

REVIEW

Open Access



Deep learning approaches in flow visualization

Can Liu¹, Ruike Jiang¹, Datong Wei¹, Changhe Yang¹, Yanda Li¹, Fang Wang² and Xiaoru Yuan^{1,3*}

*Correspondence:

xiaoru.yuan@pku.edu.cn

¹Key Laboratory of Machine Perception (Ministry of Education), and School of AI, Peking University, Beijing, China

Full list of author information is available at the end of the article

Abstract

With the development of deep learning (DL) techniques, many tasks in flow visualization that used to rely on complex analysis algorithms now can be replaced by DL methods. We reviewed the approaches to deep learning technology in flow visualization and discussed the technical benefits of these approaches. We also analyzed the prospects of the development of flow visualization with the help of deep learning.

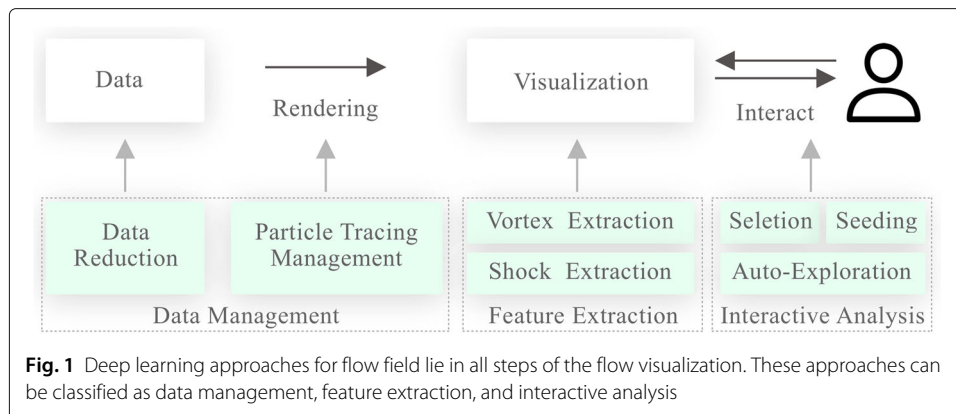
Keywords: Deep learning, Flow visualization, Data management, Feature extraction, Particle tracing

1 Introduction

With the rapid development of modern scientific computing technology, scientists use numerical simulations to study various kinds of phenomena in the real world, such as meteorology, ocean, and other research fields. Flow visualization is the visualization of the vector data generated in fluid dynamics studies, for example, the simulations of combustion models, aerodynamic models, and climate models. The flow field data consists of one or more fields, some of which have time-varying properties, representing the magnitude and direction of the velocity at each location.

As the size of flow data becomes larger, and the internal structure becomes more complex, the analysis of flow data becomes complicated. Visualization of flow data is an important method for understanding flow data. As Fig. 1 shows, the deep learning approaches can be classified into three categories, including data management for flow visualization, flow feature extraction, and interactive visual analysis.

These various tasks in flow visualization processes usually face many challenges that can be handled by DL techniques. For the **data management of flow visualization**, since the flow data usually has a very large size, how to reduce the size of the data to support faster visualization is an essential research topic. Meanwhile, during the rendering process of large-scale flow data, the non-parallelized algorithm is time-consuming. However, the traditional parallel algorithm will be low-efficient as the particles of flow data go irregular, so the data management to improve the efficiency of parallelism also captures researchers' interests [1]. During the visualization process of flow visualization, **features extraction**



including vortex and shock has long been studied using analysis algorithms. For example, Hong et al. [2] used LDA methods for analyzing flow features. These algorithms may be time-consuming, and some features are failed to be extracted using rules, whereas, deep learning methods can be employed to handle these problems. The analysis of flow visualization may not be solved with a static visualization. The **interactive analysis** is needed to help users select the parts they are interested in. With effective interactions including selecting appropriate flow streamlines and flow surfaces, users can understand the flow data better.

