit (5%,corresponding to a cutoff value of 0.025 in the detection trace). Conversely, most other detection methods were very sensitive to threshold changes (Fig. 3j and Supplementary Fig. 5) due to the comparatively low SNR ratio of the output traces (but note that the Bayesian method is only slightly threshold dependent). In addition, with output data in the interval [0, 1], the miniML threshold value is bounded with an intuitive meaning. For other detection methods, thresholds are unbounded and usually relative to some data property, such as standard deviation. These comparisons underscore that miniML requires no prior knowledge of the exact event shape and is largely threshold independent, thus allowing reliable event detection.

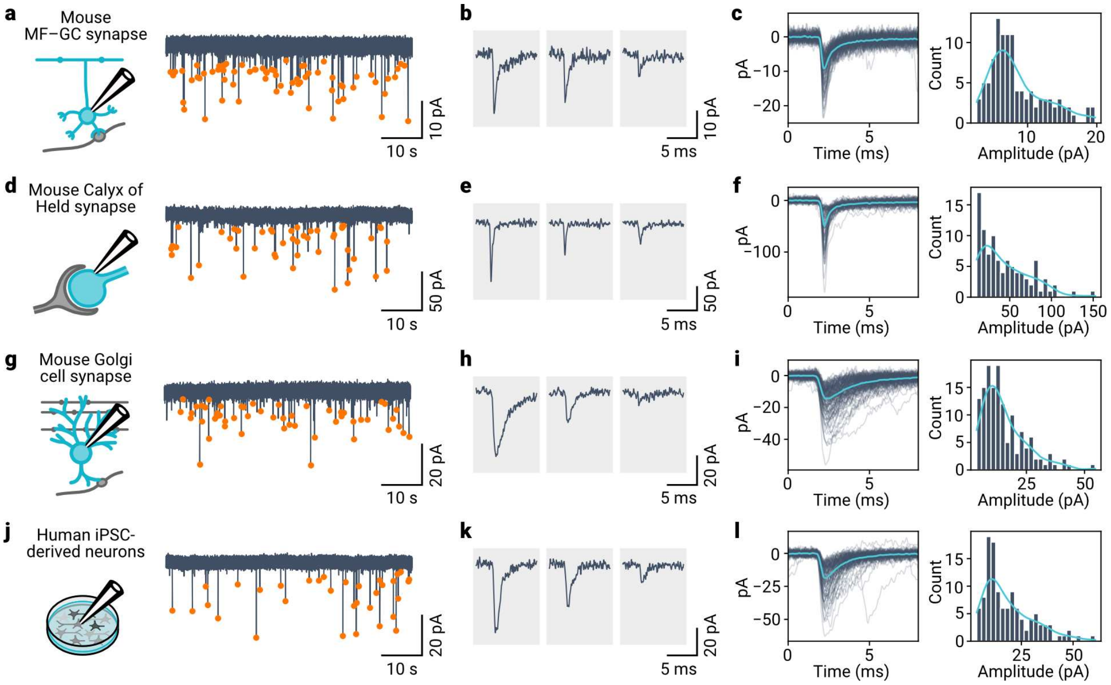


Fig. 4. Application of miniML to diverse synaptic preparations.
a, Schematic and example voltage-clamp recordings from mouse cerebellar GCs. Orange circles mark detected events. **b**, Representative individual mEPSC events detected by miniML. **c**, Left: All detected events from **a**, aligned and overlaid with average (light blue). Right: Event amplitude histogram. **d–f**, Same as in **a–c**, but for recordings from mouse Calyx of Held synapses. **g–i**, Same as in **a–c**, but for recordings from mouse cerebellar Golgi cells. **j–l**, Same as in **a–c**, but for recordings from cultured human induced pluripotent stem cell (iPSC)-derived neurons. miniML consistently retrieves spontaneous events in all four preparations.

miniML reliably detects spontaneous synaptic events in out-of-sample data

We trained miniML using data from cerebellar MF–GC synapses, which enabled reliable analysis of mEPSC recordings from this preparation (Fig. 2, Fig. 4a). However, synaptic event properties can vary considerably between preparations, for example in terms of event frequency, waveform and SNR. To test the detection performance on out-of-sample data, we recorded and analyzed data from the mouse Calyx of Held synapse, a large axosomatic synapse in the auditory brainstem that relays rate-coded information over a large bandwidth and with high temporal precision. Due to the large number of active zones, synaptic events are quite frequent. Despite being trained on cerebellar MF–GC data, miniML reliably detected mEPSCs in these recordings (Fig. 4d–f). This result may be facilitated by the fast kinetics of mEPSCs at the Calyx of Held, which approach the kinetics of MF–GC mEPSCs. To test whether miniML could also detect slightly slower events, we applied it to recordings from cerebellar Golgi cells (Fig. 4g)[36]. These interneurons in the cerebellar cortex provide surround inhibition to GCs and receive input from parallel fiber and MF synapses. miniML reliably detected synaptic events in these recordings (Fig. 4h–i), although event decay kinetics were slower compared to the training data (Golgi cell, 1.83 ms (SD 0.44 ms, n = 10 neurons), GC training data, 0.9 ms). However, simulations indicated that larger differences in event kinetics may ultimately hinder detection when using the MF–GC mEPSC model (Supplementary Fig. 6). Synaptic events are also commonly recorded from neuronal culture preparations. We determined whether miniML could be applied to this scenario using recordings from cultured human induced pluripotent stem cell (iPSC)-derived neurons. We patch-clamped eight-week-old neurons and recorded spontaneous synaptic events in voltage-clamp (Fig. 4j). Prediction on these human iPSC-derived neuron data showed that miniML reliably detected synaptic events (Fig. 4k–l), which had an average frequency of 0.15 Hz (SD 0.24 Hz, n = 56 neurons). Taken together, the consistent performance in different synaptic preparations indicates that the miniML model can be directly applied to out-of-sample data with similar kinetics.

Generalization of miniML to diverse event and data types via transfer learning

Synaptic events can vary greatly between preparations, for example due to differences in the number and type of postsynaptic neurotransmitter receptors, the number of synaptic inputs, postsynaptic membrane properties, transmitter content of synaptic vesicles or presynaptic release properties. In addition, different hardware and recording conditions may affect the characteristics of the recorded data. Because classical detection methods are comparatively sensitive to event shape and threshold settings, their application to novel preparations and/or different recording modes is challenging. We employed a transfer learning (TL) strategy to facilitate the application of miniML to different recording conditions and data types. TL is a powerful technique in machine learning that allows for the transfer of knowledge learned from one task or domain to another[37,38]. TL is widely used with CNNs to take advantage of large pretrained models and repurpose them to solve new, unseen tasks[39]. Importantly, only a part of the network needs to be trained for the novel task, which significantly reduces the number of training samples needed and speeds up training while avoiding overfitting[37]. We therefore reasoned that TL based on freezing the convolutional layers during training of our pretrained network could be used to train a new model to detect events with different shapes and/or kinetics, using a lower number of training samples (Fig. 5a).

