2023 IEEE Symposium Series on Computational Intelligence (SSCI)
Mexico City, Mexico. December 5-8, 2023

# A ResNet-9 Model for Insect Wingbeat Sound Classification

1st Béla J. Szekeres
*Department of Numerical Analysis*
*ELTE, Faculty of Informatics*
Budapest, Hungary
szekeres@inf.elte.hu

2nd Máté Natabara Gyöngyössy
*Department of Artificial Intelligence*
*ELTE, Faculty of Informatics*
Budapest, Hungary
natabara@inf.elte.hu

3rd János Botzheim
*Department of Artificial Intelligence*
*ELTE, Faculty of Informatics*
Budapest, Hungary
botzheim@inf.elte.hu

***Abstract*—Sound-based insect wingbeat classification presents a unique challenge with implications for areas such as mosquito control and the prevention of mosquito-borne diseases. This paper introduces a straightforward modified ResNet-9 model to address this challenge by utilizing one-dimensional convolutional layers. The architecture of the proposed ResNet-9 model is outlined in detail. Impressively, the model can accurately classify fruitflies and mosquitoes using raw audio data instead of relying on spectrograms. Its performance surpasses the majority of preceding models while concurrently reducing the number of trainable parameters by 90%. The results from this research carry notable significance for practical applications in insect control and disease prevention.**

***Index Terms*—Audio classification, ResNet architecture, Deep Learning, Mosquito Wingbeat**

## I. Introduction

Annually, diseases carried by mosquitoes afflict over 700 million people worldwide and result in over a million fatalities [1]. The roster of these illnesses is extensive, encompassing illnesses such as malaria, dengue fever, Zika virus fever, yellow fever, West Nile fever, and various forms of encephalitis [2]. Different species of mosquitoes are responsible for transmitting each of these diseases. Efficient monitoring plays a crucial role in controlling and preventing the spread of these diseases. Therefore, the classification of mosquitoes is of great importance and requires quick and accurate methods.

Fawazz et al. [3] provide a detailed overview of methods for time series classification. In our case, there are two main machine-learning approaches employed for this task: audio-based and image-based classification. Image-based solutions rely on analyzing images of mosquitoes to classify them [4]–[6]. Additionally, it has been observed that different mosquito species produce distinct audio characteristics through their wingbeats [7]. This observation justifies the exploration of audio or audio-like signals for classification purposes. Additionally, acoustic sensors are easier to use in practice, as locating flying insects and taking an image of them during flight would require an ultra-short exposure camera and expensive motoric equipment [8]. In audio-based methods, the raw wingbeat recordings or transformed data obtained through techniques such as short-time Discrete Fourier Transform (DFT) or Discrete Fourier Transform (DFT) are utilized [9]–[14].

Rigakis and Potamitis [15], [16] developed a system for recording insect wingbeats using large aperture optical sensors that convert light fluctuations caused by wing occlusion into acoustic signals. They created the *Wingbeats* and *Fruitflies* datasets. The *Wingbeats* dataset was subsequently investigated using various state-of-the-art deep learning architectures [14], [17]. Fanioudakis et. al. converted the audio signals into spectrograms and achieved a test accuracy of 96% using a DenseNet-121 model. However, these results could not be replicated according to the article [14].

It was shown that even inexpensive mobile phones are capable of capturing acoustic data related to mosquito wingbeat sounds [18]. The resulting *Abuzz* dataset was also examined using the so-called *WbNet* architecture proposed in [14], which was initially developed for the *Wingbeats* dataset.

In our contribution, we describe and apply a ResNet-based model to the *Wingbeats*, *Fruitflies*, and the *Abuzz* datasets, incorporating one-dimensional convolutional layers. We also support our work with numerical experiments. Our findings indicate that a small raw time-domain signal processing ResNet architecture can outperform the current state-of-the-art solutions on large benchmark datasets (Wingbeats and Fruitflies) while being on par or even more advantageous in terms of processing speed. We repeated the numerical experiments of the articles [14], [17], but using a strict train-validation-test separation to measure model generalization capabilities and avoid data leak, which is suspected to be present in the original publications according to the corresponding repositories* , this also ensures a fair comparison. For further details see Section II-B. To mention a few examples our proposed small and large models achieved 95.37% and 95.43% average test accuracies on the *Wingbeats* dataset among five different runs surpassing both the DenseNet-121 model's [17] 5-run average test accuracy of 91.92% and the test accuracy of 90.32% achieved by the WbNet architecture [14]. In the case of average test accuracies on the *Abuzz* dataset, *WbNet* models with best validation accuracies achieved 67.25% average test accuracy while our proposed models outperform it with 85.49% and 90.33%. Despite conducting numerous experiments, it was

