`SqueakOut`: Autoencoder-based segmentation of mouse ultrasonic vocalizations

Gustavo M. Santana
Laboratory of Physiology of Behavior,
Interdepartmental Neuroscience Program,
Program in Physics, Engineering and Biology,
Yale University, USA
Graduate Program in Biochemistry,
Federal University of Rio Grande do Sul, BRA
gustavo.santana@yale.edu

Marcelo O. Dietrich
Laboratory of Physiology of Behavior,
Department of Comparative Medicine,
Department of Neuroscience,
Yale University, USA
marcelo.dietrich@yale.edu

Abstract

Mice emit ultrasonic vocalizations (USVs) that are important for social communication. Despite great advancements in tools to detect USVs from audio files in the recent years, highly accurate segmentation of USVs from spectrograms (i.e., removing noise) remains a significant challenge. Here, we present a new dataset of `12,954` annotated spectrograms explicitly labeled for mouse USV segmentation. Leveraging this dataset, we developed `SqueakOut`, a lightweight (`4.6M` parameters) fully convolutional autoencoder that achieves high accuracy in supervised segmentation of USVs from spectrograms, with a *Dice* score of `90.22`. `SqueakOut` combines a MobileNetV2 backbone with skip connections and transposed convolutions to precisely segment USVs. Using stochastic data augmentation techniques and a hybrid loss function, `SqueakOut` learns robust segmentation across varying recording conditions. We evaluate SqueakOut's performance, demonstrating substantial improvements over existing methods like VocalMat (`63.82` *Dice* score). The accurate USV segmentations enabled by `SqueakOut` will facilitate novel methods for vocalization classification and more accurate analysis of mouse communication. To promote further research, we release the annotated `12,954` spectrogram USV segmentation dataset and the `SqueakOut` implementation publicly.

1 Introduction

Vocalizations are an important form of social communication among many animals, including mice [1]. Mice produce ultrasonic vocalizations (USVs) in various behavioral contexts such as maternal interactions [2, 3], social exploration [4], courtship [5], and distress situations like maternal separation [6, 7, 8, 9, 10, 11]. Although these high-frequency calls are inaudible to humans, they convey rich information about the animal's internal state and behavioral experience [12, 13, 14]. For example, the emission of USVs by mouse pups when separated from their mother evokes maternal behavior and activates selective pathways in the brain of mothers [15, 16, 17, 18, 19]. Moreover, a detailed analysis of USVs is a powerful way to phenotype mouse models of neurodevelopmental disorders [20], genomic imprinting [16], pharmacological manipulations [21], as well as to perform comparative studies among different rodent species [22]. Thus, vocalizations provide a unique window into animals' internal states and a rich framework for a better understanding of animal behavior and brain function.

The detailed analysis of USVs poses significant challenges due to the complexity of extracting precise spectrotemporal features. This general problem can be broadly divided into three components: detection, classification, and segmentation of USVs. Each component serves a specific purpose and presents unique challenges [23]. Detection involves identifying and distinguishing actual vocalization signals from periods of silence or noise in audio recordings (**Figure 1A**). Accurate detection is crucial for the subsequent stages of analysis to be correct and meaningful (**Figure 1B**). The detected USVs can be categorized into different classes, typically based on various acoustic features or specific characteristics identified on spectrograms of each vocalization [24]. Classification allows grouping similar sounds, aiding in studying the diversity and distribution of USVs across different contexts [1]. The third component is segmentation (**Figure 1B**), which involves further breaking down each detected USV into discrete spectrotemporal units [25, 26]. Effective segmentation of USVs, akin to parsing words into distinct syllables, enables detailed analysis of the structure of each vocalization [27].

Traditional methods for detecting and analyzing rodent USVs have relied heavily on manual annotation or semi-automated techniques based on predefined parameters and thresholds [28, 29]. These approaches are labor-intensive, time-consuming, and prone to human bias and error, especially when dealing with large datasets or complex vocal repertoires. Furthermore, they often fail to capture the nuanced spectrotemporal variations present in USVs, which may hold crucial information about the animal's motivational state [13, 20]. Recent work has explored the application of computer vision techniques and machine learning models to USV audio recordings and spectrograms as a data-driven alternative [30, 26, 31, 26]. In particular, convolutional neural network (CNN) architectures have shown promising performance in detecting and classifying USVs [24, 30, 32]. Despite the significant progress made in the detection and classification of USVs, accurately segmenting USVs in spectrograms remains a challenge [30, 23, 31].

Here, we present a machine learning dataset explicitly designed for the segmentation of mouse USVs. Using this dataset, we also developed `SqueakOut`, a fully convolutional autoencoder for accurate USV segmentation. We evaluate `SqueakOut`'s performance, compare it to VocalMat, and demonstrate the utility of the segmented USVs for downstream analysis tasks. Lastly, we discuss insights gained from the model and potential applications to high-throughput USV phenotyping workflows.

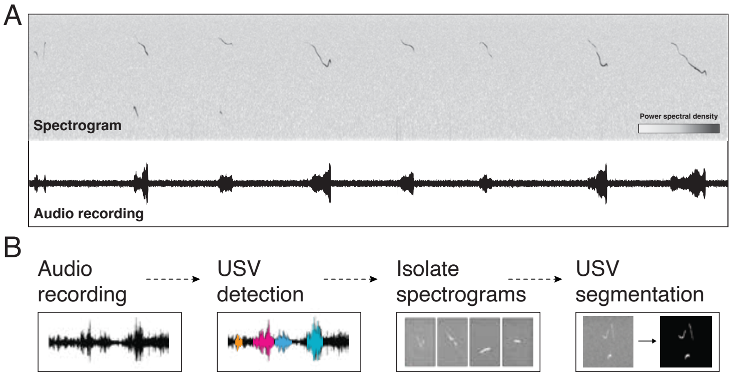

Figure 1: Overview of vocalization analysis pipeline. **(A)** The process begins with an audio recording, which is then converted into a spectrogram representation. **(B)** Individual vocalizations are detected, and isolated spectrograms are segmented and classified for downstream analysis.

