Ben-Gurion University of the Negev
Engineering Faculty
Biomedical Engineering

Segmentation and classification of ultrasonic calls as a tool for evaluation of social communication in a mouse model of autism

Supervisors:
Prof. Golan Hava
Dr. Lederman Dror

Students:
Lavy Guy - 313417909
Wolfman Vered - 308053404

Date: 15/08/2021

Abstract

1. Abstract

Autism Spectrum Disorder (ASD), is a neurological developmental disorder that causes social interaction impairment and irregular behavioral patterns. To date, ASD diagnosis is averagely done in ages 4-5, by behavioral examination. This diagnosis includes screening and based on behavioral metrics.. Early age diagnosis somewhat relies on the parents' attention to abnormalities in their child, the early age diagnosis is a key to enhance the quality of life of the children suffering from ASD, because it allows access to treatment and therapy. Therefore, a new, reliable early age diagnosis system is of great interests.

This research focuses on a mice model for autism, to investigate the biology of the disorder, in aim to develop a new diagnosis tool. Mice emit Ultrasonic Vocalizations (USV) in order to communicate with each other. These USVs function as a language for rodents and divide into 10 distinct syllables. Research shows that mice with ASD symptoms use these vocalizations in a slightly different way, regarding rate of vocalization" and different usage in the various types of syllables for any social interaction.

We developed an automatic tool for the classification of USVs, based on a CNN model trained via Transfer learning. Our model achieved 81.5% average accuracy in the classification of the ten syllable types. This tool will allow examination of changes in USV emission between healthy mice, and mice suffering from ASD-like symptoms, as we demonstrate on a large scale. Using these statistics can be a big step towards vocal based diagnosis of autism in mice. Based on the research finding and future analysis of baby cries it might be possible to implicate this tool to develop a dedicated diagnosis system for humans.

2. Introduction

2.1. Autism Spectrum Disorder

Autism Spectrum Disorder (ASD) is a term for a variety of complex neurological development disorders, that causes mild to severe impairment in social interactions and atypical repetitive patterns of behavior [1][2].

In the United States, there is a prevalence of 1 case of ASD in every 68 children, which translates to about 1.5% [3]. Early age diagnosis plays a crucial role for children who suffer

from ASD. It allows early etiologic investigation, counseling regarding recurrence risk and providing appropriate intervention. The intervention may include behavior and communication therapies, appropriate educational programs, and specific medications that can help control symptoms. These can improve long term outcomes related to cognition, daily living skills, language, social and adaptive behaviors, thus drastically improving the children and their families quality of life [2][4]. The core clinical signs of ADS are severe social skills deficits, stereotyped and repetitive patterns of behavior, activity and interests, lack of eye contact and language delays. The main challenge in the ASD diagnosis and identification is the wide range of the severity in the core clinical signs among children who suffer from ASD [4]. Diagnosis of ASD can nowadays begin before the age of 2, and the accuracy of the behavioral based diagnosis increases with age [2], and although accurate methods of diagnosis exist for early ages, the average age of ASD diagnosis is between 4 and 5 years. Currently, most of the diagnosis methods include observations and screening of the suspected childrens' behavior and development, by their caretakers at home and professional psychiatrists and doctors at the clinics. All of these methods are based on behavioral and developmental criteria only.

The exact cause of ASDs is still unknown, although it is believed to have both genetic factors and environmental factors. ASDs involve multiple genes and demonstrate great phenotypic variation, and for this reason its exact cause has eluded science to this day. Among the genes associated with the risk of ASD is "methylenetetrahydrofolate - reductase" (MTHFR). MTHFR deficiency increases the risk to create developmental delay, and autistic symptoms in both humans and mice, therefore a MTHFR deficient (MTHFR heterozygous knockout) mouse model is valid to test autistic behaviors that can be implicated to humans [5]. Golan et al. [5] suggests this mouse model of autism to deepen the understanding of ASD in humans. They tested mouse different behaviors and showed results that support the impact of deficiency in the MTHFR gene on the ASD features. This model can help to clarify the molecular mechanisms and distinguish between autonomous and in-utero factors for ASD [5].

Even though ASDs were found to be heritable, environmental factors may also modulate phenotypic expression. An example for a risk factor for ASD with both genetic and environmental properties, is the older age of the parents, yet it is unknown whether this factor

affects the infants due to a high likelihood for genetic mutations, or because of the environmental changes it may create [4].

Studies show that communication in a variety of animal species share many properties with human communication, for instance mice emit UltraSonic Vocalization (USVs) in different social contexts. In one study, a mouse model for tauopathy, a neurodegenerative disease associated with progressive language disorders, was suggested. The study reported an age-related impairment of mice USV that correlates with areas of the brain controlling vocalization [6]. Another research showed that specific inbreeds of mice (BTBR) that display ASD symptoms, communicate differently compared to ordinary mice in specific situations [7]. Based on these and other reports [5][6][7], studying mice USVs is considered to be valid for testing autistic-like behaviors in mice, deepening the understanding of the biological mechanisms and causes of the syndrome, in order to develop treatment or diagnosis systems.

2.2. UltraSonic Vocalization

In this project, we focused on analyzing one of the main symptoms of ASD - impaired communication. For this purpose, we made use of USVs emitted by mice pups and adolescents, female and male, healthy and suffering from ASD symptoms. Several studies [7][8] have shown that mice use USVs - vocal signals that surpass the human hearing, with frequencies ranging between 40-120kHz, to communicate with one another not unlike many other animal species. These calls play an important role in social communication between the pups and their mother or nest and are modulated during development. These calls can differ between a healthy mouse and a mouse showing ASD symptoms of impaired communication. Thus, by analyzing these calls it is possible to distinguish the autistic mice. One way to categorize the mice USVs is splitting them into 10 distinctive syllables (Fig.1), that show different frequencies, shape, and duration. Studies have found that specific inbreeds of mice (BTBR) that display ASD symptoms, use a different set of syllables at different rates compared to ordinary mice in specific situations such as being separated from their nest. Furthermore, mouse pups' vocalizations share some properties similar to the cry of human babies [7], thus encouraging us to investigate the disorder in a mouse model.

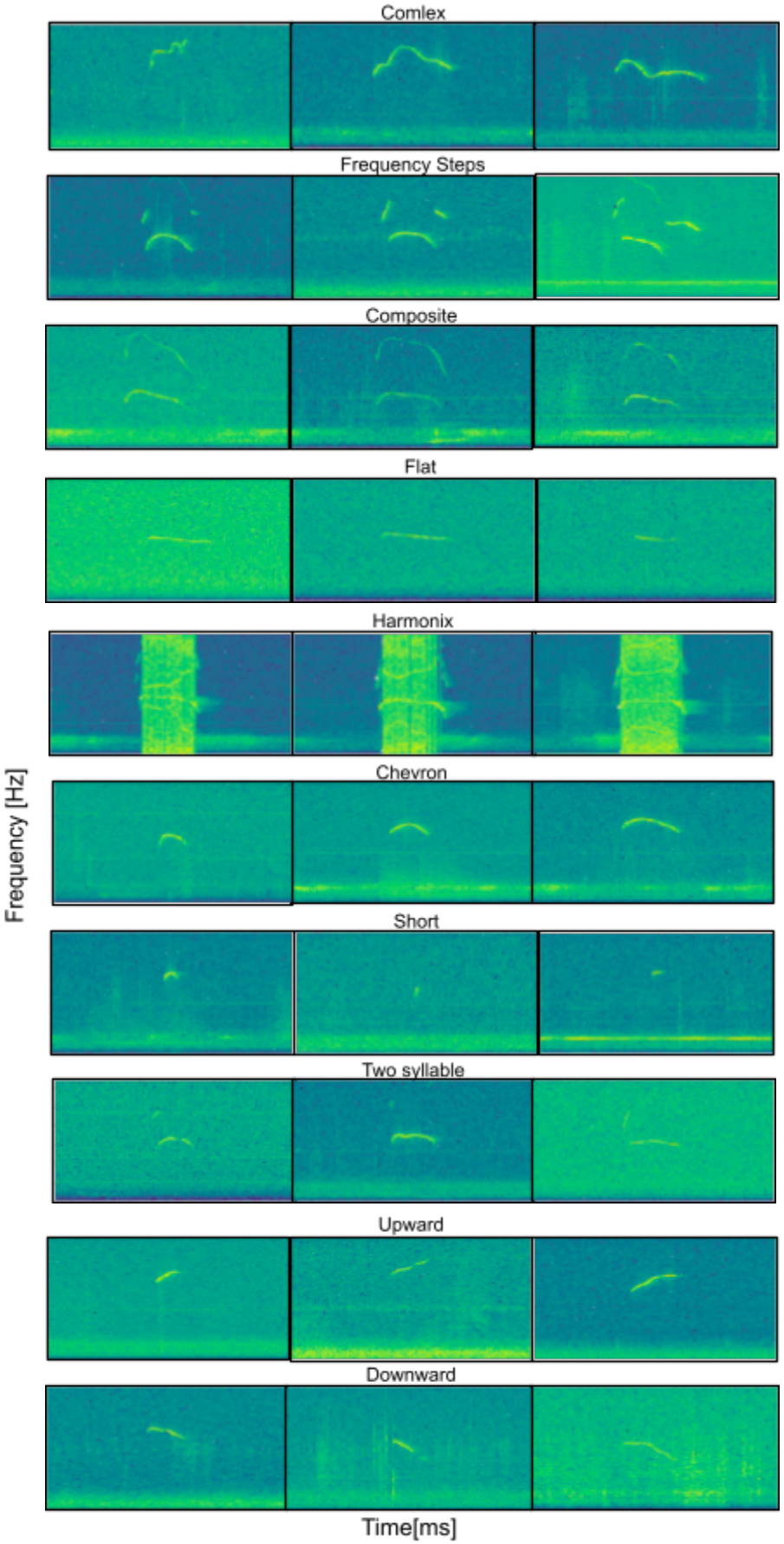


Figure 1. The ten different USV calls produced by the mice

The list of the USV syllables can be described as follows:

1. Complex - One main continuous syllable, with at least one inflection point.
2. Frequency steps - 3 or 4 main components, one longer than the others at the lowest frequency, two short components - one at the beginning of the syllable and the latter in the end. If there is a fourth component, it shall have the highest frequency.
3. Composite - Two main components in the syllable, usually the one with the higher frequency has a lower amplitude.
4. Flat - One main syllable with a totally constant frequency.
5. Harmonix - Ranges from 3 to 7 components with different frequencies, usually one component spreads over the entire syllable whilst the others are shorter in duration.
6. Chevron - One main continuous syllable, with a concave shape, can be quite similar to complex.
7. Shorts - One punctuated syllable, similar to chevron or complex only with a very short duration, about 3 milliseconds.
8. Two-syllable - Two main components in the syllable, usually one shorter than the other, the latter is located at the beginning of the longer syllable and at a higher frequency.
9. Upward - One main syllable, frequency increases over time.
10. Downward - One main syllable, frequency decreases over time.

Our dataset contains roughly 90,000 audio recordings of mice USVs (70,000 adults and 27,000 young), distributed between males, females, healthy, and mice with MTHFR gene deficiency.

2.3. Previous USVs classification works

In recent years, deep learning has been proven to be a useful tool for recognition and classification of different types of data. Specifically, auditory signal classification tasks have benefited from the high accuracy and efficiency of deep neural networks, and have been widely used in research [9][10].

Deep learning is a subset of Machine learning (ML), in which a designated software learns from an example dataset in order to make predictions on unseen data. Unlike classic ML,

deep learning does not require manual feature selection, but automatically learns to extract features from the data from its own.
In the past years many tools have been proposed for automatic rodent USVs analysis, both segmentation and classification [11-18] relying mostly on supervised or unsupervised deep learning models. Each brought an innovation in their respective research.

For instance, in 2015 an unsupervised deep learning approach and a hierarchical clustering strategy were developed in order to classify the syllables [14]. A major limitation of this approach is that the quality of clustering is highly dependent on the dataset, and the algorithm can mistakenly classify different syllables under the same category in pups.

In [16], a heuristic approach for classification of USV into 4 distinct categories, as a generalization of the 10 syllables categorization. The approach is based on k-means clustering with a fixed, pre-defined number of clusters. The authors reported good results around 85% success rate, however failed to classify USVs with gaussian white noise in some cases [12].

In 2019, Coffey et al. [12] proposed “DeepSqueak”, a model for segmentation and classification of USVs. This model uses a Regional Convolutional Neural Network architecture, termed “Faster-RCNN”, with four default detection networks for different USV frequencies made by different rodent species. This algorithm showed good results, approximately 95% success rate. However, the classification model was designed for 5 syllables only, which is not adequate for our case [12].

The most promising advancement we encountered so far was done by Fonseca et al. [18] who introduced “VocalMat”. They used a different set of classes composed of 11 different types. “VocalMat” is a software which provides accurate segmentation and classification based on a CNN model trained on a large custom dataset. The accuracy of the “VocalMat” classifier was 86% but struggled classifying syllables with more than one component.

In last year’s project, our colleagues developed and tested both supervised and unsupervised deep learning models in order to automatically classify mice USV signals that have been segmented the year before [11][13]. Unsupervised learning is a machine learning process in which the learning is done on uncategorized data, while supervised learning is done on categorized or tagged data. The unsupervised approach provided in-sufficient results, and

yielded an average classification rate of 64% whereas the supervised approach yielded an average classification rate of 96.6% when the classification was performed on 2 syllables, and 87.3% for 4 syllables [13]. Even though this model showed good results, a model that can distinguish between all syllables is necessary in order to identify a mouse with ASD.

Table 1: USV classification tools comparison

Name	Year	Syllable categories number	Main method	Average accuracy results
VoICE	2015	11	Unsupervised hierarchical tree clustering	N.A.
MUPET	2017	4	k-mean clustering	85%
DeepSqueak	2019	5	Faster-RCNN	95%
BGU proj.	2020	4	CNN	87%
VocalMat	2021	11	CNN	86%

Table 1 summarizes the methods mentioned above and compares their accuracy. Note that "VoICE" did not provide classification accuracy results.

2.4. Project goals

The main goal of this project was to develop an automatic tool to identify ASD like behavior in mice based on USV recordings and to apply it on USVs emitted by healthy and syndromic mice.

In order to achieve this goal, we first developed a Convolutional Neural Network (CNN) model for USV classification. Then used this model's output to analyze the statistical differences between mice with and without ASD symptoms, based on rate and timing of different syllable usage.

3. Material and Methods

3.1. The Dataset

The whole dataset contains about 90,000 audio recordings of mice USVs (70,000 adults and 27,000 young), distributed between males, females, healthy, and mice with MTHFR gene deficiency. We used a previously developed segmentation algorithm to create a dataset of USV syllables for the purpose of this work. This algorithm received a raw audio recording of a mouse, and output segmented USV syllables, for easier management of the dataset [11]. This algorithm, and manual labeling by crews in previous years yielded a dataset of USV syllables containing 5865 spectrograms (roughly 2000 adults and 4000 young).
For our purpose we used only the 5865 labeled syllables as our dataset.

3.2. Pre-processing

During the research, we observed unsettling results i.e, less than 66% accuracy. Concerns were raised regarding the dataset and two bugs were found:
1. Time axis normalization.
2. Inconsistent manual labeling caused by said normalization.
As mentioned in the introduction the syllables differ in duration. Each spectrogram in the labeled dataset was built from the starting point to the end point given by the segmentation algorithm. This means that both long and short syllables were compressed or stretched respectively to a fixed length as can be seen in Fig 2. This time normalization prevented credible manual and automatic classification because an important feature of the spectrogram was being ignored. It is crucial that the spectrograms remain in the exact same size in order to train a classification model but without damaging the integrity of the syllables' time axis. Our proposed solution for this bug was zero padding of the recordings before and after each syllable to a fixed size of the auditory signal. The segmentation algorithm provided a starting point and ending pointing point for each syllable. We trimmed each syllable according to these points and filled it with the same amount of zeros, from left and right, in order to keep the syllables in the center of the spectrogram. The specific amount of zeros for each syllable was calculated by the following formula.

$$(1)\ N_0 = \frac{(L_{max} - L_i) \cdot f}{2},$$

where:

- L_{max}- is the duration of the longest syllable in seconds.
- L_i- is the duration of the current syllable in seconds.
- f- is the sample frequency in Hz.

Note that the division by 2 is because the zero padding is done in two places (before and after).

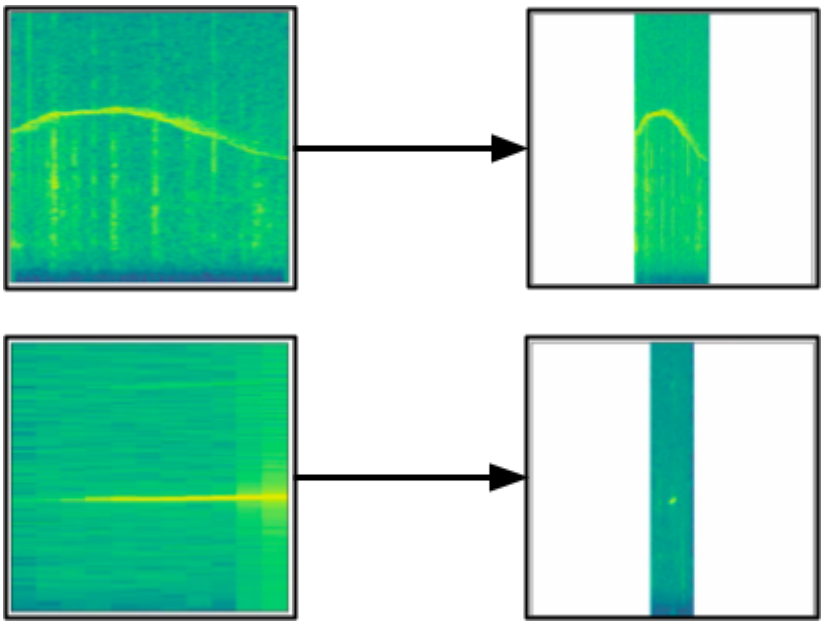


Figure 2. Illustration of time axis normalization

In Fig. 2 we can see an illustration of the time axis normalization. In the left side of the figure the spectrograms from the normalized dataset are presented. As mentioned earlier the syllables are being stretched through the entire time axis, whereas in the right side of the figure the zero padded spectrograms are presented. For example, the syllable on the bottom right is of "Short" type and the time normalization caused it to appear as "Flat" type.