*https://github.com/xutong30/WbNet-ResNet-Attention
https://www.kaggle.com/code/left13/mxnet-densenet121-0-955-acc-full-set

978-0-7381-4408-5/23/$31.00 ©2023 IEEE

| Species | Wingbeats | Abuzz |
|---|---|---|
| *Ae. aegypti* | 85553 | 324 |
| *Ae. Albopictus* | 20231 | 197 |
| *An. Gambiae* | 49471 | 171 |
| *An. Arabiensis* | 19297 | 95 |
| *Cu. pipiens* | 30415 | 66 |
| *Cu. quinquefasciatus* | 74599 | 62 |

TABLE I: Element numbers of each class in the *Wingbeats* and in the *Abuzz* datasets.

consistently found that both the DenseNet-121 and MobileNet architectures outperformed our proposed network when applied to the *Abuzz* dataset. Given the relatively small size of the *Abuzz* dataset, the fact that our proposed models demonstrate a significant performance advantage over other models on the *Fruitflies* dataset as well may mitigate this issue, i.e. our small *ResNet-9* model achieved here $97.64\%$ average test accuracy among five different runs, while the performance of other models remained below $93\%$. Our repository with the supporting source code is publicly available[†] .

The paper is structured as follows. In Section II, we provide the descriptions of the investigated *Wingbeats*, *Fruitflies*, and *Abuzz* datasets. Section III describes the proposed network architectures, methods, and experiments in detail. In Section IV the corresponding classification and efficiency-related results are presented. Finally, we summarize our findings in Section V.

## II. Datasets and preprocessing

### A. Descriptions

Three publicly accessible datasets, namely *Wingbeats* [16], *Fruitflies* [15] and *Abuzz* [18], are analyzed in this research. The *Wingbeats* and the *Abuzz* datasets encompass raw audio signals from six mosquito species spread across three different genera. These species include *Ae. aegypti*, *Ae. albopictus*, *An. arabiensis*, *An. gambiae*, *Cu. pipiens*, and *Cu. quinquefasciatus*. The data were recorded individually at the Biogents premises in Regensburg, Germany, employing large aperture optoelectronic devices for the *Wingbeats* dataset. Each audio recording maintains a duration of $0.65$s meaning it encapsulates $5000$ samples at an $8$ kHz sampling rate. The total number of recordings amounts to $279,556$. Imbalances in the dataset are observed, the specifics of which can be referred to in Table I.

The *Fruitflies* data were gathered from the areas of Gouves and Chersonisos in Crete, following the same method of collection. This dataset consists of audio signals captured from three distinct species of fruit flies: *Drosophila melanogaster*, *Drosophilia suzukii*, and *Zaprionus*. These signals were sampled at a rate of 8kHz. The dataset includes a total of $34,518$ recordings, with each recording lasting for a duration of $0.65$ seconds. The class sizes are detailed in Table II.

In the *Abuzz* dataset, data acquisition was carried out using mobile phones. The original recordings varied in length, with some lasting up to 5 minutes, and sample rates being either

[†]https://github.com/szbela87/insect_wingbeat_classification

| Species | Fruitflies |
|---|---|
| *Dr. melanogaster* | $6,064$ |
| *Dr. suzukii* | $10,142$ |
| *Zaprionus* | $18,312$ |

TABLE II: Element numbers of each class in the *Fruitflies* dataset.

8kHz or $44.1$kHz. The dataset employed in this study is the preprocessed version from the article [14], meaning that the original recordings were divided into segments, each being $10$ seconds long. Technical details can be accessed in the code repository associated with the WbNet architecture [‡]. The dataset contains a total of $915$ recordings, with the distribution of elements for each class provided in Table I. All signals were converted to 8kHz sampling rate during the numerical experiments.

### B. Data preprocessing

Raw audio signals are used in all datasets during the analysis. To ensure a fair comparison with the results from the articles [14], [17] the datasets are partitioned into training and testing sets in precisely the same manner. The training set comprises $80\%$ of the data, and a validation subset is subsequently split from this in a stratified way to measure generalization capabilities. This yields a $60/20/20\%$ split for the training/validation/testing sets, with the exact sizes of the sets available in Table III. The data are then preprocessed in the most straightforward way possible, treating the entire dataset as the measured data of a single variable. Consequently, simple standardization is performed by computing the necessary scalars, the mean and the standard deviation, of the training set.

