Contents lists available at ScienceDirect

Theoretical & Applied Mechanics Letters

Letter

Classifying wakes produced by self-propelled fish-like swimmers using neural networks

Binglin Li[a,b], Xiang Zhang[a,b], Xing Zhang[a,b,*]

[a] *The State Key Laboratory of Nonlinear Mechanics, Institute of Mechanics, Chinese Academy of Sciences, Beijing 100190, China*
[b] *School of Engineering Science, University of Chinese Academy of Sciences, Beijing 100049, China*

HIGHLIGHTS

- Several neural networks are trained to classify wakes produced by self-propelled fish- like swimmers.
- The wakes are generated by using high-fidelity numerical simulations and local measurements of flow variables are used in training the neural networks.
- A remarkably high accuracy in wake classification can be achieved if the neural networks are trained by using the combination of velocity components and vorticity.

ABSTRACT

We consider the classification of wake structures produced by self-propelled fish-like swimmers based on local measurements of flow variables. This problem is inspired by the extraordinary capability of animal swimmers in perceiving their hydrodynamic environments under dark condition. We train different neural networks to classify wake structures by using the streamwise velocity component, the crosswise velocity component, the vorticity and the combination of three flow variables, respectively. It is found that the neural networks trained using the two velocity components perform well in identifying the wake types, whereas the neural network trained using the vorticity suffers from a high rate of misclassification. When the neural network is trained using the combination of all three flow variables, a remarkably high accuracy in wake classification can be achieved. The results of this study can be helpful to the design of flow sensory systems in robotic underwater vehicles.

©2020 The Authors. Published by Elsevier Ltd on behalf of The Chinese Society of Theoretical and Applied Mechanics. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

The long-lasting wake structures left behind active swimmers and inanimate objects contain useful hydrodynamic cues about the sources that produce them. Such information can be extracted and exploited by other swimmers to accomplish a variety of tasks [1]. For example, with the help of the lateral line system, even blind fish can school [2] and avoid obstacles in the water [3, 4]. By using the whiskers, harbor seals can track the preys by detecting the wake signals [5]. This remarkable capability in animal swimmers also inspired the development of artificial flow sensory systems [6–10].

The wake structures behind a bluff body and an active swimmer are often characterized by the von Karman and reverse von Karman vortex streets, respectively. Based on the number of vortices that are shed per flapping cycle, the wakes produced by an active swimmer can also be further classified into different sub-types. The "2S" and "2P" wakes are the two most commonly seen sub-types. Here "2S" refers to the situation in which two vortices of opposite sign are shed per cycle; whereas "2P" refers to the situation in which two pairs of vortices are shed per cycle. Besides, some higher-order exotic wakes, with up to 16 vortices shed per cycle have also been observed [11]. The types of wake

* Corresponding author.
E-mail address: zhangx@lnm.imech.ac.cn (X. Zhang).

produced by the swimmers are intimately associated with the swimming characteristics.

For an animal (or artificial) swimmer, the capability of distinguishing between different wake types lays the groundwork for motion planning and navigation. However, this seemingly easy task is actually an inverse problem which poses great challenges to the flow sensory system. The reason is that the data of measurements are only available at discrete locations within a small sub-domain of the entire flow field. This is in contrast with the situation faced by an outside observer to whom the globally spacial-temporal flow patterns are accessible.

With the rapid progress of machine learning, fundamental methodologies in machine learning have been used to handle the tasks of fluid mechanics [12]. Recently, some pioneering works have been conducted to solve this inverse problem by using machine learning algorithms. Wang and Hemati [13] generated "2S" and "2P" wakes with the point vortices by using the ideal-flow model. The wake-induced velocities were measured on the surface of a fish-like body that was immersed in the vortical wakes. The K-Nearest Neighbors (KNN) algorithm was then used to classify the wake types. Colvert et al. [14] generated the wakes behind a *tethered* rigid pitching foil placed in an otherwise steady stream by solving the Navier-Stokes equations numerically. The neural networks were then trained to distinguish between three distinct wake types from the time histories of vorticity measured at discrete locations. Alsalman et al. [15] used the same wakes as those generated by Colvert et al. [14], and went one step further to investigate how the accuracy of wake classification was affected by the types of sensors (i.e., sensors measuring different flow variables). They also trained the neural networks to classify wake types from the data of "one time" measurements at a spatially-distributed array of sensors. Motivated by the aforementioned studies, in this work, we train the neural networks to classify wakes behind a *self-propelled* fish-like swimmer. Similar to the studies of Refs. [14] and [15], in the present study the wakes are also produced by high-fidelity numerical simulations. However, the wakes considered here and those in the two aforementioned works are very dissimilar, in terms of the spatial-temporal patterns. Clearly, the wakes considered here bear a much closer resemblance to those produced by an animal (or an artificial) swimmer under free-swimming condition.