2 Results

2.1 Creating a USV segmentation dataset

Creating a mouse USV segmentation dataset is a laborious task. One major problem is the intensive work required to produce accurate annotations by experts. Here, we generated an accurate segmentation dataset using both automated and manual approaches.

First, we took advantage of the publicly available dataset from VocalMat ([24]) and used it as a starting point (**Figure 2A**). The dataset consists of `12,954` spectrograms, including `2,083` spectrogram examples of noise and 10,871 spectrograms of mouse USVs. The dataset includes vocalizations from male and female mice of five different strains (C57Bl6/J, NZO/HlLtJ, 129S1/SvImJ, NOD/ShiLtJ, and PWK/PhJ), ranging from postnatal day 5 to postnatal day 15.

2.1.1 Automated approaches for creating a USV segmentation dataset

VocalMat uses computer vision and machine learning to detect, segment, and classify vocals. While VocalMat's USV detection rate achieves state-of-the-art results, its segmentation performance is suboptimal when noise is present in spectrograms (see **Figure 3C**), making the segmentation masks unsuitable for directly training a segmentation neural network. To enhance the VocalMat dataset, we used its segmentation masks as a starting point and trained an autoencoder to learn the segmentation task. We trained the autoencoder separately on spectrograms of USVs and noise to enable it to learn representations of vocalizations and noise. This approach allowed us to use the autoencoder to denoise the original segmentation masks. The following automated processing steps result in a dataset with segmentation masks of vocalizations with fewer noise segments.

(1) Unsupervised spectrogram reconstruction task: We begin by using U-Net, an autoencoder broadly used for biomedical image segmentation [33] (**Figure 2B**), and train the network on the VocalMat dataset for the unsupervised task of spectrogram reconstruction (**Figure 2C-1**). In this step, the autoencoder receives the spectrograms as input and is trained to reconstruct the input spectrogram as its output. The autoencoder has a bottleneck architecture to ensure that it learns a meaningful representation of the data instead of simply copying its input as the output. The autoencoder can reconstruct spectrograms with high accuracy ($91.67\% \pm 2.08\%$; mean ± SEM).

(2) USV segmentation task: Next, using the pre-trained autoencoder from the previous step, we trained the autoencoder for the task of spectrogram segmentation (**Figure 2C-2**). The autoencoder receives spectrograms of USVs as inputs and outputs binary segmentation masks, i.e., images containing 0 and 1, where 1 indicate USV segments. Initially, we use the segmentation produced by VocalMat as the ground truth annotations for this training step.

(3) Noise segmentation task: Similarly to the previous step, we trained the autoencoder for the task of spectrogram segmentation but only using the spectrograms of noise (**Figure 2C-3**). This allows the autoencoder to learn and distinguish between representations of noise and actual vocalizations in the spectrograms.

(4) Enhancing the dataset: Following the pre-training steps, we now use the autoencoder to perform inference over the original VocalMat dataset for the task of spectrogram segmentation (**Figure 2C-4**). Since the autoencoder has learned the task of spectrogram segmentation and representations of noise, it can produce segmentation masks similar to those generated by VocalMat but with fewer false positives (spectrograms containing only noise: VocalMat $2.85\% \pm 1.27\%$; Autoencoder $78.91\% \pm 4.86\%$; values are the *Dice* score mean ± SEM) (**Figure 3A**).

2.1.2 Manual fine-tuning of the USV segmentation dataset

The generated dataset following the automated steps is significantly better than the original dataset (**Figure 3A**) but still requires fine-tuning. VocalMat's algorithms produce segmentation masks that exceed the size of the actual vocalizations, capturing both vocal-related and surrounding pixels. (**Figure 3B**). We address this *border effect* by using the morphological image processing technique known as erosion. This thinning process effectively reduces the size of the segmentation masks, mitigating the *border effect* (**Figure 3B**). We apply erosion to all vocalization segments at least four times as large as the erosion kernel (4×2 pixels) to prevent the thinning out of already small vocalizations.

Following the *border* thinning process, we repeat automated steps (2) through (4). This sequence of automated and manual refinements is repeated until erosion can no longer be applied to any vocalization segments.

The final step in creating the dataset involves manual refinement. Despite having over two thousand noise examples from the VocalMat dataset, certain types of noise are exclusively present in spectrograms containing vocalizations. Consequently, these noise segments are incorrectly detected as

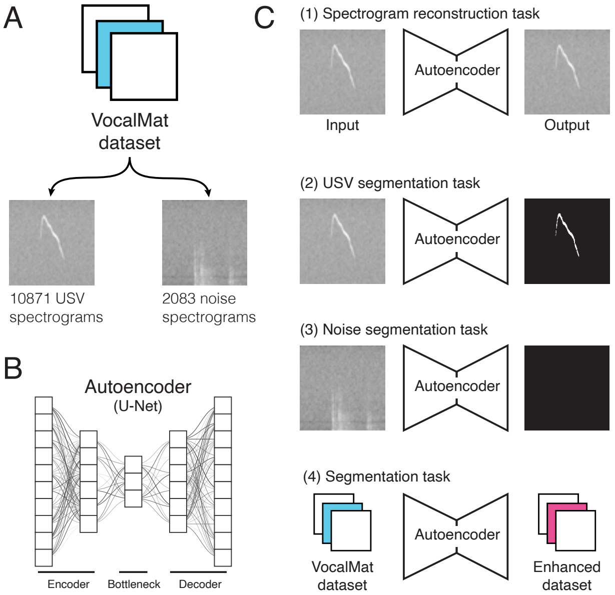

Figure 2: Automated methods for creating the USV segmentation dataset. We use the VocalMat dataset [24] which contains 10,871 USV spectrograms and 2,083 noise spectrograms **(A)**, and the U-Net autoencoder architecture [33] **(B)**. The autoencoder is trained on a spectrogram reconstruction task **(C1)**. The pre-trained autoencoder is then trained on a segmentation task using USV spectrograms **(C2)** and on noise spectrograms **(C3)**. The trained U-Net model is used to enhance the original VocalMat dataset by generating segmentation masks that are mostly devoid of noise components **(C4)**.