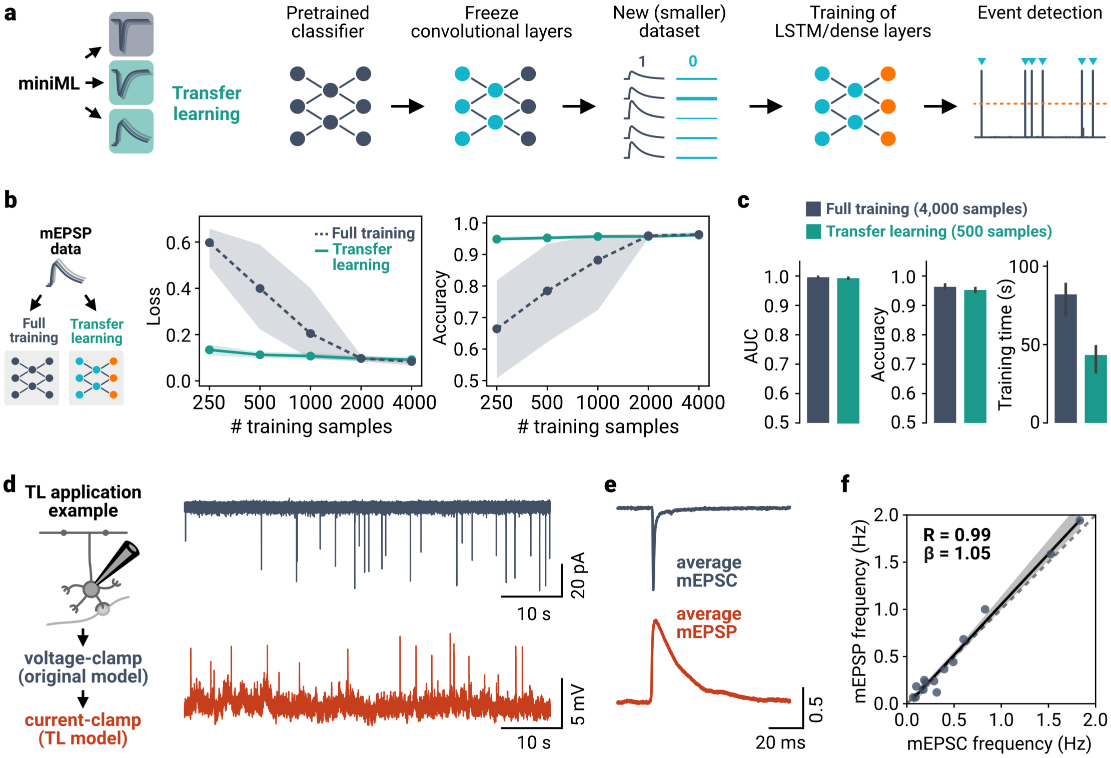


Fig. 5. Transfer learning allows analyzing different types of events with small amounts of training data.
a) Illustration of the transfer learning (TL) approach. The convolutional layers of a trained miniML model are frozen before retraining with new, smaller datasets. **b)** Comparison of TL and full training for MF–GC mEPSP data. Shown are loss and accuracy versus sample size for full training and TL. Indicated subsets of the dataset were split into training and validation sets; lines are averages of fivefold crossvalidation, shaded areas represent 95%CI. Note the log-scale of the abscissa. **c)** Average AUC, accuracy, and training time of models trained with the full dataset or with TL on a reduced dataset. TL models yield comparable classification performance with reduced training time using only 12.5% of the samples. **d)** Example recordings of synaptic events in voltage-clamp and current-clamp mode consecutively from the same neuron. **e)** Average peak normalized mEPSC and mEPSP waveform from the example in **d)**. Note the different event kinetics depending on recording mode. **f)** Event frequency for mEPSPs plotted against mEPSCs. Dashed line represents identity, solid line with shaded area represents linear regression.

We tested the use of TL for miniML with recordings of miniature excitatory postsynaptic potentials (mEPSPs) in mouse cerebellar GCs. These events have opposite polarity and slower kinetics compared to the original mEPSC training data. We compared TL-based model training to full training (with all layers trainable and reinitialized weights), varying the training sample size. While accuracy increased and loss decreased with the number of samples (Fig. 5b), TL models performed well with as few as 400 samples. Under these conditions, accuracy was only slightly lower than for full training with almost ten times the sample size (median accuracy, 95.4 versus 96.1%; Fig. 5c). This suggests that TL can significantly reduce the sample size needed for training, minimizing the time-consuming process of event extraction and labeling. Across different datasets, TL-trained models performed comparably to those trained from scratch (Supplementary Fig. 7). Taken together, these data demonstrate that TL allows model training with an order of magnitude smaller amount of training data, making the miniML method easily transferable to new datasets.

To investigate how a TL-trained model performs in event detection, we analyzed data recorded in different recording modes (voltage-clamp vs. current-clamp). The different noise conditions typically encountered in current-clamp recordings and the difference between EPSP and EPSC waveforms necessitate distinct detection schemes for common detection methods. By recording spontaneous EPSPs and EPSCs in the same cerebellar granule cells (Fig. 5d–e), we could compare detection performance via event frequency. Voltage changes upon spontaneous synaptic vesicle fusions could be approximated by a two-exponential time course, similar to postsynaptic currents. However, their kinetics were considerably slower due to the charging of the plasma membrane (Fig. 5e). We used a TL-trained model for mEPSP detection and the standard miniML model for mEPSCs. Remarkably, the average event frequencies were very similar in the two different recording modes (voltage-clamp: 0.49 Hz, SD 0.53 Hz, current-clamp: 0.54 Hz, SD 0.6 Hz, n = 15 for both) and highly correlated across neurons (Fig. 5f). This highlights the reliability of TL with small training sets for consistent detection of synaptic events in datasets with varying characteristics.

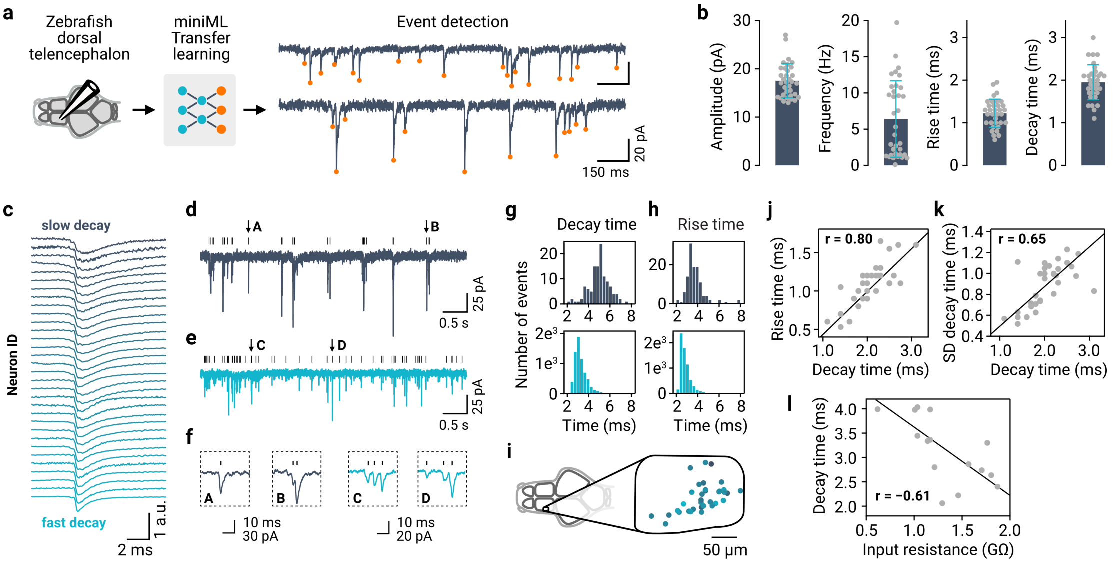


Fig. 6. Synaptic event detection for neurons in a full-brain explant preparation of adult zebrafish.
a, Application of TL to facilitate event detection for EPSC recordings. **b**, Extraction of amplitudes, event frequencies, decay times, and rise time for all neurons in the dataset (n = 34). **c**, Different typical (mean) event kinetics for the analyzed neurons, ordered by decay times. **d–e**, Example recordings with slow (dark blue) and fast (light blue) event kinetics. **f**, Examples of events taken from **d–e**, illustrating the diversity of kinetics within and across neurons. **g**, Distribution of event decay kinetics across single events for two example neurons (traces shown in **d–e**). **h**, Distribution of event rise kinetics across single events for same two example neurons. **i**, Mapping of decay times (color-coded as in **c**) onto the anatomical map of the recording subregion of the telencephalon. **j**, Mean decay and rise times are correlated across neurons. **k**, Decay time distributions are broader (SD of decay time distributions) when mean decay times are larger. **l**, Input resistance as a proxy for cell size is negatively correlated with the decay time.

Application to an *ex vivo* whole brain preparation

The presence of overlapping and highly variable event shapes pose additional challenges for event detection methods. We evaluated miniML's performance in such complex scenarios by analyzing data recorded from principal neurons of the adult zebrafish telencephalon[40]. We focused on spontaneous excitatory inputs to these neurons, characterized by diverse event shapes and frequencies[40]. Training via TL (see Methods) yielded a model that enabled the reliable detection of spontaneous excitatory currents (Fig. 6a). Analysis of event properties across cells revealed broad distributions of event statistics (Fig. 6b), including a large diversity of event rise and decay kinetics (Fig. 6b–c). Notably, miniML consistently identified synaptic events with diverse kinetics and shapes.

We next used the extracted event kinetics features of individual neurons (Fig. 6d–h) to demonstrate miniML's utility in better understanding the diversity of an existing dataset. First, we explored whether the anatomical location of each neuron could predict event decay times. We mapped the recorded neurons to an anatomical reference and plotted decay times as a function of their position, but did not find a strong relationship (Fig. 6i; correlation with position p>0.05 in all 3 dimensions). Second, we tested the hypothesis that slower event kinetics might be associated with larger cells. In large cells, EPSCs may undergo stronger filtering as they propagate from synaptic sites to the soma. Consistent with this idea, we observed correlations between decay and rise times across neurons (Fig. 6j). Furthermore, the distribution of decay times (examples shown in Fig. 6h) was broader for neurons with longer decay times (Fig. 6k), suggesting a broader distribution of distances from synapses to the cell body. Finally, for a subset of neurons, we recorded input resistance, which approximates the cell membrane resistance and is therefore a proxy for cell size. Input resistance was negatively correlated with decay times across neurons (Fig. 6l), consistent with the hypothesis that diverse event kinetics across neurons are determined by the conditions of synaptic event propagation to the soma and, more specifically, cell size. Taken together, this analysis underscores the versatility of miniML, as it can be successfully applied to new datasets with varying recording conditions. miniML consistently extracted synaptic events across a spectrum of event kinetics, enabling the identification and investigation of key factors determining event kinetics and other event-related properties across neurons.

