Using artificial neural networks to classify optimal microswimmers based on their shapes[*]

Niyizhen Jin[†][§], Xinyue Qie[‡][§], Nicole Surgent , and Wanting Huang

Project advisor: Hanliang Guo

Abstract. Studies of microswimmers have received increasing attention since the 2000s fueled by the advancements in micro-manufacturing and their potential for exciting biomedical applications. One of the popular mathematical research directions is the optimization of the flagella- or cilia-kinematics to maximize the swimming efficiency, usually for isolated microswimmer. The collective behaviors, on the other hand, are affected by the types of microswimmers (e.g., pusher, puller, or neutral). Understanding the connections between the optimal activation of a given shape and its swimming-type can have important implications on designing artificial microswimmers. In this work, we build an artificial neural network (ANN) that can predict the types of optimal microswimmers based solely on their shapes. More interestingly, we show that the tangent vector information along the microswimmer surface is important for the ANN to successfully classify the microswimmers.

1. Introduction.

Microswimmers is a class of microscopic machines that can move in fluid environments with a swimming motion. It was first observed in 1677 when Antonie Van Leeuwenhoek remarked the jiggling motion of sperm cells (see, e.g., Budrikis [2020]). Since then, microswimmers including many prokaryotes and eukaryotes have received increasing attention from scientists in various fields, from biologists [Tamm, 1972, Sleigh, 1991] to mathematicians [Taylor, 1951, Brennen and Winet, 1977, Lauga and Powers, 2009, Gaffney et al., 2011]. Some of the well-studied microswimmers are *Volvox*, spermatozoa, and *Paramecium* (see Figure 1). These microswimmers have flagella- or cilia-based swimming motions, where the flagella or cilia provide the driving force for the cell's movement. Specifically, ciliated microswimmers such as *Paramecium* and *Opalina* feature hundreds of motile cilia on the cell surface. The coordinated motions of the dense ciliary carpet creates slip velocities tangent to the cell surfaces and drive the microswimmers efficiently.

On the other hand, artificial microswimmers have gained tractions since the early 2000s benefitted by the significant advancements in micro-manufacturing. Some artificial microswimmers are inspired by their biological counterparts. For example, Dreyfus et al. [2005] designed a microswimmer by attaching an externally actuated flagellum-like artificial component to a human red blood cell. Other artificial microswimmers can swim without the external actuation. For example, diffusiophoretic particles propel themselves by exploiting the asymmetry of chemical reactions on their surfaces [Anderson, 1989, Golestanian et al., 2007]. Such particles, like the Janus particle [Howse et al., 2007], generate chemical gradients which in turn create an effective steady slip velocity on the surface by osmosis effects. Exciting biomedical applications of artificial microswimmers include cargo transport [Klumpp and Lipowsky, 2005], drug delivery [Qiu et al., 2015], artificial insemination [Medina-Sánchez et al., 2016], sensing [Sun

[*] Department of Mathematics, University of Michigan, Ann Arbor, MI.

[†] jinni@umich.edu

[‡] xqie@umich.edu

[§] These authors contributed equally to this project.

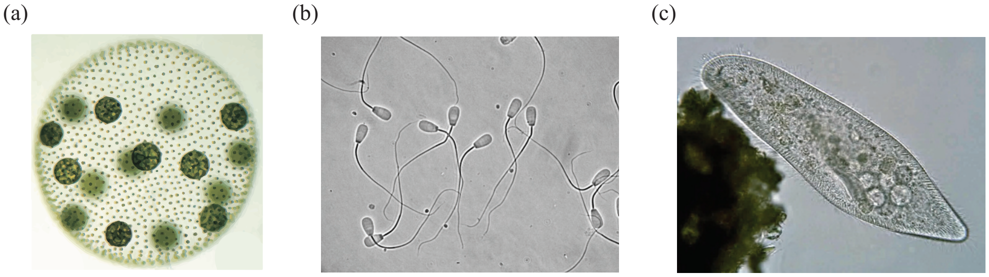


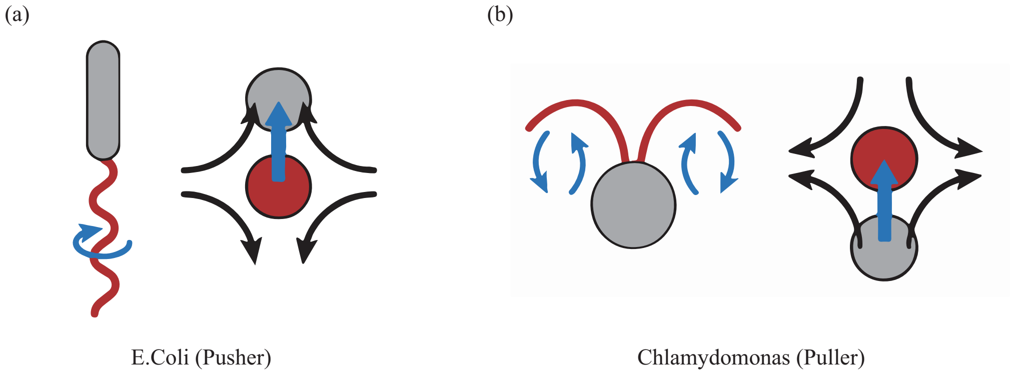
Figure 1. *Gallery of biological microswimmers under microscopes. (a): Volox [Prochnik et al., 2010]. (b): bull spermatozoa [Parrish, 2010]. (c) Paramecium [Deuterostome]. All three types of cells present flagella- or cilia-based swimming motion, and move in fluid environments by squirming their flagella.*

et al., 2020], and imaging [Yan et al., 2017].