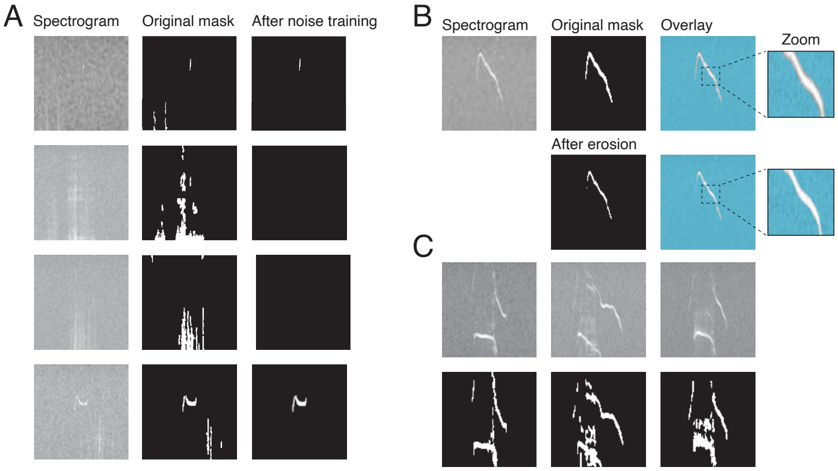

Figure 3: Dataset fine-tuning processing. **(A)** Comparison of an original VocalMat noisy mask and the trained autoencoder mask. **(B)** Illustration of the *border thinning* process through erosion operation applied to a segmentation mask. **(C)** Three examples illustrating noise overlapping with USVs in spectrograms and the segmentation produced by VocalMat, highlighting the challenges in accurate vocalization segmentation.

vocalizations (**Figure 3C**). An expert annotator manually inspected all segmentation masks using the image annotation and segmentation tool `RectLabel`[1], and corrected the masks for a small subset (approximately 15%) of the data.

[1] https://rectlabel.com

2.2 SqueakOut: Autoencoder for mouse USV segmentation

2.2.1 Network architecture

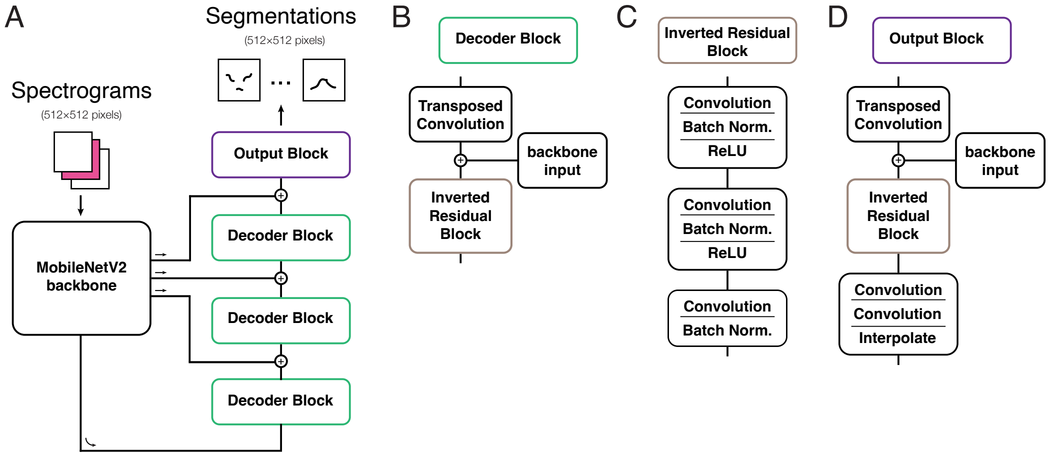

Figure 4: SqueakOut network architecture combines a MobileNetV2 backbone with an encoder-decoder structure. **(A)** SqueakOut takes input spectrograms and outputs segmentations masks (both 512×512 pixels). Spectrograms are processed through the backbone encoder, and then through a series of decoder blocks (shown in **B** and **C**) with skip connections and transposed convolutions and an output block (shown in **D**) to produce segmentation masks.

SqueakOut is a fully convolutional autoencoder that generates segmentation masks of vocalizations from spectrograms. The architecture is depicted in **Figure 4**. SqueakOut uses a modified MobileNetV2 [34] as its backbone for its small memory footprint and efficient processing using inverted residuals, depth-wise convolutions, and lack of explicit non-linearities in the narrow layers. Specifically, we removed the *average pooling* layer and added a *dropout* layer before the final bottleneck layer with a 20% rate.

The decoder path for SqueakOut uses skip connections from backbone layers and transposed convolutions to reconstruct segmentation masks. Skip connections have been demonstrated to enhance segmentation accuracy and capture fine-grained details [35, 36, 37]. Moreover, skip connections improve gradient propagation and convergence during training [37, 38]. In SqueakOut, we concatenate the input from backbone layers with the input from the previous layer in the decoder path, ensuring the propagation of intact information from early layers in the network. Lastly, we employ a series of convolutional layers in the output block and use interpolation to upsample the output, restoring it to the same spatial dimensions as the original input spectrogram.