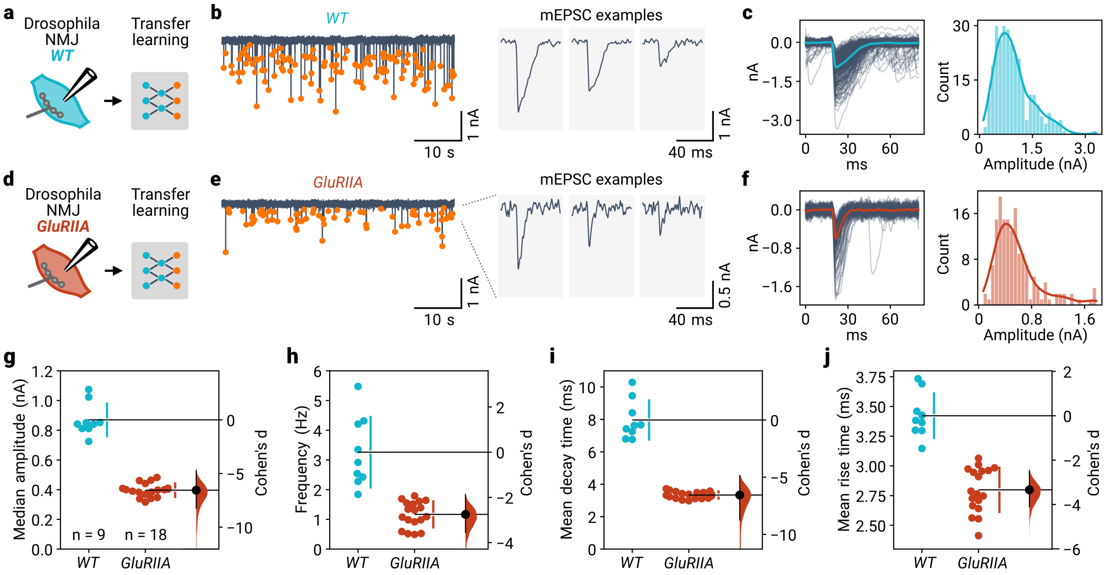


Fig. 7. Event detection at *Drosophila* neuromuscular synapses.
a, Two-electrode voltage-clamp recordings from wild-type (WT) *Drosophila* NMJ were analyzed using miniML with transfer learning. **b**, Left: Example voltage-clamp recording with detected events highlighted. Right: Three individual mEPSCs on expanded time scale. **c**, Left: All detected events from the example in **b** overlaid with average (blue line). Right: Event amplitude histogram. **d–f**, Same as in **a–c**, but for GluRIIA mutant flies. **g**, Comparison of median event amplitude for WT and GluRIIA NMJs. Both groups are plotted on the left axes; Cohen's d is plotted on a floating axes on the right as a bootstrap sampling distribution (red). The mean difference is depicted as a dot (black); the 95% confidence interval is indicated by the ends of the vertical error bar. **h**, Event frequency is lower in GluRIIA mutant NMJs than in WT. **i**, Knockout of GluRIIA speeds event decay time. **j**, Faster event rise times in GluRIIA mutant NMJs.

Application to two-electrode voltage-clamp data from neuromuscular synapses

Synaptic events can also be recorded at synapses of the peripheral nervous system. We next applied miniML to analyze data obtained from the *Drosophila melanogaster* larval neuromuscular junction

(NMJ)[41]. This synapse is characterized by a higher frequency of spontaneous synaptic events with a slower time course compared with typical brain slice recordings. In addition, mEPSC recordings are performed using two-electrode voltage-clamp, which can be associated with large leak currents and rather low SNR. Because of the different event shapes in these data, we used a small dataset of manually extracted NMJ events to train a TL model. The TL model was able to reliably detect synaptic events in wild-type (WT) NMJ recordings, with excellent performance regarding recall and precision (Fig. 7a–c). We next assessed event detection upon deletion of the non-essential glutamate receptor subunit GluRIIA, which causes a strong reduction in mEPSC amplitude and faster kinetics[42]. A separate TL model allowed reliable synaptic event detection in recordings from GluRIIA mutant larvae (Fig. 7d–f). We observed a 54% reduction in mEPSC amplitude compared to WT (Fig. 7g), consistent with previous reports[42,43]. In addition, the event frequency was reduced by 64% (Fig. 7h), probably due to some events being below the detection limit. Half decay and rise times were also shorter at GluRIIA than at WT NMJs (−58% and −18%, respectively) (Fig. 7i–j), which can be explained by the faster desensitization of the remaining GluRIIB receptors[42]. Thus, miniML can be applied to two-electrode voltage clamp recordings at the *Drosophila* NMJ and robustly detects group differences upon genetic receptor perturbation.

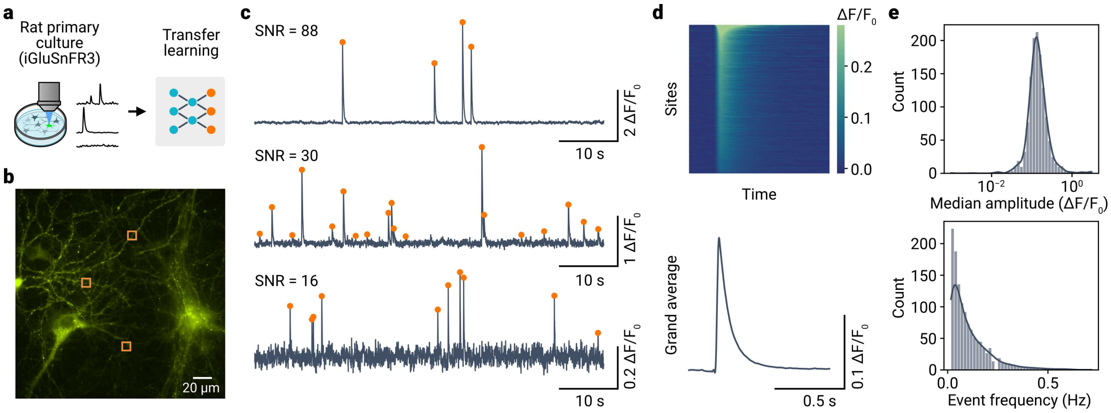


Fig. 8. Applying miniML to detect optical minis.
a, The miniML model was applied to recordings from rat primary culture neurons expressing iGluSnFR3. Data from[44]. **b**, Example epifluorescence image of iGluSnFR3-expressing cultured neurons. Image shows a single frame with three example regions of interest (ROIs) indicated. **c**, $\Delta F/F_0$ traces for the regions shown in **b**. Orange circles indicate detected optical minis. Note the different signal-to-noise ratios of the examples. **d**, Top: Heatmap showing average optical minis for all sites with detected events, sorted by amplitude. Bottom: Grand average optical mini. **e**, Top: Histogram with kernel density estimate (solid line) of event amplitudes for n=1524 ROIs of the example in **b**. Note the log-scale of the abscissa. Bottom: Inferred event frequency.

Application to glutamate imaging of miniature synaptic events

To assess the broader applicability of miniML, we tested its ability to analyze events in time-series data derived from live fluorescence imaging experiments. These imaging datasets often feature a lower sampling rate and distinct noise profile compared to electrophysiological recordings. Nevertheless, the waveforms of synaptic release events or Ca^{2+} transients closely resemble those used to train miniML. Thus, we hypothesized that miniML could also be employed for event detection in imaging data. We analyzed a previous dataset[44] from rat neuronal cultures expressing the glutamate sensor iGluSnFR3 (Fig. 8a–b). These data, recorded in the presence of TTX, contain spontaneous transient increases in fluorescence intensity representing individual release events ('optical minis'). Initially, we selected a small subset of events from the data to train a TL model. Given the low sampling rate of the imaging data (100 Hz), we upsampled the raw $\Delta F/F_0$ data by a factor of 10 to match the input shape required by the miniML model. The TL model was subsequently applied to all detected sites within the widefield recording[44]. A qualitative assessment of the imaging traces showed excellent event detection, with miniML consistently localizing the iGluSnFR3 fluorescence transients at varying SNRs (Fig. 8c). The detected optical minis had similar kinetics to those reported in[44] (10–90% rise time, median 21.8 ms; half decay time, median 48.7 ms, Fig. 8d). In addition, analysis of event frequencies across sites revealed a power-law distribution (Fig. 8e). These results demonstrate that miniML generalizes beyond electrophysiological data and can support fluorescence imaging data analysis. Future applications of miniML could improve event detection and spike inference in imaging datasets.

DISCUSSION

Here, we developed and evaluated miniML, a novel method for detecting spontaneous synaptic events using deep learning. miniML provides a reliable and versatile solution for analyzing synaptic events in time-series data with several important advantages.

We demonstrate that miniML outperforms commonly used methods for synaptic event detection, particularly with respect to false positives, which is important for the accurate quantification of synaptic transmission. In addition to its high precision, miniML's detection performance is largely threshold independent. This effectively eliminates the trade-off between false positive and false negative rates typically present in event detection methods[15,16,19]. Thus, miniML is highly reproducible and eliminates the need for laborious manual event inspection. Automated, reproducible data analysis is key to open science and the use of publicly available datasets[45,46].