Promising applications of artificial microswimmers require careful designs of these microscopic machines. One of the important metrics is the hydrodynamic efficiency. A popular hydrodynamic efficiency is proposed by Lighthill [1952] that measures the ratio between the 'useful work' and the total work, where the useful work is defined as the work needed to toll a rigid body of the microswimmer's shape at the given speed. The optimal slip velocity (the slip velocity that leads to the highest hydrodynamic efficiency) of spheroidal microswimmers can be solved semi-analytically [Leshansky et al., 2007]. A recently proposed numerical algorithm can solve the optimal slip velocity of any arbitrary axisymmetric shape [Guo et al., 2021]. This algorithm takes advantage of the linearity of the problem (Stokes equations) and formulates the optimization problem as a quadratic optimization problem, which can be solved directly in seconds on a laptop computer.

Besides optimizing individual microswimmers, it is also important to take their collective motions into consideration. Different types of microswimmers can cluster, swarm, and interact non-trivially with solid boundaries, which may further lead to interesting behaviors such as complex chaotic flows, enhanced particle velocities, efficient fluid mixing (see Saintillan and Shelley [2015] and the reference therein). Many of these behaviors can be explained by the reduced order models that classify microswimmers into three types, 'pusher', 'pullers', and 'neutrals'. The classification is based on the magnitude of *stresslet*, which is the symmetric first moment of the stresses exerted by the particle on the fluid [Kim and Karrila, 2013]. The microswimmers are called pushers, pullers, and neutrals if the magnitudes are less than, greater than, and equal to zero, respectively. Figure 2(a) shows an example pusher *E. Coli*. In this case, the flagellum (red) provides the force to push the cell body (grey) to move forward. Figure 2(b) shows an example puller *Chlamydomonas*, in which the flagella locate in front of the cell body and pull it to move forward. Note that biological microswimmers are often not hydrodynamically optimal [Guo et al., 2014], presumbly because they have other biological functions to serve such as feeding and chemical sensing. Artificial microswimmers, on the contrary, can be designed as hydrodynamically optimal. The goal of this paper is to design a framework to *predict* the classifications of optimal microswimmers, biological or artificial, for a given shape.

The conventional approach to classify optimal microswimmer with a given shape involves



E.Coli (Pusher)

Chlamydomonas (Puller)

Figure 2. *Cartoons of pusher microswimmer and puller microswimmer. The grey parts represent the cell body, and the red parts represent the activation force. Both swimmers swim upwards. Pusher microswimmer's activation force is located at the back of the cell body, and puller microswimmer's activation force is located at the front on the cell body. The black arrows indicate the induced fluid flows. Note that biological microswimmers are not necessarily hydrodyamically optimal. This cartoon only serves as an illustrative purpose for classic pusher and puller microswimmers.*

first optimizing the slip profile and then compute the stresslet induced by the microswimmer. Even though effective optimization algorithms exist, this approach is still limited by one's ability and comfort level at optimization as well as computation of the stresslets. A heuristic scalar metric is recently proposed by Guo et al. [2021] that can predict the classification using shape-information only. This heuristic metric is motivated by numerical observations and takes into account geometric features such as the tangent direction along the generating curve and the distance to the axis of symmetry. While the metric does a good job at predicting classifications (29 out of 32 microswimmers studied in the paper were correctly classified), the form of the metric is relatively arbitrary and provides little information about which geometric feature plays a more significant role in making the prediction.

In order to gain more insights on designing microswimmers and to leverage the ever-so-growing strength of data-driven sciences, we look into machine learning approaches. Machine learning and fluid dynamics research are tightly knitted historically. Researchers have applied various machine learning algorithms including proper orthogonal decomposition (POD), dynamic mode decomposition (DMD), genetic algorithms (GA) in fluid dynamics to extract flow feature, model flow dynamic, and optimize and control flows [Brunton et al., 2020]. In particular, artificial neural networks (ANN) presents promising applications in classifying canonical behaviors and dynamic regimes. At the same time, ANN has also greatly succeeded in classification tasks outside of the conventional fluid dynamics field. Some examples include hand-written digits recognition [LeCun et al., 1989a,b], facial recognition [Lawrence et al., 1997], medical diagnosis [Kononenko, 2001]. The power and flexibility of ANN stem from its modular structure. ANN consists of layers of artificial neurons, categorized into the input, hidden, and output layers. Each neuron in the ANN receives an input, processes it through an activation function, and produces an output [Nielsen, 2015]. The ANN is trained by processing the inputting examples and forming probability-weighted associations between each layer