Experiments based on the works of Fanioudakis et. al. [17] and Wei et. al. [14] use the same preprocessing steps as in the corresponding supporting code published by the authors. Therefore, the DenseNet-121 and MobileNet experiments [17] are executed on the non-standardized spectrograms, while the WbNet architecture [14] uses the same standardization procedure on the resulting spectrograms. We keep the spectrum processing parameters (e.g.: FFT bins, hop length, etc) unchanged and stick to the published configuration.

| Dataset | Training | Validation | Testing |
|---|---|---|---|
| *Wingbeats* | $167,739$ | $55,914$ | $55,913$ |
| *Abuzz* | $549$ | $184$ | $182$ |
| *Fruitflies* | $20,710$ | $6,904$ | $6904$ |

TABLE III: Sizes of the training, validation, and testing sets.

## III. Proposed methods

### A. The proposed model

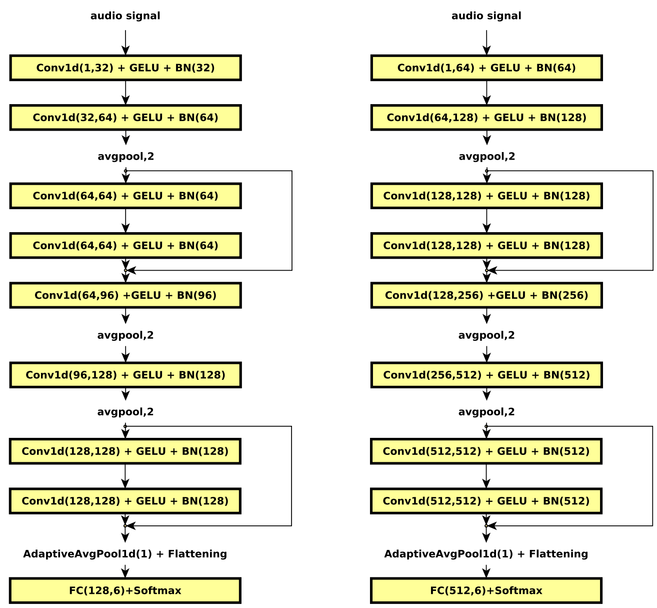
We propose a straightforward *ResNet-9* model [19]. The smaller network contains a total of approximately 670K trainable parameters and the larger one has around 8M parameters. Graphical representations of the architecture are presented in

[‡]https://github.com/xutong30/WbNet-ResNet-Attention

Figures 1. Further abbreviations in Figures 1 are explained as follows:

- Conv1D$(I, O)$: one-dimensional convolutional layer with $I$ input channels and with $O$ output channels,
- GELU: The GELU activation function
- BN$(I)$: Batch normalization in $I$ channels,
- FC$(I, O)$: Fully connected layer between $I$ and $O$ neurons.
- avgpool,2: Performs average pooling operation on the input by the kernel size 2, this is also the size of the stride here.
- AdaptiveAvgPool1d$(1)$: Performs adaptive average pooling operation, in this case, the size of the output is 1 by each channel.

The kernel size is 11 and the size of the padding is 5 in the one-dimensional convolution layers. The pooling size was set to 5 only in the case of the *Abuzz* dataset. Furthermore, the two sideways arrows in Figure 1 represent the residual connections.




(a) Smaller model with 670K trainable parameters

(b) Larger model with 8M trainable parameters

Fig. 1: The architecture of the proposed *ResNet-9* models.

### B. Experiments

We conduct experiments on all three datasets introduced in Section II using the proposed *ResNet-9* models of both sizes.

Stochastic Gradient optimizer and One-cycle learning rate scheduler [20] are used with Nesterov momentum during the training. The selection and configuration of the learning rate scheduler were also vital elements of our research.

The additional parameter configurations for the training are available in Table IV. These parameters were chosen empirically through model evaluations under varying settings on the validation set to circumvent both overfitting and underfitting. This philosophy was employed for the selection of the optimizer as well.

We conduct baseline experiments with regards to *DenseNet121* and *MobileNet* architectures proposed by Fanioudakis et. al. [17] and the *WbNet* architecture by Wei et. al. [14]. We investigated both the original papers and the released source code and found that these solutions were using the test set for evaluations during training, which is a form of data leak according to Goodfellow et. al. [21]. To solve this problem we use the same three-fold data split as with the *ResNet-9* models, but keep all the training parameters the same as the original authors set them. Additionally, to comply with our *Python 3.8* system we reimplement the code from Fanioudakis et. al. [17] to use *PyTorch* instead of *MxNet* while keeping all the algorithmic aspects unchanged. During training on the *Fruitflies* dataset we adopt *Wingbeat*'s training configuration for the three baseline models mentioned above.

