

ORIGINAL RESEARCH ARTICLE

Multi-task super resolution method for vector field critical points enhancement

Yilun Yang, Zhou Hao, Weilong Peng, Keke Tang, Meie Fang*

School of Computer Science and Cyber Engineering, Guangzhou University, Guangzhou 511442, Guangdong, China. E-mail: fme@gzhu.edu.cn

ABSTRACT

It is a challenging task to handle the vector field visualization at local critical points. Generally, topological based methods firstly divide critical regions into different categories, and then process the different types of critical regions to improve the effect, which pipeline is complex. In the paper, a learning based multi-task super resolution (SR) method is proposed to improve the refinement of vector field, and enhance the visualization effect, especially at the critical region. In detail, the multi-task model consists of two important designs on task branches: one task is to simulate the interpolation of discrete vector fields based on an improved super-resolution network; and the other is a classification task to identify the types of critical vector fields. It is an efficient end-to-end architecture for both training and inferencing stages, which simplifies the pipeline of critical vector field visualization and improves the visualization effect. In experiment, we compare our method with both traditional interpolation and pure SR network on both simulation data and real data, and the reported results indicate our method lower the error and improve PSNR significantly.

Keywords: critical point; vector field visualization; multiple tasks; super resolution

1. Introduction

Vector field visualization has wide application in the field of science and engineering, and plays an important role in meteorology, aircraft design, automotive design, and computational fluid dynamics^[1]. Although significant progress has been made in vector field visualization after years of research and development, quite a few questions worth exploring still exist. For example, during data collection tasks, due to the limited precision of sensors and radical vector changes near critical points, the collected discrete data do not represent well the local features.

Currently, vector field visualization related works are mainly focused on improving the visualization quality and effectiveness, including the research, application, and refinement of streamline generation strategies, color enhancements^[2], GPU load balancing^[3], and seeding strategies^[4]. However, these methods do not work well for the problem of vector field visualization near critical points. Since the critical points in the vector field contain very distinct structural information and can be explicitly characterized by topological features, the topological feature-based approach is better for visualizing the critical region of the vector field^[1]: vector fields are first classified according to the type of critical points, the

ARTICLE INFO

Received: May 5, 2022 | Accepted: June 5, 2022 | Available online: June 21, 2022

CITATION

Yang Y, Hao Z, Peng W, et al. Multi-task super resolution method for vector field critical points enhancement. *Metaverse* 2022; 3(1): 8 pages.

COPYRIGHT

Copyright © 2022 by author(s). *Metaverse* is published by Asia Pacific Academy of Science Pte. Ltd. This is an Open Access article distributed under the terms of the Creative Commons Attribution License (<https://creativecommons.org/licenses/by/4.0/>), permitting distribution and reproduction in any medium, provided the original work is cited.

visualization is then enhanced using a special interpolation according to its classification. This process however involves numerical-based classification algorithms, which is computationally intensive. At the same time, the visualization process is complicated, and error accumulation is still too high for radical vector direction changes.

Super-resolution algorithm achieves image enhancement by enlarging the local details of an image. It is a mature end-to-end machine learning algorithm in the image processing field. Since vector field data and image have the same characteristics, i.e. a fixed boundary and number of channels, this paper draws on the super-resolution algorithm to perform super-resolution enhancement on the critical region of the vector field. In addition, combined with the idea of multi-tasking, and the fact that multiple tasks in the network model can enhance each other, we consider adding multiple supervised signals based on the super-resolution model to achieve further enhancement of the super-resolution effect of the vector field.

Therefore, this paper proposes a multi-task based super-resolution neural network model to achieve the improvement of interpolation and visualization near the critical region of vector fields. For the super-resolution task, interpolation of discrete vector fields is implemented to improve data accuracy in order to highlight more detailed features of the data fields. For the classification task, the classification of critical points is implemented, replacing the numerical solving classification of critical points in traditional topology-based methods. Effective information is extracted at the shallow layers of the super-resolution neural network instead of interpolation of specific types of vector fields.

In summary, the approach this paper presents has two main novel contributions:

(1) A deep learning based multi-task super-resolution network model is designed to achieve a refined enhancement of the critical region of the vector field.

(2) The proposed vector field enhancement method is an end-to-end efficient inference approach

that helps simplify the process of vector field visualization.