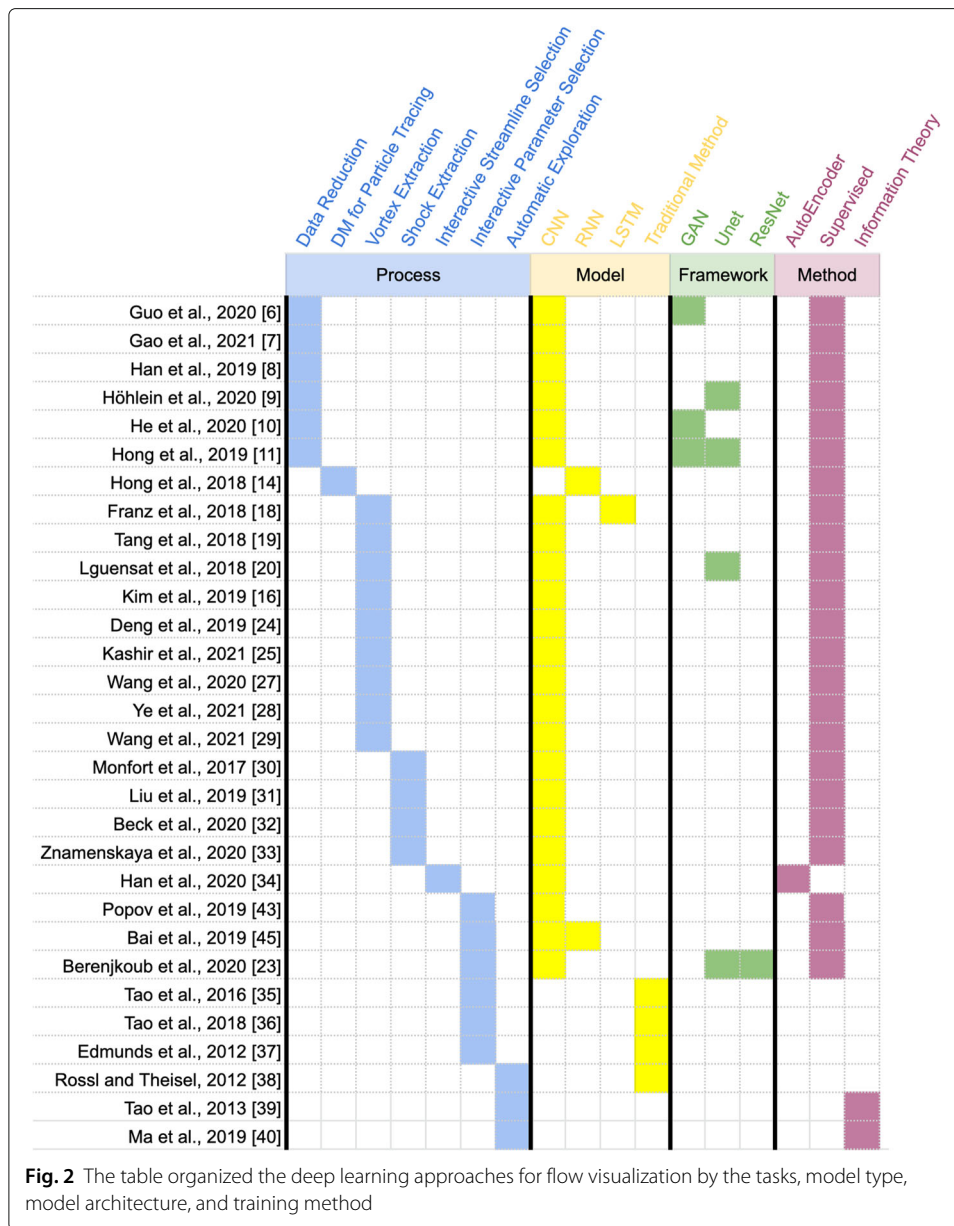
The existing problems above can be addressed with the help of deep learning techniques. Deep learning has shown advantages in representation [3] and feature extraction [4]. Such abilities can help to improve the effectiveness of flow visualization tasks. For example, the presentation can help with data compression, and the feature extraction ability benefits the flow feature extraction. Therefore, these deep learning techniques can have some applications in data management for flow field data visualization, auxiliary analysis in interaction analysis, and automatic feature analysis. We systematically classified and summarized the deep learning-driven flow visualization methods in the category of flow visualization process and the deep learning method and framework employed. Figure 2 organized the deep learning approaches for flow visualizations in the category of the tasks in flow visualization, the method and framework used, and the training method.

2 Data management for flow visualization

The rendering of the large-scale flow visualization requires heavy computation. Thus, a highly efficient data management framework is necessary. There are two ways to enhance the rendering process of the flow data. One is to reduce the size of the data. The other is to speed up the calculation process of the flow data. Deep learning methods show their strong capability in data presentation and prediction, which can be used to help the rendering of the flow data.

2.1 Data reduction

The flow data may have a large volume size because of the huge 2D or 3D space and the time dimension. Deep learning-based methods have shown the representation capability in various tasks [5]. When the representation output of the original data is smaller than the original one, the process of the representation can be regarded as the reduction.



There are two ways to reduce the size of the data. One is to represent the original data information explicitly. For example, use the deep learning model to recover the original data from the low-resolution one [6, 7] or from the streamline result [8]. With the help of a deep learning model, the low-resolution data or the flow visualization results can preserve the most information of the high-resolution data. The other is to encode the data information of the data with the deep learning model implicitly. In such cases, the deep learning model can be seen as the compressed data. The model can be used to synthesize the result of the visualization but the original data is not given explicitly.

To reproduce the high-resolution data information from the low-resolution one, SSR-VFD [6] produces super-resolution (SSR) 3D vector field data (VFD) using a deep learning framework from low-resolution field data, which is the first machine learning-based method to produce super-resolution results. SSR-VFD employs the convolutional neural

networks, as Fig. 3 shows, to synthesize the high-resolution data from the low-resolution one. Gao et al. [7] used the CNN-based method to generate high-resolution flow from low-resolution input. The CNN-SR model is able to denoise the flow field data. In the area of the wind field, Höhle et al. [9] compared several convolutional neural networks to downscale the flow data and proposed the DeepRU model based on U-Net, which can reconstruct the wind structure that other models can not reproduce. To recover the data information from the flow visualization, Han et al. [8] proposed a deep learning method to recover original vector field data from the streamline of the flow data.

In recent years, several works try to represent the content of the original data with the model [10, 11] implicitly, i.e., do not preserve the original data but replace the data with the trained model. For example, He et al. [10] used the CNN model to directly synthesize the rendering results given the parameter as input. DNN-VolVis [11] takes the rendering images representing the viewpoint and style as input to directly synthesize the rendering results. These two approaches are not directly for the vector format data, but there is no limitation to the underlying volume data. The key is to synthesize the resulting image from the deep model.

These methods are able to reduce the size of the data in an explicit or implicit way.

2.2 Data management in particle tracing

In the flow visualization based on parallel particle tracking, data management involves the organization of data in external storage, data access, and the dynamic scheduling of data in parallel particle tracking.

At the algorithmic level, Zhang et al. [12] combine higher-order access dependencies into data management for more accurate data prediction and prefetching. However, such a pre-calculation process in the high-order algorithm still requires large storage. Data prefetching strategy is a proven strategy for solving large data I/O waits in large-scale flow field particle tracking applications. The prefetch process is a prediction problem given the past track of the particles. This process is a traditional sequence to sequence [13] prediction problem, which has been researched in many areas including natural language translation and so on.