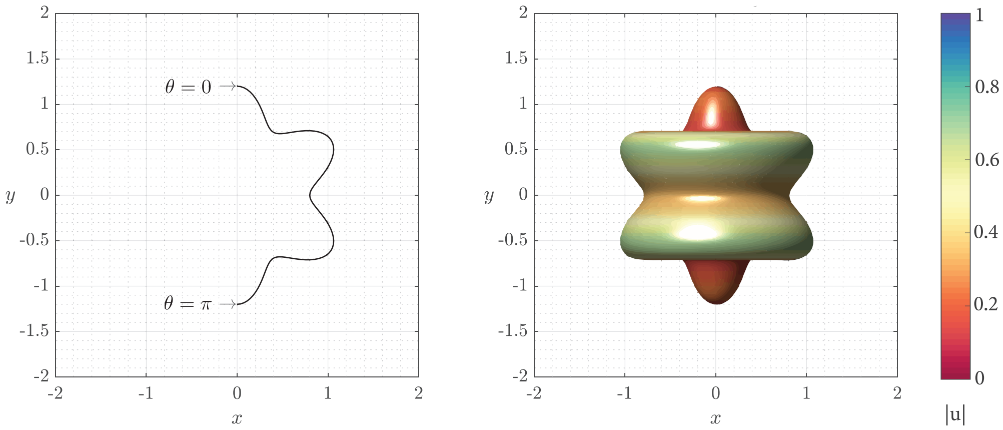


Figure 3. *Generate a microswimmer by rotating the generating curve. The figure on the left is a generating curve, and the figure on the right is its corresponding microswimmer. We obtain the axisymmetric microswimmer on the right by rotating the generating curve around the y-axis by 360 degrees. $|u|$ measures the optimal slip velocity on the surface of the microswimmer. A color towards red signifies smaller magnitude of the optimal slip, and a color towards blue represents a larger magnitude. Note that the shown microswimmer after optimization is neutral, and is generated using equation 2.1 with a = 0.15 and k = 6. Sample generated microswimmers classified as pusher and puller are given in 4. Figures are generated using the Matlab App Xlip [Guo, 2021].*

of neurons. It is worth noting that although naively adopting ANN structures in a black-box fashion can yield respectful accuracies for our classification problem, it is still unable to discern the more important geometric feature in the prediction process. To this end, we carefully design experiments with different input vectors to ANNs with the same structure and compare the classification results from both the accuracy and efficiency point of view. Our results help us to understand which geometric feature is more useful to this classification problem using ANN.

The paper is organized as follows: In section 2, we introduce the two shape families used in our experiment and our choice of the ANN structure; we present the classification results using four different ways to structure the input vector in section 3; we discuss our results and their implications in section 4.

2. Model and Methods.

In this project, we focus on studying axisymmetric microswimmers. By symmetry, any axisymmetric shape can be obtained by rotating a generating curve around the y-axis by 360 degrees. See Figure 3 for a visual representation.

In general, we consider two families of axisymmetric shapes: the wavy shapes

$$Z(\theta) = [1 + a\cos(k\theta)]e^{i(\pi/2-\theta)}, \tag{2.1}$$

and the spherical harmonic shapes

$$Z(\theta) = [1 + rY_n^m(\theta, 0)]e^{i(\pi/2 -\theta)}. \tag{2.2}$$

In both formulas, $\theta \in [0, \pi]$ is the polar angle. a and k are the amplitude and the wavenumber for the wavy shapes; $Y_n^m(\theta, 0)$ is the spherical harmonic of degree n and order m evaluated at colatitude θ and longitude 0, and r represents the perturbation size. In the remaining of the paper, we use x and y to denote the real and the imaginary parts of the complex number Z along the generating curve.

Note that the shapes corresponding to different shape parameters do not necessarily have the same north and south poles, which may add another level of difficulty in training the neural network. To eliminate such effects, we scale and translate the generating curve such that the poles are always a unit distance away from the origin along the y-axis. The training labels of the microswimmers are obtained following the algorithm documented in Guo et al. [2021]. Simply speaking, the algorithm first computes the optimal slip profile given a microswimmer's generating curve, then integrate the resulting stress tensor to classify the microswimmers. The geometric information along the generating curves and the computed labels are used to train the neural network, which will be discussed in the following paragraphs.

We use Tensorflow to construct a dense neural network. The neural network contains four layers, including one input layer, one output layer, and two hidden layers. The input layer takes in the geometric information of the generating curves, and the output layer presents the classification result. We discretize the generating curve into pieces of equal arclength by N points along the curve. Depending on the choice of models, we either construct the ANN's input vectors by using only the (x, y) coordinates of the points or by using both the (x, y) coordinates and the tangent directions of the discretization points, resulting in input vectors of lengths $2N$ and $4N$ respectively. Figure 4 is a visual representation of the structure of our neural network in which only the (x, y) coordinates are used.