The evaluation metric applied to the test set was accuracy. More precisely, this implies that the weights of the model that yielded the best validation accuracy during training were saved, and the model was subsequently evaluated on the test set. To mitigate the stochastic effect of weight initialization and batch ordering we perform a total of 5 runs in each experiment and report the mean accuracy of them. All models were executed on a single *NVIDIA GeForce RTX 3090 GPU*, using *Python 3.8* with supported libraries *PyTorch*, *Librosa*, *Pandas*, and *NumPy*.

| Dataset | Batch size | Epochs | Learning rate | Weight decay |
|---|---|---|---|---|
| *Wingbeats* | 32 | 25 | 0.0005 | 0.005 |
| *Abuzz* | 4 | 150 | 0.0002 | 0.005 |
| *Fruitflies* | 32 | 25 | 0.0001 | 0.005 |

TABLE IV: Training configuration for *ResNet-9*.

## IV. Results

### A. Classification experiments

We found that our proposed *ResNet-9* models have outperformed other models on the *Wingbeats* and the *Fruitflies* datasets. The numerical experiments in [14], [17] were repeated on all datasets with the same splitting for training/validation/testing sets to ensure a fair comparison among the different models. We summarize these test accuracy results in Table V, the two highest values in each column are marked in bold.

The small *ResNet-9* model achieved an average test accuracy of $95.37\%$ on the *Wingbeats* dataset, $85.49\%$ on the *Abuzz* dataset, and $97.64\%$ on the *Fruitflies* dataset. These results are computed as the mean accuracy over five separate runs. These five results were obtained by saving the model with the best validation accuracy during the training process and then evaluating them on the testing set. While it's important to point out that *ResNet-9* doesn't deliver the highest average test accuracy on the *Abuzz* dataset it outperforms the other models on the other two datasets. It is important to highlight that *Abuzz* is very small compared to *Wingbeats* and *Fruitflies*.

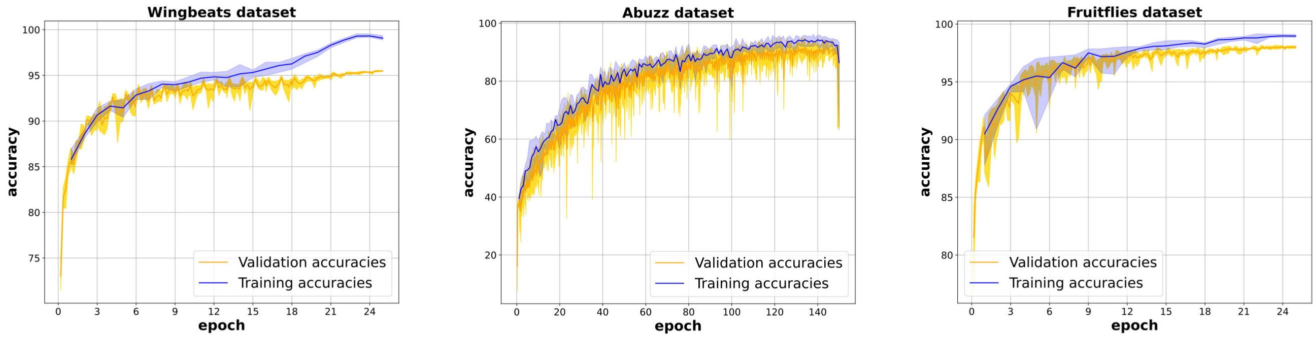



Fig. 2: Performance on the different datasets for the accuracy generated from 5 independent runs by the small *ResNet-9* model. The shaded region is between the maximum and minimum values over the runs, while the boldface curve displays the average.

The evaluation on the validation set was performed every 1000 training batch and at the end of each epoch for the *Wingbeats* dataset. For the *Abuzz* dataset, the evaluation occurred after every 30 batch and at the end of each epoch, and for the *Fruitflies* dataset, it was performed every 100 batch and at the end of each epoch.

The evolution of the accuracies on the training and the validation sets are shown in Fig. 2 for the small *ResNet-9* model. The exact details about the test results based on each species for the small and large *ResNet-9* models can be seen in Table VII and Table VIII, respectively. These results were obtained from the models with the best validation accuracy. In Figures 3 we present the confusion matrices for the *Wingbeats*, *Abuzz* and *Fruitflies* datasets respectively achieved by the best small *ResNet-9* model. For the *Wingbeats* and the *Abuzz* datasets the classes are arranged by genus in pairs, i.e. *Anopheles*, *Aedes*, and *Culex*. For the dataset *Fruitflies*, there are also two species of the same genus, namely *Drosophila*. It can be observed that most misclassifications occur between species of the same genus in all three cases.