We consider continuous and intermittent self-propelled swimming of a flexible fish-like swimmer which is driven by fluid-structure interaction (FSI) [16, 17]. The localised actuation consists of the coupled heaving and pitching motions that are imposed at the leading edge (see Fig. 1). For the continuous swimming, the driving motions are:

$$y_{s=0}(t) = A\cos(2\pi f t), \tag{1}$$

$$\theta_{s=0}(t) = A_\theta \cos(2\pi f t - \phi), \tag{2}$$

where $y_{s=0}$ and $\theta_{s=0}$ are the instantaneous heaving and pitching displacement of the leading edge. A and A_θ are the heaving and pitching amplitudes, respectively. f is the actuation frequency, and ϕ is the phase lag between the heaving and pitching motions. In this study, the phase lag is set to 90 degrees. For the intermittent swimming, certain time duration of passive coasting is interspersed between two half-periods of active bursting (flapping) [16]. The degree of intermittency is characterised by the duty cycle (*DC*), which is defined as the ratio of the time duration of active flapping to the time duration of one full cycle of burst-and-coast locomotion.

The incompressible Naiver-Stokes equations and nonlinear dynamics equations constitute the governing equations of the system. These equations can be written in a dimensionless form as:

$$\frac{\partial \boldsymbol{u}}{\partial t} + \nabla \cdot (\boldsymbol{u}\boldsymbol{u}) = -\nabla p + \frac{1}{Re_f}\nabla^2 \boldsymbol{u} + \boldsymbol{f}, \tag{3}$$

$$\nabla \cdot \boldsymbol{u} = 0, \tag{4}$$

$$\beta(s)\frac{\partial^2 \boldsymbol{X}}{\partial t^2} - \frac{\partial}{\partial s}\left(\xi(s)\frac{\partial \boldsymbol{X}}{\partial s}\right) + \frac{\partial^2}{\partial^2 s}\left(\gamma(s)\frac{\partial^2 \boldsymbol{X}}{\partial s^2}\right) = -\boldsymbol{F}, \tag{5}$$

$$\frac{\partial \boldsymbol{X}}{\partial \mathrm{s}} \cdot \frac{\partial \boldsymbol{X}}{\partial \mathrm{s}} = 1, \tag{6}$$

where $\boldsymbol{u}$ is the fluid velocity, p is the pressure, $\boldsymbol{X}$ is the position vector for the Lagrangian (marker) point on the flexible swimmer and s is the Lagrangian coordinate along the arc. $\boldsymbol{f}$ and $\boldsymbol{F}$ are the dimensionless Eulerian and Lagrangian forces which represent the interaction between the flow and the structure. We select the chord length L and $U_{ref} = Lf$ as the reference length and reference velocity for scaling the system. The four dimensionless quantities that arise in the governing equations are: flapping Reynolds number Re_f, mass ratio β, dimensionless tension ξ and dimensionless rigidity γ. The definitions of these four dimensionless quantities as:

$$Re_f = \frac{U_{ref}L}{\nu}, \beta = \frac{\rho_s \delta}{\rho_f L}, \xi = \frac{T}{\rho_f (U_{ref})^2 L}, \gamma = \frac{B}{\rho_f (U_{ref})^2 L^3}, \tag{7}$$

where ν is the kinematic viscosity of the fluid, ρ_f and ρ_s are the densities of the fluid and the filament, respectively. δ is the
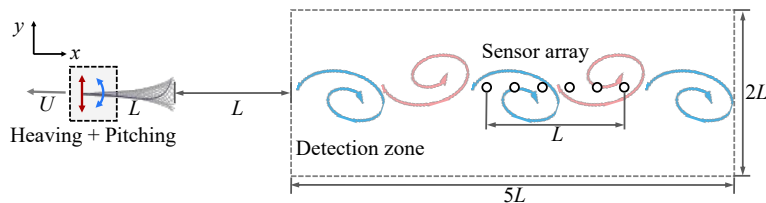
Fig. 1. Schematic diagram of the wake classification problem: an array of sensors is placed in the wake behind a self-propelled fish-like swimmer.