While training the model, we split the data into 80% training and 20% test datasets while keeping the same ratio of pushers, pullers, and neutrals between the training set and the test set. We apply the adaptive moment estimation for optimization, sparse categorical cross-entropy as the loss function, relu as the activation function for both hidden layers, and softmax as the activation function for the output layer. We use the default parameters for all other variables in the function. To prevent the model from overfitting, we draw 30% data from the end of the randomly shuffled training set as the validation set prior to training. The validation set is used for the early stopping criterion, which monitors the validation losses with patience of 10. That is to say, we will halt the training process if the validation loss is greater than the current minimum loss for ten consecutive epochs. Note that the model is *never trained* using the validation set. The validation set is only *evaluated* at the end of each epoch as an indicator of potential overfitting.

We generate 515 wavy shapes using equation (2.1) and 513 spherical harmonic shapes using equation (2.2). Among the total of 1028 microswimmers, 430 are neutrals, 310 are pushers, and 288 are pullers. We use two hidden layers with 100 and 50 nodes respectively in our ANN structure. As will be shown in Section 3, this structure yields high accuracies for both training and test sets.

3. Results.

We start by training an ANN that takes the (x, y) coordinates of $N = 100$ discretization points along the generating curve as the input data. As a result, each input vector is 200-dimensional. We refer to this model as the *base model*. In addition to the base

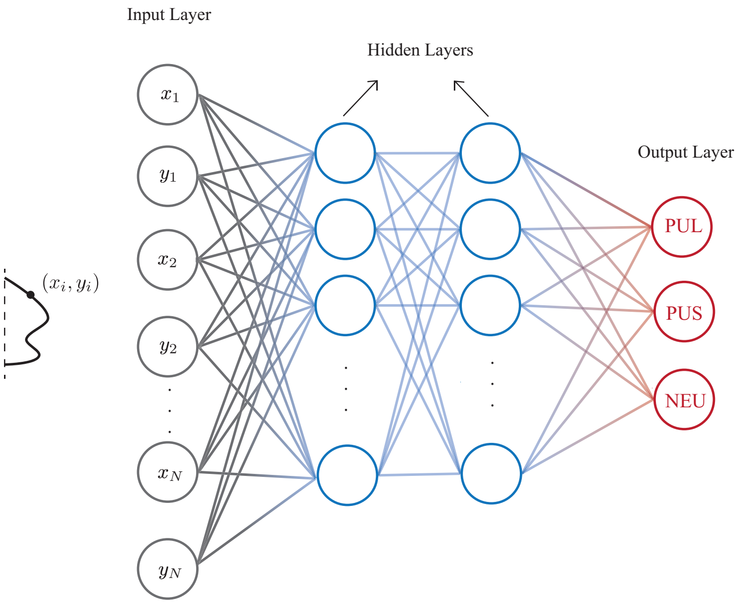


Figure 4. *Neural Network structure. A neural network has three types of layers: the input layer, the hidden layers, and the output layer. The input layer takes in the data then passes the data to the hidden layer for processing. Lastly, the model spits out the result from the output layer. The lines connecting each layer show the training process, while each layer feeds the data to its subsequent layer.*

model, we construct input vectors with both the (x, y) coordinates and the tangent directions at each point. The tangent vector at each discretization point is computed using a forward difference scheme. Note that appending the tangent vectors doubles the dimension of the input vectors. To make a fair comparison, we use a coarser discretization on the generating curve with 50 evenly distributed points, which yields the same dimension of the input vector as the base model. We refer to this case as the *test model*.

The accuracies and model losses of the base model are shown in Figure 5(a-b). Specifically, the training accuracy increases from 42.26% to 86.78% and the validation accuracy increases from 48.99% to 85.43% within 79 epochs, when the training ends due to the early stopping criteria based on the validation losses. The model losses decrease from about 1.1 to 0.4 for both the training set and the validation set. The test accuracy for the trained model is 83.98%.

Figure 5(c) shows that the training accuracy of the test model increases from 47.65% to 91.13% and the validation accuracy increases from 59.51% to 89.47% within 45 epochs. The test accuracy of this model is 88.35%, about 5% higher than the base model. The final model losses for the test model are also lower than those of the base model (Fig. 5(d)).

To further understand the results of the trained neural network for each microswimmer type, we generate a confusion matrix of the base model in Figure 6(a). The diagonal elements show the percentage of the microswimmers being correctly classified, and the off-diagonal ones demonstrate the mis-classified percentages. As we can see, the classifications of neutral microswimmers is significantly more accurate than that of pullers and pushers. Specifically, all