Hong et al. [14] first introduced a deep learning model to model the particle trajectories through LSTM networks [15], which can predict the access of particles to data blocks more accurately and thus improve the efficiency of large scale particle tracking algorithms. In this method, the coordinate sequences of particle trajectories are transformed into sequences of data blocks visited by particles, and the historical access records are used as data input to the long short-term memory network, which outputs the data block prediction for particle movement during parallel particle tracing. This method can

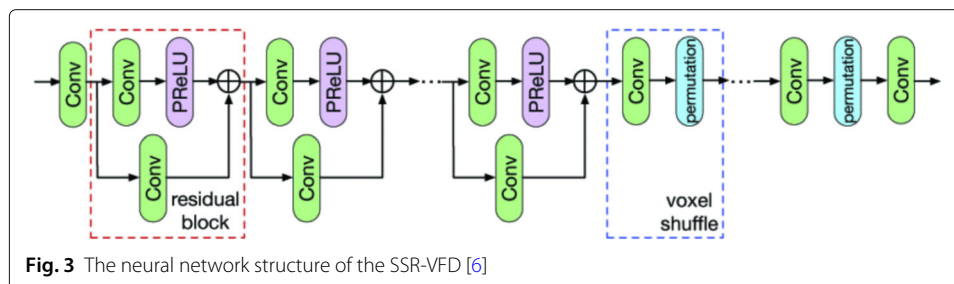


Fig. 3 The neural network structure of the SSR-VFD [6]

obtain the same accuracy of prediction results and significantly reduce the storage cost compared with the higher-order access-dependent data prefetching method proposed by Zhang et al. [12].

2.3 Summary

Data management for flow visualization is a practical and challenging topic. Deep learning methods help to reduce the size of flow data in implicit and explicit ways. In the traditional methods, analysis algorithms are proposed to organize the parallel rendering, and the algorithm may need large storage to store the rule. Deep learning method [14] uses the data-driven method to implicitly perceive the rules with a trained model. The trained model has better generalization than the analysis algorithms. We should also notice that there are still many tasks in the data organization of the flow visualization that have not been explored, for example, using deep learning to guide the data partition in the parallel computation. Methods are introduced to reduce the size through both implicit and explicit ways. In the future, flow data can be reduced by combining with more representation methods in visualization. During the rendering process, the deep model can be used to present the streamline and stream surface of the flow data.

3 Automatic flow feature extraction

Features can be extracted and displayed explicitly in flow visualization to release users' burden of finding the features by themselves. Traditional feature extraction methods relied on experts who provided rules and parameters as optimal settings of extraction algorithms. However, these approaches may need to determine different parameters for different conditions manually and lack the ability to deal with noisy data well. For example, Kim et al. [16] trained their model on noisy data to find reference frames that cannot be detected robustly by traditional methods [17]. Traditional methods also lack the scalability needed in large-scale flow field visualization. Deep learning-based methods could accelerate the calculation process while achieving comparable performance. Recent works introduce deep learning methods to handle the challenges in feature extraction of flow field data.

3.1 Vortex feature extraction

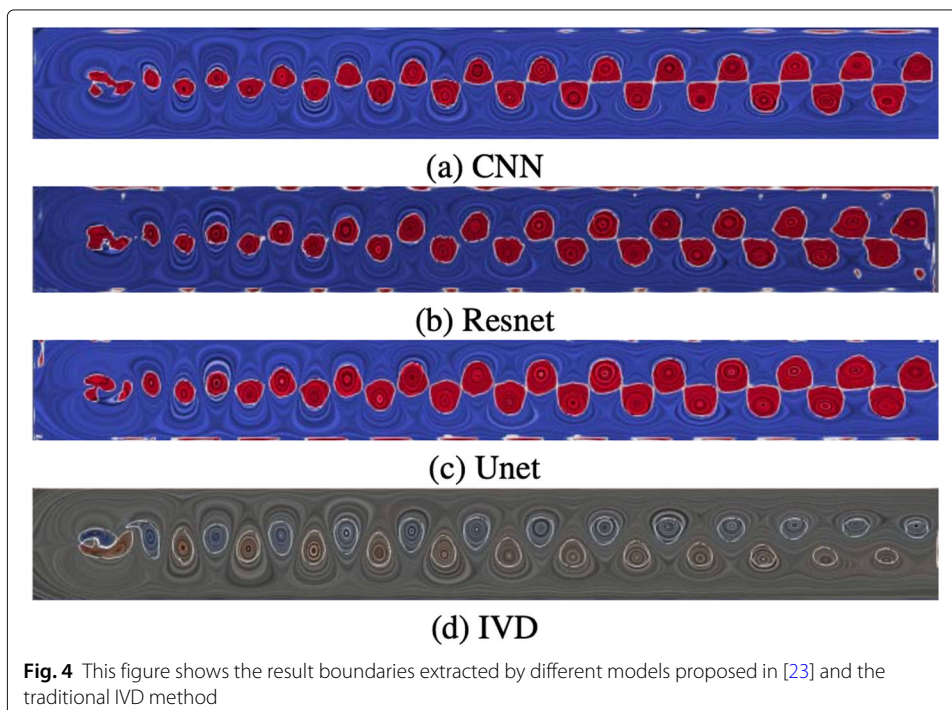
Franz et al. [18] focus on detecting and tracking mesoscale ocean eddies using deep learning methods. The vortex structures affect the global circulation of the ocean, and can further influence global climate change. Their detection model takes an encoder-decoder architecture and uses convolutional layers to extract features. The input data is sea level anomaly maps which are discretized into grids. The daily dataset is divided into training and testing datasets based on date. The output is a labeled image of the same size as the input. The labels are calculated based on the results of the Okubo-Weiss method combined with threshold-based filtering. They test two methods for eddy tracking, namely the image processing method KLT-tracker and a convolutional LSTM. The CNN model could identify eddy cores with probabilities. LSTM is a promising direction as it could track eddies jointly rather than the independent tracking of eddy cores using KLT-tracker. Tang and Li [19] also use CNN to extract flow field features. Lguensat et al. [20] see eddy detection as a pixel-wise classification problem and use a deep neural network to solve the problem.