According to the literature, the maximum frequency of mice USV is 120kHz, for this reason the recordings were taken at a sampling rate of 250kHz via an ultrasonic microphone. Before feeding the data into our model we first examined the spectrum of the different syllables in order to design an optimal filter. The spectrum presented in the following figure was procured using Fast Fourier Transform (FFT) algorithm.

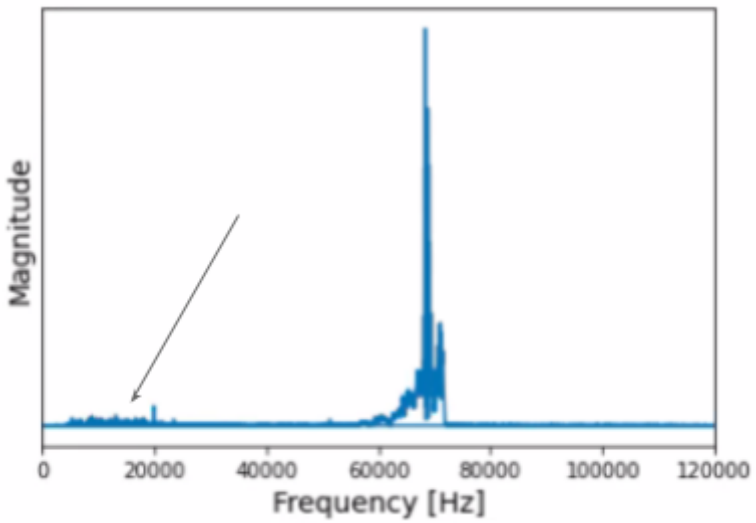


Figure 3. The spectrum of an unfiltered arbitrary syllable from the dataset

In Fig. 3, a spectrum of an unfiltered arbitrary syllable is presented. The main frequency is around 70kHz, and lower frequencies up to 30kHz are noisy.

We designed an analog Butterworth High Pass Filter (HPF) and converted it to a discrete filter using the bilinear transform. We chose a $f_c = 30kHz$ cutoff frequency, because the minimum frequency of USV in our dataset was around 35kHz, and calculated the minimum filter order (N=6), then implemented the filter in code.

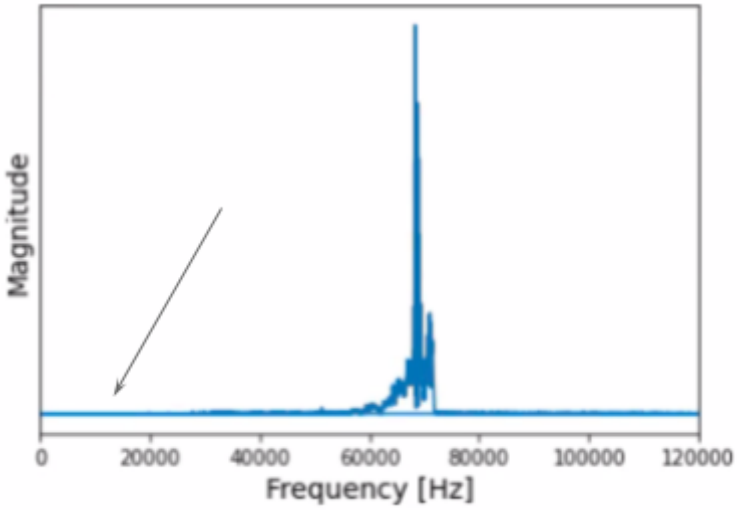


Figure 4. The spectrum of the same syllable from the dataset after HPF

In Fig. 4, the spectrum of the same syllable from Fig. 3 after HPF is presented. The main frequency, 70kHz, is untouched and lower frequencies up to 30kHz were filtered from the signal.

We then performed a Short Time Fourier Transform (STFT), with a Hamming window in size 2048 and overlap of 1920, and stored the spectrograms in files.

As previously mentioned, the time axis normalization also caused inconsistent manual labeling due the distortion of the syllables that made them difficult to distinguish as seen in Fig 5.

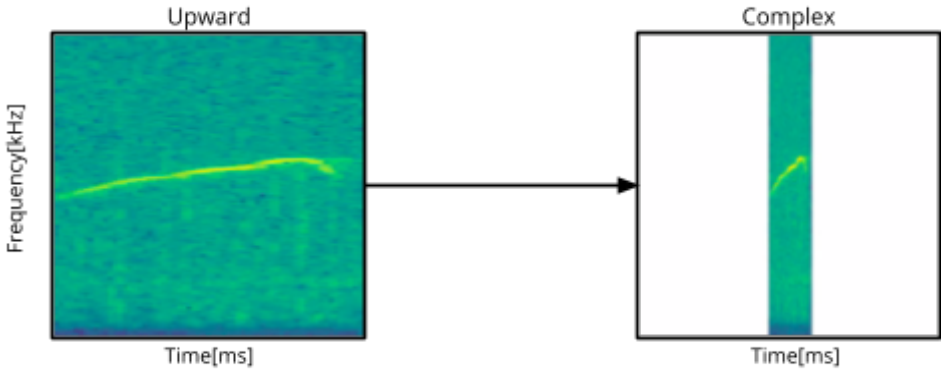


Figure 5. Illustration of incorrect labeling

In Fig. 5, we can see the “Complex” syllable (at the bottom) after our correction, mistakenly labeled as “Upward” (at the top) due to the distortion of the inflection points caused by the stretch of the syllable, which made them less visible.

Because of these phenomena we had to re-classify the labeled dataset. For this purpose, we created a basic interface in python and went over the entire labeled dataset in order to ensure correct and consistent labeling.

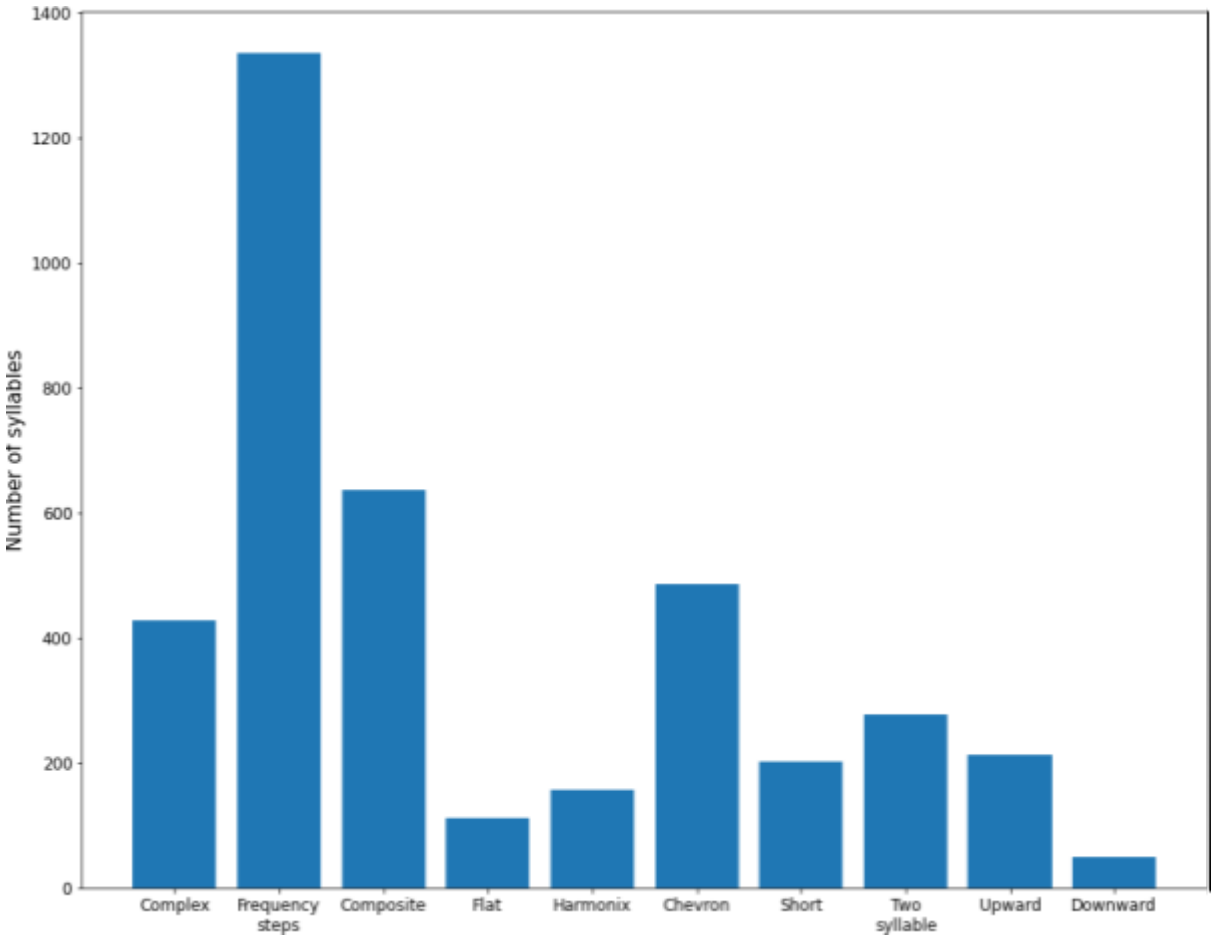


Figure 6. Distribution of the re-labeled dataset

In Fig. 6, a bar plot of the distribution of the re-labeled dataset is shown. As seen at the figure some syllables such as “Frequency steps”, “Composite” and “Chevron” contain a large number of examples whilst other syllable types such as “Flat” or “Downward” contain a very small amount.