Our results from a wide variety of preparations, covering different species and neuronal preparations, demonstrate the robustness of miniML and its general applicability to the study of synaptic physiology. miniML generalizes well to different experimental preparations, conditions, or data types. Data with event waveforms that are similar to the original miniML training data (mouse cerebellar GC recordings) can be analyzed immediately. For more distinct event waveforms or data properties, data resampling and/or retraining of a classifier via transfer learning (TL) enables event detection optimized for the respective recording conditions. Importantly, TL requires only a few hundred labeled events for training, facilitating its application to novel datasets. For example, an existing miniML model can be easily retrained to eliminate potential false positives, e.g., due to sudden changes in noise characteristics.

Furthermore, miniML's approach to detect events is not restricted to electrophysiological data, but can also be applied to time-series data derived from live imaging experiments using, e.g., reporters of glutamate[44], Ca^{2+}[47], or membrane voltage[48]. miniML could also be applied to the analysis of evoked postsynaptic responses, such as failure analysis or quantification of unitary events, and to functional connectivity studies[49]. In addition, the method could be extended to other areas of biology and medicine where signal detection is critical, such as clinical neurophysiology or imaging.

Comprehensive performance comparison is essential for method evaluation and selection. Benchmarking requires standardized ground truth data, but unlike, e.g., spike inference from Ca^{2+} imaging[50], these are usually not available for spontaneous synaptic event recordings. Furthermore, the results can vary between simulated and real data[50]. We therefore established a benchmarking pipeline using event-free recordings with simulated events, circumventing differences in the noise power spectrum of simulations vs. recordings[16,19]. This approach may provide a general framework for evaluating the effectiveness of different synaptic event detection methods.

The supervised learning method of miniML requires labeled training data. Although TL requires only a few hundred training samples, these data need to be collected and annotated by the user. Future directions may explore unsupervised learning techniques such as autoencoders to reduce the dependence on annotated training data. In addition, at very high event frequencies, individual synaptic events may overlap, making their separation difficult. Although miniML can generally detect overlapping events, very close events may not always be detected and there may be a lower bound. Implementing additional techniques such as wavelet transformation, or spectral domain or shapelet analysis[51,52] may improve the accuracy of event detection, especially for overlapping events.

miniML presents an innovative data analysis method that may help advance the field of synaptic physiology. Its open-source Python software design ensures seamless integration into existing data analysis pipelines and enables widespread use of the method, fostering the development of new applications and further innovation. Remarkably, despite its deep learning approach, miniML runs at relatively rapid speed on commodity hardware. Thanks to its robust, generalizable, and unbiased detection performance, miniML allows researchers to perform more accurate and efficient synaptic event analysis. A standardized, more efficient, and reproducible analysis of synaptic events will promote important new insights into synaptic physiology and dysfunction[53,54] and help improve our understanding of neural function.

METHODS

Electrophysiological recordings

Animals were treated according to national and institutional guidelines. All experiments were approved by the Cantonal Veterinary Office of Zurich (approval number no. ZH206/2016 and ZH009/2020).

Experiments were performed in male and female C57BL/6J mice (Janvier Labs, France). Mice were 1–5-month-old, but for recordings from the Calyx of Held, which were performed in P9 animals. Animals were housed in groups of 3–5 in standard cages under a 12h-light/12h-dark cycle with food and water ad libitum. Mice were sacrificed by rapid decapitation after isoflurane anesthesia. The cerebellar vermis was quickly removed and mounted in a chamber filled with chilled extracellular solution. Parasagittal 300-µm thick slices were cut with a Leica VT1200S vibratome (Leica Microsystems, Germany), transferred to an incubation chamber at 35 °C for 30 minutes, and then stored at room temperature until experiments. The extracellular solution (artificial cerebrospinal fluid, ACSF) for slicing and storage contained (in mM): 125 NaCl, 25 $NaHCO_3$, 20 D-glucose, 2.5 KCl, 2 $CaCl_2$, 1.25 NaH_2PO_4, 1 $MgCl_2$, bubbled with 95% O_2 and 5% CO_2.

Slices were visualized using an upright microscope with a 60×, 1 NA water immersion objective, infrared optics, and differential interference contrast (Scientifica, UK). The recording chamber was continuously perfused with ACSF. For event-free recordings, we blocked excitatory and inhibitory transmission using ACSF supplemented with 50 µM D-APV, 10 µM NBQX, 10 µM bicuculline, and 1 µM strychnine. Patch pipettes (open-tip resistances of 3–8 MΩ) were filled with solution containing (in mM): 150 K-D-gluconate, 10 NaCl, 10 HEPES, 3 MgATP, 0.3 NaGTP, 0.05 ethyleneglycol-bis(2-aminoethylether)-N,N,N',N'-tetraacetic acid (EGTA), pH adjusted to 7.3 with KOH. Voltage-clamp and current-clamp recordings were made using a HEKA EPC10 amplifier controlled by Patchmaster software (HEKA Elektronik GmbH, Germany). Voltages were corrected for a liquid junction potential of +13 mV. Experiments were performed at room temperature (21–25 °C). Miniature EPSCs (mEPSCs) were recorded at a holding potential of −100 mV or −80 mV, and miniature EPSPs (mEPSPs) at the resting membrane potential. Data were filtered at 2.9 kHz and digitized at 50 kHz. Synaptic event recording periods typically lasted 120 s.

Spontaneous EPSCs were recorded from human iPSC-derived neurons in a similar fashion, but ACSF consisted of (in mM): 135 NaCl, 10 HEPES, 10 D-glucose, 5 KCl, 2 $CaCl_2$, 1 $MgCl_2$. We recorded from 8-week-old neuronal cultures as described in[55]. Synaptic events were observed in ~ 51% of neurons.

Training data and annotation

We used synaptic event recordings from a previous publication[31] to generate the training dataset. mEPSCs were extracted based on low-threshold template-matching. Corresponding sections of data without events were randomly selected from the recordings. The extracted windows had a length of 600 datapoints. Given our sampling rate of 50 kHz, this corresponds to 11.98 ms. We subsequently manually scored data sections as event-containing (label=1) or not event-containing (label=0). The ratio of events to non-events was kept close to one to ensure efficient training. Based on empirical observations of model performance, we included relatively small amplitude events that are often missed by other methods. Similarly, including negative examples that are commonly picked up as false positives, resulted in more accurate prediction traces. We used data augmentation techniques to further improve model discrimination. We simulated waveforms of non-synaptic origin, which are occasionally encountered during recordings, and superimposed them onto noise recordings. Examples included rapid current transients that can be caused by peristaltic perfusion pump systems often used in brain slices recordings, or slow currents with a symmetric rise and decay timecourse. A total of 4500 segments were created and labeled as 0. To maintain the ratio of negative to positive examples, we added an equivalent number of simulated synaptic events. The biexponential waveform described in the section *Benchmarking detection methods* was used for event simulation. The final training dataset contained 30,140 samples (21,140 from recorded traces and 9,000 simulated samples).

Machine learning model architecture

We built a discriminative end-to-end deep learning model for one-dimensional time-series classification[23]. The neural network architecture comprised multiple convolutional layers, an LSTM layer, and a fully connected layer. The approach is related to networks designed for event detection in audio[56] and image data[57], or classification of genomic data[58]. The classifier was built using TensorFlow, an open-source machine learning library for Python[59]. The network takes batches of one-dimensional (1D) univariate time-series data as input, which are converted into a tensor of shape (batch size, data length, 1). The data is passed to three convolutional blocks. Each block consists of a 1D convolutional layer with a leaky Rectified Linear Unit (leaky ReLU) activation function followed by an average pooling layer. To avoid overfitting, each convolutional block uses batch normalization[60]. Batch normalization reduced training time by about two times and improved the accuracy of the resulting model. We added a fourth convolutional block that includes a convolutional layer, batch normalization and a leaky ReLU activation function, but no average pooling layer. The output of the final convolutional layer passes through a bidirectional recurrent layer of LSTM units. The final layers consist of a fully connected dense unit layer and a dropout layer[61], followed by a single unit with sigmoid activation. The output of the neural network is a scalar between [0, 1]. The layers and parameters used including output shape and number of trainable parameters are summarized in Supplementary Table 1.The total number of trainable parameters was 190,913.

Training and evaluation

The network was trained with Tensorflow 2.12 and Python 3.10 with CUDA 11.4. Datasets were scaled between zero and one, and split into training and validation data (.75/.25). The model was compiled using the Adam optimizer[62] with AMSGrad[63]. We trained the classifier using a learning rate $\eta = 0.00002$ and batch size of 128 on the training data. Training was run for maximum 100 epochs with early stopping to avoid overfitting. Validation data was used to evaluate training performance. Early stopping was applied when the validation loss did not improve for eight consecutive epochs. Typically, training lasted for 20–40 epochs. We used binary cross-entropy loss and binary accuracy as measures of performance during training. The best performing model was selected from a training run, and a receiver-operating characteristic (ROC) was calculated. We used accuracy and area under the ROC curve to evaluate training performance. To accelerate training, we used a GPU; training time for the neural network was ~8 minutes on a workstation with NVIDIA Tesla P100 16 GB GPU (Intel Xeon 2.2 GHz CPU, 16 GB RAM) and ~30 minutes on a laptop computer (Apple M1 Pro, 16 GB RAM).

