A Mosquito is Worth 16x16 Larvae: Evaluation of Deep Learning Architectures for Mosquito Larvae Classification

Aswin Surya,[1] David Backer Peral,[2] Austin VanLoon,[3] Akhila Rajesh[4]
[1]*Bellarmine College Preparatory, 960 West Hedding St, San Jose, CA 95126*
[2]*La Cañada High School, 4463 Oak Grove Dr, La Cañada Flintridge, CA 91011*
[3]*Middleton High School, 4801 North 22nd St, Tampa, FL 33610*
[4]*Westwood High School, 12400 Mellow Meadow Dr, Austin, TX 78750*

Mosquito-borne diseases (MBDs), such as dengue virus, chikungunya virus, and West Nile virus, cause over one million deaths globally every year. Because many such diseases are spread by the *Aedes* and *Culex* mosquitoes, tracking these larvae is critical to mitigating the spread of MBDs. Even as citizen science projects to obtain large mosquito image datasets continuously grow, the manual annotation of mosquito images is becoming ever more time-consuming and inefficient. Previous research has seen computer vision used to identify mosquito species, and the Convolutional Neural Network (CNN) has become the de-facto for image classification. But, these models typically require substantial computational resources. This research introduces the application of the Vision Transformer (ViT) in a comparative study to improve image classification on *Aedes* and *Culex* larvae. By using mosquito larvae image data from the GLOBE Observer Mosquito Habitat Mapper, two ViT models, ViT-Base and CvT-13, and two CNN models, ResNet-18 and ConvNeXT, from the HuggingFace library were trained and compared to determine the most effective model to classify mosquito larvae as *Aedes*, *Culex*, or neither. Testing revealed that ConvNeXT obtained the greatest values across all four classification metrics, making it a viable method for mosquito larvae image classification. Based on the results of this work, future research could include creating and implementing a model specifically designed for mosquito larvae classification by combining elements of CNN and transformer architecture.

Keywords: Vision Transformer, Convolutional Neural Network, Image Classification, Mosquito-Borne Disease.

RESEARCH QUESTION

This work aims to answer the following research question: Between vision transformers (ViTs) and convolutional neural networks (CNNs), which machine learning model is best to classify *Aedes* and *Culex* mosquito larvae to prevent the spread of mosquito-borne diseases?

1. INTRODUCTION

Among Earth's most deadly species, mosquitoes are among the deadliest. In fact, mosquito-borne diseases (MBDs) are responsible for at least 725,000 deaths annually (Barcelona Institute for Global Health, 2017). While MBDs have challenged humans for generations, factors like urbanization, climate change, and population growth have only exacerbated the issue (Sutherst, 2004).

MBDs are especially dangerous because mosquitoes transmit viruses easily. The three types of mosquitos are the *Aedes*, *Culex*, and *Anopheles* mosquitoes. *Aedes* and *Culex* mosquitoes are especially deadly because they can breed anywhere, not just in natural environments. For instance, the female *Aedes aegypti* mosquito can lay eggs in any moist and warm environment. In Brazil, the *Aedes aegypti* species alone started a Zika epidemic and caused 2,500 cases of microcephaly, a condition where a baby's brain has not developed properly (LaFrance, 2016). Additionally, *Culex* mosquitoes rapidly spread the West Nile virus, the leading cause of MBD in the United States (CDC, 2022).

The spread of MBDs can be prevented by classifying mosquito larvae, which are distinguishable by their siphon. Larvae classification allows health officials to track mosquito populations in an area, learn which species thrive in certain environments, and forecast the presence of invasive species to prevent outbreaks, as only certain mosquito species transmit certain viruses (Joshi, 2021). This preventative approach is important because many MBDs like dengue virus and West Nile Virus, which has become endemic in the U.S., have no vaccine or treatment (CDC, 2021; Petersen, 2017).

Recent research has focused on the use of artificial intelligence (AI) as an alternative to manual classification. For image classification, convolutional neural networks (CNNs) and vision transformers (ViTs) are the most common. CNNs perform convolution to locally extract features from an image and produce a feature map, from which the network can classify an image. Many past works have previously applied CNNs to identify mosquito-specific tasks. Goodwin et al. (2021) achieved an accuracy of 89.50% when identifying unknown mosquito species and 88.72% when identifying known species with CNNs. Elango et al. (2022) compared CNNs with the You-Only-Look-Once (YOLO) algorithm to predict mosquito habitats, and the YOLOv4 CNN worked best. The study concluded that CNNs were the most efficient and cost-effective approach to predict large-scale mosquito habitats, but were unable to identify small-scale habitats such as footprints, tires, or puddles.