However, this is not entirely true for the *Abuzz* dataset. In the confusion matrix, we can observe that most misclassifications arise from predictions for the *An. Gambiae* species class for the small *ResNet-9* model while in the case of the large *ResNet-9* model, the problematic class is the *Ae. Albopictus*. Based on the Tables VII-VIII, we can also notice that the predictions for the other classes are good based on the evaluation metrics, similar to the *Wingbeats* dataset.

In general, our proposed models performed better for the *Wingbeats* and the *Fruitflies* datasets compared to the *Abuzz* dataset. Of course, this may also be due to the fact that *Wingbeats* and *Fruitflies* contain much more recordings than the *Abuzz*. It is also important to note that the audios were short in both of the *Wingbeats* and *Fruitflies*, only $0.65s$ long, and captured using advanced audio equipment, while the *Abuzz* data were longer and recorded using mobile devices in noisy environments, varying up to 5 minutes in length. Therefore, *Abuzz* audio recordings were divided into multiple $10s$ long segments. We also find that by solving the data leak problem of previously proposed architectures [14], [17], their test performance dramatically drops compared to the originally reported. We believe that this is due to the deceiving practice of using the test set during training for internal evaluations.

| Wingbeats | | | | | |
|---|---|---|---|---|---|
| Architecture | *TP* | Data | *BVA %* | *TA %* | *ATA %* |
| small *ResNet-9* (ours) | 0.7M | *RS* | **95.55** | **95.43** | **95.37** |
| large *ResNet-9* (ours) | 8M | *RS* | **95.65** | **95.35** | **95.43** |
| *DenseNet121* [17] § | 7M | *SP* | 92.16 | 91.97 | 91.92 |
| *MobileNet* [17] § | 2M | *SP* | 91.41 | 91.14 | 91.20 |
| *WbNet* [14] § | 11M | *SP* | 87.62 | 91.09 | 90.32 |
| **Abuzz** | | | | | |
| Architecture | *TP* | Data | *BVA %* | *TA %* | *ATA %* |
| small *ResNet-9* (ours) | 0.7M | *RS* | 95.11 | 87.36 | 85.49 |
| large *ResNet-9* (ours) | 8M | *RS* | 97.28 | 92.86 | 90.33 |
| *DenseNet121* [17] § | 7M | *SP* | **99.79** | **97.09** | **95.94** |
| *MobileNet* [17] § | 2M | *SP* | **100.00** | **92.98** | **93.69** |
| *WbNet* [14] § | 11M | *SP* | 62.50 | 73.63 | 67.25 |
| **Fruitflies** | | | | | |
| Architecture | *TP* | Data | *BVA %* | *TA %* | *ATA %* |
| small *ResNet-9* (ours) | 0.7M | *RS* | **98.16** | **97.75** | **97.64** |
| large *ResNet-9* (ours) | 8M | *RS* | **98.32** | **97.71** | **97.67** |
| *DenseNet121* [17] § | 7M | *SP* | 92.65 | 92.68 | 92.91 |
| *MobileNet* [17] § | 2M | *SP* | 91.73 | 91.01 | 91.57 |
| *WbNet* [14] § | 11M | *SP* | 86.37 | 86.15 | 86.67 |

TABLE V: Accuracies for different architectures. The abbreviations stand for the following: *TP* denotes the number of the trainable parameters, *BVA* means the best validation accuracy, *TA* denotes the corresponding test accuracy for the model with the best validation accuracy, *ATA* marks the average test accuracy by the saved models with the highest validation accuracies among five different runs, *RS* means raw samples and *SP* shortens spectrogram.

## B. Efficiency

As observed in the results of Table V, the small *ResNet-9* model achieves a stable performance which is comparable to the large *ResNet-9* model. To provide further analysis of the practical usability of such models Table VI compiles the training (including both forward and backward steps) and inference (only forward steps) batch processing times. The two lowest values are written in bold. To account for memory

§Reevaluated experiment solving data leak in the original implementation.