Applying the classifier for event detection

The trained miniML classifier takes sections of data with predefined length as input. To detect events in arbitrarily long recordings, a sliding window procedure is used. Time-series data from voltage-clamp or current-clamp recordings is segmented using a sliding window with stride. The resulting 2D-tensor is min-max scaled and then used as model input for inference. To overcome the potential limitation of long computation times, we used a sliding window with stride procedure. Using a stride > 1 significantly reduces the inference time of the model (Supplementary Fig. 3), especially for data with high sampling rates or long recording times. This approach results in a prediction trace with data being spaced at $samplinginterval * stride$. With $stride = 20$, a 120-s recording at 50 kHz sampling rate can be analyzed in ~15 s on a laptop computer (Apple M1Pro, 16 GB RAM, Supplementary Fig. 3).

Events in the input data trace result in distinct peaks in the prediction trace. We applied a maximum filter to enhance post-processing of the data. Thus, by applying a threshold to the prediction trace, synaptic event positions can be detected. Our analyses indicate that the absolute threshold value is not important in the [0.05, 0.95] range (Fig. 5). Data following a threshold crossing in the prediction trace are cut from the raw data and aligned by steepest rise. To find the point of steepest rise, a peak-search is performed in the first derivative of the short data segment. If multiple peaks are detected, any peak that has a prominence $\geqslant 0.25$ relative to the largest detected prominence is treated as an additional event that is in close proximity or overlapping. The resulting 2D-array with aligned events can be further analyzed to obtain descriptive statistics on, e.g., amplitude, charge, or rise and decay kinetics of individual events.

Benchmarking detection methods

We compared the deep learning-based miniML with the following previously described detection methods: template-matching[15], deconvolution[16], a finite threshold-based approach[12], the commercial MiniAnalysis software (version 6.0.7, Synaptosoft), and a Bayesian inference procedure[19]. Detection methods were implemented in Python 3.9, with the exception of the Bayesian method, which was run using Matlab R2022a, and MiniAnalysis running as stand-alone software on Windows.

We used generated standardized data to benchmark the detection performance of different methods. Event-free recordings (see section *Electrophysiological Recordings* for details) were superimposed with simulated events having a biexponential waveform:

$$I(t) = (1 - e^{(\frac{-t}{\tau_{rise}})}) * e^{(\frac{-t}{\tau_{decay}})}$$

Where $I(t)$ is the current as a function of time, and τ_{rise} and τ_{decay} are the rise and decay time constants, respectively. Simulated event amplitudes were drawn from a log-normal distribution with variance 0.4. Mean amplitude was varied to generate diverse signal-to-noise ratios. Decay time constants were drawn from a normal distribution with mean 1.0 ms and variance 0.25 ms[31]. Generated events were randomly placed in the event-free recording with an average frequency of 0.7 Hz and a minimum spacing of 3 ms. Generated traces provided ground-truth data for the evaluation of the different methods.

To quantify detection performance of different methods over a range of signal-to-noise ratios, we calculated the number of true positives (TP), false positives (FP) and false negatives (FN). From these, the following metrics were calculated:

$$Precision = \frac{TP}{TP+FP}$$

$$Recall = \frac{TP}{TP+FN}$$

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall}$$

For all metrics, higher values indicate a better model performance.

We also evaluated detection performance under non-optimal conditions (i.e., events that did not precisely match the event template or the training data). To do this, we varied the kinetics of simulated events by either increasing (mean = 4.5 ms) or decreasing (mean = 0.5 ms) the average decay time constant (Supplementary Fig. 5).

Hyperparameter settings of detection methods

For all conditions, threshold settings were $4 * SD$ of the detection trace for template-matching and deconvolution, and –4 pA for the finite-threshold method. Both template-based methods used a template waveform corresponding to the average of the simulated events. For the Bayesian detection approach, we used the code provided by the authors[19] with the following adjustments to the default hyperparameters: minimum amplitude = 3.5, noise $\phi = [.90; -.52]$, rate = 0.5. We chose a cutoff of $6 * SD$ of the event time posteriors as threshold. MiniAnalysis (Synaptosoft) was run in the automatic detection mode with default settings for EPSCs and amplitude and area thresholds of 4. We used a threshold of 0.5 for miniML.

Transfer learning

To make the miniML method applicable to data with different characteristics, such as different event kinetics, noise characteristics, or recording mode, we used a transfer learning (TL) approach[64]. We froze the convolutional layers of the fully trained MF–GC miniML model, resulting in a new model with only the LSTM and dense layers being trainable. With this approach, the convolutional layers act as pretrained feature detectors and much fewer training samples are required. Hyperparameters and training conditions were the same as for full-training (see *Training and evaluation*) with the following exceptions: learning rate $\eta = 0.00000002$, patience = 15, batch size = 32, dropout rate = 0.5. The training data were resampled to 600 datapoints to match the input shape of the original model.

To compare TL with full training, random subsets of the training data (mEPSPs in cerebellar GCs, spontaneous EPSCs in zebrafish, or mEPSCs at the *Drosophila* neuromuscular junction) with increasing size were generated and used to train models using a fivefold cross-validation. We always used the same size of the validation dataset. After comparing TL with full training (Fig. 5, Supplementary Fig. 7), we trained separate models to analyze the different datasets using subsets of the available training data.

Quantification

Computing times were quantified using the performance counter function of the built-in Python package *time*, and are given as wall times. Statistical comparisons were made using permutation t-tests with 5000 reshuffles[65]. Effect sizes are reported as Cohen's d with 95% confidence intervals, obtained by bootstrapping (5000 samples; the confidence interval is bias-corrected and accelerated)[65]. Event amplitudes were quantified as difference between detected event peaks and a short baseline window before event onset. Decay times refer to half decay times, and rise times were quantified as 10–90% rise times.

Data and code availability

Datasets used for model training are available from Zenodo [Link will be made available upon publication]. miniML source code and pretrained models are available online [https://github.com/delvendahl/miniML], including analysis code. All code was implemented in Python, using the following open-source libraries: TensorFlow, Scipy, Numpy, Matplotlib, Pandas, h5py, sklearn, pyABF, dabest.

We used data from previous publications to generate training sets and to assess the application of miniML[31,40,41]. In addition, a previously published dataset[44] was used [https://doi.org/10.25378/janelia.21985406].

ACKNOWLEDGMENTS

We thank Mark D. Robinson and Anu G. Nair for helpful discussions.

This work received funding by the Swiss National Science Foundation (grant PZ00P3_174018 to I.D., grant PZ00P3_209114 to P.R., grant 310030B_152833/1 to R.F.), the Novartis Research Foundation (to R.F.), the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement No 742576 to R.F.), a fellowship from the Boehringer Ingelheim Fonds (to P.R.), and the UZH Alumni Research Talent Development fund (to I.D.). The funding bodies had no role in study design, data collection and interpretation, or the decision to submit the work for publication.

REFERENCES

1. Kaeser, P. S. & Regehr, W. G. Molecular Mechanisms for Synchronous, Asynchronous, and Spontaneous Neurotransmitter Release. *Annual Review of Physiology* **76,** 333–363. https://doi.org/10.1146/annurev-physiol-021113-170338 (Feb. 2014).
2. Kavalali, E. T. The mechanisms and functions of spontaneous neurotransmitter release. *Nature Reviews Neuroscience* **16,** 5–16. https://doi.org/10.1038/nrn3875 (Jan. 2015).
3. McKinney, R. A., Capogna, M., Dürr, R. & Gähwiler, B. Miniature synaptic events maintain dendritic spines via AMPA receptor activation. *Nature Neuroscience* **2,** 44–49. https://doi.org/10.1038/4548 (Jan. 1999).
4. Banerjee, S. *et al.* Miniature neurotransmission is required to maintain Drosophila synaptic structures during ageing. *Nature Communications* **12.** https://doi.org/10.1038/s41467-021-24490-1 (July 2021).
5. Abbott, L. F. & Regehr, W. G. Synaptic computation. *Nature* **431,** 796–803. https://doi.org/10.1038/nature03010 (Oct. 2004).
6. Holler, S., Köstinger, G., Martin, K. A. C., Schuhknecht, G. F. P. & Stratford, K. J. Structure and function of a neocortical synapse. *Nature* **591,** 111–116. https://doi.org/10.1038/s41586-020-03134-2 (Jan. 2021).
7. Huganir, R. L. & Nicoll, R. A. AMPARs and Synaptic Plasticity: The Last 25 Years. *Neuron* **80,** 704–717. https://doi.org/10.1016/j.neuron.2013.10.025 (Oct. 2013).
8. Malinow, R. & Malenka, R. C. AMPA Receptor Trafficking and Synaptic Plasticity. *Annual Review of Neuroscience* **25,** 103–126. https://doi.org/10.1146/annurev.neuro.25.112701.142758 (Mar. 2002).
9. Turrigiano, G. G., Leslie, K. R., Desai, N. S., Rutherford, L. C. & Nelson, S. B. Activity-dependent scaling of quantal amplitude in neocortical neurons. *Nature* **391,** 892–896. https://doi.org/10.1038/36103 (Feb. 1998).
10. O'Brien, R. J. *et al.* Activity-Dependent Modulation of Synaptic AMPA Receptor Accumulation. *Neuron* **21,** 1067–1078. https://doi.org/10.1016/s0896-6273(00)80624-8 (Nov. 1998).
11. Kim, Y. G., Shin, J. J. & Kim, S. J. Minhee Analysis Package: an integrated software package for detection and management of spontaneous synaptic events. *Molecular Brain* **14.** https://doi.org/10.1186/s13041-021-00847-x (Sept. 2021).