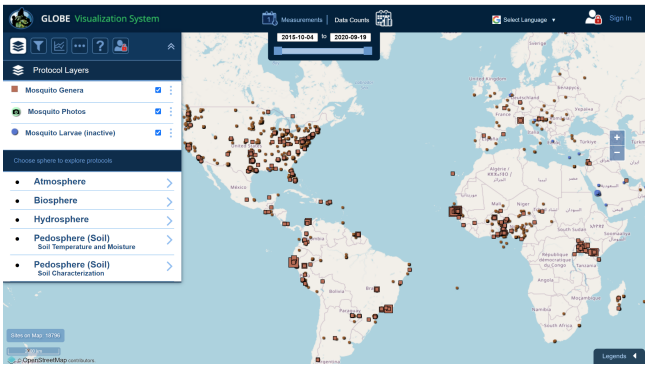

FIG. 1. The GLOBE Mosquito Habitat Mapper database, used to access the larvae data.

While CNNs were the longtime state-of-the-art machine learning for image classification, Dosovitskiy et al. (2021) proposed a novel architecture that outperformed CNNs: the ViT.

Rather than convolutions, the ViT uses self-attention to integrate all features of the data. Unlike CNNs, ViTs have not been extensively applied to mosquito-related tasks. Thus far, Sengar et al. (2022) reached an accuracy of 90.03% in using ViTs to predict malaria using thin blood smear microscopic images.

This work seeks to compare the CNN to the ViT for mosquito larvae classification to proactively prevent the spread of MBDs. This study compares four machine learning models to classify larvae as *Aedes*, *Culex*, or neither, a distinction from many previous works.

2. METHODS

2.1. Mosquito Larvae Dataset

All mosquito larvae data were obtained using the GLOBE Mosquito Habitat Mapper database (depicted in Figure 1 above), an

app-based citizen science initiative to collect nationwide data on mosquito habitats and identification. Data were collected from May 31, 2017 to July 7, 2022, from North America, Latin America, and Africa.

2.2. *Data Preparation*

This work required a multistep data preparation process to maintain the quality, accuracy, and consistency of the data. The data was downloaded using a spreadsheet builder and then converted into comma-separated values (CSV) files. Each image had to be carefully classified as *Aedes*, *Culex*, or neither, based on the length and shape of the siphon, color of the larvae, and amount of hair. The main feature that distinguishes larvae is their siphon, the organ used to breathe. *Aedes* larvae have a shorter and wider siphon, while *Culex* larvae have a longer and thinner siphon. After classification, the additional commas in the "location" column needed to be removed due to the data loader assuming that those were extraneous blank columns.

Pre-processing also included removing several image links that were no longer supported by the server. When loading the image data, a separate script was used to identify and remove URLs with erroneous HTTP status codes. Finally, null values were also reviewed and validated.

After cleaning the data, the dataset was uploaded to the HuggingFace Hub to be easily imported into code. The train data was the mosquito larvae data from Africa, consisting of 7107 rows, while the test data was from North America and Latin America, consisting of 3439 rows. The image links were cast as PIL image data and converted to pixel values. Prior to training each model, feature extraction was performed on the data using the Feature Extractor class in the HuggingFace transformers library. Finally, the classification head of each model was altered to three to account for the three individual labels.

2.3. *Convolutional Neural Network*

A Convolutional Neural Network (CNN) is a deep learning (DL) architecture widely used for image classification that extracts local features and learns directly from them using kernel convolutions. The kernel convolution is the process of taking a small matrix of numbers, called a filter, and passing it over the image in a specified stride and padding. This process repeats, constantly sliding until the entire image has been covered. The result of this process is a feature map, from which the network can comprehend features and make a classification. The formula to calculate the dimensions of the output matrix, n_{out}, where p is the padding, f is the filter dimension, s is the stride, and n_{in} is the input image is:

$$n_{out} = \left[\frac{n_{in} + 2p - f}{s} + 1 \right]$$

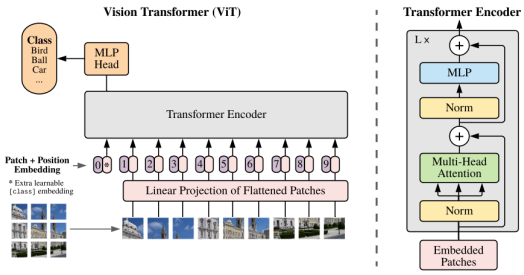

FIG. 2. A side-by-side comparison of the Vision Transformer and the Transformer Encoder. The attention-based Transformer Encoder is utilized in the ViT, but it also includes a classification token and multi-layer perceptron to perform class prediction. Figure acquired from Adaloglou (2021).