thickness of the filament. T and B are the tension and bending rigidity of the structure, respectively. Here ξ is a Lagrangian multiplier which is determined to enforce the inextensibility condition (Eq. (6)) [18]. β and γ are two exponential functions with respect to s [16, 17]. In this study, Re_f is fixed to 6000, which is the same as the highest Reynolds number considered in Ref. [16]. The Reynolds numbers based on the swimming velocity lie in the range of 3000 to 4000. The coefficients in the two aforementioned exponential functions are tunable.

This FSI problem is simulated by using a loosely-coupled solver in which the flow equations and structure equations are advanced sequentially in time. In solving the flow equations, the immersed boundary method is used to impose the no-slip condition on the surface of the swimmer. In solving the structure equations, a finite difference method is used for the spatial discretization and a three-time-level scheme is used for the temporal advancement. This FSI solver has been extensively validated in the earlier works of our group [16–20]. After solving for the velocity field $\boldsymbol{u}(\boldsymbol{x}, t)$, the scalar vorticity field is computed by $\omega_z = (\nabla \times \boldsymbol{u}) \cdot \boldsymbol{k}$, where $\boldsymbol{k}$ is the unit normal vector which points towards the "outside" of the two-dimensional plane. Here the second-order central-difference scheme is used to discretize the curl operator ($\nabla \times$).

We conducted nine cases of simulation by tuning some control parameters. These parameters include: coefficients of the exponential functions that govern the distributions of β and γ, dimensionless heaving amplitude A/L, pitching amplitude A_θ, and duty cycle DC. Among all simulation cases, three distinct wake types, namely, "2S", "2P", "2P+2S", are identified. The values of the control parameters and wake type corresponding to each simulation case are listed in Table 1. It is seen that each wake type can be observed in three different simulation cases. The representative vortex structures for the "2S", "2P" and "2P+2S" wakes are shown in Fig. 2.

The wake detection zone is of rectangular shape and has the dimensions of $5L \times 2L$. It is symmetric with respect to the horizontal centerline and is placed downstream of the swimmer with a distance of L from the trailing edge. Unlike the previous studies in Refs. [14] and [15], where the time histories of flow variables were used for wake classification, here we only use the data of "one time" measurements. The reasons for this choice in the present study are twofold. First, in the wake produced by a self-propelled swimmer, "decaying with time" is the salient feature in the time histories of flow variables, while temporal periodicity does not show up in the signals. Second, the wake classification based on the data of "one time" measurements implies that there is no time delay. This is certainly more favorable in the design of flow sensory systems in animal (or artificial) swimmers. For each simulation case listed in Table 1, the data sampling frequency is 20 times per flapping cycle, and the duration of sampling is 5 flapping cycles. As a result, the data of 100 snapshots are collected. For each snapshot, the vorticity (ω_z) and two velocity components (u_x and u_y) at the grid points of the computational mesh are saved. Before later usage in training and testing the neural networks, all flow variables are preprocessed by scaling with the root-mean-square (RMS) values of the collected data.

A spatially distributed array of senors with the length of L is then used to measure the flow variables (see Fig. 1). The sensor array is in align with the x-axis and the number of sensors is set to 6. For each snapshot, the measurements are conducted 861 times, by placing the first (leftest) sensor on the vertices of a 41×21 lattice which occupies part of the detection zone. The vertices of the lattice are uniformly distributed and the interspace between two neighboring vertices is $0.1L$. Thus, for each simulation case, $861 \times 20 \times 5 = 86100$ measurements are conducted by the sensor array. Since the positions of sensors are not necessarily in coincidence with the positions of grid points, bilinear interpolations are performed to obtain the values of flow variables at the positions of sensors.

We train fully connected neural networks (FCNs) to classify the wake structures. The hyperparameters of the neural networks are listed in Table 2. Specifically, the FCNs consist of 3 hidden layers and 1 output layer and each hidden layer has 32 units (neurons). We test the effects of the numbers of hidden layers and units on the performance of the networks. It is found that the networks exhibit little variation in performance if the number of hidden layers or the number of units is further increased. The output of each unit is the input of ReLU activation function. The output layer is a softmax function which can map the output of the last hidden layer to a probability distribution vector. This vector has three components which indicate the probabil-

Table 1 The control parameters of the fish-like swimmers used to produce the wakes.