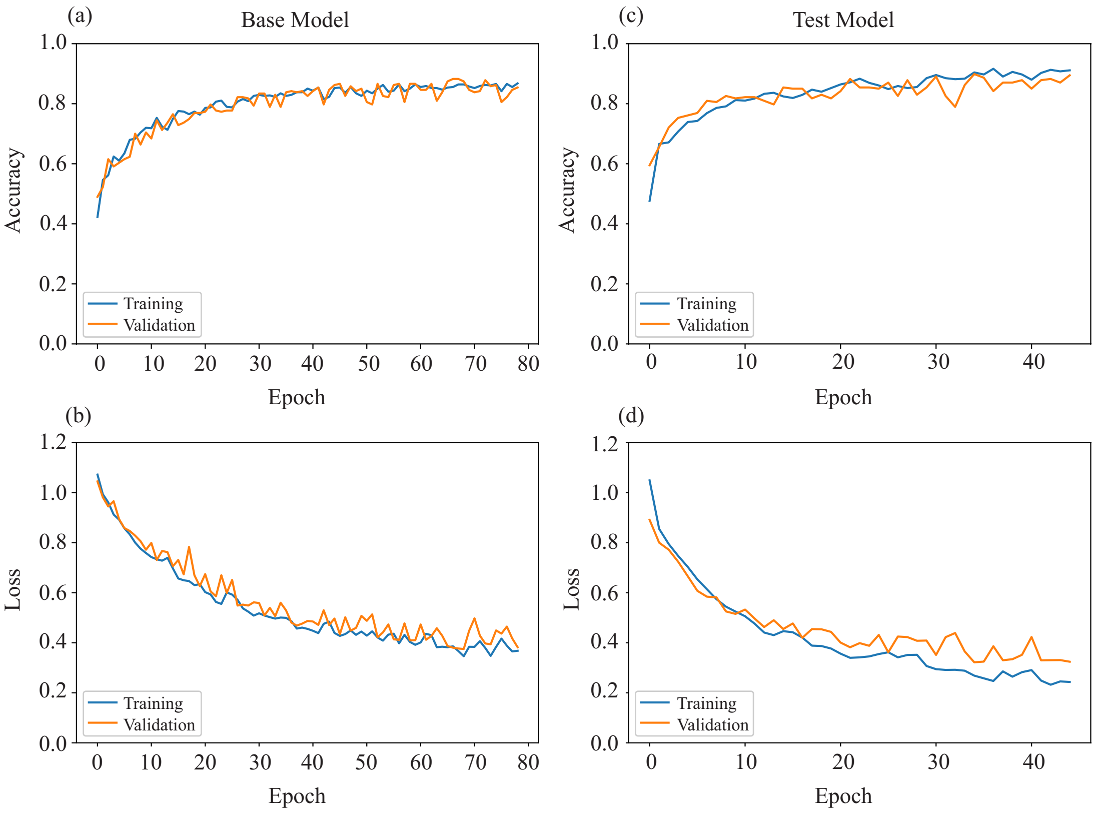


Figure 5. *Model Accuracy and loss for the base model and the test model. (a) and (b) Training, validation accuracy and model loss for model trained with 100 discretization points as input. (c) and (d) Model accuracy and loss for model trained with 50 discretization points and corresponding tangent vectors.*

neutral microswimmers are correctly classified, while the classification accuracies for pullers and pushers are 71% and 74% respectively. Among the 29% mis-classified pullers, 16% are classified as neutral and 14% as pullers.[1] Similarly, 15% of pushers are mis-classified as neutrals and 11% as pushers.

To reduce the high-accuracy bias introduced by the neutral microswimmers, we separate each type of microswimmers and train three more models with two types of microswimmers at a time. The confusion matrices of these three models are shown in Figure 6(b-d). The classification accuracies for pushers and pullers are significantly improved in all three models, and the advantages of neutral microswimmers become less prominent.

The confusion matrices of the test model are shown in Figure 6(e-h). The accuracies for pullers and pushers are improved from the base model and the accuracy for the neutral ones remains high. Additionally, including the tangent vectors significantly reduces the percentage

[1] up to round-off error.