$$\text{MultiHead}\left(\mathbf{Q}, \mathbf{K}, \mathbf{V}\right) = \text{Concat}\left(\text{head}_1, \ldots, \text{head}_\text{h}\right)\mathbf{W}^O$$
$$\text{where head}_\text{i} = \text{Attention}\left(\mathbf{Q}\mathbf{W}_i^Q, \mathbf{K}\mathbf{W}_i^K, \mathbf{V}\mathbf{W}_i^V\right)$$

FIG. 3. Multi-head attention using query, key, and values. Each independent head is concatenated and transformed using a square weight matrix, allowing the model to perform self-attention computation in parallel. Figure acquired from Adaloglou (2020).

2.4. *Vision Transformer*

Transformers were first introduced as a novel approach to Natural Language Processing (NLP) using a sequential, purely attention-based mechanism (Vaswani et al., 2017). Later, the transformer was applied to computer vision by Dosovitskiy et al. (2021), creating the vision transformer (ViT). Inspiring the title of this paper, the ViT sought to utilize nothing but attention on sequential patches of images. It splits an input image into patches and uses positional embeddings to capture the relative location of the patch. The ViT uses a mechanism known as self-attention instead of convolutions to integrate all features of the data across the model, from the lowest to highest layers, enabling the network to learn the local and global features of the image (see Figure 2).

The ViT architecture also employs multi-head attention, running through the attention mechanism several times (see Figure 3). The query, key, and value matrices are mapped into lower-dimensional spaces, and attention is continually computed, with each output referred to as a head. The independence of each head allows for parallelization of self-attention computation, facilitating the model to attend to different parts of the sequence differently and capture unique positional information.

2.5. *ResNet-18*

One of the most-employed CNN models is ResNet, which adopted residual learning to stacked layers. Deep CNNs face the vanishing gradient problem, where gradients calculated from the loss function shrink to zero because of repeated applications of the chain rule, hindering training in early layers. This saturates the network's accuracy and increases the network's training error, effectively reducing its generalization capability. ResNet adds skip connections to allow the gradients to flow from the final layers to the initial filters. The skip connections with residual mapping also allowed ResNets to more easily optimize and gain accuracy from increased depth, removing the previous bottleneck. ResNets were shown to perform better than comparative CNNs such as VGG, GoogLeNet, and Inception networks (He et al., 2015).

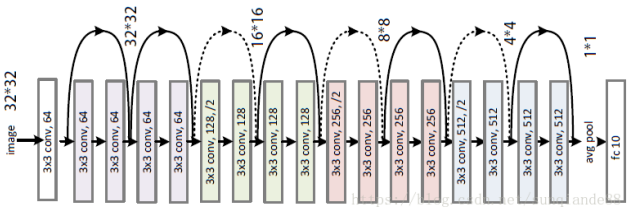

FIG. 4. The ResNet-18 with 18 parameter layers with inserted shortcut connections for performing identity mapping. Table 1 has more details for other architectures. Figure acquired from Duchesnay et al. (2020).

ResNet-18 is the CNN architecture of choice in this work as it is a widely-used CNN benchmark in image classification. The ResNet-18 used was pre-trained on ImageNet-1k and fine-tuned on the mosquito larvae dataset.