Vortex Extraction in Unsteady Flow

In the traditional methods [21], the frames in the unsteady flow are transformed into near-steady reference frames to extract vortex structures in the flow field data. However, this is a challenging task to deal with noises and sampling artifacts in the input data. Kim and Günther [16] propose to use the convolutional neural network to find the reference frames robustly. To create a benchmark dataset, they define the steady flow primitive based on Vatisias velocity profile [22], and combine multiple flow primitives as their parametric model. The model is then fitted into a simulated flow field dataset to get distributions of each parameter. Finally, the training dataset is created by sampling the parameters from these distributions independently, transforming the reference frames, adding noises, and sampling. The convolutional network used consists of two convolutional layers to extract features and two fully connected layers to predict the transformations represented by 6 first or second-order derivatives. The network outperforms the previous optimization-based method in both synthetic data and numerical simulation data containing different levels of noise.

Local Vortex Extraction

Vortex identification is important in the flow field analysis. Machine learning-based methods try to combine the benefits of local and global methods. However, these methods can not generalize well and suffer from the scalability issue. Berenjkoub et al. [23] use a parametric method to generate training data with different configurations and test the identification performance of different models, as shown in Fig. 4. Convolutional operation is well-known for its locality and translation invariance, so it is suitable for solving problems in vortex detection. Deng et al. [24] propose Vortex-Net that uses the convolutional neural network to classify whether the central point of a local patch is inside a vortex structure. The model takes local patches of the flow field as input. There are four



convolutional layers using convolutional kernels which are location-invariant. Three fully connected layers are used to classify the features returned by the convolutional layers. The input data patches contain three components of velocity and are normalized while the label of each patch is calculated based on the IVD method. The model is compared with local methods, machine-learning-based methods, and the multiple level perception methods using fully connected layers. Results show that Vortex-Net achieves higher precision and recall than other methods, and reduces both the false positives and false negatives which can be found in local methods.

Global Vortex Extraction

Convolutional-network-based work usually predicts the label of a local patch, which may influence the precision of the method. Instead, Kashir et al. [25] propose to extract features in the flow field at the pixel level. They see the problem as a semantic segmentation problem and use the symmetric fully convolutional network to extract vortex structures in the fluid flow field. Before being fed into the model, the computational grids are converted into image pixels. The network consists of convolutional blocks followed by symmetric deconvolutional blocks. Each block consists of two convolutional layers and one residual layer between them. There is a max-pooling layer at the end of each convolutional layer and an upsampling layer at the front of the deconvolutional layer. The segmentation result is of the same size as the input images with each pixel to be the predicted label. The training dataset is generated using Q-criterion under conditions of different Reynolds number and velocity boundary values. The model is tested on datasets generated with parameters different from the training stage. The model achieves high accuracy under the measurement of precision, Jaccard metric, and Dice metric.

Combination of Local and Global Extraction

Previous methods combining local and global vortex detection methods are mainly based on supervised learning. These methods need large-scale labeled datasets. However, as it is at the beginning stage for using machine learning or deep learning in flow visualization, there are few available large-scale benchmark repositories. Deng et al. [26] proposed an unsupervised learning method to identify important vortex structures. The method consists of three parts, namely data pre-processing, data clustering, and rendering. At the pre-processing stage, the work chooses to use a physical metric, IVD vector, as it reflects the vorticity. In addition, to make the method well generalized to different IVD value ranges, standardization and normalization are used to get normalized IVD vectors. The method uses the K-means algorithm and uses Canopy clustering to determine the optimal number of clusters. The cluster with fewer data points is considered as the vortex set as there are fewer vortex areas in the flow field data. The rendering stage embeds label information into the original mesh. Results show that the method outperforms previous work in F1 score and execution time. The method could also be applied to the low-resolution dataset to reduce memory usage.

Deep learning-based methods balance the fast computation time of local method and high accuracy of global method, and outperform traditional methods. However, they require a large amount of time to train the models as these models contain many parameters and need back propagation to get optimal parameter settings. Wang et al. [27] propose Vortex-ELM-Net to reduce the training time. Vortex-ELM-Net exploits the extreme learning machine method which does not need backpropagation. The model consists of one input layer, several convolutional layers to extract features from the flow

field patches, fully connected layers, and the ELM network to conduct binary classification. The training data is generated by calculating the vorticity and normalizing the results based on z-score and sigmoid, while the labels are generated by the global IVD method. Results on 2D flow field datasets show that the proposed method achieves both high precision and recall. The training time of the proposed Vortex-ELM-Net is smaller than other machine learning and deep learning methods. The visualization result of the method is consistent with the global method, and could reflect vortex phenomena like vortex shedding. Ye et al. [28] use CNN to model the pressure distribution of a flow field data.