Wake type	Case ID	$\beta(s)$	$\gamma(s)$	A/L	A_θ	DC
	A1	$0.05\mathrm{e}^{-3s^2}$	$0.10\mathrm{e}^{-9s^2}$	0.010	8°	1
2S	A2	$0.10\mathrm{e}^{-3s^2}$	$0.30\mathrm{e}^{-9s^2}$	0.005	5°	1
	A3	$0.15\mathrm{e}^{-3s^2}$	$0.20\mathrm{e}^{-9s^2}$	0.009	6°	1
	B1	$0.15\mathrm{e}^{-3s^2}$	$0.30\mathrm{e}^{-9s^2}$	0.005	6°	0.9
2P	B2	$0.15\mathrm{e}^{-3s^2}$	$0.30\mathrm{e}^{-9s^2}$	0.007	6°	0.9
	B3	$0.15\mathrm{e}^{-3s^2}$	$0.30\mathrm{e}^{-9s^2}$	0.009	6°	0.9
	C1	$0.10\mathrm{e}^{-3s^2}$	$0.20\mathrm{e}^{-9s^2}$	0.005	8°	0.6
2P+2S	C2	$0.10\mathrm{e}^{-3s^2}$	$0.25\mathrm{e}^{-9s^2}$	0.005	7°	0.6
	C3	$0.10\mathrm{e}^{-3s^2}$	$0.30\mathrm{e}^{-9s^2}$	0.005	6°	0.6
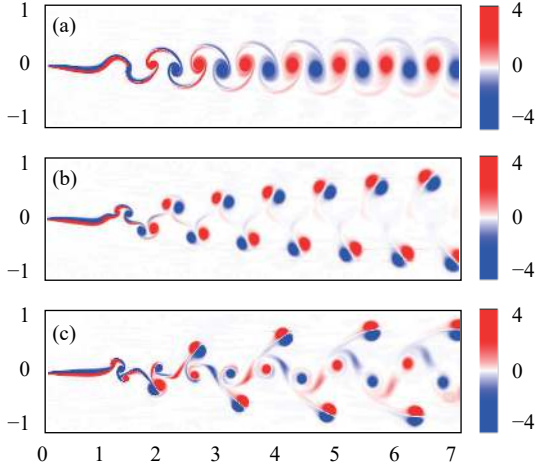
Fig. 2. Representative vortex structures for the three types of wake: **a** "2S" type; **b** "2P" type; **c** "2P+2S" type. The vortex structures are visualized using the contours of dimensionless vorticity (vorticity scaled by f).

Table 2 The hyperparameters of the FCNs used in the wake classification.

hidden layer	activation	unit	optimizer	batch size	epoch
3	relu	32	adam	256	100

Table 3 Different combinations of simulation cases in the three-fold cross-validation for training and testing the neural networks.

	Data set 1	Data set 2	Data set 3
Training data	A1, B1, C1	A1, B1, C1	A2, B2, C2
	A2, B2, C2	A3, B3, C3	A3, B3, C3
Testing data	A3, B3, C3	A2, B2, C2	A1, B1, C1

ity of the three predicted wake types. Thus, the values of the three components lie in the range of 0 to 1. The loss function of the FCNs is the cross-entropy function which can be written as:

$$L(\mathbf{p},\mathbf{q}) = -\sum_{i=1}^{n} p_i(x) \log q_i(x), \tag{8}$$

where $p_i(x)$ is the true probability distribution vector by using one-hot encoding that identifies the wake type and $q_i(x)$ is the predicted distribution vector obtained from the FCNs. The training of the FCNs is performed by minimizing the loss function to approach the real distribution of the target. In this work, the neural networks are implemented using the open source machine learning package TensorFlow.

The FCNs are trained and tested by using the approach of three-fold cross-validation. More specifically, for each wake type, the measurement data obtained from two simulation cases are used for training and the data from the remaining case is reserved for testing. Altogether, three distinct data sets (Data set 1, Data set 2 and Data set 3) are created by different combinations of the simulation cases for training and testing (see Table 3).