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
		3×3 max pool, stride 2				
conv2_x	56×56	[3×3, 64; 3×3, 64] ×2	[3×3, 64; 3×3, 64] ×3	[1×1, 64; 3×3, 64; 1×1, 256] ×3	[1×1, 64; 3×3, 64; 1×1, 256] ×3	[1×1, 64; 3×3, 64; 1×1, 256] ×3
conv3_x	28×28	[3×3, 128; 3×3, 128] ×2	[3×3, 128; 3×3, 128] ×4	[1×1, 128; 3×3, 128; 1×1, 512] ×4	[1×1, 128; 3×3, 128; 1×1, 512] ×4	[1×1, 128; 3×3, 128; 1×1, 512] ×8
conv4_x	14×14	[3×3, 256; 3×3, 256] ×2	[3×3, 256; 3×3, 256] ×6	[1×1, 256; 3×3, 256; 1×1, 1024] ×6	[1×1, 256; 3×3, 256; 1×1, 1024] ×23	[1×1, 256; 3×3, 256; 1×1, 1024] ×36
conv5_x	7×7	[3×3, 512; 3×3, 512] ×2	[3×3, 512; 3×3, 512] ×3	[1×1, 512; 3×3, 512; 1×1, 2048] ×3	[1×1, 512; 3×3, 512; 1×1, 2048] ×3	[1×1, 512; 3×3, 512; 1×1, 2048] ×3
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

TABLE 1. ResNet architectures for ImageNet. Table acquired from He et al. (2019).

2.6. *ConvNeXT*

After their success, the application of the ViT's design in CNNs was limited, leading to the ConvNeXT.

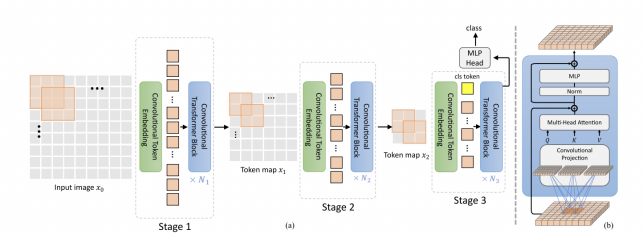

FIG. 6. CvT architecture pipeline. a) Overall architecture, demonstrating hierarchical structure with the Convolutional Token Embedding layer. b) Expanded view of the Convolutional Transformer Block, containing convolutional projection. Figure acquired from Wu et al. (2021).

Beginning with ResNet-50, Liu et al. (2022) gradually modernized the architecture to the design of a transformer by training the ResNet using vision transformer techniques like the AdamW Optimizer, extending epochs to 300 instead of the usual 90, and several regularization schemes. ConvNeXT replaced the ResNet's 4x4 downsampling stem cell with a "patchify" strategy found in Vision Transformers, with larger kernel size and non-overlapping convolution. Depthwise convolutions were also utilized, while at the same time, increasing the network width. The ReLu activation function was replaced with the Gaussian Error Linear Unit (GELU), which drops all residual layers from the block except the one between 1x1 Conv layers, replicating the style of the Transformer block. The equation is given by:

$$GELU(x) = 0.5x(1 + tanh(\sqrt{2/\pi}(x + 0.044175x^3)))$$

The ConvNeXT model used was pre-trained on ImageNet-22k at resolution 224x224 and fine-tuned on the mosquito-larvae dataset.

2.7. *Convolutional Vision Transformer (CvT)*

The Convolutional Vision Transformer (CvT) introduces convolutions to the vision transformer architecture (see Figure 6). A Convolutional Token Embedding layer applies convolutions on the tokenized patches and reshapes them, effectively reducing the number of tokens and increasing their depth, similar to conventional CNNs. Additionally, Convolutional Transformer Blocks are applied instead of the usual position-wise linear projection in the ViT, where a depth-wise separable convolution operation is employed on the query, key, and value embeddings.

The CvT-13 model used was pre-trained on ImageNet-1k at resolution 224x224, and then fine-tuned on the mosquito larvae dataset.

2.8. *Evaluation Metrics*

All models were evaluated using four metrics: accuracy, precision, recall, and F1 score, all calculated from four values. True positives (TP) occur when the model correctly classified the test image as it was actually labeled. False positives (FP) occur when the model categorizes an image with another, incorrect classification. False negatives (FN) occur when the model does not label a mosquito larva image with the right label. True negatives (TN) occur when the model correctly classifies another image as another label as opposed to the class considered.

Accuracy describes the total number of images correctly classified:

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

Precision measures how often the model is correct when it classifies images:

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

Recall calculates how many actual positives the model captures out of the total "positive" images labeled:

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

F1 Score is the harmonic mean of precision and recall, and it applies the same level of importance to both precision and recall, indicating the overall performance of the model:

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

3. RESULTS AND DISCUSSION

The results are shown in Table 2 below. All four models had similar performances around 60%, with much lower values than expected. This general finding could be due to the regional difference between the training and testing data. In the Africa train data, there were 3917 rows of *Aedes*, 2405 rows of *Culex*, and 785 rows of "neither." In the Americas test data, there were 2062 rows of *Aedes*, 1253 rows of *Culex*, and only 124 rows of "neither." The percentage of "neither" rows in the Africa data was 11.05% while the percentage in the Americas data was 3.61%. This regional difference in the number of mosquito larvae that were not *Aedes* or *Culex* might have contributed to the lack of accuracy from training to testing.