12. Kudoh, S. N. & Taguchi, T. A simple exploratory algorithm for the accurate and fast detection of spontaneous synaptic events. *Biosensors and Bioelectronics* **17,** 773–782. https://doi.org/10.1016/s0956-5663(02)00053-2 (Sept. 2002).

13. Ankri, N., Legendre, P., Faber, D. & Korn, H. Automatic detection of spontaneous synaptic responses in central neurons. *Journal of Neuroscience Methods* **52,** 87–100. https://doi.org/10.1016/0165-0270(94)90060-4 (Apr. 1994).

14. Jonas, P., Major, G. & Sakmann, B. Quantal components of unitary EPSCs at the mossy fibre synapse on CA3 pyramidal cells of rat hippocampus. *The Journal of Physiology* **472,** 615–663. https://doi.org/10.1113/jphysiol.1993.sp019965 (Dec. 1993).

15. Clements, J. & Bekkers, J. Detection of spontaneous synaptic events with an optimally scaled template. *Biophysical Journal* **73,** 220–229. https://doi.org/10.1016/s0006-3495(97)78062-7 (July 1997).

16. Pernía-Andrade, A. J. *et al.* A Deconvolution-Based Method with High Sensitivity and Temporal Resolution for Detection of Spontaneous Synaptic Currents In Vitro and In Vivo. *Biophysical Journal* **103,** 1429–1439. https://doi.org/10.1016/j.bpj.2012.08.039 (Oct. 2012).

17. Shi, Y., Nenadic, Z. & Xu, X. Novel Use of Matched Filtering for Synaptic Event Detection and Extraction. *PLoS ONE* **5** (ed Degtyar, V. E.) e15517. https://doi.org/10.1371/journal.pone.0015517 (Nov. 2010).

18. Zhang, X., Schlögl, A., Vandael, D. & Jonas, P. MOD: A novel machine-learning optimal-filtering method for accurate and efficient detection of subthreshold synaptic events in vivo. *Journal of Neuroscience Methods* **357,** 109125. https://doi.org/10.1016/j.jneumeth.2021.109125 (June 2021).

19. Merel, J., Shababo, B., Naka, A., Adesnik, H. & Paninski, L. Bayesian methods for event analysis of intracellular currents. *Journal of Neuroscience Methods* **269,** 21–32. https://doi.org/10.1016/j.jneumeth.2016.05.015 (Aug. 2016).

20. LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521,** 436–444. https://doi.org/10.1038/nature14539 (May 2015).

21. Richards, B., Tsao, D. & Zador, A. The application of artificial intelligence to biology and neuroscience. *Cell* **185,** 2640–2643. https://doi.org/10.1016/j.cell.2022.06.047 (July 2022).

22. Wang, Z., Yan, W. & Oates, T. *Time Series Classification from Scratch with Deep Neural Networks: A Strong Baseline* 2016. https://arxiv.org/abs/1611.06455.

23. Fawaz, H. I., Forestier, G., Weber, J., Idoumghar, L. & Muller, P.-A. Deep learning for time series classification: a review. *Data Mining and Knowledge Discovery* **33,** 917–963. https://doi.org/10.1007/s10618-019-00619-1 (Mar. 2019).

24. Iqbal, A., Khan, R. & Karayannis, T. Developing a brain atlas through deep learning. *Nature Machine Intelligence* **1,** 277–287. https://doi.org/10.1038/s42256-019-0058-8 (June 2019).

25. Imbrosci, B., Schmitz, D. & Orlando, M. Automated Detection and Localization of Synaptic Vesicles in Electron Microscopy Images. *eneuro* **9,** ENEURO.0400–20.2021. https://doi.org/10.1523/eneuro.0400-20.2021 (Jan. 2022).

26. Rupprecht, P. *et al.* A database and deep learning toolbox for noise-optimized, generalized spike inference from calcium imaging. *Nature Neuroscience* **24,** 1324–1337. https://doi.org/10.1038/s41593-021-00895-5 (Aug. 2021).

27. Yip, M. C., Gonzalez, M. M., Valenta, C. R., Rowan, M. J. M. & Forest, C. R. Deep learning-based real-time detection of neurons in brain slices for in vitro physiology. *Scientific Reports* **11.** https://doi.org/10.1038/s41598-021-85695-4 (Mar. 2021).

28. Denis, J., Dard, R. F., Quiroli, E., Cossart, R. & Picardo, M. A. DeepCINAC: A Deep-Learning-Based Python Toolbox for Inferring Calcium Imaging Neuronal Activity Based on Movie Visualization. *eneuro* **7,** ENEURO.0038–20.2020. `https://doi.org/10.1523/eneuro.0038-20.2020` (July 2020).

29. Sitá, L. *et al.* A deep-learning approach for online cell identification and trace extraction in functional two-photon calcium imaging. *Nature Communications* **13.** `https://doi.org/10.1038/s41467-022-29180-0` (Mar. 2022).

30. Donahue, J. *et al.* Long-Term Recurrent Convolutional Networks for Visual Recognition and Description. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **39,** 677–691. `https://doi.org/10.1109/tpami.2016.2599174` (Apr. 2017).

31. Delvendahl, I., Kita, K. & Müller, M. Rapid and sustained homeostatic control of presynaptic exocytosis at a central synapse. *Proceedings of the National Academy of Sciences* **116,** 23783–23789. `https://doi.org/10.1073/pnas.1909675116` (Nov. 2019).

32. Van der Ploeg, T., Austin, P. C. & Steyerberg, E. W. Modern modelling techniques are data hungry: a simulation study for predicting dichotomous endpoints. *Bller Medical Research Methodology* **14.** `https://doi.org/10.1186/1471-2288-14-137` (Dec. 2014).

33. Bailly, A. *et al.* Effects of dataset size and interactions on the prediction performance of logistic regression and deep learning models. *Computer Methods and Programs in Biomedicine* **213,** 106504. `https://doi.org/10.1016/j.cmpb.2021.106504` (Jan. 2022).

34. Lesperance, L. S., Yang, Y.-M. & Wang, L.-Y. Delayed expression of activity-dependent gating switch in synaptic AMPARs at a central synapse. *Molecular Brain* **13.** `https://doi.org/10.1186/s13041-019-0536-2` (Jan. 2020).

35. Ishii, T., Stolz, J. R. & Swanson, G. T. Auxiliary Proteins are the Predominant Determinants of Differential Efficacy of Clinical Candidates Acting as AMPA Receptor Positive Allosteric Modulators. *Molecular Pharmacology* **97,** 336–350. `https://doi.org/10.1124/mol.119.118554` (Feb. 2020).

36. Kita, K. *et al.* GluA4 facilitates cerebellar expansion coding and enables associative memory formation. *eLife* **10.** `https://doi.org/10.7554/elife.65152` (July 2021).

37. Yosinski, J., Clune, J., Bengio, Y. & Lipson, H. *How transferable are features in deep neural networks?* in *NIPS* (2014).

38. Caruana, R. *Learning Many Related Tasks at the Same Time with Backpropagation* in *Advances in Neural Information Processing Systems* (eds Tesauro, G., Touretzky, D. & Leen, T.) **7** (MIT Press, 1994). `https://proceedings.neurips.cc/paper_files/paper/1994/file/0f840be9b8db4d3fbd5ba2ce59211f55-Paper.pdf`.

39. Theodoris, C. V. *et al.* Transfer learning enables predictions in network biology. *Nature* **618,** 616–624. `https://doi.org/10.1038/s41586-023-06139-9` (May 2023).

40. Rupprecht, P. & Friedrich, R. W. Precise Synaptic Balance in the Zebrafish Homolog of Olfactory Cortex. *Neuron* **100,** 669–683.e5. `https://doi.org/10.1016/j.neuron.2018.09.013` (Nov. 2018).

41. Baccino-Calace, M., Schmidt, K. & Müller, M. The E3 ligase Thin controls homeostatic plasticity through neurotransmitter release repression. *eLife* **11.** `https://doi.org/10.7554/elife.71437` (July 2022).