Four types of FCNs are trained using the measurements of different flow variables as the input data. The first three FCNs are trained by using u_x, u_y and ω_z, respectively, while the fourth FCN is trained by using the combination of all three flow variables. In Fig. 3, we show the progress to convergence in training the FCN using all three flow variables as the input. It is seen that 100 epochs are sufficient for the network to reach its best performance, since both the loss function and accuracy exhibit little variation with the increase of epoch number. Similar behaviors are also observed in the training of other three FCNs (the progress to convergence for them are not shown here for brevity).

We then evaluate the performances of the four FCNs in classifying the wake structures. The performances of the FCNs are visualized using the normalized confusion matrices. The definition of confusion matrix is as follows. Each row of the confusion matrix represents the instances in an actual class, while each column represents the instances in a predicted class. Thus, the diagonal and off-diagonal entries of the matrix show the instances of correct classification and misclassification, respectively. A normalized confusion matrix is constructed by normalizing the entries of each row of the confusion matrix using the total instances in the actual class (i.e., the summation of all entries in that row). The normalized confusion matrices for the three FCNs trained by using u_x, u_y and ω_z are shown in Fig. 4. The accuracies shown in the normalized confusion matrices are the av-
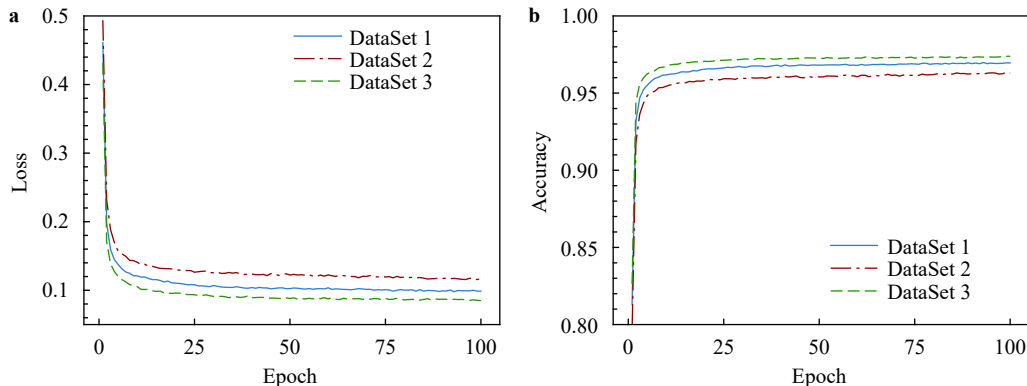
Fig. 3. The values of loss function and network accuracy versus the number of training epochs in training the FCN using the combination of all three flow variables: **a** loss function; **b** network accuracy.
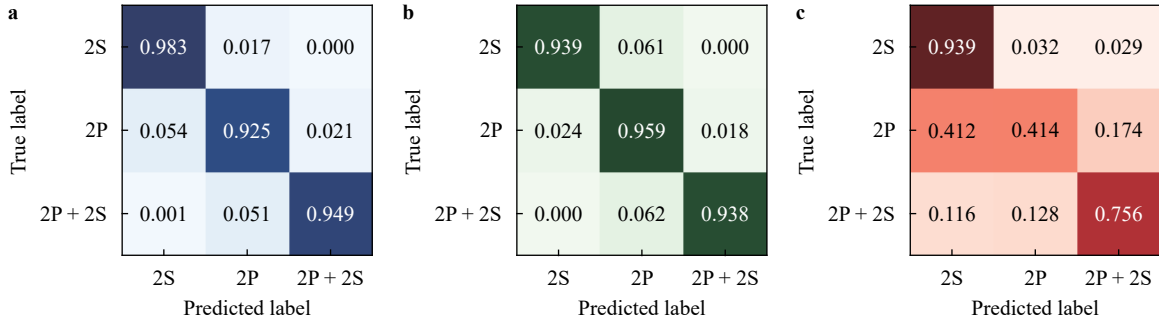
Fig. 4. Normalized confusion matrices for the three FCNs trained using **a** u_x; **b** u_y; **c** ω_z.
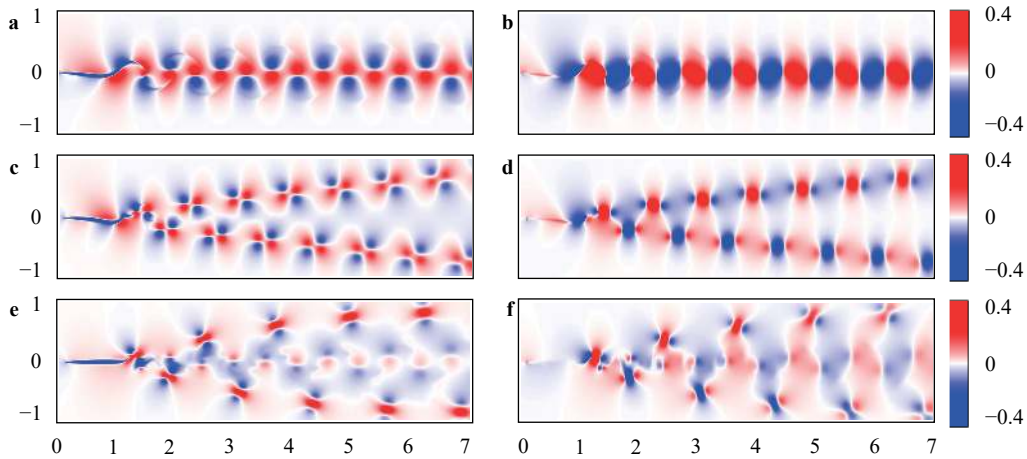