Algorithm Results				
Model	*Accuracy*	*Precision*	*Recall*	*F1 Score*
ViT-Base	0.6374	0.6061	0.6374	0.5868
CvT-13	0.6400	0.6292	0.6400	0.6209
ConvNeXT	**0.6563**	**0.6386**	**0.6563**	**0.6355**
ResNet-18	0.5967	0.6034	0.5967	0.5756

TABLE 2. Results from all four algorithms. The highest performing model across all metrics is the ConvNeXT.

Another issue that was faced during the training of the models was overfitting, when a model fits exactly to its training data so it is unable to classify on new, unseen test data. One reason for overfitting could be the size of the models. The ViT-Base model had 86 million trainable parameters, the CvT-13 had 19.98 million parameters, the ConvNeXT had 89 million trainable parameters, and the ResNet-18 had 11 million parameters. These large numbers of parameters means overfitting is likely. Additionally, overfitting could also be observed during training, as the training loss kept decreasing at a steady rate, while the evaluation loss decreased for a short amount of time before rising up and continually increasing.

The ConvNeXT scored the highest on all four classification metrics. This was likely due to the combination of standard transformer techniques with state-of-the-art CNN models like ResNet-50 (see section 2.6). Depthwise convolutions would also play a role in the robustness of the network by providing additional network width for the data to be more integrated than the other models.

The CvT-13 model performed the second best overall, within 1.63% of the ConvNeXT. The implementation of convolutions improved performance. Although it had significantly fewer trainable parameters, it still managed to achieve similar metric values. This could be due to the multi-head attention mechanism (see section 2.4).

Although ViTs have been shown to outperform CNNs with large datasets such as CIFAR-100 and ImageNet, the ViT-Base model resulted in an accuracy of 63.74%. ViTs generally require a large amount of data to perform well, and since the mosquito larvae dataset only had 10,546 rows of data, it could have been insufficient to train. Additionally, all models were trained for 4 epochs, whereas Dosovitskiy et al. (2021) trained the ViT for 7 epochs. This was not done due to limited computational resources. Moreover, it seems that depthwise separable convolutions are a critical component in image classification due to their ability to increase the depth of an image.

ResNet-18 had the least values across all metrics. This could be due to its significantly lower number of parameters, 11 million, resulting

in underfitting, when the model is unable to capture the features of the image due to limited parameters or training time. Additionally, depthwise convolutions are not included in the ResNet architecture. Because ResNet-18 was a simpler architecture in comparison to the other three models, this result is expected. Excluding the skip connections, the ResNet-18 is simply a basic CNN model, explaining the low accuracy.

4. CONCLUSIONS

Although all four models performed similarly for classifying mosquito larvae images as *Aedes*, *Culex*, or neither, the metric values were much lower than expected. ConvNeXT performed with the highest accuracy among the 4 models with an accuracy of 65.63%. This is likely because of the variation of regional data from the training and testing dataset. Another factor could be the images themselves, which are often only a part of the mosquito larvae instead of the whole body. From this result, it seems that CNNs with depthwise separable convolutions perform better in complex image classification tasks than purely attention-based models.

Future research could include the integration of mosquito species as opposed to genus. Mapping could also be performed by correlating the mosquito species to the most likely disease it spreads. The dataset could be further expanded to higher-quality images that reveal the whole larvae body as opposed to a single part. Testing with various sizes of the same model would allow for a comparison of parameters and identification of the optimal model capacity for image classification. This could result in a portable tool that can quickly and efficiently classify mosquito larvae while conducting fieldwork. A novel model could also be created with the sole purpose of mosquito larvae image classification by utilizing depthwise convolutions and aspects of the transformer architecture, creating a hybrid CNN and ViT model like the CvT-13 and the ConvNeXT. During the training process of these models, a finer, more thorough analysis could be conducted using methods such as increasing the number of epochs, raising the probability of dropout, and adding specific data preparation techniques. Through this approach, epidemics can be controlled efficiently and at a rapid pace by locating and identifying potentially dangerous mosquito larvae. There are several cases where rapid mosquito larvae identification and classification are necessary for public health control. Extrapolation of key features from vision transformers and convolutional neural networks to create a more efficient model would prove as a viable, cost-effective, and autonomous approach to controlling the spread of mosquito-borne diseases.