3.3. CNN model & Transfer learning

We developed a model based on a convolutional neural network (CNN) in order to automatically identify mouse USVs’ syllables. In the deep learning field, CNN is a neural network architecture that integrates the mathematical operator convolution between matrices, instead of matrix multiplication as used in standard neural networks [19][20]. It is mostly used for vision applications, but has been successfully used in auditory signal classification tasks, such as in our case, by using a visual representation of the auditory signals like the STFT.

A CNN consists of three main parts. The first component of the network is the input layer in which the inputs, vectors or matrices, are being received. The second part is the "hidden layers". In these layers the inputs are broken down to various features like edges, shapes, shades, colour variations etc. The third part is the output layer that depends on the feature extraction from the hidden layers and an activation function [20].

The CNN parameters are estimated during a training procedure. The dataset is being divided into three subsets - training, validation and test sets. During training each item in the training sets is passed through the model, and the output label is fed to a predefined loss function. The loss function is dependent on the true label, and the aim of the training process is to minimize this function's output through backpropagation and gradient descent. In every hidden layer the model stores values called "model weights", and in training the model adjusts the weights in a way that lowers the gradient of the loss function towards a global minimum. The algorithm propagates backwards from the output layer to the input layer and updates the weights in the manner we described. This method ensures that the prediction of the model will be as close as possible to the actual label. The validation set provides an unbiased evaluation of the model fitting during training in order to tune the model hyperparameters and track the learning process [19][20]. The test set on the other hand, provides an assessment of the model performance after the training is complete. Note that only the training set affects the weights of the model.

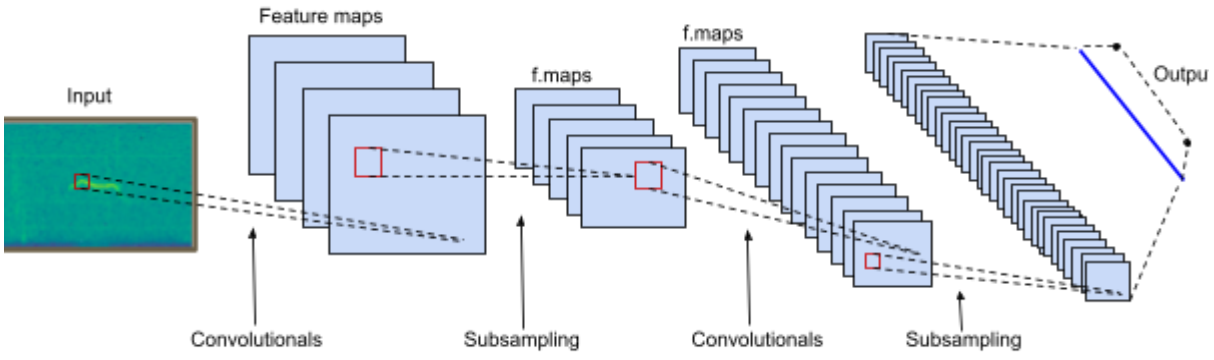


Figure 7. Example of CNN scheme

Fig. 7 shows a scheme of a generic CNN architecture. As described above the visual input is mapped into features by the convolutional layers, and these contribute to the classification output.

CNN models require a vast amount of data to achieve high accuracy results. Our dataset is relatively small, so we decided to implement specific approaches meant to cope with the problem. The first approach is transfer learning. Transfer learning is a method in which a CNN model is being trained on a massive dataset for a different classification task and achieves high accuracy. The output layer of the trained model is then replaced, and the model is re-trained on the dataset for the classification task at hand. The initial training process teaches the model to extract different features and this knowledge is then transferred to the new classification task. This method has been shown to provide great results in the field for classification tasks with small dataset [21][22].

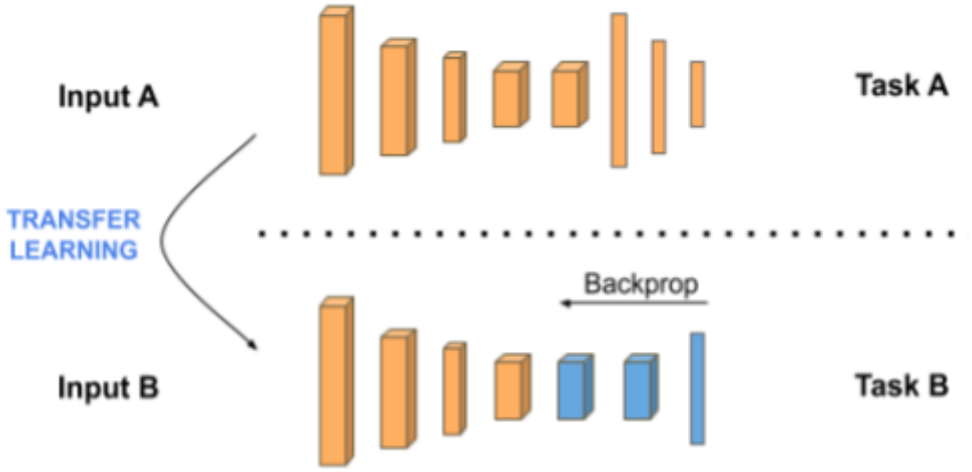


Figure 8. Visual description of transfer learning process

The transfer learning method and its process are described in Fig 8. As seen in the figure, the top layers of a pre-trained model (for task A) are being replaced, and the model re-trains for task B through backpropagation.

We tested several architectures including variations of MobileNet, ResNet and EfficientNet. The best results were obtained using Big Transfer (BiT-M) R50X3: a variation of ResNet V2-50 containing three 50 layers - ResNet V2 architecture connected in parallel. This network is specifically designed for transfer learning applications and each sub-architecture extracts a different set of features. The model was pre-trained on a dataset called ImageNet-21K, containing 14 million labeled images divided into 21,000 categories. ResNet V2-50 is a 50-layer Residual Network with 26M parameters. The residual network is a CNN model that was introduced by Microsoft in 2015 [22]. In a residual network beside learning features, the model learns from residuals that sums the input and output of each block of layers. Each block contains 3 convolutional layers with a different kernel size (1x1, 3x3).

Before the first block there is another convolutional layer with a kernel size of 7x7 and a maxpooling layer. The final two layers are a convolutional layer with a kernel size of 7x7, and a classification dense layer (the output layer). The residuals prevent a state “dead neurons” - a phenomenon that occurs in large networks in which some neurons maintain a value of zero and remain inactivated.

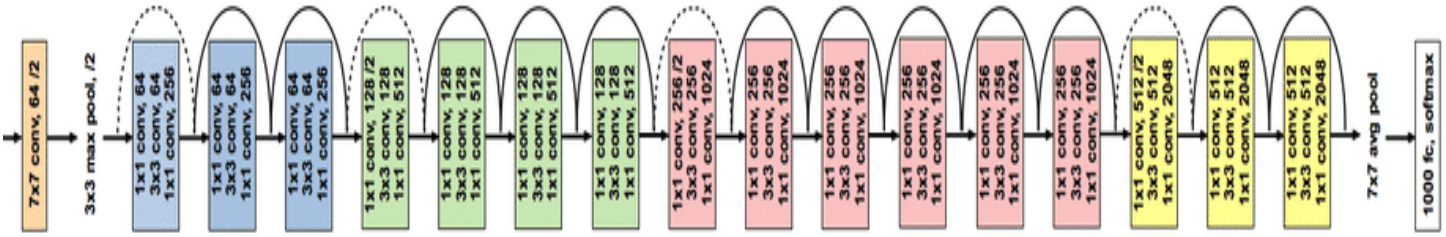


Figure 9. ResNet 50, architecture [22]

In Fig. 9, we can see the architecture of ResNet 50 and its different blocks and residuals as described before.

A critical concern in any machine learning task, and in particular when the available dataset is small, is overfitting. In overfitting the model gets too familiar with the training data, and a big gap between the training accuracy and the test accuracy is formed. This occurs mostly in tasks with limited data as ours. During training we encountered this state and took two different approaches to avoid this gap. The first approach is data augmentation, which is a strategy for artificially increasing the quantity and complexity of the data by adding modifications to our images. These modifications include flipping, rotation, and brightness adjustments. It increased our training data size and our model had considered each of these small changes as a distinct image, and this helped the model to learn better and perform well on unseen data. Using augmentations we improved performance in the model’s accuracy, and a reduction in overfitting. However, some difference between the test accuracy and the training accuracy still remained, therefore we used dropout regularization. This method randomly disables a pre-defined percentage of neurons in the model. Dropout prevents the model from relying on any specific feature or connection and by doing so, the model is forced to use the entire set of features. Using this regularization and the augmentations minimized the overfitting to a satisfying point.