Deep learning methods for vortex identification try to take advantage of the high accuracy of global methods and fast computation of local methods. To achieve this goal, deep models take local patches as input and results from the global method as input in the training stage. Previous deep neural networks did not achieve comparable speed as local methods, as they suffer from several drawbacks. First, the large number of parameters in the fully connected layers require a large amount of computation. Second, the patches overlap with each other resulting in repeat computations. Wang et al. [29] propose Vortex-Seg-Net which utilizes fully convolutional layers for segmentation. The output is designed to be a patch instead of a single point to avoid overlapping patches and redundant computations. The loss function for training the network consists of two parts, i.e., the cross-entropy loss which calculates point-wise correctness, and dice coefficient loss which calculates the global correctness based on predicted and ground-truth vortex areas. To generate training data, the authors of the work propose to first transform the non-uniform mesh into a rectangular array. Then the data will go through mesh padding to add velocity patches for points on the boundary. Finally, the raw data will be sampled to create the training dataset. In the testing stage, all patches will be calculated and the final result of vortex areas will be calculated based on the predicted results on local patches. The testing results show that Vortex-Seg-Net achieves comparable accuracy as previous deep-learning-based methods and faster speed. It also outperforms local methods and machine-learning-based methods in accuracy.

3.2 Automatic shock detection

Supersonic turbulent combustion simulation creates large amounts of data and consumes much time. Feature extraction, such as shock extraction, is used to filter the dataset. However, previous filtering methods could not deal with abnormalities in the flow data without domain knowledge.

Supervised Learning for Shock Detection

Monfort et al. [30] demonstrated the feasibility of using deep learning for extracting shock features. The datasets are discretized into volumes or regions, and each volume or region will be used to calculate the strain tensor and the Schlieren value to serve as input and output of the model, respectively. The model consists of three convolutional layers to create feature vectors, and three deconvolutional layers to construct the final images. The model achieves low mean square errors at the testing datasets. It also reduces the time used to calculate the result. Although the model output will contain some noise, it can be reduced by increasing the size of the training dataset. The method could support anomaly detection by comparing the model result and the Schlieren values.

Liu et al. [31] proposed the Shock-Net to detect the shock waves in the flow visualization, which are the locations where aerodynamics variables change abruptly. The data

is generated with important attributes including scalar attributes, pressure and density, and the vector attribute, velocity. The labels of the dataset are calculated using a widely adopted method for detecting shock waves. The structure of the Shock-Net is based on CNN and consists of one input layer and six convolutional layers. The output layer consists of the shock value and the entropy estimation. The loss function is the weighted combination of the shock value loss and entropy. As the shock value only considers the gradient of the pressure, which is not sufficient for the accurate prediction of the shock. The results show that the Shock-Net outperforms other methods in the accuracy and computation time.

Treat Flow Data as Image

Beck et al. [32] decouple the shock capture problem into shock detection stage and stabilization stage. They treat the shock detection problem as an edge detection problem where the flow solution data is seen as the input image, and the presence or location of a shock is seen as the final edge output. They designed the network which is inspired by the holistically-nested edge detection network but is adapted to the small number of pixels of their aimed problem. The network consists of multiple convolution layers and edge map prediction modules for each convolutional layer. Different edge maps contain information at different scales and will be fused to get the final result. The network is trained on both shock indicator data and shock location data. Experiments show high accuracy and robustness. Such a method is particularly useful for high-order conditions as it can fully leverage the high-order information.

Techniques including Schlieren and shadowgraph are used to measure the flow structure. A modern high-speed camera records a large number of images. Traditional methods including edge detection require manually selected thresholds, which could cost a lot of labor and time. Znamenskaya et al. [33] propose a machine learning method based on the convolutional neural network. They first train a CNN model to classify the input images as images containing shock or plume, or empty images. Then they use a transfer learning method to train another regression CNN to predict the position of shock in the classified images. Results show that both networks achieve good accuracy. However, the regression model would produce errors in around half of the cases when the structures are complex due to the training strategy of transfer learning.

3.3 Summary

Deep-learning-based methods can be used to solve other feature extraction problems. For example, Tang and Li [19] trained a unified model to both extract saddle-shaped areas and different types of vortices.

Deep learning models could achieve a balance between computational cost and accuracy for extracting features including vortices and shock values. Detection and tracking are common goals of these models. The models could be further used to model the measurements of flow field data. Typically, the training data is generated using established numerical methods. The models do not need human-selected parameters and are more accurate.

4 Interactive analysis

Interactive exploration and analysis of the features can help users better understand the flow field, and deep learning can play a variety of roles in this process. For example, selec-

tion of streamlines and stream surfaces [34], seeding [35–37], and automatic exploration generation [38–40]. In this section, we will discuss the application of deep learning in interactive feature analysis of flow field data.

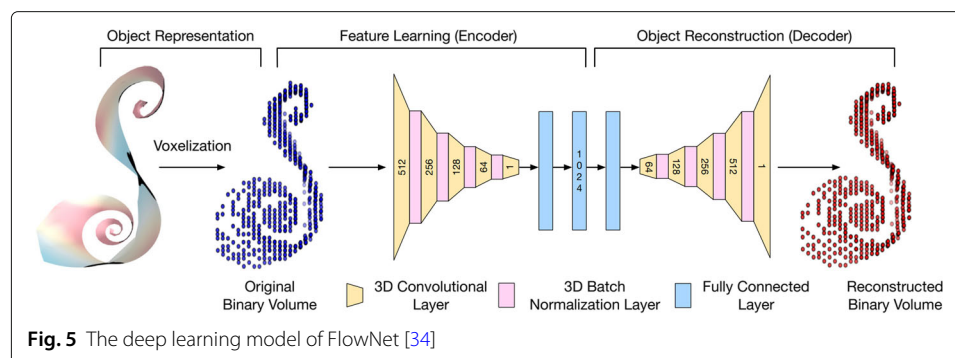
4.1 Interactive streamline selection

The interactive selection in flow field visualization is challenging. Han et al. [34] used a deep learning framework to support the selection of representative streamlines and stream surfaces. FlowNet uses a deep neural network, shown in Fig. 5, to cluster streamlines and stream surfaces in a flow field, allowing users to quickly and intuitively select them in a projection plane. The network uses an auto-encoder model, where the encoder feeds the voxelized streamlines or stream surfaces into a 3D convolutional neural network, generating a 1024-dimensional feature vector; the decoder recovers the feature vector into streamline or stream surface data. After experimenting with different methods, they use t-SNE [41] to downscale the feature vectors to a two-dimensional space and cluster each streamline or stream surface using the DBSCAN[42] algorithm.