DATA AND CODE

Data and code to replicate the results of this experiment are available at the following public Github repo: https://github.com/thenerd31/vit-cnn-mosquito-image-classification.

ACKNOWLEDGEMENTS

We would like to thank the NASA SEES Internship program for giving us the background knowledge and support needed to start this project. We acknowledge our mentors Russanne Low, Peder Nelson, Matteo Kimura, and Cassie Soeffing for their tremendous support and guidance in the research and publication process. We would also like to graciously thank Ria Jain for her incredible guidance through the writing process and her expertise in AI. We would like to thank HuggingFace and PyTorch for giving us the software capabilities of organizing our dataset and importing them into Jupyter Notebook.

The authors would like to acknowledge the support of the 2022 Earth Explorers Team, NASA STEM Enhancement in the Earth Sciences (SEES) Virtual High School Internship program. The NASA Earth Science Education

Collaborative leads Earth Explorers through an award to the Institute for Global Environmental Strategies, Arlington, VA (NASA Award NNX6AE28A). The SEES High School Summer Intern Program is led by the Texas Space Grant Consortium at the University of Texas at Austin (NASA Award NNX16AB89A), or The SEES High School Summer Intern Program is in partnership with NASA Cooperative Agreement Notice NNH15ZDA004C Award NNX16AB89A.

REFERENCES

(1) Adaloglou, N. (2020, December 24). *How transformers work in Deep Learning and NLP: An intuitive introduction*. AI Summer. Retrieved July 24, 2022, from https://theaisummer.com/transformer/#the-core-building-block-multi-head-attention-and-parallel-implementation.

(2) Adaloglou, N. (2021, January 28). *How the vision transformer (VIT) works in 10 minutes: An image is worth 16x16 words*. AI Summer. Retrieved July 10, 2022, from https://theaisummer.com/vision-transformer.

(3) Centers for Disease Control and Prevention. (2021, September 20). *Symptoms and treatment*. Centers for Disease Control and Prevention. Retrieved July 27, 2022, from https://www.cdc.gov/dengue/symptoms/index.html#:~:text=There%20is%20no%20specific%20medicine,and%20see%20your%20healthcare%20provider.

(4) Centers for Disease Control and Prevention. (2022, June 2). *West Nile virus*. Centers for Disease Control and Prevention. Retrieved July 25, 2022, from https://www.cdc.gov/westnile/index.html.

(5) Chetoui, M., & Akhloufi, M. A. (2022, May 26). *Explainable vision transformers and Radiomics for COVID-19 detection in chest X-rays*. Journal of Clinical Medicine. Retrieved July 25, 2022, from https://www.mdpi.com/2077-0383/11/11/3013.

(6) Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2021, June 3). *An Image is Worth 16x16 words: Transformers for Image Recognition at Scale*. arXiv.org. Retrieved July 15, 2022, from https://arxiv.org/abs/2010.11929.

(7) Duchesnay, E., Löfstedt, T., & Younes, F. (2020). *Convolutional neural network¶*. Convolutional neural network - Statistics and Machine Learning in Python 0.5 documentation. Retrieved July 27, 2022, from https://duchesnay.github.io/pystatsml/deep_learning/dl_cnn_cifar10_pytorch.html

(8) Elango, S., Ramachandran, N., & Low, R. (2022, March 12). *Autonomous Mosquito habitat detection using satellite imagery and convolutional neural networks for disease risk mapping*. arXiv.org. Retrieved July 17, 2022, from https://arxiv.org/abs/2203.04463.

(9) Goodwin, A., Padmanabhan, S., Hira, S., Glancey, M., Slinowsky, M., Immidisetti, R., Scavo, L., Brey, J., Sai Sudhakar, B. M. M., Ford, T., Heier, C., Linton, Y.-M., Pecor, D. B., Caicedo-Quiroga, L., & Acharya, S. (2021, July 1). *Mosquito species identification using convolutional neural networks with a multitiered ensemble model for novel species detection*. Nature News. Retrieved July 24, 2022, from https://www.nature.com/articles/s41598-021-92891-9.