4. Results

4.1. Model results

In order to evaluate the model performance, we present in this chapter the convergence plots of the model accuracy and loss function during the training process, and the confusion matrix of the accuracy on the test set. We also compare the results of 7 syllables from the previous time normalized dataset to the dataset that went through the revised pre-processing procedure and re-labeling. For simplicity they will be referred to as dataset A and dataset B respectively.

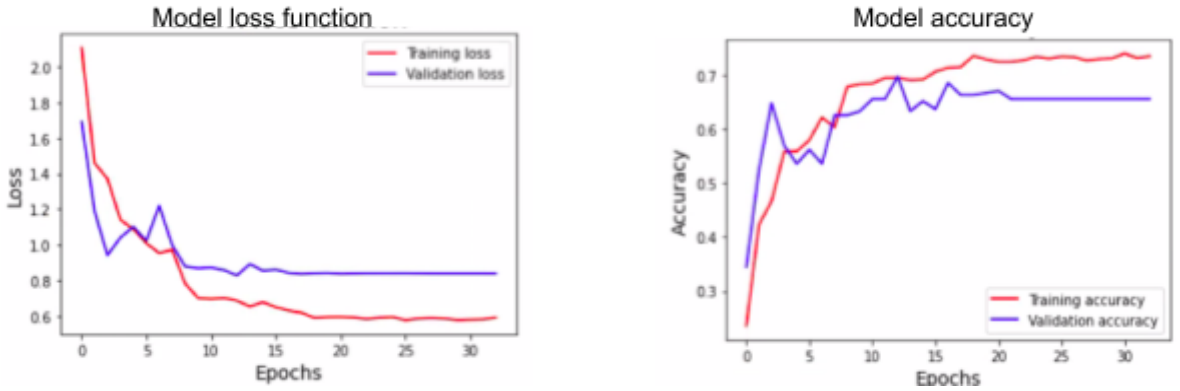


Figure 10. Convergence plots of the model on 7 syllables from dataset A.

The left side of Fig. 10 contains the model's loss function on 7 syllables from dataset A and the right side contains the model's accuracy plot on those syllables. The red lines represent the results of the training set and the blue lines represent the same on the validation set. The descent and plateau of the loss plot represent the convergence to a minimum. The model on these data achieved 65% accuracy on the validation set.

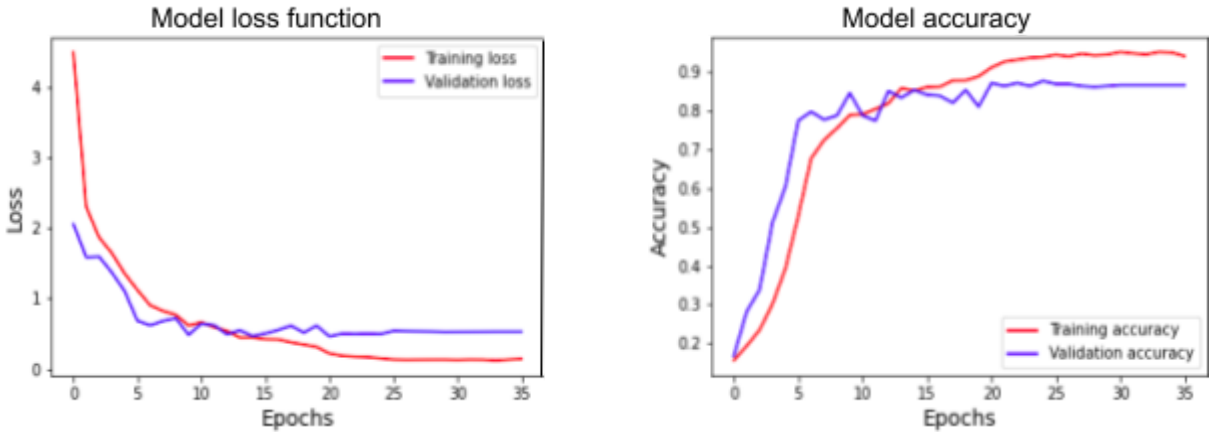
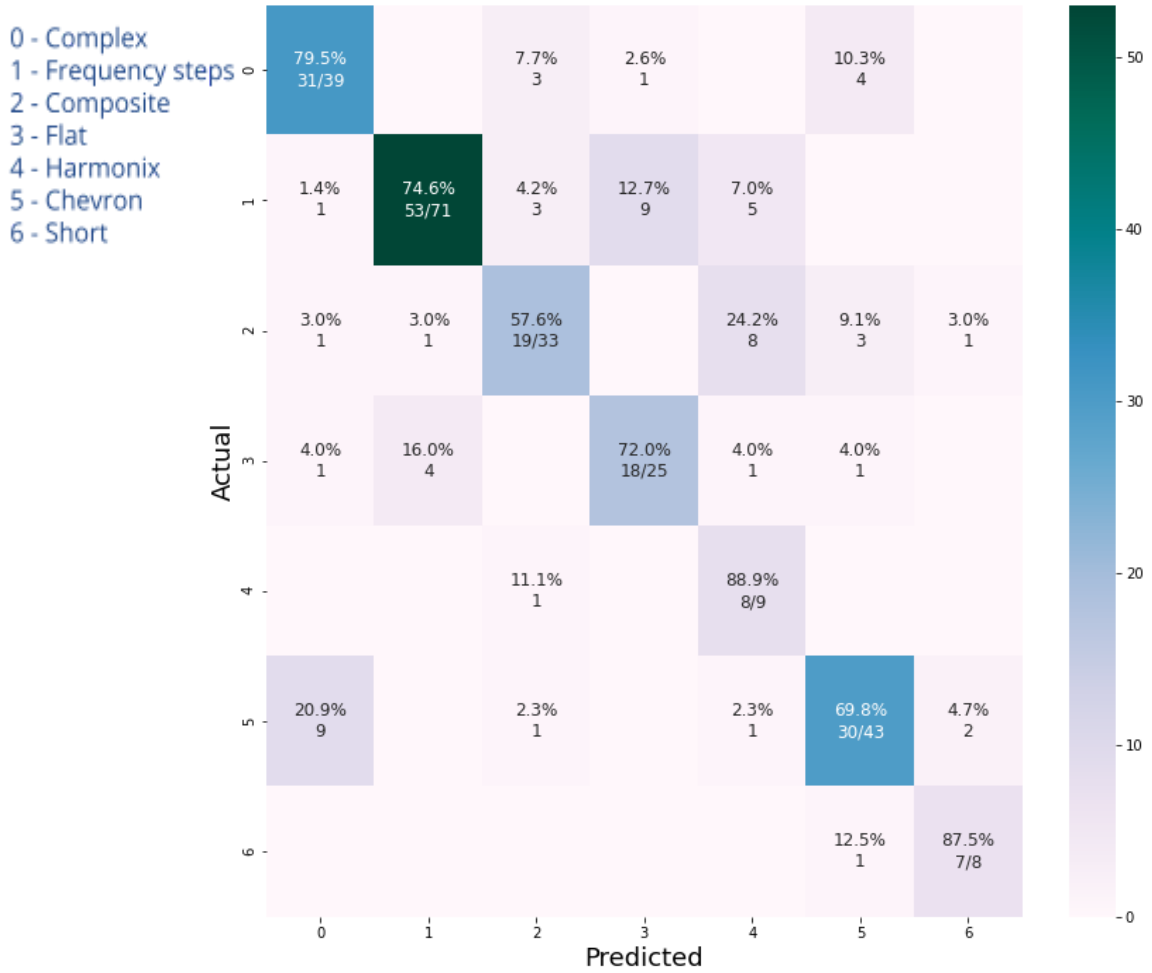


Figure 11. Convergence plots of the model on 7 syllables from dataset B.