4.2 Interactive parameter selection

Seeding is essential for the generation of representative stream surfaces. Tao et al. [35] proposed an interactive stream surface generation method based on users' sketching. A sketch-based interface is designed to allow the user to draw strokes over the streamline visualization. A corresponding 3D seeding curve can be determined, and a stream surface that captures the outermost flow pattern of the streamlines is generated. Then, the streamlines whose patterns are covered by the stream surface are removed. By repeating this process, the streamlines are replaced with customized stream surfaces. Furthermore, Tao et al. [36] proposed a scheme to identify the optimal seeding curve in the neighborhood of an original seeding curve based on surface quality measures. In order to support interactive optimization, a parallel surface quality estimation strategy is designed to estimate the quality of the seeding curve without generating the surface. Edmunds et al. [37] proposed a framework for automatic stream surface seeding. The framework is based on vector field clustering. Users are provided with the flexibility to guide the seeding by controlling the density of surfaces and prioritizing the formation of vector field clusters.

In the simulation of laser-induced breakdown (LIB), the ignition should be calculated for each focal point, which makes it computationally costly. Popov et al. [43] proposed to predict the ignition result to avoid long simulations. They compared machine learning methods and deep learning methods. The machine learning method takes 5 metrics at 3-time steps as input, and a two-layer neural network to train the model. To improve



the final result, they designed an ignition pointer as the output and used the bagging method to use the mean value of different models as the results. The deep learning model is designed as a convolutional neural network with three convolutional and pooling layers. The network uses 5 metrics from 2 time steps and the increasing ratio as input. The output does not need to use a manually designed pointer, while still achieving comparable result.

Some traditional methods [44] rely on manually selected thresholds, which makes the accuracy affected by human judgment. Finding the vortex boundary is important for understanding the flow behavior. The extent of a vortex could also be used to compare vortex structures at different time steps or in different ensemble members. However, the IVD method [44] relies on the threshold manually selected, which makes it difficult to use in large-scale datasets. Deep learning methods trained on synthetic data calculated with the IVD method cannot deal with the vorticity concentration. Bai et al. [45] propose to combine features from multiple layers of the neural network with features of eddies. They designed an object detection neural network called streampath-based region-based convolutional neural networks (SP-RCNN). The authors create a large-scale image dataset based on data about ocean current and use it to train the model. The final result is better than previous work showing the effectiveness of the method. The work also enhances the eddy visualization to help users detect eddies.

Berenjkoub et al. [23] design a new parametric model and fit it with numerically generated data to get their training data. For the neural network, they compare UNet [46], ResNet [47], and CNN. In the experiments conducted on both synthetic and numerical datasets, results show that UNet achieves the best among all other methods.

4.3 Automatic exploration

To further reduce manual interference, there is work on fully automatic exploration for flow fields. Rossl and Theisel [38] proposed a method for interactive exploration of streamlines by mapping streamlines to points in 3D as it can reduce visual clutter than visualizing streamlines directly. The map is based on the preservation of the Hausdorff metric in streamline space. Tao et al. [39] formulated streamline selection and viewpoint selection into a unified information-theoretic framework. Two interrelated information channels between a set of candidate streamlines and a set of sample viewpoints are built with mutual information, shape characteristics, and conditional probability. The streamlines that best capture flow features by passing through the vicinity of critical points or interesting regions are chosen. A camera path that passes through all selected viewpoints is then generated. Ma et al. [40] proposed an automatic method for tour generation of non-constant flow fields. They adopt entropy-based methods for the determination of critical regions to focus on during the tour. The traversal order of selected regions is derived with energy minimization and dynamic programming strategies. After that, the best viewpoints are selected from candidate viewpoints which are created with a mesh enclosing each focal region. Finally, a view path traversing all selected viewpoints is generated.

4.4 Summary

The above methods adopt deep learning to optimize different parts of interactive feature analysis of flow fields, such as selection and seeding. Compared with traditional

methods, deep learning has better performance in extracting the effective features of the flow field. At the same time, through training with large-scale data, deep learning models are able to understand the focus of the users. However, current work on the automatic exploration of flow fields using deep learning is relatively limited, and this aspect needs to be explored more.

5 Conclusion and future work

We classified and summarized the deep learning techniques for flow visualization. In the process of flow visualization rendering, flow feature extraction, and interactive flow visualization exploration, deep learning can be introduced to reduce the data size, accelerate the rendering process, improve the feature extraction accuracy, and automatize interactions.

Most of these approaches used the CNN-based methods as the flow data can be well handled by the convolutional method to extract the local and global features. Some methods use the LSTM model that includes the serial information, for example, to prefetch the data block according to the previous trajectory. Frameworks including GAN, U-Net are employed to preserve more information among deep modules. In the training method, most works focus on supervised learning, while some works aim to extract information from the information itself.

We also have some suggestions on the possible research direction for the deep learning methods in flow visualization. In the data reduction, the novel deep learning model designed considering the properties of the vector field data can be proposed to represent the information of the flow data in a more compact way. When rendering the flow field visualization, deep models can not only help to reduce the extra storage for prefetching data blocks but can also help to organize the partition of the particles or data blocks. During the feature extraction process, deep learning approaches can be introduced to find more kinds of features. Meanwhile, better support for customized feature detection is also needed. Furthermore, in the exploration of flow visualization, deep learning approaches can be proposed to increase the degree of automation.

Acknowledgements

We would like to thank the funding support from NNW2018-ZT6B12 and NSFC No. 61872013, and the reviewers' feedback.