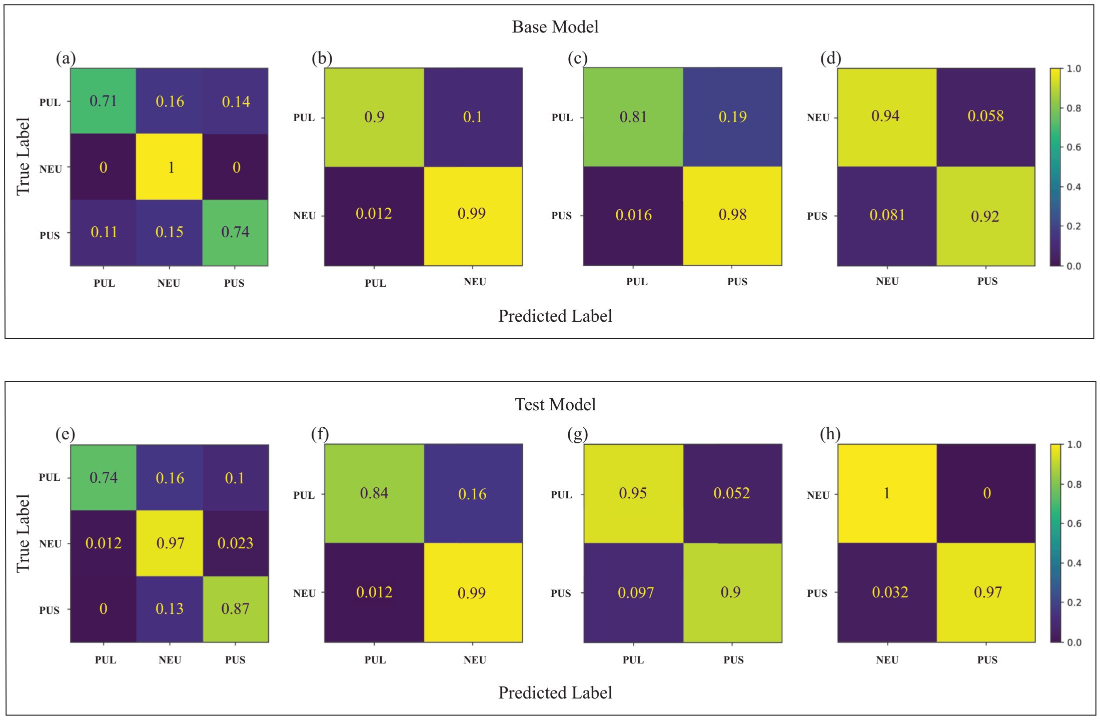


Figure 6. *Confusion matrices for the base model and the test model. (a) base model trained with pullers, neutrals, and pushers; (b)-(d) base model trained with only pullers and neutrals, pullers and pushers, or neutrals and pushers, respectively. (e) test model trained with pullers, neutrals, and pushers; (f)-(h) test model trained with only pullers and neutrals, pullers and pushers, or neutrals and pushers, respectively.*

of mis-classified pushers (pullers) as pullers (pushers).

To reduce random error, we perform 20 Monte Carlo simulations for the base and test models. The average test accuracies and the average number of epochs required for the model to converge are summarized in Table 1. While both models share the same dimension of input vectors, including the corresponding tangent vector indeed noticeably increases the model accuracies—average test accuracy increases from 82.06% (base model) to 87.99% (test model). Additionally, the average number of epochs required is reduced from 73 to 55, a roughly 25% computational save.

Table 1

Average Test Accuracy (TA) and the Number of Epochs (NOE) among 20 model trains for the four models.

TA, NOE	**Without tangent vector**	**With tangent vector**
100 points	82.06% 72.6 (base model)	85.32% 50.15
50 points	83.86% 85.05	87.99% 55.15 (test model)

In addition to the base and test models, we investigate two other models with input data containing 100 points with tangent vectors and 50 points without tangent vectors as shown in Table 1 for references. In general, when the number of discretization points are fixed, including the tangent vectors increases the test accuracies and reduces the number of epochs required; when the type of input is fixed (with or without tangent vector), halving the number of discretization points from 100 to 50 also increases the test accuracies yet requires more number of epochs to converge.

4. Discussions.

In our work, we constructed an artificial neural network using TensorFlow in Python to classify optimal microswimmers based solely on their shapes. We showed that, using a simple dense neural network with two hidden layers, the trained neural network achieved a high prediction accuracy consistently. The datasets are taken from two families of microswimmers that cover a wide range of shapes and the corresponding labels are computed using a numerical method documented in a recent paper [Guo et al., 2021].

Compared to the conventional approaches to classify optimal microswimmers that fluid dynamicists are familiar with, the trained neural network bypassed the optimization as well as the stress integration steps and required only the microswimmers shape information. For example, the input vectors of the neural network consist of solely the positions of the discretization points along the generating curve in the base model.

We further explored the shape features that can help the artificial neutral network to classify the microswimmers. Specifically, we examined whether providing the tangent vectors at each discretization point in addition to the position vectors as input of the artificial neural network can increase the classification accuracy. Keeping the hyperparameter combination used in the base model model (e.g., number of hidden layers, number of nodes in each hidden layer), we further trained three models with the same set of microswimmers yet provided the ANN with different input information. The trained models yielded different training, validation, and test accuracies as well as number of epochs needed for convergence.

Specifically, we showed that providing tangent vector information while keeping the number of discretization points increases the test accuracy (Table 1). On the other hand, halving the number of discretization points from 100 to 50 also increases the test accuracy. The test model (50 points with tangent vector information) boasts the highest test accuracy among all four models. In addition to the increase of accuracy, reducing the number of discretization points also increases the average epoch needed to convergence. These two observations combined likely means that the additional 50 discretization points on the generating curve do not add useful information for model training yet increases the potential of overfitting, as the number of model parameters increase with the dimension of the input vector.

It is interesting to note that the model trained with the highest dimension of input information (100 points with tangent vector case) needs the fewest number of epochs to convergence. This may allude one to believe that this model has the least computational cost. However, as the computational cost scales with the number of model parameters, this model actually requires about 15% higher computational cost than the base model and 51% higher than the test model, which has the least computational cost among the four models.

Both the base and test models are extremely accurate when classifying the neutral microswimmers, while the classification accuracies for pullers and pushers are similar but noticably lower than the classification accuracy for neutrals, as shown in Figure 6. Additionally, the mis-classified labels have a bias towards the neutrals. We believe that this model performance can be explained by how the true labels for the training data are generated. As discussed before, the pusher/puller labels are determined by the sign of the stress integral along the generating curve, and the neutral is a middle state that the stress integral is 0. Therefore, when the model prediction fails for a puller or a pusher, it is more likely that the microswimmer is predicted as a neutral as it lies between the puller and pusher.

