

Received June 14, 2019, accepted July 9, 2019, date of publication July 29, 2019, date of current version August 15, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2931781

# A Streakpath-Based RCNN Approach to Ocean Eddy Detection

XUE BAI, CHANBO WANG, AND CHENHUI LI<sup>ID</sup>

School of Computer Science and Technology, East China Normal University, Shanghai 200062, China

Corresponding author: Chenhui Li (chli@sei.ecnu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61802128, Grant 61672237, and Grant 61532002, and in part by the Fundamental Research Funds for Central Universities.

**ABSTRACT** An eddy is a circular current of water in the ocean that affects the fields of maritime transport, ocean data analysis, and so on. Traditional eddy detection methods are based on numerical simulation data and satellite images and their accuracy is affected greatly by manual threshold adjustment. In this paper, we present a new eddy detection approach via deep neural networks to improve eddy detection accuracy. First, we present a streakpath-based approach to build a large-scale eddy image dataset from ocean current data and apply our dataset to eddy detection. Second, by combining the multilayer features in the neural network with the characteristics of the eddies, we achieve a competitive detection result with an mAP of 90.64% and an average SDR of 98.91%, which performs better than the previous methods. Third, through our enhanced eddy visualization approach, we solve the problem that eddies are difficult to detect in the sparse streakpath region.

**INDEX TERMS** Eddies detection, deep neural network, flow visualization, object detection.

## I. INTRODUCTION

The ocean is full of eddies with radii from a few kilometers to hundreds of kilometers. They have a significant impact on weather conditions, shipping, marine ecology, multiscale movements in the ocean, etc. [1]–[3]. Therefore, the detection and analysis of ocean eddies have great significance and scientific value.

In the past few decades, researchers have developed several methods to detect eddies through numerical simulations and satellite image measurements [4], [5]. These methods are mainly based on the characteristics of the data for manual detection. The detection accuracy is greatly affected by the preset threshold, and the detection result is not intuitive. It is difficult for nonprofessionals to understand. Recently, object detection approaches based on deep learning, such as faster region-based convolutional neural network (Faster RCNN) [6], you only look once (YOLO) [7] and single shot detection (SSD) [8], have achieved good results. Although these approaches can be applied to eddy detection applications, there is no satisfactory eddy image dataset.

Eddies in satellite images and ocean velocity data are difficult to identify. The ground truth labeling is also a challenge

The associate editor coordinating the review of this manuscript and approving it for publication was Songwen Pei.

for satellite image data and ocean velocity data. As objects in the ocean, the eddies are different sizes and shapes. There may be many small eddies distributed in one area. Detecting small objects in the image is challenging for these methods. Thus, eddy detection applications require a feasible dataset and an improved traditional object detection method.

In this paper, we propose an eddy dataset generation pipeline and a new approach to automatically detecting ocean eddies from flow field data, so-called streakpath-based region-based convolutional neural networks (SP-RCNN). The experimental results show that our approach achieves higher accuracy than previous eddy detection approaches. We further employ our approach to the visualization of flow field data, which enhance the display of the eddies by placing more particles in the eddy domain. Compared to previous object detection methods, SP-RCNN is more competitive for small objects and suitable for eddy detection on streakpath-based eddy images. We believe our method is the first work to apply deep learning techniques to detect eddies in flow field data.

The main contributions of our work are as follows. 1) We present a streakpath approach to transform flow field data to streakpath-based eddy images. Then, a large labeled dataset is provided for training and eddy detection verification. 2) We propose a robust region-based

convolutional neural network for accurate ocean eddy detection. 3) We propose an iterative approach to enhance the streampath visualization of the eddies to further improve the eddy detection accuracy. It is beneficial for humans to detect the eddy directly on the current streampath video, and people will not fail to detect unclear streampath eddies.

The rest of this paper is organized as follows. In Section III, we present an overview of our approach. We first introduce our streampath transformation process, then we describe the implementation of our eddy detection method and provide an explanation of our network structure. We also perform cross-validation to illustrate the effectiveness of our approach. Finally, we describe our enhanced methods. Experimental results are shown in Section IV to demonstrate the practicability and effectiveness of our method. We present the discussions and conclusions in Section V.

## II. RELATED WORKS

In the last decade, researchers have proposed many visualization methods based on velocity field data. Arrow plots are the original method, but they are not intuitive enough. Vatistas [9] proposed an empirical formula for the tangential velocity that is better for the observations. Many methods are based on the extension of this method. Cabral and Leedom [10] proposed the line integral convolution (LIC), which enables velocity field data to be better visualized. Recently, Kim and Gunther proposed a robust reference frame extraction method [11] based on the Vatistas velocity profile. It uses neural networks to extract a steady reference frame from a given unsteady 2D vector field. Under increasing noise and resampling artifacts, this method is robust on real-world data.