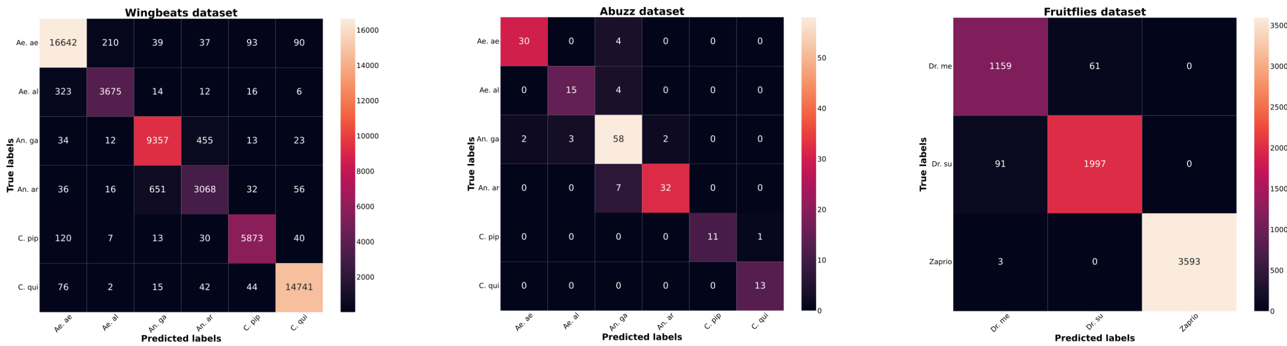



Fig. 3: Confusion matrices on the testing datasets for the small *ResNet-9* model which has the best validation accuracy among 5 different runs.

| Wingbeats | | | | | |
|---|---|---|---|---|---|
| Architecture | *TP* | Features | *BS* | *TTPS* | *ITPS* |
| small *ResNet-9* (ours) | 0.7M | *RS* | 512 | **0.45** | **0.14** |
| large *ResNet-9* (ours) | 8M | *RS* | 256 | 1.83 | 0.43 |
| *DenseNet121* [17] ¶ | 7M | *SP* | 256 | 0.85 | 0.30 |
| *MobileNet* [17] ¶ | 2M | *SP* | 512 | **0.38** | **0.14** |
| *WbNet* [14] ¶ | 11M | *SP* | 128 | 1.20 | 0.38 |
| Abuzz | | | | | |
| Architecture | *TP* | Features | *BS* | *TTPS* | *ITPS* |
| small *ResNet-9* (ours) | 0.7M | *RS* | 64 | **3.00** | **1.08** |
| large *ResNet-9* (ours) | 8M | *RS* | 32 | 20.97 | 5.11 |
| *DenseNet121* [17] ¶ | 7M | *SP* | 32 | 13.08 | 4.20 |
| *MobileNet* [17] ¶ | 2M | *SP* | 32 | **5.40** | **1.86** |
| *WbNet* [14] ¶ | 11M | *SP* | 8 | 22.04 | 5.11 |
| Fruitflies | | | | | |
| Architecture | *TP* | Features | *BS* | *TTPS* | *ITPS* |
| small *ResNet-9* (ours) | 0.7M | *RS* | 512 | **0.44** | **0.14** |
| large *ResNet-9* (ours) | 8M | *RS* | 256 | 1.92 | 0.40 |
| *DenseNet121* [17] ¶ | 7M | *SP* | 256 | 1.67 | 0.56 |
| *MobileNet* [17] ¶ | 2M | *SP* | 512 | **0.37** | **0.13** |
| *WbNet* [14] ¶ | 11M | *SP* | 128 | 1.22 | 0.42 |

TABLE VI: Sample processing speeds of different architectures using the largest power of two batch sizes which fit into $24GB$ memory. *TP* denotes the number of the trainable parameters, *BS* marks the batch size, *TTPS* is training time per sample in milliseconds, *ITPS* denotes inference time per sample in milliseconds, *RS* means raw samples and *SP* shortens spectogram. Processing speeds were averaged over a full epoch of training and validation.

efficiency we use the largest power of two as batch size which fits into a $24GB$ VRAM limit. This memory limitation matches the memory limit of consumer-grade desktop GPUs.

This analysis thus illustrates the speed with which a widely applicable cost-effective solution can perform this task, which could help in controlling the spread of insect-borne infections. To account for statistical instability our batch processing time results are averaged over a full epoch of training and validation.

¶Reevaluated experiment solving data leak in the original implementation.

| Wingbeats | | | |
|---|---|---|---|
| Species | Precision | Recall | F1-score |
| *Ae. aegypti* | 96.58 | 97.26 | 96.92 |
| *Ae. Albopictus* | 93.70 | 90.83 | 92.24 |
| *An. Gambiae* | 92.74 | 94.57 | 93.65 |
| *An. Arabiensis* | 84.19 | 79.50 | 81.78 |
| *Cu. pipiens* | 96.74 | 96.55 | 96.64 |
| *Cu. quinquefasciatus* | 98.56 | 98.80 | 98.68 |
| Abuzz | | | |
| Species | Precision | Recall | F1-score |
| *Ae. aegypti* | 93.75 | 88.24 | 90.91 |
| *Ae. Albopictus* | 83.33 | 78.95 | 81.08 |
| *An. Gambiae* | 79.45 | 89.23 | 84.06 |
| *An. Arabiensis* | 94.12 | 82.05 | 87.67 |
| *Cu. pipiens* | 100 | 91.67 | 95.65 |
| *Cu. quinquefasciatus* | 92.86 | 100 | 96.30 |
| Fruitflies | | | |
| Species | Precision | Recall | F1-score |
| *Dr. melanog.* | 92.49 | 95.00 | 93.73 |
| *Dr. suzukii* | 97.04 | 95.64 | 96.33 |
| *Zaprionus* | 100 | 99.92 | 99.96 |