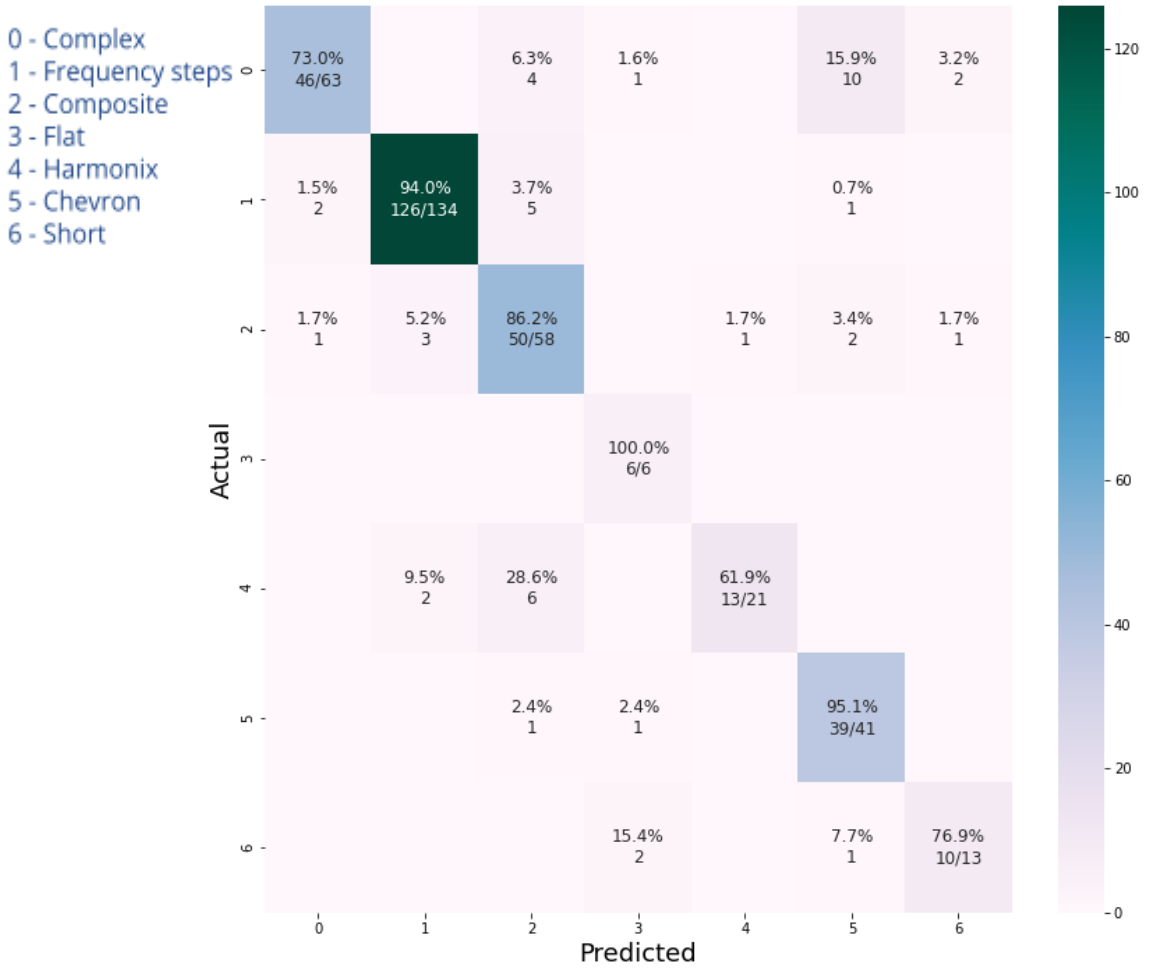
The left side of Fig. 11 contains the model's loss function on 7 syllables from dataset B and the right side contains the model's accuracy plot on those syllables. The model on this data achieved 86% accuracy on the validation set.

Table 2: Confusion matrix of the model on 7 syllables from dataset A.



In Table 2, we see the confusion matrix of the 7 syllables from dataset A. This matrix contains the accuracy results for each syllable on the test set. The colorbar represents the amount of true positive classifications in each cell in the main diagonal, and the amount of false positive in each cell outside the main diagonal. An empty cell means no false positive for this specific syllable type. The model's errors are affected by the time axis normalization, for instance the model mistakes "Short" syllables as "Chevron", and vice versa, because they mostly differ in time (Fig 1).

Table 3: Confusion matrix of the model on 7 syllables from dataset B.



In Table 3, we see the confusion matrix of the 7 syllables from dataset B. The colorbar represents the amount of true positive classifications in each cell in the main diagonal, and the amount of false positives in each cell outside the main diagonal. This matrix contains the accuracy results for each syllable on the test set. Note that the model's errors are inline with expectation, for example the model classified 15.9% of the "Comlex" syllables as "Chevron", and 28.6% of "Harmonix" syllables as "Composite". This makes sense because of the similarities between these types (Fig 1).

As shown in Figs. 10, 11, and in Tables 2, 3, dataset B yielded an improvement in accuracy of over 20%. Furthermore, from the confusion matrices we learn that the mistakes of the model on dataset A were seemingly random as opposed to dataset B on which the model's mistakes were as expected and made sense with the properties of the different syllables.

After we saw the model got satisfying results on the 7 syllables, we examined it on all 10 syllables. The model's results for the 10 syllables-

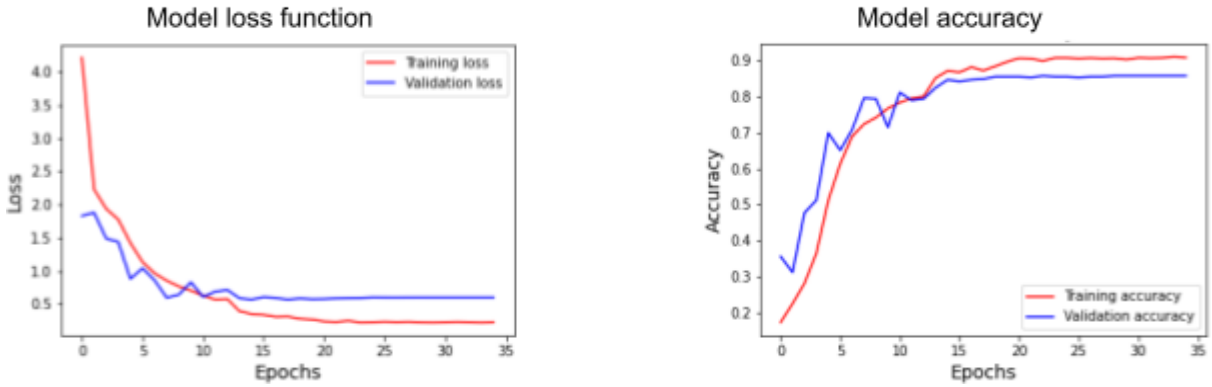


Figure 12. Convergence plots of the model on 10 syllables from dataset B.

The left side of Fig. 12 contains the model's loss function on 10 syllables from dataset B and the right side contains the model's accuracy plot on those syllables. The model on this data achieved 85% accuracy on the validation set.

Table 4: Confusion matrix of the model on 10 syllables from dataset B.