Fig. 5. Contours of the two velocity components in the "2S", "2P" and "2P+2S" wakes. **a** and **b**: (dimensionless) streamwise and crosswise velocity components for the "2S" wake; **c** and **d**: (dimensionless) streamwise and crosswise velocity components for the "2P" wake; **e** and **f**: (dimensionless) streamwise and crosswise velocity components for the "2P+2S" wake.

eraged results obtained by using three different data sets. It is seen that the FCN trained using u_x performs very well. For all three wake types, the percentage of correct identification are higher than 90%. For the "2S" wake, the rate of correct identification can even reach 98.3%. The FCN trained by using u_y performs comparably with the FCN trained using u_x. For these two FCNs, the highest rate of misclassification is around 6% to 7%. The performance of the FCN trained using ω_z is much worse. In identifying the "2S", "2P" and "2P+2S" wakes, the percentages of misclassification are 6.1%, 58.6%, and 24.4%, respectively.
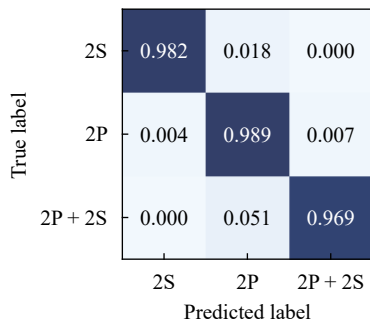

Fig. 6. Normalized confusion matrix for the FCN trained using the combination of u_x, u_y and ω_z.

The poor performance of the FCN trained using ω_z (especially in identifying the "2P" wakes) can be explained as follows. By examining the vorticity contours shown in Fig. 2, it is seen that for the "2P" wake, in a large portion of the wake detection zone, the vorticity magnitude is very small (close to zero). This makes it very difficult for the FCN to extract the intrinsic feature of the wake from local measurements of ω_z. In contrast with the contours of ω_z, the contours of u_x and u_y look very different (see Fig. 5). It is seen that the contours with high magnitudes of u_x and u_y are less concentratedly distributed in the wake detection zone. This makes it easy for the FCNs to uncover the salient feature of each wake type.

Figure 6 shows the normalized confusion matrix for the FCN trained by using the combination of all three flow variables (i.e., u_x, u_y and ω_z) as the input data. It is seen that remarkably high accuracies in identifying all three wake types have been achieved. The highest rate of misclassification is reduced to 3% (which is observed in identifying the "2P+2S" wakes). The superior performance of the FCN trained using all three variables can be explained as follows. The spatial distributions of the three flow variables measured by the sensor array can complement each other and result in reduction of confusion among the three wake types.

In this work, we trained four types of FCNs to distinguish among three types of wakes produced by a self-propelled fish-

like swimmer. We used the "one time" measurements of some flow variables (including the two velocity components and the vorticity) as the input data in training the neural networks. It turned out that the FCN trained using streamwise or crosswise component of velocity performs much better than the FCN trained using vorticity. The FCN trained by using the combination of all three flow variables outperforms the other three FCNs and achieves a remarkably high accuracy in wake classification.