2. Related works

2.1. Vector field visualization

Vector field visualization is part of visualization in scientific computing. In practice, vector field visualization methods can be divided into icon-based, geometry-based, texture-based, topology-based^[1], etc. Among them, texture-based methods are usually used for two-dimensional vector field visualization, and the most representative ones are the Line Integral Convolution (LIC) algorithm^[5] and the improved algorithms derived from the Line Integral Convolution algorithm^[2]. Extensive research has already been done on LIC, and as such LIC related works are abundant and mature. Since the LIC algorithm usually encounters problems of occlusion and low efficiency on 3D data, 3D vector fields currently are mostly visualized by flow lines or flow surfaces based methods.

2.2. Topology-based methods

The theoretical basis for the vector field visualization based on topological features is the Critical Point Theory, in which the critical point refers to a point in the vector field where each component is zero. The Critical Point Theory considers the topology of an arbitrary vector field consisting of critical points, and surfaces and curves connecting those critical points. As a result, it can help people extract the important information of the vector field by extracting the topological skeleton and ignoring the secondary information, thus improving the visualization and enhancing the understanding of the vector field data. The topology-based approach has a sound mathematical basis and can therefore be applied to any vector field. The classification of critical points in this paper is based on publication^[6].

A very important step for topology-based of vector field visualization is to classify the critical points so that the topological characteristics of the vector fields can be better studied. The eigenvalues

and eigenvectors of the Jacobi matrix are calculated by evaluating the partial derivatives of the vector field for each critical point. Classification is then performed based on the eigenvalue distribution on the complex plane^[7]. Generally, they can be classified into saddle points, nodes, foci, centers, etc. Better visualization can be done according to critical point classifications.

2.3. Deep learning and visualization

General data visualization methods have been commonly used in the field of deep learning, usually for visualizing the training process of deep learning models, assisting researchers in debugging models, and enhancing the interpretability of models^[8], etc. On the contrary, there are only a few related works on data field visualization using deep learning methods. For example, publication^[9] trains a model for viewpoint recommendation in volumetric data visualization, and publication^[10] uses deep learning methods to reconstruct vector fields from generated streamlines, etc. In conclusion, the intersection of artificial intelligence and visualization is currently a research focus in computer science, and is early in its infancy. There are still many issues worth studying^[11].

2.4. Super-resolution

Super-resolution technology was first used in image processing to restore low-resolution images to high-resolution images, increasing image clarity in the process. Over the past decade, deep learning-based image super-resolution enhancement algorithms have been emerging. SRCNN method was first proposed in the publication^[12], on the basis of which many improved super-resolution models were proposed, such as FSRCNN^[13], which improved the input of SRCNN, changed the feature dimension, and accelerated the training speed. Other improved models such as VDSR^[14], ESPCN^[15], and more were also proposed. The deep learning-based image super-resolution technology has been developed into a full-fledged and mature image enhancement solution over the years.

Publication^[16] used the SRCNN method in the scalar field sampling, publication^[17] simulated and generated a huge fluid flow dataset and used a simple super-resolution model for interpolation training, and publication^[18] trained a deep convolutional neural network to interpolate 3D vector fields. This is the first time someone proposed a neural network specifically for vector field interpolation. The three components of the vector are separately trained and then combined, the network consists of several residual blocks.

3. Multi-task super-resolution network model

3.1. Problem statement

Objective Given a low-resolution vector field X , our goal is to synthesize a high resolution vector field Y using some function F .

Definition and Types of F Essentially, $F: X \rightarrow Y$ is an interpolation process, since given a low-resolution domain U with uniformly distributed discrete values in $[1, \dots, M] \times [1, \dots, N]$, the objective of F is to infer the discrete values in a higher resolution domain $[1, \dots, sM] \times [1, \dots, sN]$. F can be a non-machine learning-based function, such as bilinear interpolation, or an ML-based function, such as a super-resolution neural network. They all have their own characteristics and bottlenecks in ensuring reasonable continuity of high-resolution discrete values.

Non-ML Based F Issues The sampling information which F uses to generate Y is from X only. This is more effective for discrete values with good smoothness on U . If complex ridge and line features emerge, the generated result will be unsatisfactory.