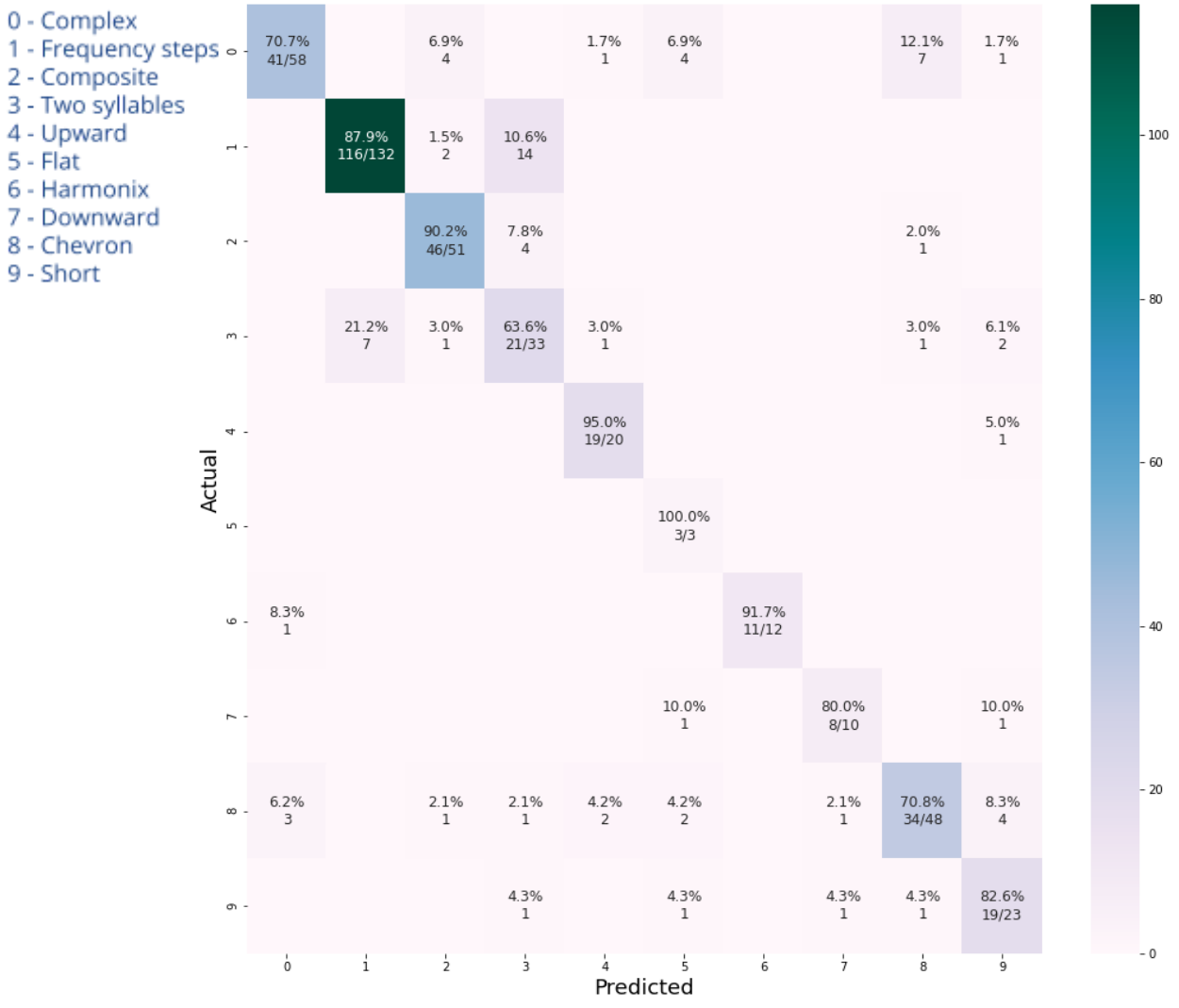


Table 4, presents a summary of results for the 10 syllables from dataset B set. The colorbar represents the amount of true positive classifications in each cell in the main diagonal, and the amount of false positive in each cell outside the main diagonal. An empty cell means no false positive for this specific syllable type. This matrix contains the accuracy results for each syllable on the test set. From the matrix we can see that “Complex”, “Two syllable” and ”Chevron” syllables were classified below 80% accuracy. This can be explained by the fact that “Two syllables” is quite similar to “Frequency steps” but the latter contains the biggest amount of data. Therefore the model tends to classify “Two syllables” as “Frequency steps”, with 21.2% error. The relatively low accuracy of “Complex” and ”Chevron” can be explained by the similarity of these syllables but in this case the classification error occurs on both sides due to almost equal amount of data for each of them. The model classified 12.1% “Complex” as ”Chevron” and 6.2% the other way around. The rest of the syllable types were classified above 80% accuracy. The variations in the accuracy percentages depend on the amount of data for each class (Fig 6), and the uniqueness of the syllable type.

4.2. Statistical analysis of USV calls

In this chapter we present the usage of the segmentation [11] and our classification model on 248 recordings containing ~1500 USV syllables from mice pups differing in gender, genotype and age. The classification model was not trained with a “Noise” class, therefore in order to avoid errors such as classifying noise as a syllable, a threshold was set. The threshold requires the probability of a syllable classification to be over 50% certainty, otherwise the syllable is classified as “Undefined”.

Table 5: Number of USV calls divided by the different groups

Age	**Male**		**Female**	
	WT	**HT**	**WT**	**HT**
4 days	311	0	379	73
8 days	2	122	54	15
10 days	112	41	87	49
12 days	36	29	42	43

The following figures present visualization of the data: distribution of the different syllable usage in females and males with different genotypes and at different ages. The genotypes of

mice in the recording are “WT” (Wild Type)- ordinary healthy mice, and “HT” (Heterozygous Type)- MTHFR knockout mice.

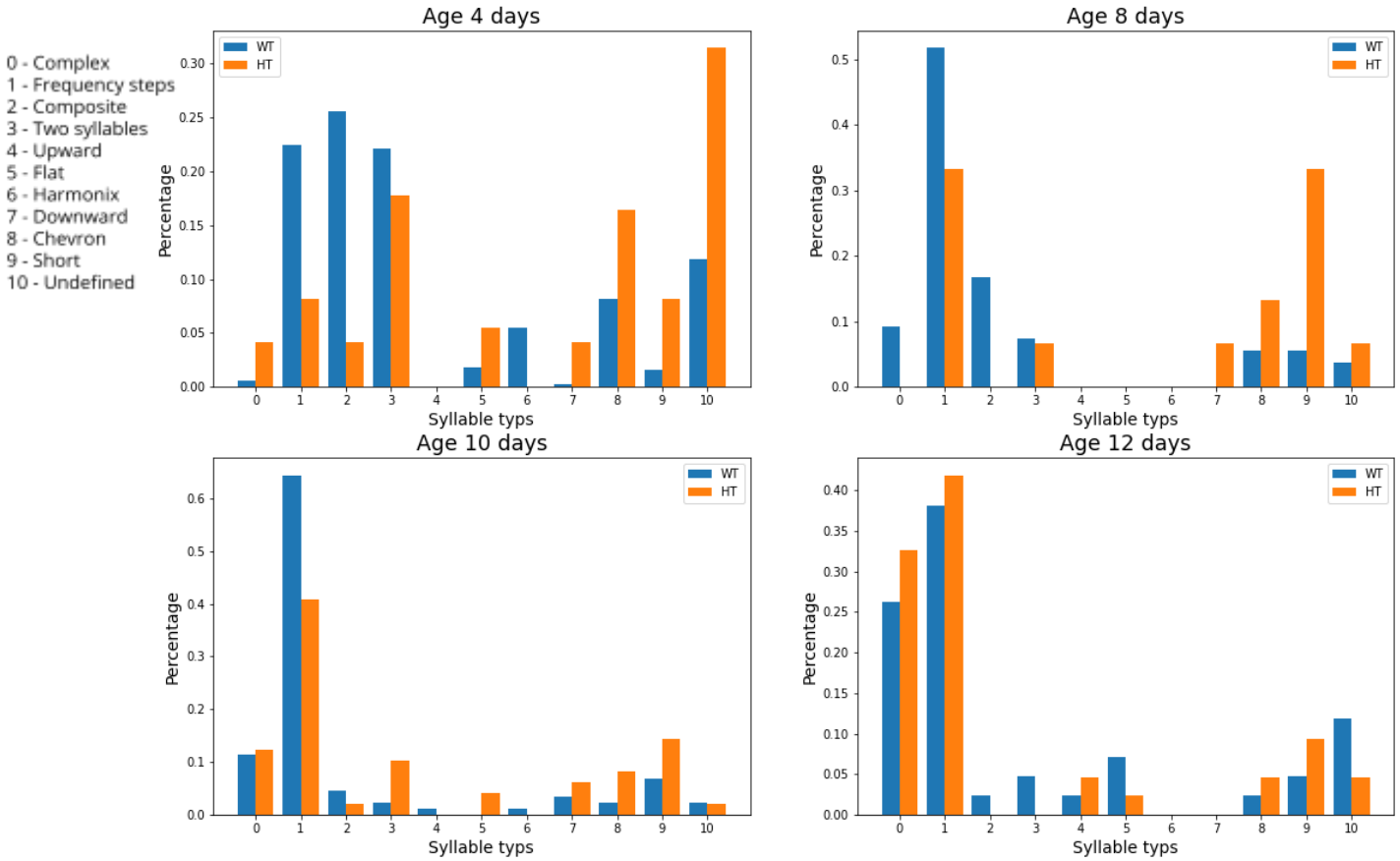


Figure 13. Syllable usage of female pups- comparison by age and genotypes

Fig 13. shows the distribution of syllable usage of female mice pups, with different genotypes. The y axis represents the percentage of each syllable out of all syllables recorded for each group. Ages are shown from left to right, top to bottom- 4 days, 8 days, 10 days and 12 days. The blue bars represent the percentage of syllable usage in “WT” genotype, the orange represent the percentage of syllable usage in “HT” genotype. At the age of 4 days, we can see that female pups of “WT” genotype mainly use “Frequency steps”, “Composite” and “Two syllables”, all of which are multicomponent syllables. “HT” genotype on the other hand mainly uses “Two syllables” and “Chevron” at this age. We also see that a lot of the syllables by this genotype at this age are undefined and it might suggest a developmental delay in communication skills. At the ages of 8 days and 10 days we can see that “Frequency steps” dominates the female “WT” genotype vocabulary. “HT” genotype mainly uses “Frequency steps” and “Short” at the age of 8, and at the age of 10 the usage of “Short” decreases. At the age of 12 days, we can see that both genotypes mostly use “Complex” and “Frequency steps” and no major differences are seen by eye.

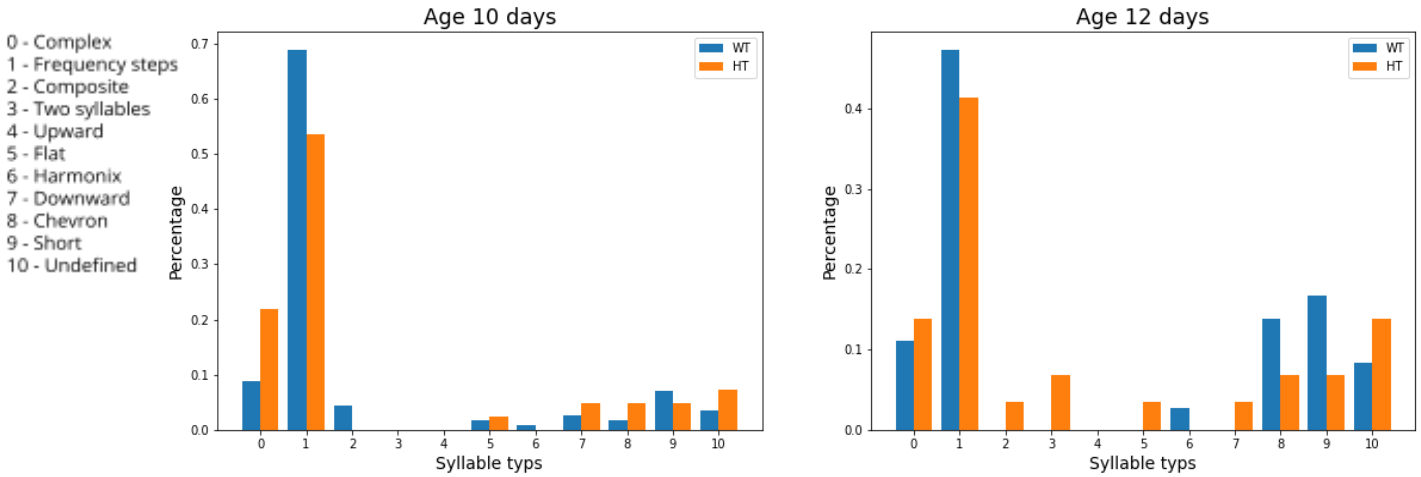


Figure 14. Syllable usage of male pups- comparison by age and genotypes

Fig 14. shows the distribution of syllable usage of male mice pups, with different genotypes. The y axis represents the percentage of each syllable out of all syllables recorded for each group. Ages are shown from left to right- 10 days and 12 days. Due to lack of data days 4 and 8 are not presented for male pups. The blue bars represent the percentage of syllable usage in "WT" genotype, the orange represent the percentage of syllable usage in "HT" genotype. At the age of 10 days the male pups of "WT" genotype mostly use "Frequency steps", and those of "HT" genotype mainly use "Frequency steps" but also use "Complex". At the age of 12 days, we can see an increase of "Chevron" and "Short" usage, and decrease of "Frequency steps" in "WT" males. On the other hand, "HT" males start to use "Composite" and "Two syllables".