2.2.2 Training

SqueakOut is implemented in PyTorch [39] using the PyTorchLightning framework [40] and was trained on the enhanced VocalMat dataset. A subset of the dataset containing 849 USVs was used as the *test* set. The remaining dataset was randomly split into *training* (90%) and *validation* (10%) sets. SqueakOut was trained using Adam [41] with a learning rate of $1e^{-4}$. The learning rate was reduced by a factor of 0.1 if performance on the *validation* set did not improve for five consecutive iterations. Training was halted if the performance did not improve for fifteen consecutive iterations to avoid overfitting. A batch size of eight samples was used. The loss function was a weighted sum of the Focal loss (FL) [42]—which emphasizes hard data points and prevents easy negatives from dominating the loss during training—, and the Dice loss (DL) [43]—a measure of the similarity between the network output and the ground-truth segmentation. We chose this hybrid loss function because of the extreme imbalance in class labels for the segmentation task. The majority of pixels in a spectrogram represent background, with USVs accounting for, on average, less than 5% of pixels. Briefly,

$$\mathbf{FL} = -(1 - p_n)^\gamma log(p_n) \tag{1}$$

$$\mathbf{DL} = 1 - 2 * \frac{\sum_{n=1}^{N} p_n s_n + \epsilon}{\sum_{n=1}^{N} p_n + \sum_{n=1}^{N} s_n + \epsilon} \tag{2}$$

$$\mathbf{L} = \alpha FL + (1 - \alpha) DL \tag{3}$$

where p_n is the predicted segmentation probability map, and s_n is the ground-truth segmentation map for a spectrogram. We include a small term ϵ in the *Dice* loss to prevent division by 0. In SqueakOut, we use $\gamma = 2$ and $\alpha = 0.3$.

2.2.3 Data augmentations

To further enrich our dataset, we utilized data augmentation techniques. The segmentation task is unique in that any augmentations made on the spectrogram can be identically applied to its matching segmentation mask. Importantly, this would not hold for a classification task. For example, if a spectrogram is randomly warped such that the morphological characteristics of a USV changes, then the corresponding classification of that USV would also likely change in unpredictable ways.

To make SqueakOut robust to changes in spectrogram quality and better generalize, we use the data augmentations depicted in **Figure 5**. Briefly, augmentations were applied during training on a batch-by-batch basis with the following conditions:

(a) 75% chance of applying a random affine transformation

(b) 25% chance of applying contrast normalization

(c) 25% chance of applying a Gaussian blur

(d) 33% chance of applying one of the following:

- Additive noise
- Gaussian noise
- Frequency noise
- Elastic transformations

Augmentations for each of the four conditions (*a-d*) have an independent probability of being applied to a given batch of spectrograms. This means that for each training batch, any combination of the four augmentations can occur, ranging from no augmentations to all four being applied simultaneously. The stochastic nature of this augmentation strategy helps to increase the diversity of the training data and improve SqueakOut's robustness to variations in the input spectrograms. For example, noise and contrast augmentations especially improve segmentation in spectrograms with low signal-to-noise ratios. Affine and elastic transformations are intended to make SqueakOut robust to USV morphologies not present in our dataset and improve USV contour segmentation quality.

2.3 USV segmentation performance

SqueakOut is a lightweight autoencoder model at only 18MB (4.6M parameters) that is fast and achieves high accuracy. Inference on a batch of 64 512×512 pixels spectrograms on a gaming GPU takes less than 0.035 seconds, and about 8 seconds on a CPU.

Vocalizations on spectrograms are generally small, leading to an imbalance in class labels (0 or 1) for the segmentation task. We therefore created a null model, which treats every spectrogram as if it contains no USVs, i.e., it always generates blank segmentation masks (all values are 0). The null model achieves 99.09% accuracy, showing that most pixels in a spectrogram are indeed background. Similarly, all models achieve relatively high pixel-wise accuracy (**Table 1**). In contrast, the null model achieves a 4.95 *Dice* score. The *Dice* score measures the overlap between two segmentation masks by weighing their intersection against their union (**Equation 2**), where a score of 100 means perfect overlap. It effectively balances false positive and false negative rates, and, therefore, we use the *Dice* coefficient as our primary performance score metric.

We first compared SqueakOut's performance with VocalMat and applied the same metrics to the output of both tools (**Table 1**). VocalMat is accurate in segmenting any high-intensity segments in