42. DiAntonio, A., Petersen, S. A., Heckmann, M. & Goodman, C. S. Glutamate Receptor Expression Regulates Quantal Size and Quantal Content at the *Drosophila* Neuromuscular Junction. *The Journal of Neuroscience* **19,** 3023–3032. `https://doi.org/10.1523/jneurosci.19-08-03023.1999` (Apr. 1999).

43. Petersen, S. A., Fetter, R. D., Noordermeer, J. N., Goodman, C. S. & DiAntonio, A. Genetic Analysis of Glutamate Receptors in Drosophila Reveals a Retrograde Signal Regulating Presynaptic Transmitter Release. *Neuron* **19,** 1237–1248. `https://doi.org/10.1016/s0896-6273(00)80415-8` (Dec. 1997).

44. Aggarwal, A. *et al.* Glutamate indicators with improved activation kinetics and localization for imaging synaptic transmission. *Nature Methods* **20,** 925–934. https://doi.org/10.1038/s41592-023-01863-6 (May 2023).

45. Ascoli, G. A., Maraver, P., Nanda, S., Polavaram, S. & Armañanzas, R. Win–win data sharing in neuroscience. *Nature Methods* **14,** 112–116. https://doi.org/10.1038/nmeth.4152 (Feb. 2017).

46. Ferguson, A. R., Nielson, J. L., Cragin, M. H., Bandrowski, A. E. & Martone, M. E. Big data from small data: data-sharing in the 'long tail' of neuroscience. *Nature Neuroscience* **17,** 1442–1447. https://doi.org/10.1038/nn.3838 (Oct. 2014).

47. Tran, V. & Stricker, C. Spontaneous and action potential-evoked Ca2+ release from endoplasmic reticulum in neocortical synaptic boutons. *Cell Calcium* **97,** 102433. https://doi.org/10.1016/j.ceca.2021.102433 (July 2021).

48. Li, B. *et al.* Two-Photon Voltage Imaging of Spontaneous Activity from Multiple Neurons Reveals Network Activity in Brain Tissue. *iScience* **23,** 101363. https://doi.org/10.1016/j.isci.2020.101363 (Aug. 2020).

49. Campagnola, L. *et al.* Local connectivity and synaptic dynamics in mouse and human neocortex. *Science* **375.** https://doi.org/10.1126/science.abj5861 (Mar. 2022).

50. Theis, L. *et al.* Benchmarking Spike Rate Inference in Population Calcium Imaging. *Neuron* **90,** 471–482. https://doi.org/10.1016/j.neuron.2016.04.014 (May 2016).

51. Ye, L. & Keogh, E. *Time series shapelets* in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining* (ACM, June 2009). https://doi.org/10.1145/1557019.1557122.

52. Batal, I. & Hauskrecht, M. *A Supervised Time Series Feature Extraction Technique Using DCT and DWT* in *2009 International Conference on Machine Learning and Applications* (IEEE, Dec. 2009). https://doi.org/10.1109/icmla.2009.13.

53. Lepeta, K. *et al.* Synaptopathies: synaptic dysfunction in neurological disorders - A review from students to students. *Journal of Neurochemistry* **138,** 785–805. https://doi.org/10.1111/jnc.13713 (Sept. 2016).

54. Zoghbi, H. Y. & Bear, M. F. Synaptic Dysfunction in Neurodevelopmental Disorders Associated with Autism and Intellectual Disabilities. *Cold Spring Harbor Perspectives in Biology* **4,** a009886–a009886. https://doi.org/10.1101/cshperspect.a009886 (Jan. 2012).

55. Asadollahi, R. *et al.* Pathogenic *SCN2A* variants cause early-stage dysfunction in patient-derived neurons. *Human Molecular Genetics* **32,** 2192–2204. https://doi.org/10.1093/hmg/ddad048 (Apr. 2023).

56. Passricha, V. & Aggarwal, R. K. A Hybrid of Deep CNN and Bidirectional LSTM for Automatic Speech Recognition. *Journal of Intelligent Systems* **29,** 1261–1274. https://doi.org/10.1515/jisys-2018-0372 (Mar. 2019).

57. Islam, M. Z., Islam, M. M. & Asraf, A. A combined deep CNN-LSTM network for the detection of novel coronavirus (COVID-19) using X-ray images. *Informatics in Medicine Unlocked* **20,** 100412. https://doi.org/10.1016/j.imu.2020.100412 (2020).

58. Tasdelen, A. & Sen, B. A hybrid CNN-LSTM model for pre-miRNA classification. *Scientific Reports* **11.** https://doi.org/10.1038/s41598-021-93656-0 (July 2021).

59. Martín Abadi *et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems* Software available from tensorflow.org. 2015. https://www.tensorflow.org/.

60. Ioffe, S. & Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *arXiv,* 10.48550/ARXIV.1502.03167. https://arxiv.org/abs/1502.03167 (2015).

61. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. & Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research* **15,** 1929–1958 (2014).

62. Kingma, D. P. & Ba, J. Adam: A Method for Stochastic Optimization. *arXiv,* 10.48550/ARXIV.1412.6980. https://arxiv.org/abs/1412.6980 (2014).

63. Reddi, S. J., Kale, S. & Kumar, S. On the Convergence of Adam and Beyond. *arXiv,* 10.48550/ARXIV.1904.09237. arXiv: 1904.09237 [cs.LG] (2019).

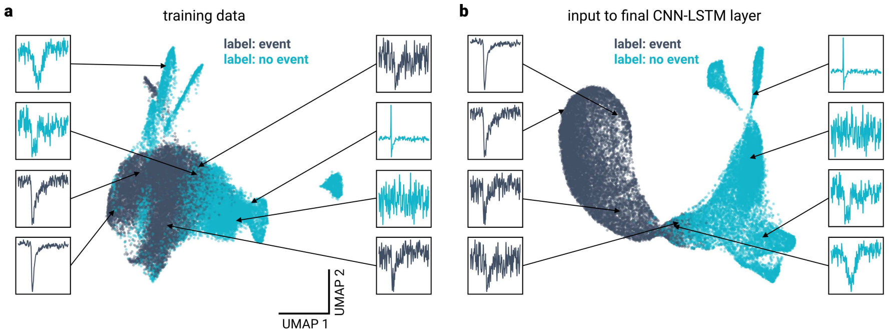
64. Pratt, L. Y., Mostow, J. & Kamm, C. A. *Direct Transfer of Learned Information Among Neural Networks* in *AAAI Conference on Artificial Intelligence* (1991).

65. Ho, J., Tumkaya, T., Aryal, S., Choi, H. & Claridge-Chang, A. Moving beyond P values: data analysis with estimation graphics. *Nature Methods* **16,** 565–566. https://doi.org/10.1038/s41592-019-0470-3 (June 2019).

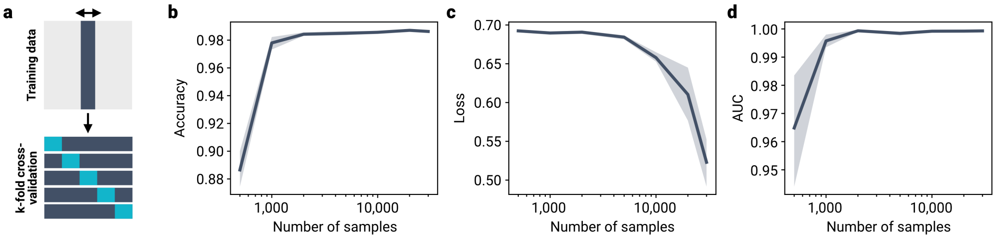
SUPPLEMENTARY DATA

Table 1. Overview of model architecture

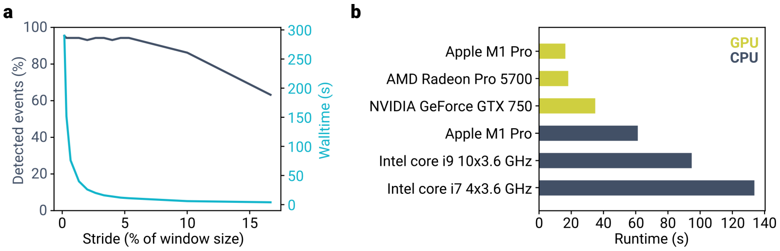
Block	Layer	Output shape	Parameters	Settings	Values
Convolutional Block I	1D Convolutional Layer	(None, 600, 32)	320	Filters Kernel Size Padding	32 9 'same'
	Batch Normalization	(None, 600, 32)	128	All	default
	Leaky ReLU	(None, 600, 32)	0	α	0.3
	Avg. Pooling	(None, 200, 32)	0	Pool Size Strides	3 3
Convolutional Block II	1D Convolutional Layer	(None, 200, 48)	10'800	Filters Kernel Size Padding	48 7 'same'
	Batch Normalization	(None, 200, 48)	192	All	default
	Leaky ReLU	(None, 200, 48)	0	α	0.3
	Avg. Pooling	(None, 100, 48)	0	Pool Size Strides	2 2
Convolutional Block III	1D Convolutional Layer	(None, 100, 64)	15'424	Filters Kernel Size Padding	64 5 'same'
	Batch Normalization	(None, 100, 64)	256	All	default
	Leaky ReLU	(None, 100, 64)	0	α	0.3
	Avg. Pooling	(None, 50, 64)	0	Pool size Strides	2 2
Convolutional Block IV	1D Convolutional Layer	(None, 50, 80)	15'440	Filters Kernel Size Padding	64 5 'same'
	Batch Normalization	(None, 50, 80)	320	All	default
	Leaky ReLU	(None, 50, 80)	0	α	0.3
LSTM Layer	Bidirectional LSTM	(None, 96)	135'936	Units Dropout Rate Merge Mode Activation	96 0.2 'sum' tanh
Fully Connected Layers	Dense	(None, 128)	12'416	Units Activation	128 Leaky ReLU
	Dropout	(None, 128)	0	Dropout Rate	0.2
	Dense	(None, 1)	129	Units Activation	1 sigmoid