Note that we employed an ANN with a relatively simple structure–two hidden layers and a set of fixed hyperparameters–that yielded a desirable model performance. Although fine-tuning the ANN structure may lead to even better model performance, it is not our intension of this paper to construct the neural network with the highest prediction accuracy. Rather, we used this simple-structured model to examine and found that providing tangent information can boost the model accuracy at reduced computational cost. Given our ANN's minimalist structure, we believe the conclusion made in the paper can be generalized to other ANN with more complex structures.

Several extensions to the work can be interesting. In this work, we only considered two families of axisymmetric microswimmers—the wavy shape and the spherical harmonic shape. However, our approach can be naturally generalized to include other axisymmetric families as well. The extension to non-axisymmetric shapes is theoretically manageable but significantly more challenging. One of the foreseeable challenges is that the result will likely be dependent on the surface discretization schemes, not to mention the very high dimensional input vectors. Moreover, we only focused on microswimmers in zero-Reynolds number Newtonian fluids in this work. It will be of great interest to apply similar method to classifications of optimal microswimmers in other complex fluids.

Changing hyperparameters such as the number of hidden layers, the number of nodes per layer, or even applying more advanced neural networks such as the convolutional neural network (CNN) will all influence the model accuracy. Additionally, our work only focused on one type of shape feature besides the position, the tangent vector, further works can explore other shape features of the microswimmers such as the area of the cross section and curvatures. It follows that another exciting improvement to make on our approach is to have the model to extract the most useful features automatically with advanced algorithms such as SINDy [Brunton et al., 2016]. Lastly, although we do not currently have access to experimental data, we will be excited to see future experiments related to this work.

Acknowledgments. We would like to acknowledge the *Laboratory of Geometry at Michigan* program, which provided this undergraduate research opportunity. All authors contributed in the semester-long research project, under the mentorships of Dr. Hanliang Guo and Ms. Kashvi Srivastava. NS programmed a major part of the initial Python codes. NJ and XQ refined the codes, analyzed the results, and wrote the paper with the help from HG. We would like to thank reviewers for taking the time and effort necessary to review the manuscript. We sincerely appreciate all valuable comments and suggestions, which helped us to improve the quality of the manuscript. Codes will be available by contacting the corresponding authors.

Appendix. In addition to the neutral example shown in Figure 3, we present two additional microswimmers in Figure 7, where one of which is classified as pusher after optimization and the other is classified as puller after optimization.

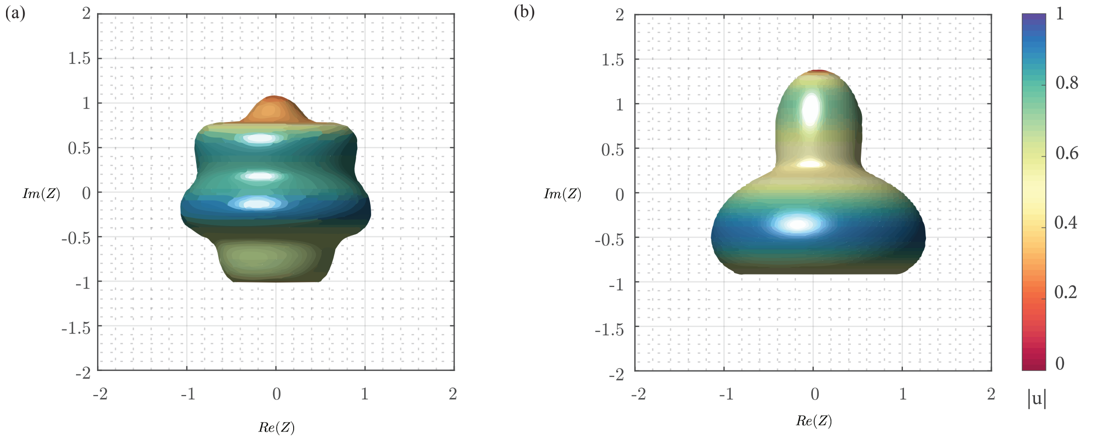


Figure 7. *Pusher and puller microswimmers generated by equation 2.1. (a) shows a pusher microswimmer generated by 2.1, with a = 0.1, k = -7. (b) shows a puller microswimmer generated by equation 2.1, with a = 0.4, k = 3. Figures are generated using the Matlab App Xlip [Guo, 2021].*

References.

J. L. Anderson. Colloid transport by interfacial forces. *Annual review of fluid mechanics*, 21 (1):61–99, 1989.

C. Brennen and H. Winet. Fluid mechanics of propulsion by cilia and flagella. *Annual Review of Fluid Mechanics*, 9(1):339–398, 1977.

S. L. Brunton, J. L. Proctor, and J. N. Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the national academy of sciences*, 113(15):3932–3937, 2016.

S. L. Brunton, B. R. Noack, and P. Koumoutsakos. Machine learning for fluid mechanics. *Annual Review of Fluid Mechanics*, 52:477–508, 2020.

Z. Budrikis. Sperm swimming is more complicated than thought. *Nature Reviews Physics*, 2 (9):461–461, 2020.

Deuterostome. The ciliated protozoan paramecium caudatum. https://en.wikipedia.org/wiki/Paramecium_caudatum#/media/File:Paramecium_caudatum_Ehrenberg,_1833.jpg.