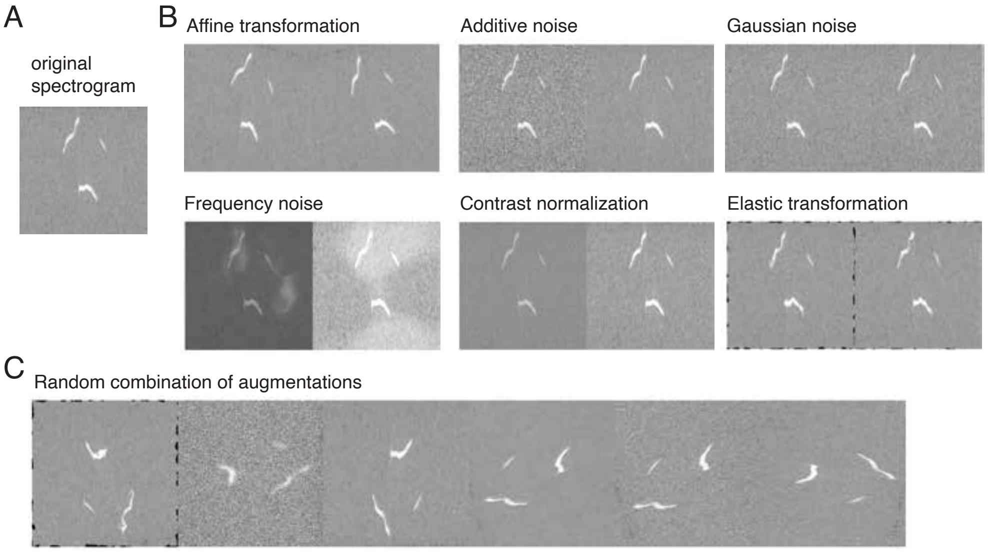

Figure 5: Dataset augmentation techniques applied to spectrograms for training robust segmentation models. **(A)** An example USV spectrogram. **(B)** Illustration of individual data augmentation techniques applied to the example spectrogram. **(C)** Illustration of augmentation techniques when jointly applied to the example spectrogram.

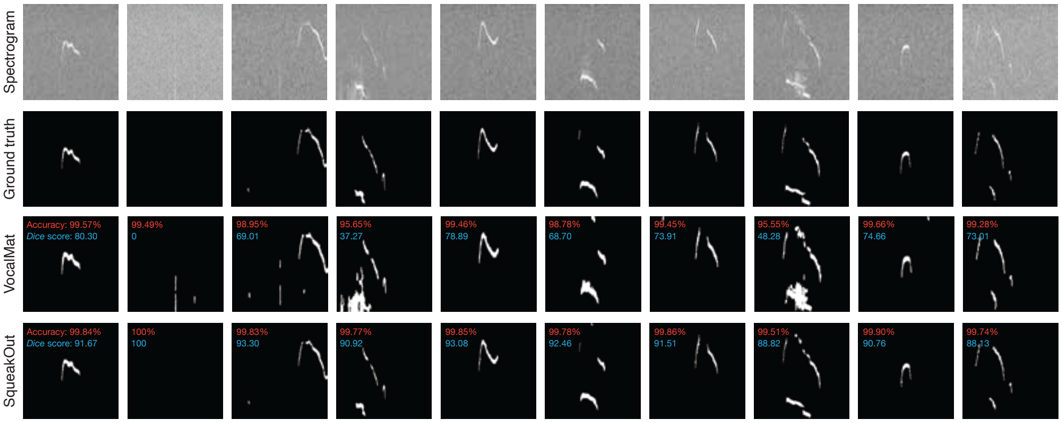

Figure 6: Comparison of USV spectrogram segmentation performance between the ground truth annotations, the VocalMat dataset, and the proposed `SqueakOut` method. The spectrograms illustrate `SqueakOut`'s ability to accurately segment USVs compared to the ground truth and the baseline VocalMat dataset. Inset values in red represent the pixel-wise accuracy, while those in blue indicate the *Dice* score.

spectrograms, including noise (**Figure 6**), resulting in a significantly lower *Dice* score. In noise-free recordings, we expect that VocalMat's performance would be qualitatively similar to `SqueakOut`. We also compared `SqueakOut` with another autoencoder architecture for image segmentation, U-Net. We trained the U-Net model and `SqueakOut` using the same dataset, without any data augmentations. Nevertheless, the U-Net model performs worse than `SqueakOut` (no augmentations) (**Table 1**), showing that `SqueakOut`'s high *Dice* score is not solely due to our refined dataset, but also to its architecture. Importantly, the data augmentation techniques we employed improved `SqueakOut`'s *Dice* score by 8.72% (last two rows in **Table 1**).

Table 1: USV segmentation performance across models. Values are mean ± SEM.

	Performance	
Model	**Accuracy (%)**	***Dice* score**
Null Model	`99.10` ± `0.02`	`4.95` ± `0.74`
VocalMat [24]	`98.83` ± `0.06`	`63.82` ± `0.71`
U-Net [33]	`98.91` ± `0.04`	`72.31` ± `0.83`
`SqueakOut` (no augmentations)	`99.03` ± `0.02`	`82.98` ± `0.65`
`SqueakOut`	`99.84` ± `0.01`	`90.22` ± `0.41`

3 Discussion

Here, we first present a new mouse USV segmentation dataset that is made publicly available and can be used by any group for machine learning applications. We leveraged this unique dataset to develop `SqueakOut`, a fully convolutional autoencoder for supervised segmentation of USV spectrograms. Our results demonstrate `SqueakOut`'s ability to segment USVs with very high accuracy. Our segmentation dataset joins the VocalMat dataset to provide a single high-quality annotated dataset for mouse USV detection and segmentation.

Autoencoders have been used across a broad spectrum of research in various species to analyze acoustic communication [44, 45, 30, 46, 47]. The popularity of autoencoders is partly due to their inherent ability to learn latent data structures in an unsupervised fashion, i.e., without requiring annotated datasets. This can be advantageous when the features of the data relevant for analysis are unknown or for unbiased data analysis. However, the extracted latent features are often hard to interpret. On the other hand, supervised methods that use pre-defined features are interpretable but can lead to biased analysis depending on the features chosen by the experimenter.

Unsupervised methods that attempt to learn latent features of vocals using their spectrograms often suffer from variability in the quality of recordings. Any variability in recording conditions will result in drastically different background noise levels and signal-to-noise ratios in spectrograms. This variability will affect what the network learns and can make latent features even harder to interpret. Unsupervised methods have to be carefully tuned to the specifics of a dataset, and an extensive list of methods have been developed to deal with varying quality in spectrograms [47, 23]. `SqueakOut` can produce accurate USV segmentations, effectively removing any variability due to recording conditions. The resulting segmentation masks can be used for downstream analysis using unsupervised methods such as Variational Autoencoders [45, 30] and dimensionality reduction techniques such as UMAP [48] or diffusion maps [49] to exploit the combined advantages of unsupervised and supervised methods.