ML-Based F Issues The sampling information which F uses to generate Y is not only from X , but also from a sample domain $\{X\}$ to which X belongs. This can interpolate features that are not visible in X but are present in the sample domain. However, if the critical points present in $\{X\}$ is diverse and have radical changes, a single-task holistic

deep learning approach based on $\{X\}$ will have great difficulty achieving high accuracy.

In a two-dimensional vector field, the direction of vectors near the critical points is highly dependent on the critical point type. The network architecture we propose will add subcategorized supervised signals of critical point type to the SR model, and use the shared network to split the sample domain $\{X\}$ into smaller subdomains $\{X_1\}\{X_2\}\{X_3\}\{X_4\}$ during

the learning process, which facilitates further improvement of the SR task on each subdomain.

3.2. Neural network architecture

The multi-tasking network architecture we proposed for two-dimensional vector field critical points is shown in **Figure 1**. It is a multi-task super-resolution network model that includes a super-resolution branch and a classification branch, as well as a base network shared by both branches.

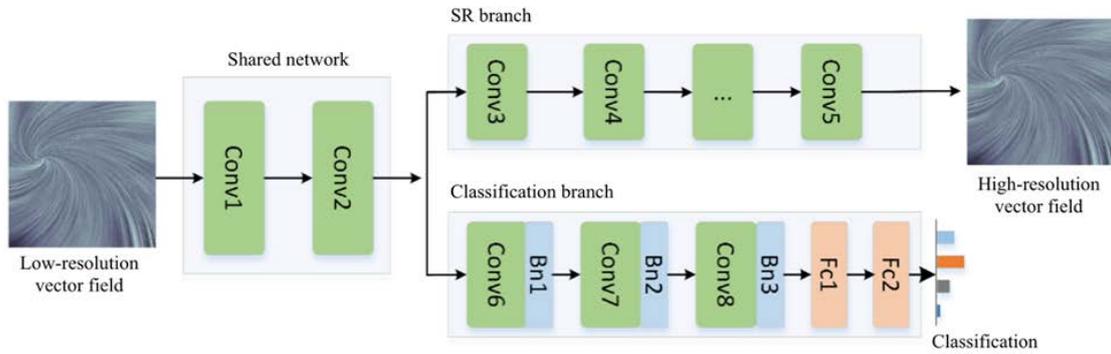


Figure 1. Neural network architecture.

Input. The super-resolution branch of this model is modified from VDSR and adopts a similar approach to VDSR in terms of data input. The low-resolution vector field is first scaled up to the same size as the high-resolution vector field and subsequently fed into the network.

Shared Network. Since our approach is multi-task prediction, we adopt the end-to-end learning strategy considering the inference speed and model learning capability. The shared network is intended to find common features across the two branches. Specifically, common features are extracted by convolving the input data twice, and then fed into the super-resolution branch and the classification branch for their respective prediction tasks.

Super-resolution Branch. The super-resolution branch is composed of several convolutional layers, in which the Conv4 layer can be followed by several identical convolutional layers. In this experiment, we used three Conv4 layers. To improve the computational efficiency of the classification branch, the number of channels before the introduction of the classification branch is reduced. The original number of channels is then restored. We used the Leaky

ReLU activation function in order to prevent the vanishing gradient problem in this experiment.

Classification Branch. We employ a conventional classification architecture. Having elicited data from the super-resolution branch, a four-classification result is acquired after three convolutional layers and two fully connected layers.

Under the supervision of the classification branch, this architecture uses a shared network to split the input sampling domain of the vector field space domain into small subdomains. Data enhancement is performed via each super-resolution branch. The types and parameters used in each layer of the architecture are shown in **Table 1**.

3.3. Loss function

For the super-resolution of images, we usually use the root mean square error (MSE) to measure the accuracy of the model output results. Thus, we continue to employ MSE as the loss function for our model. However, in vector field super-resolution based on the critical point classification, it is not enough to consider only the root mean square error.

Table 1. Model parameters

Name	#Input channel	#Output channel	Convolution kernel size
Conv1	2	64	3
Conv2	64	16	3
Conv3	16	64	3
Conv4	64	64	3
Conv5	64	2	3
Conv6	16	16	3
Conv7	16	24	3
Conv8	24	24	3
Fc1	13824	600	-
Fc2	600	4	-

As the loss function of the classification also needs to be taken into consideration, we have made the decision of combining multiple loss functions.