We used the chi square test for each gender at every age, and examined if there is a significant statistical difference between the two genotype groups. The tests were performed on the percentage of each syllable out of the entire recorded syllables of each group. The null hypothesis is that there is no difference between mice pusps of genotype "WT" and "HT", and alpha is equal to 0.05.

Table 6: p-values - female comparison different genotype each age

	Age 4 days	**Age 8 days**	**Age 10 days**	**Age 12 days**
p-value	$0.4 \cdot 10^{-7}$	$0.1 \cdot 10^{-9}$	$0.6 \cdot 10^{-2}$	0.045

From table 6, which represents the chi square p-value results of females with different genotypes at different ages, we can see that the results were significant ($p < 0.05$). This means

we can reject the null hypothesis and conclude that there is a difference between the female mice pups with the two genotypes at every age tested.

Table 7: P values - male comparison different genotype each age

	Age 10 days	**Age 12 days**
P value	$0.8 \cdot 10^{-3}$	0.035

Table 7 shows the chi square p-value results of males with different genotypes, “WT” and “HT”, at different ages. We can see that the results were significant ($p < 0.05$), meaning we can reject the null hypothesis and conclude that there is a difference between the male mice pups with the two genotypes at every age tested.

Note: The significant results presented here might point to differences between individual mice, and in order to make a statistical statement these tests must be done on a far larger sample size. This is just an example for the possible use of the model proposed in this report.

5. Discussion

We report the development of our automatic classification model, to identify mice USVs. After preprocessing including HPF and zero padding we refined our dataset and used it to train and evaluate the performance of our proposed CNN model. The classifier achieved 81.5% accuracy - an improvement of ~20% over previous attempts. One paper [18] proposed a similar approach for USV classification based on a CNN model, and reported 86% percent accuracy, however it struggled to distinguish multicomponent syllables unlike our model. Furthermore, with a larger training dataset we believe our model could achieve similar or even better results.

These results are sufficient to combine the segmentation algorithm with the classification model. To date, USV analysis requires tedious and time-consuming manual labor, and our model provides a fast and efficient automatic tool for USV analysis. With this tool, it is possible to gather a massive amount of data from the recorded dataset, about the differences in USV emission between the mice genotypes.

We presented a demonstration for the usage of our model in order analyze USV recordings and provided a proof of concept for the analysis of the differences in USV emission between the mice genotypes. Our project lays the groundwork for future development of an automatic tool for ASD diagnosis in mice, based on the statistical differences we observed in USV usage.

6. Reference

[1] A. Masi, M. DeMayo, N. Glozier, and A. Guastella, "An Overview of Autism Spectrum Disorder, Heterogeneity and Treatment Options," vol. 33, no. 2, pp. 183–193, 2017, doi: 10.1007/s12264-017-0100-y.

[2] J. H. Elder, C. M. Kreider, S. N. Brasher, and M. Ansell, "Clinical impact of early diagnosis of autism on the prognosis and parent-child relationships," vol. 10, p. 283, 2017, doi: 10.2147/PRBM.S117499.

[3] A. S. Heinsfeld, A. R. Franco, R. C. Craddock, A. Buchweitz, and F. Meneguzzi, "Identification of autism spectrum disorder using deep learning and the ABIDE dataset," vol. 17, p. 16, 2018, doi: 10.1016/j.nicl.2017.08.017.

[4] C. P. Johnson and S. M. Myers, "Identification and evaluation of children with autism spectrum disorders," vol. 120, no. 5, p. 1183, 2007, doi: 10.1542/peds.2007-2361.

[5] N. Sadigurschi and H. M. Golan, "Maternal and offspring methylenetetrahydrofolate-reductase genotypes interact in a mouse model to induce autism spectrum disorder-like behavior," vol. 18, no. 1, p. n/a, 2019, doi: 10.1111/gbb.12547.

[6] C. Menuet et al., "Age-Related Impairment of Ultrasonic Vocalization in Tau.P301L Mice: Possible Implication for Progressive Language Disorders (Vocalization Impairment in Aging Tau.P301L Mice)," ,PLoS ONE, vol. 6, no. 10, p. e25770, 2011, doi: 10.1371/journal.pone.0025770.

[7] M. L. Scattoni, S. U. Gandhy, L. Ricceri, and J. N. Crawley, "Unusual Repertoire of Vocalizations in the BTBR T+tf/J Mouse Model of Autism (Atypical USVs in BTBR Mice)," vol. 3, no. 8, p. e3067, 2008, doi: 10.1371/journal.pone.0003067.

[8] M. L. Scattoni, L. Ricceri, and J. N. Crawley, "Unusual repertoire of vocalizations in adult BTBR T+tf/J mice during three types of social encounters," vol. 10, no. 1, p. 44, 2011, doi: 10.1111/j.1601-183X.2010.00623.x.

[9] G. Eysenbach et al., "Detecting Screams From Home Audio Recordings to Identify Tantrums: Exploratory Study Using Transfer Machine Learning," vol. 4, no. 6, 2020, doi: 10.2196/18279.

[10] R. P. Yagya, D. Kim, and J. Lee, "Domestic Cat Sound Classification Using Learned Features from Deep Neural Nets," vol. 8, no. 10, p. 1949, 2018, doi: 10.3390/app8101949.

[11] S. Tzur and R. Altshular, "Analysis of vocal signals in mice to identify Autistic behavior",2019

[12] K. R. Coffey, R. G. Marx, and J. F. Neumaier, "DeepSqueak: a deep learning-based system for detection and analysis of ultrasonic vocalizations," vol. 44, no. 5, p. 859, 2019, doi: 10.1038/s41386-018-0303-6.

[13] O. Malbin and Y. Rotaru, "Identification and classification of ultrasonic vocalizations in mice using Deep Neural Networks in order to predict autistic behavior", 2020

[14] Z. D. Burkett, N. F. Day, O. Peñagarikano, D. H. Geschwind, and S. A. White, "VoICE: A semi-automated pipeline for standardizing vocal analysis across models," vol. 5, no. 1, p. 10237, 2015, doi: 10.1038/srep10237.

[15] S. Hertz, B. Weiner, N. Perets, and M. London, "Temporal structure of mouse courtship vocalizations facilitates syllable labeling," vol. 3, no. 1, 2020, doi: 10.1038/s42003-020-1053-7.

[16] M. Van Segbroeck, A. T. Knoll, P. Levitt, and S. Narayanan, “MUPET—Mouse Ultrasonic Profile ExTraction: A Signal Processing Tool for Rapid and Unsupervised Analysis of Ultrasonic Vocalizations,” vol. 94, no. 3, pp. 465-485.e5, 2017, doi: 10.1016/j.neuron.2017.04.005.

[17] R. O. Tachibana, K. Kanno, S. Okabe, K. I. Kobayasi, and K. Okanoya, “USVSEG: A robust method for segmentation of ultrasonic vocalizations in rodents,” vol. 15, no. 2, p. e0228907, 2020, doi: 10.1371/journal.pone.0228907.

[18] A. H. O. Fonseca, G. M. Santana, B. O. Gabriela,M., S. Bampi, and M. O. Dietrich, “Analysis of ultrasonic vocalizations from mice using computer vision and machine learning,” vol. 10, 2021, doi: 10.7554/eLife.59161.

[19] E. Alpaydin, Introduction to machine learning. Cambridge, Massachusetts: Cambridge, Massachusetts : MIT Press, 2010.

[20] T. J. Sejnowski, “The deep learning revolution”, MIT press, vol. 9, pp. 127–142, 2018.

[21] V. A. Patel and M. V. Joshi, “Convolutional neural network with transfer learning for rice type classification,” vol. 10696. pp. 1069613-1069613–8, 2018, doi: 10.1117/12.2309482.

[21] M. M. Leonardo et al, "Deep Feature-Based Classifiers for Fruit Fly Identification (Diptera: Tephritidae)," Sibgra, pp. 41-47, 2018. doi: 10.1109/SIBGRAPI.2018.00012.