Vocalizations in mice are powerful indicators of their emotional and behavioral states, and classification of these vocalizations is important for linking behavior with brain function across different contexts [1]. Historically popular methods have used hand-crafted features such as the duration, frequency modulation, amplitude, and other characteristics of each USV. These methods heavily relied on the quality of the segmentation and were therefore not incredibly precise nor high-throughput. Recently, CNNs have become the standard models for supervised image classification and are widely used for USV call type classification but lack spatiotemporal measurements (e.g., USV duration or average frequency). However, with accurate segmentations such as those produced by `SqueakOut`, the use of hand-crafted features and traditional machine learning methods such as random forests can reemerge as efficient and powerful alternatives for studying USV diversity across behaviors using interpretable features.

In conclusion, this work presents a new publicly available dataset for mouse USV segmentation, which we believe will be a valuable resource for the research community. We demonstrate the utility of this dataset by developing `SqueakOut`, a fully convolutional autoencoder that achieves high accuracy in supervised USV segmentation. The combination of our segmentation dataset with the existing VocalMat dataset provides a comprehensive, high-quality annotated resource for USV detection and segmentation. By providing accurate segmentation tools, we aim to enable more precise and powerful methods for USV classification and analysis, facilitating novel approaches for studying mouse communication and neurobiology.

4 Materials and methods

Mouse USV dataset The annotated dataset for mouse USV segmentation is openly available on Open Science Framework [50] at https://osf.io/f9sbt/. We welcome contributions from the community. Anyone may submit corrections or newly annotated audio recordings to be included in the dataset. A similar approach to this work can be used to generate segmentation labels.

SqueakOut architecture SqueakOut was implemented in PyTorch `v1.7.0` [39]. The network implementation is available at https://github.com/gumadeiras/squeakout. Here we provide a brief overview of the PyTorch functions utilized to implement each layer: convolution (`Conv2d`), batch normalization (`BatchNorm2d`), ReLU6 (`ReLU6`), dropout (`Dropout`), transposed convolution (`ConvTranspose2d`), and upsampling (`Interpolate`).

Pre-trained SqueakOut network The SqueakOut network implementation and pre-trained weights are available at https://github.com/gumadeiras/squeakout. We provide scripts to train SqueakOut on a new dataset or perform inference on your data. We developed and have tested SqueakOut using Python `3.7.10`, NumPy `v1.21.5` [51], Scikit-Learn `v1.0.2` [52], PyTorch `v1.7.0` [39], Torchvision `0.8.0`, PyTorchLightning `v1.4.0` [40], and the image augmentation library `imgaug v0.4.0`.

Segmentation metrics We used two metrics to quantify segmentation performance: pixel-wise accuracy, and the *Dice* score. Pixel-wise accuracy was computed as follows:

$$\textbf{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

where TP are the true positives (correctly predicting a pixel belongs to a vocalization), TN the true negatives (correctly predicting a pixel belongs to the background), FP the false positives (incorrectly predicting a pixel belongs to a vocalization), and FN the false negatives (incorrectly predicting a pixel belongs to the background). The *Dice* score was computed as described in **Equation 2**.

Acknowledgments and Disclosure of Funding

We thank Antonio Fonseca for insightful discussions on the early development of the segmentation dataset. We thank Sarah Mohr for suggesting the name `SqueakOut`. M.O.D. was supported by the National Institute of Mental Health of the National Institutes of Health (R01MH125008 and R01MH130825), by the Smith Family Foundation, and by discretionary funds from the Yale School of Medicine. G.S. was supported in part by the Coordination for the Improvement of Higher Education Personnel – Brazil (CAPES) – Finance Code 001.

References

[1] Kylie Yao et al. "A review of ultrasonic vocalizations in mice and how they relate to human speech." In: *The Journal of the Acoustical Society of America* 154 2 (2023), pp. 650–660. URL: https://api.semanticscholar.org/CorpusID:260485538.

[2] E. Noirot. "Serial order of maternal responses in mice." In: *Animal behaviour* 17 3 (1969), pp. 547–50.

[3] K. Kuroda et al. "ERK-FosB signaling in dorsal MPOA neurons plays a major role in the initiation of parental behavior in mice". In: *Molecular and Cellular Neuroscience* 36 (2007), pp. 121–131.

[4] A. Moles et al. "Ultrasonic vocalizations emitted during dyadic interactions in female mice: A possible index of sociability?" In: *Behavioural Brain Research* 182 (2007), pp. 223–230.

[5] J. Nyby. "Ultrasonic vocalizations during sex behavior of male house mice (Mus musculus): a description." In: *Behavioral and neural biology* 39 1 (1983), pp. 128–34.

[6] M. Scattoni and I. Branchi. "Vocal repertoire in mouse pups: strain differences". In: *Handbook of Behavioral Neuroscience* 19 (2010), pp. 89–95.

[7] Igor Branchi, Daniela Santucci, and Enrico Alleva. "Ultrasonic vocalisation emitted by infant rodents: a tool for assessment of neurobehavioural development". In: *Behavioural Brain Research* 125.1 (2001), pp. 49–56. ISSN: 0166-4328. DOI: https://doi.org/10.1016/S0166-4328(01)00277-7. URL: https://www.sciencedirect.com/science/article/pii/S0166432801002777.