Vector Loss Measurement. We utilize MSE to measure the loss of vectors ($LOSS_{mse}$), it's defined as follows:

$$LOSS_{mag}(V, V') = \frac{1}{2 \times W \times H} \sum_{y=0}^H \sum_{x=0}^W (V_{x,y} - V'_{x,y})^2 \quad (1)$$

In Equation (1), V represents real data, and V' represents the model prediction result. We calculate the root mean square error by comparing the square of the difference between each component of the output vector and each component of the real value.

Classification Loss Measurement. We use the cross-entropy loss function to measure the accuracy of the classification ($LOSS_{ce}$), defined as follows:

$$LOSS_{ce} = -\frac{1}{m} \sum_{i=1}^m \sum_{j=1}^n y_{ij} \log(p_{ij}) \quad (2)$$

In Equation (2), m is the number of samples fed into the model for each training, n is the number of categories for classification, and y_{ij} is the indicator variable, which is 1 if the i th data is equivalent to category j and 0 otherwise. p_{ij} is the predicted probability distribution of belonging to category j for data i .

Based on the above two loss functions, our final loss function is defined as:

$$LOSS_{final} = LOSS_{mag}(V, V') + LOSS_{ce} \quad (3)$$

In practice, each component of the loss function can be weighted accordingly. In this experiment, we assign the same weight to both loss functions.

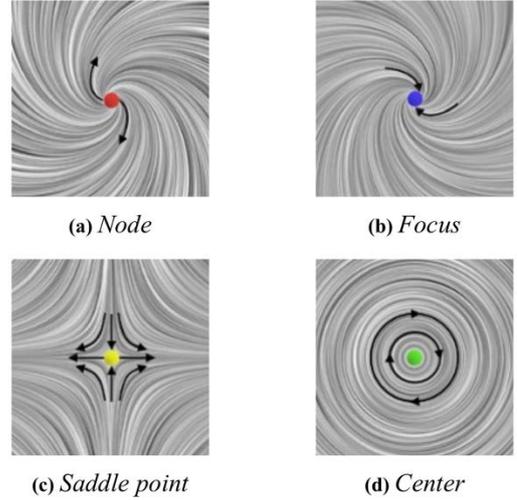


Figure 1. Critical point types.

Source: [6].

4. Experiment results

4.1. Data preparation

Data Simulation. Since we need to perform SR to the vector field according to the critical point type, and there is a lack of large-scale datasets for vector fields, the training data can only be generated via simulation. The theoretical basis of our simulation is derived from the publication^[19], where an experiment-based formula is given that can simulate various types of critical points. Different types of critical points can be generated by adjusting the parameters. We use this formula to generate the critical points of two-dimensional vector fields. Types of generated critical points include saddle points, centers, foci, and nodes. The characteristics of vector fields near different critical points are shown in **Figure 2**. 9,345 synthetic data entries were used for training and 763 entries were used for testing, each with a size of 200×200 and a channel number of 2. The data were listed in categories according to data type, as shown in **Table 2**:

Table 2. The information about adopted dataset

Type	# of training entries	# of test entries
Nodes	2645	243
Foci	2645	243
Saddle points	1396	48
Centers	2659	229
Total	9345	763

4.2. Implementation details

The experiment was run on a server equipped with an Intel Xeon E5 2678 v3 processor with 64GB of memory, and two NVIDIA GeForce RTX 2080Ti GPUs with 11GB of memory. The model was trained for 7.2 hours.

In this experiment, in order to achieve a more accurate training effect, we also adopt the adaptive learning rate. The initial learning rate is set to 0.001, and thereafter the learning rate becomes one-tenth of the previous one every 30 epochs. Adam optimizer is utilized, and 60 shuffled data entries are fed into the neural network every epoch, for 120 a total of 120 epochs.