Authors' contributions

Can Liu, Ruike Jiang, Datong Wei, Changhe Yang and Yanda Li worked together on the reference collection. Can Liu, Ruike Jiang, and Datong Wei also contribute to the technical writing. Fang Wang is a domain expert in flow visualization and helps on designing the overall structure of the paper. Xiaoru Yuan initiated and oversaw the project, and made the overall design of work. The authors read and approved the final manuscript.

Funding

This work is supported by NNW2018-ZT6B12 (National Numerical Windtunnel project) and NSFC No. 61872013.

Availability of data and materials

This is a survey paper. All materials (references) are included in the reference list.

Declarations

Competing interests

The authors declare that they have no competing interests.

Author details

¹Key Laboratory of Machine Perception (Ministry of Education), and School of AI, Peking University, Beijing, China. ²China Aerodynamics Research and Development Center, Mianyang, China. ³National Engineering Laboratory for Big Data Analysis and Application, Peking University, Beijing, China.

Received: 11 October 2021 Accepted: 22 March 2022

Published online: 14 April 2022

References

- Guo H, Zhang J, Liu R, Liu L, Yuan X, Huang J, Meng X, Pan J (2014) Advection-based sparse data management for visualizing unsteady flow. *IEEE Trans Vis Comput Graph* 20(12):2555–2564
- Hong F, Lai C, Guo H, Shen E, Yuan X, Li S (2014) FLDA: Latent dirichlet allocation based unsteady flow analysis. *IEEE Trans Vis Comput Graph* 20(12):2545–2554
- Sun Y, Wang X, Tang X (2014) Deep learning face representation from predicting 10,000 classes. In: *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, 23–28 June 2014. IEEE. pp 1891–1898
- Liang H, Sun X, Sun Y, Gao Y (2017) Text feature extraction based on deep learning: a review. *EURASIP J Wirel Commun Netw* 2017(1):211
- Bengio Y, Courville A, Vincent P (2013) Representation learning: A review and new perspectives. *IEEE Trans Pattern Anal Mach Intell* 35(8):1798–1828
- Guo L, Ye S, Han J, Zheng H, Gao H, Chen DZ, Wang J-X, Wang C (2020) SSR-VFD: Spatial super-resolution for vector field data analysis and visualization. In: *Proceedings of the 2020 IEEE Pacific Visualization Symposium (PacificVis)*, Tianjin, 3–5 June 2020. IEEE. pp 71–80
- Gao H, Sun L, Wang J-X (2021) Super-resolution and denoising of fluid flow using physics-informed convolutional neural networks without high-resolution labels. *Phys Fluids* 33(7):073603
- Han J, Tao J, Zheng H, Guo H, Chen DZ, Wang C (2019) Flow field reduction via reconstructing vector data from 3-D streamlines using deep learning. *IEEE Comput Graph Appl* 39(4):54–67
- Höhlein K, Kern M, Hewson T, Westermann R (2020) A comparative study of convolutional neural network models for wind field downscaling. *Meteorol Appl* 27(6):e1961
- He W, Wang J, Guo H, Wang K-C, Shen H-W, Raj M, Nashed YS, Peterka T (2020) InSituNet: Deep image synthesis for parameter space exploration of ensemble simulations. *IEEE Trans Vis Comput Graph* 26(1):23–33
- Hong F, Liu C, Yuan X (2019) DNN-VolVis: Interactive volume visualization supported by deep neural network. In: *Proceedings of the 2019 IEEE Pacific Visualization Symposium (PacificVis)*, Bangkok, 23–26 April 2019. IEEE. pp 282–291
- Zhang J, Guo H, Yuan X (2016) Efficient unsteady flow visualization with high-order access dependencies. In: *Proceedings of the 2016 IEEE Pacific Visualization Symposium (PacificVis)*, Taipei, 19–22 April 2016. IEEE. pp 80–87
- Sutskever I, Vinyals O, Le QV (2014) Sequence to sequence learning with neural networks. In: *Proceedings of the 27th International Conference on Neural Information Processing Systems*, vol 2. MIT Press, Cambridge. pp 3104–3112
- Hong F, Zhang J, Yuan X (2018) Access pattern learning with long short-term memory for parallel particle tracing. In: *Proceedings of the 2018 IEEE Pacific Visualization Symposium (PacificVis)*, Kobe, 10–13 April 2018. IEEE. pp 76–85
- Gers FA, Schraudolph NN, Schmidhuber J (2003) Learning precise timing with lstm recurrent networks. *J Mach Learn Res* 3:115–143
- Kim B, Günther T (2019) Robust reference frame extraction from unsteady 2D vector fields with convolutional neural networks. *Comput Graph Forum* 38(3):285–295
- Günther T, Gross M, Theisel H (2017) Generic objective vortices for flow visualization. *ACM Trans Graph (TOG)* 36(4):1–11
- Franz K, Roscher R, Milioto A, Wenzel S, Kusche J (2018) Ocean eddy identification and tracking using neural networks. In: *Proceedings of the 2018 IEEE International Geoscience and Remote Sensing Symposium*, Valencia, 22–27 July 2018. IEEE. pp 6887–6890
- Tang B, Li Y (2018) CNN-based flow field feature visualization method. *Int J Performability Eng* 14(3):434–444
- Lguensat R, Sun M, Fablet R, Tandeo P, Mason E, Chen G (2018) EddyNet: A deep neural network for pixel-wise classification of oceanic eddies. In: *Proceedings of the 2018 IEEE International Geoscience and Remote Sensing Symposium*, Valencia, 22–27 July 2018. IEEE. pp 1764–1767
- Lugt HJ (1979) The dilemma of defining a vortex. In: Müller U, Roesner KG, Schmidt B (eds). *Recent Developments in Theoretical and Experimental Fluid Mechanics*. Springer, Berlin, Heidelberg. pp 309–321
- Vatistas GH, Kozel V, Mih W (1991) A simpler model for concentrated vortices. *Exp Fluids* 11(1):73–76
- Berenjkoub M, Chen G, Günther T (2020) Vortex boundary identification using convolutional neural network. In: *Proceedings of the 2020 IEEE Visualization Conference (VIS)*, Salt Lake City, 25–30 October 2020. IEEE. pp 261–265
- Deng L, Wang Y, Liu Y, Wang F, Li S, Liu J (2019) A CNN-based vortex identification method. *J Vis* 22(1):65–78
- Kashir B, Ragone M, Ramasubramanian A, Yurkiv V, Mashayek F (2021) Application of fully convolutional neural networks for feature extraction in fluid flow. *J Vis* 24:771–785
- Deng L, Wang Y, Chen C, Liu Y, Wang F, Liu J (2020) A clustering-based approach to vortex extraction. *J Vis* 23(3):459–474
- Wang J, Guo L, Wang Y, Deng L, Wang F, Li T (2020) A vortex identification method based on extreme learning machine. *Int J Aerosp Eng* 2020:8865001
- Ye S, Zhang Z, Song X, Wang Y, Chen Y, Huang C (2020) A flow feature detection method for modeling pressure distribution around a cylinder in non-uniform flows by using a convolutional neural network. *Sci Rep* 10:4459
- Wang Y, Deng L, Yang Z, Zhao D, Wang F (2021) A rapid vortex identification method using fully convolutional segmentation network. *Vis Comput* 37(2):261–273
- Monfort M, Luciani T, Komperda J, Ziebart B, Mashayek F, Marai GE (2017) A deep learning approach to identifying shock locations in turbulent combustion tensor fields. In: Schultz T, Özarslan E, Hotz I (eds). *Modeling, Analysis, and Visualization of Anisotropy*. Mathematics and Visualization. Springer, Cham. pp 375–392
- Liu Y, Lu Y, Wang Y, Sun D, Deng L, Wang F, Lei Y (2019) A CNN-based shock detection method in flow visualization. *Comput Fluids* 184:1–9
- Beck AD, Zeifang J, Schwarz A, Flad DG (2020) A neural network based shock detection and localization approach for discontinuous Galerkin methods. *J Comput Phys* 423:109824
- Znamenskaya I, Doroshchenko I, Tatarenkova D (2020) Edge detection and machine learning approach to identify flow structures on schlieren and shadowgraph images. In: Bykovskii S, Kustarev P, Mourontsev D (eds). *Proceedings*