(10) He, K., Zhang, X., Ren, S., & Sun, J. (2015, December 10). *Deep residual learning for image recognition*. arXiv.org. Retrieved July 24, 2022, from https://arxiv.org/abs/1512.03385.

(11) Joshi, A., & Miller, C. (2021, January 26). *Review of machine learning techniques for Mosquito Control in urban environments*. Ecological Informatics. Retrieved July 24, 2022, from https://www.sciencedirect.com/science/article/pii/S1574954121000327.

(12) LaFrance, A. (2016, April 26). *What makes the zika-spreading mosquito such a menace*. The Atlantic. Retrieved July 25, 2022, from https://www.theatlantic.com/science/archive/2016/04/aedes-aegypti/479619/.

(13) Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., & Guo, B. (2021, August 17). *Swin Transformer: Hierarchical vision transformer*

using shifted windows. arXiv.org. Retrieved July 24, 2022, from https://arxiv.org/abs/2103.14030.

(14) Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., & Xie, S. (2022, March 2). *A ConvNet for the 2020s*. arXiv.org. Retrieved July 24, 2022, from https://arxiv.org/abs/2201.03545.

(15) Mishra, N. (2015, September 16). *Why is there still no cure or vaccine for dengue fever?* TheQuint. Retrieved July 25, 2022, from https://www.thequint.com/fit/health-news/why-is-there-still-no-cure-or-vaccine-for-dengue-fever-2#read-more

(16) *Mosquitoes: World's Deadliest animal*. ISGlobal. (2017, August 18). Retrieved July 20, 2022, from https://www.isglobal.org/en_GB/-/mosquito-el-animal-mas-letal-del-mundo.

(17) Petersen, L. R. (2017, November 21). *Decades later, still no vaccine or treatment for West Nile virus*. Healio. Retrieved July 25, 2022, from https://www.healio.com/news/infectious-disease/20171115/decades-later-still-no-vaccine-or-treatment-for-west-nile-virus.

(18) Sengar, N., Burget, R., & Dutta, M. K. (2022, July 1). *A Vision Transformer-based approach for analysis of Plasmodium vivax life cycle for malaria prediction using thin blood smear microscopic images*. Computer Methods and Programs in Biomedicine. Retrieved July 24, 2022, from https://www.sciencedirect.com/science/article/abs/pii/S0169260722003789.

(19) Skalski, P. (2019, April 14). *Gentle dive into math behind Convolutional Neural Networks*. Medium. Retrieved July 24, 2022, from https://towardsdatascience.com/gentle-dive-into-math-behind-convolutional-neural-networks-79a07dd44cf9.

(20) Sutherst, R. W. (2004, January). *Global change and human vulnerability to vector-borne diseases*. Clinical microbiology reviews. Retrieved July 27, 2022, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC321469/.

(21) Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017, December 6). *Attention is all you need*. arXiv.org. Retrieved July 27, 2022, from https://arxiv.org/abs/1706.03762.

(22) Wu, H., Xiao, B., Codella, N., Liu, M., Dai, X., Yuan, L., & Zhang, L. (2021, March 29). *CvT: Introducing Convolutions to Vision Transformers*. arXiv.org. Retrieved July 24, 2022, from https://arxiv.org/abs/2103.15808.

International Virtual Science Symposium Badges

Badges Applied For:

Data Science: This badge is being applied for due to the large amount of data collected from GLOBE Observer from 2017 to 2022 in North America, South America, and Africa. Several data preparation techniques were utilized to ensure high-quality data of mosquito larvae. We employed this data to compare state-of-the-art vision transformer models and CNN models on how well they did in classifying each image as *Aedes*, *Culex*, or neither. We also released this data to the public as a database of mosquito larvae images, which is available at https://huggingface.co/datasets/TheNoob3131/mosquito-data.

Engineer: This badge is being applied because we utilized feature extraction and data augmentation to improve the performance of our ViT-base model, ConvNeXT model, ResNet-18 model, and CvT model. This, in turn, allows us to determine which model is most capable to classify mosquito larvae species, providing a solution to scientists and researchers out in the field to quickly identify mosquito larvae and prevent potentially dangerous diseases.

Impact: This badge is being applied as a result of our models being the first to classify *Aedes*, *Culex*, or other species as larvae, as opposed to adult classification. Therefore, our research enables scientists to identify disease-carrying mosquitoes before they fully develop into adults and contributes to the active prevention of mosquito epidemics worldwide.