[8] Igor Branchi, Patrizia Campolongo, and Enrico Alleva. "Scopolamine effects on ultrasonic vocalization emission and behavior in the neonatal mouse". In: *Behavioural Brain Research* 151.1 (2004), pp. 9–16. ISSN: 0166-4328. DOI: https://doi.org/10.1016/S0166-4328(03)00277-8. URL: https://www.sciencedirect.com/science/article/pii/S0166432803002778.

[9] M. Hofer. "Multiple regulators of ultrasonic vocalization in the infant rat". In: *Psychoneuroendocrinology* 21 (1996), pp. 203–217.

[10] M. Myers et al. "Brief maternal interaction increases number, amplitude, and bout size of isolation-induced ultrasonic vocalizations in infant rats (Rattus norvegicus)." In: *Journal of comparative psychology* 118 1 (2004), pp. 95–102.

[11] H. Shair et al. "Social, thermal, and temporal influences on isolation-induced and maternally potentiated ultrasonic vocalizations of rat pups." In: *Developmental psychobiology* 42 2 (2003), pp. 206–22.

[12] G. Ehret. "Infant Rodent Ultrasounds – A Gate to the Understanding of Sound Communication". In: *Behavior Genetics* 35 (2005), pp. 19–29.

[13] Talmo D. Pereira, Joshua W. Shaevitz, and Mala Murthy. "Quantifying behavior to understand the brain". In: *Nature Neuroscience* 23 (2020), pp. 1537–1549. URL: https://api.semanticscholar.org/CorpusID:226295697.

[14] Jaehong Park et al. "Brainstem control of vocalization and its coordination with respiration". In: *Science* 383 (2024). URL: https://api.semanticscholar.org/CorpusID:268263624.

[15] Jennifer K. Schiavo et al. "Innate and plastic mechanisms for maternal behaviour in auditory cortex". In: *Nature* 587 (2020), pp. 426–431. URL: https://api.semanticscholar.org/CorpusID:222213659.

[16] Gabriela Bosque Ortiz, Gustavo M. Santana, and Marcelo O. Dietrich. "Deficiency of the paternally inherited gene Magel2 alters the development of separation-induced vocalization and maternal behavior in mice". In: *Genes, Brain, and Behavior* 21 (2021). DOI: 10.1111/gbb.12776. URL: https://api.semanticscholar.org/CorpusID:233175640.

[17] William P. Smotherman et al. "Maternal responses to infant vocalizations and olfactory cues in rats and mice". In: *Behavioral Biology* 12.1 (1974), pp. 55–66. ISSN: 0091-6773. DOI: https://doi.org/10.1016/S0091-6773(74)91026-8. URL: https://www.sciencedirect.com/science/article/pii/S0091677374910268.

[18] Eliane Noirot. "Ultrasounds and maternal behavior in small rodents". In: *Developmental Psychobiology* 5.4 (1972), pp. 371–387. DOI: https://doi.org/10.1002/dev.420050410. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/dev.420050410. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/dev.420050410.

[19] G. Ehret. "Categorical perception of mouse-pup ultrasounds in the temporal domain". In: *Animal Behaviour* 43.3 (1992), pp. 409–416. ISSN: 0003-3472. DOI: https://doi.org/10.1016/S0003-3472(05)80101-0. URL: https://www.sciencedirect.com/science/article/pii/S0003347205801010.

[20] María Luisa Scattoni, Jacqueline N. Crawley, and Laura Ricceri. "Ultrasonic vocalizations: A tool for behavioural phenotyping of mouse models of neurodevelopmental disorders". In: *Neuroscience & Biobehavioral Reviews* 33 (2009), pp. 508–515. URL: https://api.semanticscholar.org/CorpusID:5114496.

[21] Marcelo R. Zimmer et al. "Functional Ontogeny of Hypothalamic Agrp Neurons in Neonatal Mouse Behaviors". In: *Cell* 178 (2019), 44–59.e7. URL: https://api.semanticscholar.org/CorpusID:206570794.

[22] Nicholas Jourjine et al. "Two pup vocalization types are genetically and functionally separable in deer mice". In: *Current Biology* 33 (2022), 1237–1248.e4. URL: https://api.semanticscholar.org/CorpusID:253524825.

[23] Dan Stowell. "Computational bioacoustics with deep learning: a review and roadmap". In: *PeerJ* 10 (2021). URL: https://api.semanticscholar.org/CorpusID:245123578.

[24] Antonio HO Fonseca et al. "Analysis of ultrasonic vocalizations from mice using computer vision and machine learning". In: *eLife* 10 (Mar. 2021), e59161. ISSN: 2050-084X. DOI: 10.7554/eLife.59161.

[25] Kevin R. Coffey, Russell G. Marx, and John F. Neumaier. "DeepSqueak: a deep learning-based system for detection and analysis of ultrasonic vocalizations". In: *Neuropsychopharmacology* 44 (2019), pp. 859–868. URL: https://api.semanticscholar.org/CorpusID:58558514.

[26] Vasiliki Stoumpou et al. "Analysis of Mouse Vocal Communication (AMVOC): a deep, unsupervised method for rapid detection, analysis and classification of ultrasonic vocalisations". In: *Bioacoustics* 32 (2022), pp. 199–229. URL: https://api.semanticscholar.org/CorpusID:237099964.

[27] Yarden Cohen et al. "Automated annotation of birdsong with a neural network that segments spectrograms". In: *eLife* 11 (2022). URL: https://api.semanticscholar.org/CorpusID:246078371.

[28] Maarten Van Segbroeck et al. "MUPET—Mouse Ultrasonic Profile ExTraction: A Signal Processing Tool for Rapid and Unsupervised Analysis of Ultrasonic Vocalizations". In: *Neuron* 94 (2017), 465–485.e5. URL: https://api.semanticscholar.org/CorpusID:207218731.