R. Dreyfus, J. Baudry, M. L. Roper, M. Fermigier, H. A. Stone, and J. Bibette. Microscopic artificial swimmers. *Nature*, 437(7060):862–865, 2005.

E. A. Gaffney, H. Gadêlha, D. J. Smith, J. R. Blake, and J. C. Kirkman-Brown. Mammalian sperm motility: observation and theory. *Annual Review of Fluid Mechanics*, 43:501–528, 2011.

R. Golestanian, T. Liverpool, and A. Ajdari. Designing phoretic micro-and nano-swimmers. *New Journal of Physics*, 9(5):126, 2007.

H. Guo. Xlip. https://www.mathworks.com/matlabcentral/fileexchange/93335-xlip, 2021.

H. Guo, J. Nawroth, Y. Ding, and E. Kanso. Cilia beating patterns are not hydrodynamically optimal. *Physics of Fluids*, 26(9):091901, 2014.

H. Guo, H. Zhu, R. Liu, M. Bonnet, and S. Veerapaneni. Optimal slip velocities of microswimmers with arbitrary axisymmetric shapes. *Journal of Fluid Mechanics*, 910, 2021.

J. R. Howse, R. A. Jones, A. J. Ryan, T. Gough, R. Vafabakhsh, and R. Golestanian. Self-motile colloidal particles: from directed propulsion to random walk. *Physical review letters*, 99(4):048102, 2007.

S. Kim and S. J. Karrila. *Microhydrodynamics: principles and selected applications.* Courier Corporation, 2013.

S. Klumpp and R. Lipowsky. Cooperative cargo transport by several molecular motors. *Proceedings of the National Academy of Sciences*, 102(48):17284–17289, 2005.

I. Kononenko. Machine learning for medical diagnosis: history, state of the art and perspective. *Artificial Intelligence in medicine*, 23(1):89–109, 2001.

E. Lauga and T. R. Powers. The hydrodynamics of swimming microorganisms. *Reports on progress in physics*, 72(9):096601, 2009.

S. Lawrence, C. L. Giles, A. C. Tsoi, and A. D. Back. Face recognition: A convolutional neural-network approach. *IEEE transactions on neural networks*, 8(1):98–113, 1997.

Y. LeCun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard, and L. Jackel. Handwritten digit recognition with a back-propagation network. *Advances in neural information processing systems*, 2, 1989a.

Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989b.

A. M. Leshansky, O. Kenneth, O. Gat, and J. E. Avron. A frictionless microswimmer. *New Journal of Physics*, 9(5):145, 2007.

M. Lighthill. On the squirming motion of nearly spherical deformable bodies through liquids at very small reynolds numbers. *Communications on pure and applied mathematics*, 5(2): 109–118, 1952.

M. Medina-Sánchez, L. Schwarz, A. K. Meyer, F. Hebenstreit, and O. G. Schmidt. Cellular cargo delivery: Toward assisted fertilization by sperm-carrying micromotors. *Nano letters*, 16(1):555–561, 2016.

M. A. Nielsen. *Neural networks and deep learning*, volume 25. Determination press San Francisco, CA, 2015.

J. Parrish. Animal science 434 (course material). http://www.ansci.wisc.edu/jjp1/ansci_repro/lab/procedures/sperm/bull_morp/bull_morphology.html, 2010. Accessed: 2022-02-14.

S. E. Prochnik, J. Umen, A. M. Nedelcu, A. Hallmann, S. M. Miller, I. Nishii, P. Ferris, A. Kuo, T. Mitros, L. K. Fritz-Laylin, et al. Genomic analysis of organismal complexity in the multicellular green alga volvox carteri. *Science*, 329(5988):223–226, 2010.

F. Qiu, S. Fujita, R. Mhanna, L. Zhang, B. R. Simona, and B. J. Nelson. Magnetic helical microswimmers functionalized with lipoplexes for targeted gene delivery. *Advanced Functional Materials*, 25(11):1666–1671, 2015.

D. Saintillan and M. J. Shelley. Theory of active suspensions. In *Complex Fluids in biological systems*, pages 319–355. Springer, 2015.

M. A. Sleigh. *Protozoa and other protists.* Cambridge University Press, 1991.

Z. Sun, P. F. Popp, C. Loderer, and A. Revilla-Guarinos. Genetically engineered bacterial biohybrid microswimmers for sensing applications. *Sensors*, 20(1):180, 2020.

S. L. Tamm. Ciliary motion in paramecium: a scanning electron microscope study. *The Journal of cell biology*, 55(1):250, 1972.

G. I. Taylor. Analysis of the swimming of microscopic organisms. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 209(1099):447–461, 1951.

X. Yan, Q. Zhou, M. Vincent, Y. Deng, J. Yu, J. Xu, T. Xu, T. Tang, L. Bian, Y.-X. J. Wang, et al. Multifunctional biohybrid magnetite microrobots for imaging-guided therapy. *Science Robotics*, 2(12), 2017.