4.3. Evaluation metrics

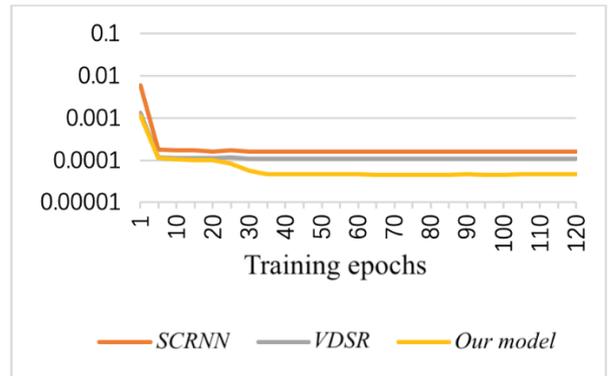
We use the peak signal-to-noise ratio (PSNR) here to measure the accuracy of the output results. PSNR was originally used to evaluate the effectiveness in image super-resolution tasks, and the formula for calculating PSNR is based on the root mean square error. Here we can also use it to measure the results of our model output. In the task of evaluating vector field super-resolution models, PSNR is defined as:

$$PSNR(V, V') = 10 \log_{10} \left(\frac{I(V)}{MSE(V, V')} \right) \quad (4)$$

We have made some modifications to the PSNR formula, where $I(V)$ is the difference between the maximum and minimum values of the vector field values, and in Equation (4) we evaluate the truthfulness of the model output by calculating the root mean square error between the predicted result V and the true value V' .

4.4. Comparative analysis of training process and end results

Here we compare the model in this paper with some other algorithms, including the SRCNN model for SR imaging, and we also compare the training effectiveness of this model with the VDSR model in order to verify the effect of multi-objective optimization. **Figure 3** shows the changes of the root mean square error of the SRCNN model, the VDSR model, and the model proposed in this paper during the training process:

**Figure 2.** Changes in RMSE during training (vertical axes).

As can be seen from **Figure 3**, during the training process, the root mean square error decreases faster due to the large initial learning rate, and then decreases by varying degrees later as the learning rate adjusts. Overall, after adding the prediction branch for multi-task training, the model converges better, and the error is lower than that of the model without the prediction branch. The adaptive learning rate also has a positive effect on multi-objective optimization. The output of the model tested on different kinds of critical points is shown below (**Figure 4**).

Subsequently, to validate the model, we tested the model using untrained test data, while adding a portion of the double gyres to verify the generalization ability of the model. The MSE and PSNR derived from each of these methods are shown in **Tables 3** and **Table 4**, and the best results are marked in bold font.

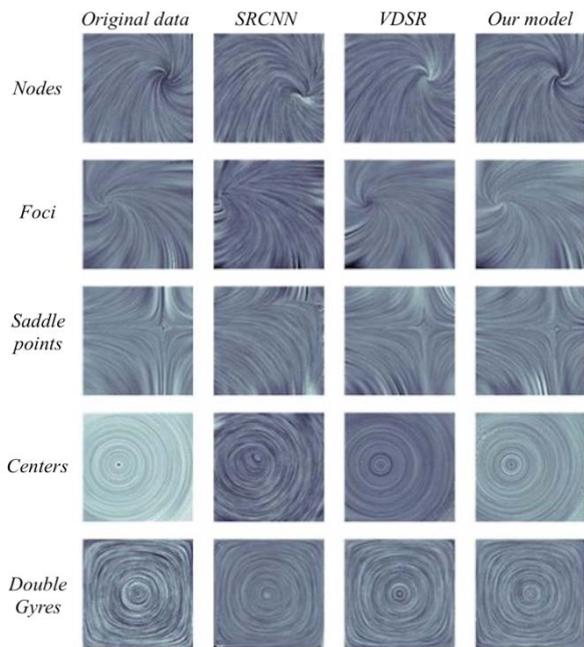


Figure 3. Experiment results.

Table 3. MSE values of different models

Type	SRCNN	VDSR	Our model
Nodes	0.0032	0.00011	0.000047
Foci	0.0066	0.00009	0.000042
Saddle points	0.0031	0.00006	0.000028
Centers	0.0035	0.00018	0.000083
Double gyres	0.00053	0.00014	0.00006

Table 4. PSNR values (dB) of different models

Type	SRCNN	VDSR	Our model
Nodes	20.79	35.43	39.12
Foci	17.59	35.83	39.55
Saddle points	18.39	35.31	38.83
Centers	19.51	32.40	35.76
Double gyres	30.73	36.51	40.20

From the above evaluation data, it can be seen that from the SRCNN model to the VDSR model to the multi-task SR model, the SR end result of the input vector field is getting better and better. Meanwhile, the VDSR model can perform the SR task more accurately compared to SRCNN due to the higher number of layers. Therefore, under certain conditions, the deeper the neural network is, the better the training result will be. Under the condition that model depth is fixed, the model with a prediction branch outperforms the model without a prediction branch.