TABLE VII: Evaluation metrics for different mosquito species by the best small *ResNet-9* models on the investigated datasets.

| Wingbeats | | | |
|---|---|---|---|
| Species | Precision | Recall | F1-score |
| *Ae. aegypti* | 96.63 | 97.48 | 97.06 |
| *Ae. Albopictus* | 93.31 | 90.36 | 91.81 |
| *An. Gambiae* | 92.29 | 94.27 | 93.27 |
| *An. Arabiensis* | 83.49 | 78.23 | 80.78 |
| *Cu. pipiens* | 97.02 | 96.84 | 96.93 |
| *Cu. quinquefasciatus* | 98.67 | 98.81 | 98.74 |
| Abuzz | | | |
| Species | Precision | Recall | F1-score |
| *Ae. aegypti* | 91.89 | 100.00 | 95.77 |
| *Ae. Albopictus* | 77.27 | 89.47 | 82.93 |
| *An. Gambiae* | 95.08 | 89.23 | 92.06 |
| *An. Arabiensis* | 97.30 | 92.31 | 94.74 |
| *Cu. pipiens* | 100 | 91.67 | 95.65 |
| *Cu. quinquefasciatus* | 92.86 | 100 | 96.30 |
| Frutflies | | | |
| Species | Precision | Recall | F1-score |
| *Dr. melanog.* | 91.63 | 95.98 | 93.75 |
| *Dr. suzukii* | 97.59 | 94.88 | 96.21 |
| *Zaprionus* | 99.94 | 99.94 | 99.94 |

TABLE VIII: Evaluation metrics for different mosquito species by the best large *ResNet-9* models on the investigated datasets.

## V. Conclusion and future work

In this study, we presented a simple approach for insect classification based on audio signals using a straightforward ResNet-9 model incorporating one-dimensional convolutional layers.

We investigated three benchmark datasets, the *Wingbeats*, the *Fruitflies*, and the *Abuzz*, which were recorded using different kinds of sensors. Our proposed architecture achieved high test accuracy on the *Wingbeats* and the *Fruitflies* datasets using the raw audio data, outperforming previously reported spectrum-based results on the *Wingbeats* dataset and the numerical experiments carried out in this study, based on the same architectures on the *Fruitflies* dataset. However, it was not effective for the very small-sized *Abuzz* dataset. Apart from that, the proposed smaller model is also streamlined and efficient, it consists of $90\%$ less trainable parameters than the above-mentioned deep learning models to process spectrograms. This leads to a processing speed that is on par with, or sometimes even faster than a MobileNet-v2 architecture working on spectral data. Our research makes a substantial contribution to advancing the development of effective and precise methods for classifying for example mosquitoes, thereby providing valuable support for controlling and intervening in mosquito-borne diseases.

Future research topics include applications of industrial time-series datasets, such as energy-efficient fault detection which could benefit from the parameter-efficient models.

The exploration of multi-domain neural operators for time series processing poses another question. Here Fourier-transform-based neural operators would provide a possibly strong, yet unexplored competitor to vanilla 1D convolutions.

## VI. Declarations

### A. Availability of supporting data

The *Wingbeats* dataset is publicly available in Kaggle https://www.kaggle.com/datasets/potamitis/wingbeats. The *Abuzz* dataset is publicly available at https://github.com/xutong30/WbNet-ResNet-Attention. The *Fruitflies* dataset is publicly available at https://timeseriesclassification.com/description.php?Dataset=FruitFlies. The codes and the datasets used in this article are available on the GitHub page https://github.com/szbela87/insect_wingbeat_classification.

### B. Acknowledgments

Béla J. Szekeres was supported by the project No. 2019-1.3.1-KK-2019-00011 financed by the National Research, Development and Innovation Fund of Hungary under the Establishment of Competence Centers, Development of Research Infrastructure Programme funding scheme.

## References

[1] H. Caraballo and K. King, "Emergency department management of mosquito-borne illness: malaria, dengue, and west nile virus," *Emergency medicine practice*, vol. 16, no. 5, p. 1—23; quiz 23—4, May 2014. [Online]. Available: http://europepmc.org/abstract/MED/25207355

[2] J. Palmer, A. Oltra, F. Collantes, J. Delgado, J. Lucientes, S. Delacour, M. Bengoa, R. Eritja, and F. Bartumeus, "Citizen science provides a reliable and scalable tool to track disease-carrying mosquitoes," *Nature Communications*, vol. 8, 10 2017.