There are a number of interesting avenues for future research. First, the sensitivity of the performance of the neural networks to the number and arrangement of sensors needs to be studied. Second, the neural networks can be used to discern more useful information of the swimmer from the wake, such as the body size, species, swimming speed and swimming style. This is achievable only if a solid link between such information and the localized flow measurements can be established. Third, one limit of the current study is the assumption that the placement of sensors in the detection zone does not disturb the flow. The two-way interactions between the sensors and the flow need to be considered.

Acknowledgement

References

[1]   G. R. Spedding, Wake signature detection, Annual Review of Fluid Mechanics 46 (2014) 273–302.

[2]   T. J. Pitcher, B. L. Partridge, C. S. Wardle, A blind fish can school, Science 194 (1976) 963–965.

[3]   S. P. Windsor, S. E. Norris, S. M. Cameron, et al., The flow fields involved in hydrodynamic imaging by blind mexican cave fish (astyanax fasciatus). part i: open water and heading towards a wall, Journal of Experimental Biology 213 (2010) 3819–3831.

[4]   S. P. Windsor, S. E. Norris, S. M. Cameron, et al., The flow fields involved in hydrodynamic imaging by blind mexican cave fish (astyanax fasciatus). part ii: gliding parallel to a wall, Journal of Experimental Biology 213 (2010) 3832–3842.

[5]   G. Dehnhardt, B. Mauck, W. Hanke, et al., Hydrodynamic trail-following in harbor seals (phoca vitulina), Science 293 (2001) 102–104.

[6]   M. S. Triantafyllou, G. D. Weymouth, J. M. Miao, Biomimetic survival hydrodynamics and flow sensing, Annual Review of Fluid Mechanics 48 (2016) 1–24.

[7]   A. Gao, M. Triantafyllou, Bio-inspired pressure sensing for active yaw control of underwater vehicles, 2012 Oceans, Hampton Roads, VA (2012) 1–7.

[8]   A.M.K. Dagamseh, R.J. Wiegerink, T.S.J. Lammerink, et al., Towards a high-resolution flow camera using artificial hair sensor arrays for flow pattern observations, Bioinspiration & Biomimetics 7 (2012) 046009.

[9]   J. Dusek, A.G.P. Kottapalli, M.E. Woo, et al., Development and testing of bio-inspired microelectromechanical pressure sensor arrays for increased situational awareness for marine vehicles, Smart Materials and Structures 22 (2012) 014002.

[10] P.V.Y. Alvarado, V. Subramaniam, M. Triantafyllou, Design of a bio-inspired whisker sensor for underwater applications, Sensors, 2012 IEEE, Taipei (2012) 1–4.

[11] T. Schnipper, A. Andersen, T. Bohr, Vortex wakes of a flapping foil, Journal of Fluid Mechanics 633 (2009) 411–423.

[12] S. L. Brunton, B. R. Noack, P. Koumoutsakos, Machine learning for fluid mechanics, Annual Review of Fluid Mechanics 52 (2020) 477–508.

[13] M.Y. Wang, M.S. Hemati, Detecting exotic wakes with hydrodynamic sensors, Theoretical and Computational Fluid Dynamics 33 (2019) 235–254.

[14] B. Colvert, M. Alsalman, E. Kanso, Classifying vortex wakes using neural networks, Bioinspiration & Biomimetics 13 (2018) 025003.

[15] M. Alsalman, B. Colvert, E. Kanso, Training bioinspired sensors to classify flows, Bioinspiration & Biomimetics 14 (2019) 016009.

[16] L.Z. Dai, G.W. He, X. Zhang, et al., Intermittent locomotion of a fish-like swimmer driven by passive elastic mechanism, Bioinspiration & Biomimetics 13 (2018) 056011.

[17] L.Z. Dai, G.W. He, X. Zhang, et al., Stable formations of self-propelled fish-like swimmers induced by hydrodynamic interactions, Journal of Royal Society Interface 15 (2018) 20180490.

[18] W.X. Huang, S.J. Shin, H.J. Sung, Simulation of flexible filaments in a uniform flow by the immersed boundary method, Journal of Computational Physics 226 (2007) 2206–2228.

[19] X.J. Zhu, G.W. He, X. Zhang, Numerical study on hydrodynamic effect of flexibility in a self-propelled plunging foil, Computers & Fluids 97 (2014) 1–20.

[20] L.Z. Dai, G.W. He, X. Zhang, Self-propelled swimming of a flexible plunging foil near a solid wall, Bioinspiration & Biomimetics 11 (2016) 046005.