For the double gyres, our proposed model is shown to be able to accurately determine the critical point classification, and combined with the evaluation metrics, we can see that our model has better results than other models for SR task of this data.

5. Conclusions

This paper proposes a new method for vector field visualization based on a multi-task super-resolution model. After experiments, it is proved that this model achieves better results than the conventional numerical interpolation methods and single-task super-resolution models, has higher accuracy for super-resolution from low-resolution vector fields to high-resolution vector fields, and has higher accuracy for the classification of critical points. The model also has good extensibility and is we believe it can be generalized to visualize 3D vector fields, which is one of the future research possibilities. Besides, our model can only identify a single critical point in a vector field, and further research is needed to classify and identify more complex vector fields with more complex topology.

Conflict of interest

The authors declare no conflict of interest.

References

1. Chen W, Shen Z, Tao Y. Big data series: Data visualization. 2nd ed. Beijing: Publishing House of Electronics Industry; 2019.
2. Tang B, Shi H. Intelligent vector field visualization based on line integral convolution. Cognitive Systems Research 2018; 52: 828–842.
3. Zhang J, Guo H, Hong F, et al. Dynamic load balancing based on constrained k-d tree decomposition for parallel particle tracing. IEEE Transactions on Visualization and Computer Graphics 2018; 24(1): 954–963.
4. Engelke W, Lawonn K, PREIM B, et al. Autonomous particles for interactive flow visualization. Computer Graphics Forum 2019; 38(1): 248–259.
5. Cabral B, Leedom LC. Imaging vector fields using line integral convolution. Association for Computing Machinery 1993; 6: 263–270.
6. Günther T, Rojo I. Introduction to Vector field topology. In: Hotz I, Masood TB, Sadlo F, Tierny J (editors). Topological methods in data analysis and visualization VI. Switzerland: Springer International

- Publishing; 2021.
7. Ba Z, Dan G, Liu J, et al. A feature-based seeding method for multi-level flow visualization. *Journal of Computer-Aided Design & Computer Graphics* 2016; 28(1): 32–40.
 8. Hohman F, Kahng M, Pienta R, et al. Visual analytics in deep learning: An interrogative survey for the next frontiers. *IEEE Transactions on Visualization and Computer Graphics* 2019; 25(8): 2674–2693.
 9. Yang C, Li Y, Liu C, et al. Deep learning-based viewpoint recommendation in volume visualization. *Journal of Visualization* 2019; 22(5): 991–1003.
 10. Han J, Tao J, Zheng H, et al. Flow field reduction via reconstructing vector data from 3-D streamlines using deep learning. *IEEE Computer Graphics and Applications* 2019; 39(4): 54–67.
 11. Xia J, Li J, Chen S, et al. A survey on interdisciplinary research of visualization and artificial intelligence. *Scientia Sinica (Informationis)* 2021; 51: 1777–1801.
 12. Dong C, Loy C, He K, et al. Learning a deep convolutional network for image super-resolution. *European Conference on Computer Vision*; 2014 Sep 6. Springer International Publishing; 2014. p. 184–199.
 13. Dong C, Loy C, Tang X. Accelerating the super-resolution convolutional neural network. *arXiv*; 2016.
 14. Kim J, Lee J, Lee K. Accurate image super-resolution using very deep convolutional networks. *IEEE Conference on Computer Vision & Pattern Recognition*; 2015.
 15. Shi W, Caballero J, Huszár F, et al. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. *arXiv*; 2016.
 16. Zhou Z, Hou Y, Wang Q, et al. Volume upscaling with convolutional neural networks. *Proceedings of the Computer Graphics International Conference*. New York: Association for Computing Machinery; 2017. p. 1–6.
 17. Jakob J, Gross M, Gunther T. A fluid flow data set for machine learning and its application to neural flow map interpolation. *IEEE Transactions on Visualization and Computer Graphics* 2021; 27(2): 1279–1289.
 18. Guo L, Ye S, Han J, et al. SSR-VFD: Spatial super-resolution for vector field data analysis and visualization. *2020 IEEE Pacific Visualization Symposium (Pacific Vis)*; 2020.
 19. Kim B, Günther T. Robust reference frame extraction from unsteady 2D vector fields with convolutional neural networks. *Computer Graphics Forum* 2019; 38(3): 285–295.