[3] H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Muller, "Deep learning for time series classification: a review," *Data Mining and Knowledge Discovery*, vol. 33, no. 4, pp. 917–963, Jul 2019. [Online]. Available: https://doi.org/10.1007/s10618-019-00619-1

[4] K. Okayasu, K. Yoshida, M. Fuchida, and A. Nakamura, "Vision-based classification of mosquito species: Comparison of conventional and deep learning methods," *Applied Sciences*, vol. 9, no. 18, 2019. [Online]. Available: https://www.mdpi.com/2076-3417/9/18/3935

[5] D. Motta, A. Santos, Bandeira, B. A. S. Machado, O. G. V. Ribeiro-Filho, L. O. A. Camargo, M. A. Valdenegro-Toro, F. Kirchner, and R. Badaró, "Optimization of convolutional neural network hyperparameters for automatic classification of adult mosquitoes," *PLOS ONE*, vol. 15, no. 7, pp. 1–30, 07 2020. [Online]. Available: https://doi.org/10.1371/journal.pone.0234959

[6] V. Kittichai, T. Pengsakul, K. Chumchuen, Y. Samung, P. Sriwichai, N. Phatthamolrat, T. Tongloy, K. Jaksukam, S. Chuwongin, and B. S, "Deep learning approaches for challenging species and gender identification of mosquito vectors," *Scientific Reports*, vol. 11, 03 2021.

[7] W. H. Offenhauser and M. C. Kahn, "The sounds of disease-carrying mosquitoes," *Journal of the Acoustical Society of America*, vol. 21, pp. 462–463, 1949.

[8] D. C. Amarathunga, J. Grundy, H. Parry, and A. Dorin, "Methods of insect image capture and classification: A systematic literature review," *Smart Agricultural Technology*, vol. 1, p. 100023, 2021.

[9] B. Arthur, K. Emr, R. Wyttenbach, and R. Hoy, "Mosquito (aedes aegypti) flight tones: Frequency, harmonicity, spherical spreading, and phase relationships," *The Journal of the Acoustical Society of America*, vol. 135, 01 2014.

[10] Y. Chen, A. Why, G. Batista, A. Mafra-Neto, and E. Keogh, "Flying insect classification with inexpensive sensors," *Journal of Insect Behavior*, vol. 27, 03 2014.

[11] T.-H. Ouyang, E.-C. Yang, J.-A. Jiang, and T.-T. Lin, "Mosquito vector monitoring system based on optical wingbeat classification," *Computers and Electronics in Agriculture*, vol. 118, pp. 47–55, 10 2015.

[12] D. Vasconcelos, N. Nunes, M. Ribeiro, C. Prandi, and A. Rogers, "Locomobis: a low-cost acoustic-based sensing system to monitor and classify mosquitoes," 01 2019, pp. 1–6.

[13] J. Luna-Gonzalez, D. Robles-Camarillo, M. Nakano-Miyatake, H. Lanz-Mendoza, and H. Perez-Meana, *A CNN-Based Mosquito Classification Using Image Transformation of Wingbeat Features*, 09 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:234786525

[14] X. Wei, M. Hossain, and K. A. Ahmed, "A resnet attention model for classifying mosquitoes from wing-beating sounds," *Scientific Reports*, vol. 12, p. 10334, 06 2022.

[15] I. Rigakis, I. Potamitis, N.-A. Tatlas, I. Livadaras, and S. Ntalampiras, "A multispectral backscattered light recorder of insects' wingbeats," *Electronics*, vol. 8, no. 3, 2019. [Online]. Available: https://www.mdpi.com/2079-9292/8/3/277

[16] I. Potamitis and I. I. Rigakis, "Large aperture optoelectronic devices to record and time-stamp insects' wingbeats," *IEEE Sensors Journal*, vol. 16, pp. 6053–6061, 2016.

[17] E. Fanioudakis, M. Geismar, and I. Potamitis, "Mosquito wingbeat analysis and classification using deep learning," in *2018 26th European Signal Processing Conference (EUSIPCO)*, 2018, pp. 2410–2414.

[18] H. Mukundarajan, F. J. H. Hol, E. A. Castillo, C. Newby, and M. Prakash, "Using mobile phones as acoustic sensors for high-throughput mosquito surveillance," *eLife*, vol. 6, p. e27854, oct 2017. [Online]. Available: https://doi.org/10.7554/eLife.27854

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[20] L. N. Smith and N. Topin, "Super-convergence: very fast training of neural networks using large learning rates," in *Defense + Commercial Sensing*, 2017.

[21] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.