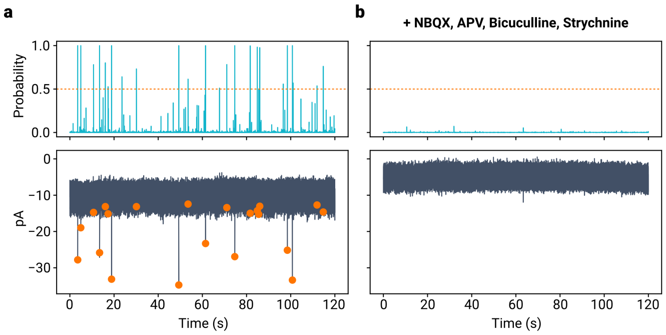
Supplementary Fig. 1. Network architecture and visualization of model training.
a, The miniML model transforms input to enhance separability. Shown is a Uniform Manifold Approximation and Projection (UMAP) dimensionality reduction of the original training dataset. **b**, UMAP of the input to the final ML model layer. Examples of labeled training samples are illustrated. Model training greatly improves linear separability of the two labeled event classes.



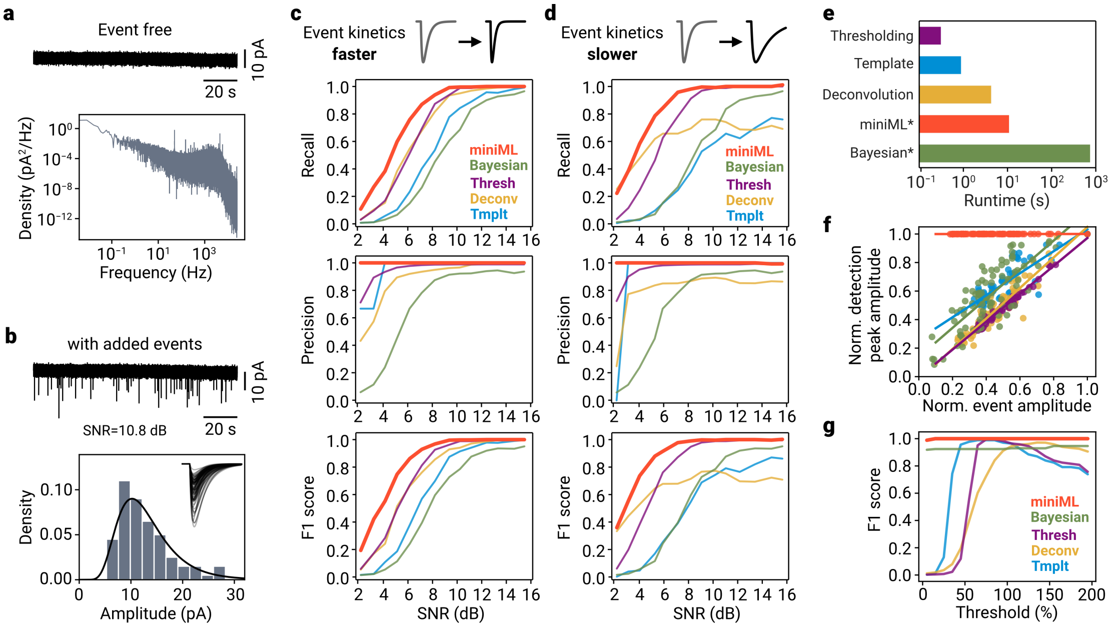
Supplementary Fig. 2. Dependence of training performance on dataset size.
a, To test how size of the training dataset impacts model training, we took random subsamples from the MF–GC dataset and trained miniML models using fivefold cross validation. **b–d**, Comparison of loss (**b**), accuracy (**c**) and area under the ROC curve (AUC; **d**) across increasing dataset sizes. Data are means of model training sessions with k-fold cross-validation. Shaded areas represent SD.



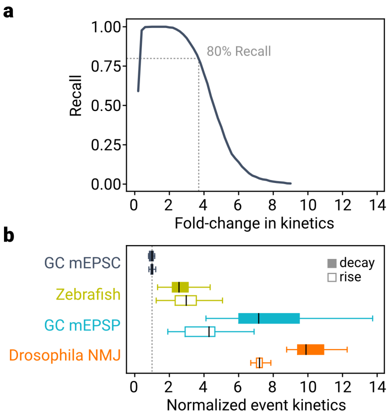
Supplementary Fig. 3. Fast computation time for event detection using miniML.
a, Detected events and analysis runtime plotted versus stride. Note that runtime can be minimized by using stride sizes up to 5% of the event window size without impacting detection performance. **b**, Analysis runtime with different computer hardware for a 120-s long recording at 50 kHz sampling (total of 6,000,000 samples). Runtime is given as wall time including event analysis. GPU computing enables analysis runtimes shorter than 20 s.



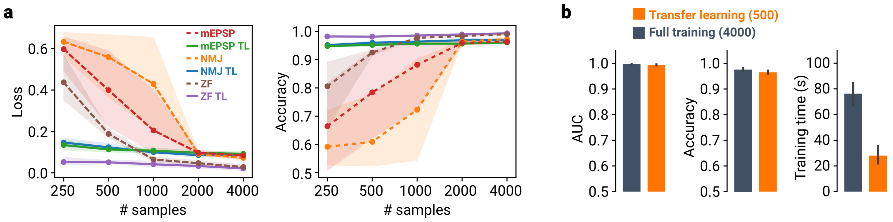
Supplementary Fig. 4. miniML performance on event-free data.
a, Probability (top) and raw data with detected events from a MF–GC recording. Dashed line indicates miniML threshold. **b**, Recording from the same cell after addition of blockers of synaptic transmission (NBQX, APV, Bicuculline, Strychnine). miniML does not detect any events under these conditions. Note that addition of Bicuculline blocks tonic inhibition in cerebellar GCs, causing a reduction in holding current and reduced noise[36].



Supplementary Fig. 5. Extended benchmarking and threshold dependence of event detection.
a, Example event-free recording (Top) and power density spectrum of the data (Bottom). **b**, Top: Data trace from **a** superimposed with simulated synaptic events. Bottom: Amplitude histogram of simulated events for a signal-to-noise ratio (SNR) of 10.8 dB. Solid line represents the log-normal distribution from which event amplitudes were drawn; Inset shows individual simulated events. **c**, Recall, precision and F1 score for five detection methods with simulated events that have a 2× faster decay than in Figure3. **d**, Same as in **c**, but for events with a 3.5× slower decay time constant. Note that we did not adjust detection hyperparameters for the comparisons in **c** and **d**. **e**, Runtime of five different synaptic event detection methods for a 120-s section of data recorded with 50 kHz sampling rate. Average wall-time of five runs. * miniML was run using a GPU, and data were down-sampled to 20 kHz for the Bayesian analysis. **f**, Normalized amplitude of the peaks in the different method's detection traces ("detection peak amplitude") versus normalized event amplitude. A linear dependence indicates that detection traces contain implicit amplitude information, which will cause a strong threshold dependence of the detection result. Note that miniML's output does not contain amplitude information. **g**, F1 score versus threshold (in % of default threshold value, range 5–195) for different methods.



Supplementary Fig. 6. Recall depends on event kinetics.
a, Recall versus event kinetics for the MF–GC model. Kinetics (i.e., rise and decay time constants) of simulated events were changed as indicated. miniML robustly detects events with up to ~ 4-fold slower kinetics (dashed line indicates 80% recall). **b**, Event kinetics for different preparations and/or recording modes. Data are normalized to MF–GC mEPSCs (dashed line).



Supplementary Fig. 7. Transfer learning facilitates model training across different datasets.
a, Loss and accuracy versus number of training dataset samples for three different datasets (MF–GC mEPSPs, *Drosophila* NMJ mEPSCs, zebrafish spontaneous EPSCs). Dashed lines indicate transfer learning (TL), whereas solid lines were obtained by full training. Points are averages of fivefold cross-validation and shaded areas represent 95% CI. **b**, Average AUC, accuracy, and training time for TL using 500 samples, and full training using 4,000 samples. Error bars denote 95% CI.