[29] Joshua P. Neunuebel et al. "Female mice ultrasonically interact with males during courtship displays". In: *eLife* 4 (2015). URL: https://api.semanticscholar.org/CorpusID:18787332.

[30] Tim Sainburg, Marvin Thielk, and Timothy Q. Gentner. "Finding, visualizing, and quantifying latent structure across diverse animal vocal repertoires". In: *PLoS Computational Biology* 16 (2020). URL: https://api.semanticscholar.org/CorpusID:219700149.

[31] Ryosuke O. Tachibana et al. "USVSEG: A robust method for segmentation of ultrasonic vocalizations in rodents". In: *PLoS ONE* 15 (2020). URL: https://api.semanticscholar.org/CorpusID:211078737.

[32] Daniele Baggi et al. "Extended performance analysis of deep-learning algorithms for mice vocalization segmentation". In: *Scientific Reports* 13 (2023). URL: https://api.semanticscholar.org/CorpusID:259833038.

[33] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. "U-Net: Convolutional Networks for Biomedical Image Segmentation". In: *ArXiv* abs/1505.04597 (2015). URL: https://api.semanticscholar.org/CorpusID:3719281.

[34] Mark Sandler et al. "MobileNetV2: Inverted Residuals and Linear Bottlenecks". In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2018), pp. 4510–4520. URL: https://api.semanticscholar.org/CorpusID:4555207.

[35] Evan Shelhamer, Jonathan Long, and Trevor Darrell. "Fully convolutional networks for semantic segmentation". In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2014), pp. 3431–3440. URL: https://api.semanticscholar.org/CorpusID:1629541.

[36] Zongwei Zhou et al. "UNet++: A Nested U-Net Architecture for Medical Image Segmentation". In: *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support : 4th International Workshop, DLMIA 2018, and 8th International Workshop, ML-CDS 2018, held in conjunction with MICCAI 2018, Granada, Spain, S...* 11045 (2018), pp. 3–11. URL: https://api.semanticscholar.org/CorpusID:50786304.

[37] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39 (2015), pp. 2481–2495. URL: https://api.semanticscholar.org/CorpusID:60814714.

[38] Simon Jégou et al. "The One Hundred Layers Tiramisu: Fully Convolutional DenseNets for Semantic Segmentation". In: *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (2016), pp. 1175–1183. URL: https://api.semanticscholar.org/CorpusID:206597918.

[39] Adam Paszke et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *ArXiv* abs/1912.01703 (2019). URL: https://api.semanticscholar.org/CorpusID:202786778.

[40] William Falcon and The PyTorch Lightning team. *PyTorch Lightning*. Version 1.4. Mar. 2019. DOI: 10.5281/zenodo.3828935. URL: https://github.com/Lightning-AI/lightning.

[41] Diederik P. Kingma and Jimmy Ba. "Adam: A Method for Stochastic Optimization". In: *CoRR* abs/1412.6980 (2014). URL: https://api.semanticscholar.org/CorpusID:6628106.

[42] Tsung-Yi Lin et al. "Focal Loss for Dense Object Detection". In: *2017 IEEE International Conference on Computer Vision (ICCV)* (2017), pp. 2999–3007. URL: https://api.semanticscholar.org/CorpusID:47252984.

[43] Carole Helene Sudre et al. "Generalised Dice overlap as a deep learning loss function for highly unbalanced segmentations". In: *Deep learning in medical image analysis and multimodal learning for clinical decision support : Third International Workshop, DLMIA 2017, and 7th International Workshop, ML-CDS 2017, held in conjunction with MICCAI 2017 Quebec City, QC,...* 2017 (2017), pp. 240–248. URL: https://api.semanticscholar.org/CorpusID:21957663.

[44] Paul Best et al. "Deep audio embeddings for vocalisation clustering". In: *PLOS ONE* 18 (2023). URL: https://api.semanticscholar.org/CorpusID:257509372.

[45] Jack Goffinet et al. "Low-dimensional learned feature spaces quantify individual and group differences in vocal repertoires". In: *eLife* 10 (2021). URL: https://api.semanticscholar.org/CorpusID:232085524.

[46] Tim Sainburg, Marvin Thielk, and Timothy Q. Gentner. "Latent space visualization, characterization, and generation of diverse vocal communication signals". In: *bioRxiv* (2019). URL: https://api.semanticscholar.org/CorpusID:212811036.

[47] Ralph E. Peterson et al. "Unsupervised discovery of family specific vocal usage in the Mongolian gerbil". In: *bioRxiv* (2023). URL: https://api.semanticscholar.org/CorpusID:257535741.

[48] Leland McInnes and John Healy. "UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction". In: *ArXiv* abs/1802.03426 (2018). URL: https://api.semanticscholar.org/CorpusID:3641284.

[49] Ronald R. Coifman et al. "Geometric diffusions as a tool for harmonic analysis and structure definition of data: diffusion maps." In: *Proceedings of the National Academy of Sciences of the United States of America* 102 21 (2005), pp. 7426–31. URL: https://api.semanticscholar.org/CorpusID:15926341.

[50] Gustavo M Santana and Marcelo O Dietrich. *SqueakOut: Autoencoder-based segmentation of mouse ultrasonic vocalizations*. Apr. 2024. DOI: `10.17605/OSF.IO/F9SBT`. URL: `osf.io/f9sbt`.

[51] Charles R. Harris et al. “Array programming with NumPy”. In: *Nature* 585.7825 (Sept. 2020), pp. 357–362. DOI: `10.1038/s41586-020-2649-2`. URL: `https://doi.org/10.1038/s41586-020-2649-2`.

[52] F. Pedregosa et al. “Scikit-learn: Machine Learning in Python”. In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.