- of the 30th International Conference on Computer Graphics and Machine Vision, Saint Petersburg, 22-25 September 2020
34. Han J, Tao J, Wang C (2020) FlowNet: A deep learning framework for clustering and selection of streamlines and stream surfaces. *IEEE Trans Vis Comput Graph* 26(4):1732–1744
 35. Tao J, Wang C (2016) Peeling the flow: a sketch-based interface to generate stream surfaces. In: *Proceedings of the SIGGRAPH ASIA 2016 Symposium on Visualization (SA'16)*, Macau, 5-8 December 2016. Association for Computing Machinery, New York
 36. Tao J, Wang C (2018) Semi-automatic generation of stream surfaces via sketching. *IEEE Trans Vis Comput Graph* 24(9):2622–2635
 37. Edmunds M, Laramée RS, Malki R, Masters I, Croft TN, Chen G, Zhang E (2012) Automatic stream surface seeding: A feature centered approach. *Comput Graph Forum* 31(3pt2):1095–1104
 38. Rossl C, Theisel H (2012) Streamline embedding for 3D vector field exploration. *IEEE Trans Vis Comput Graph* 18(3):407–420
 39. Tao J, Ma J, Wang C, Shene C-K (2013) A unified approach to streamline selection and viewpoint selection for 3D flow visualization. *IEEE Trans Vis Comput Graph* 19(3):393–406
 40. Ma J, Tao J, Wang C, Li C, Shene C-K, Kim SH (2019) Moving with the flow: an automatic tour of unsteady flow fields. *J Vis* 22(6):1125–1144
 41. Van der Maaten L, Hinton G (2008) Visualizing data using t-SNE. *J Mach Learn Res* 9(86):2579–2605
 42. Ester M, Kriegel H-P, Sander J, Xu X (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. In: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, Portland, August 1996. AAAI Press. pp 226–231
 43. Popov PP, Buchta DA, Anderson MJ, Massa L, Capecehatro J, Bodony DJ, Freund JB (2019) Machine learning-assisted early ignition prediction in a complex flow. *Combust Flame* 206:451–466
 44. Haller G, Hadjighasem A, Farazmand M, Huhn F (2016) Defining coherent vortices objectively from the vorticity. *J Fluid Mech* 795:136–173
 45. Bai X, Wang C, Li C (2019) A streampath-based RCNN approach to ocean eddy detection. *IEEE Access* 7:106336–106345
 46. Ronneberger O, Fischer P, Brox T (2015) U-Net: Convolutional networks for biomedical image segmentation. In: Navab N, Hornegger J, Wells W, Frangi A (eds). *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Lecture Notes in Computer Science, vol 9351. Springer, Cham. pp 234–241
 47. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, Las Vegas. pp 770–778

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Ready to submit your research? Choose BMC and benefit from:

- fast, convenient online submission
- thorough peer review by experienced researchers in your field
- rapid publication on acceptance
- support for research data, including large and complex data types
- gold Open Access which fosters wider collaboration and increased citations
- maximum visibility for your research: over 100M website views per year

At BMC, research is always in progress.

Learn more biomedcentral.com/submissions