It is difficult for people to extract information directly from high-dimensional data, but they can identify the velocity field behavior directly from the picture. There are many techniques for visualizing ocean data. Grant [12] used Lagrangian-Eulerian time surfaces to visualize ocean flow vertical motions. Banesh *et al.* [13] introduced a visualizing and tracking mesoscale ocean eddies methods using contours and moments that illustrated the vortex motion event intuitively. Williams [14] and Liu [2] visualized three-dimensional eddy structures to discover the eddies' physical properties and motion patterns. These methods all require eddy detection.

Traditional eddy detection methods can be divided into three categories: 1) methods based on physical parameters; 2) methods based on ocean velocity data; and 3) hybrid methods. The Okubo-Weiss (OW) [15], [16] method is the most widely used eddy detection method based on physical parameters. The parameter is defined as  $OW = Sn^2 + Ss^2 - w^2$ , where  $Sn$  and  $Ss$  are the normal strain and shear strain, respectively, and  $w$  is the relative vorticity. The eddies usually exist where the rotation dominates the flow field where  $OW$  is negative. Although this method is widely used to detect eddies, it still has some limitations [17], [18] in that it is difficult to determine the optimal threshold, and the physical

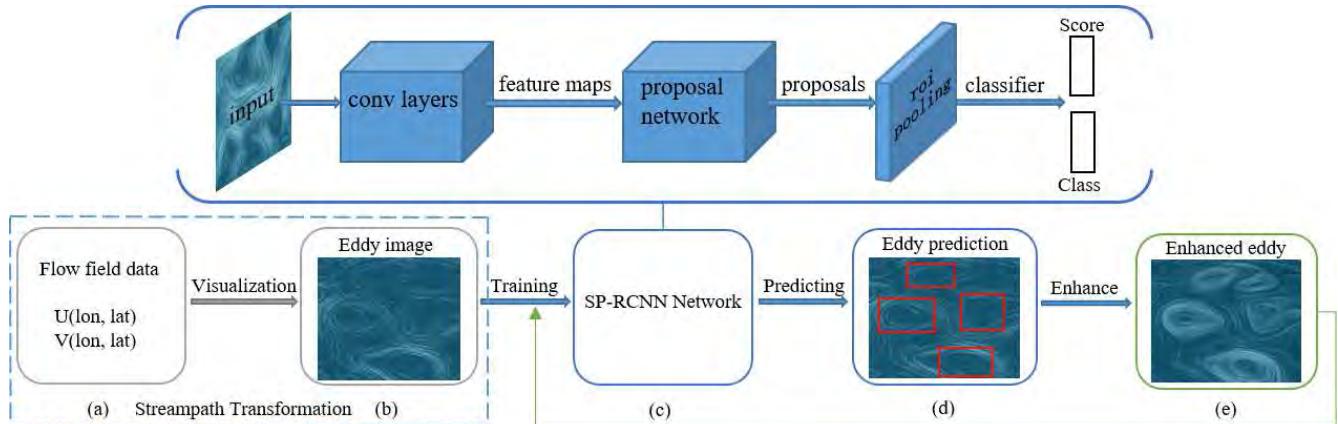
parameter derivation process will create noise and increase the eddy false detection rate.

The most representative methods based on ocean velocity data are winding-angle (WA) [17] and vector geometry (VG) [19] methods. The WA method, which identifies eddies by a vortex curve, was proposed by Sadarjoen and Post in 2000. Chaigneau *et al.* [18] compared OW and WA methods on the same data from the South Pacific. The results show that the WA method has a higher success rate than the OW method, but it requires more computational resources. The VG method is based entirely on the geometry of velocity streampaths, which detect eddy centers with four predefined constraints according to the velocity. These traditional detection methods are not only complex but also require artificial parameter adjustment, and the application scope of these methods is very limited.

Recently, deep learning has been applied to various research fields and has achieved superior results, some of which are even beyond human experts. Deep learning approaches are also widely used in object detection applications. Since 2014, there have been a series of object detection methods, such as RCNN [20], Fast RCNN [21], Faster RCNN [6], and YOLO [7], [22], [23], SSD [8].

RCNN was the first to show that a CNN can lead to a higher object detection performance than methods based on histogram oriented gradient (HOG)-like features, but it is relatively slow because it requires a large computational capacity without sharing computations. Fast RCNN was proposed based on RCNN; instead of feeding each object proposal to the CNN, it feeds the entire input image to the CNN to compute a convolutional feature map and then identifies each object proposal from the convolutional feature map. Fast RCNN is significantly faster than RCNN in training and testing sessions, but using selective search to generate the region proposals is a slow and time consuming process. To solve this bottleneck, Ren *et al.* [6] proposed the Faster RCNN, which introduces region proposal networks to replace the selective search algorithm to reduce the proposal computational costs. Further improvements on the object detection approaches include YOLO and SSD. These approaches are different from the two-stage detection algorithms seen above in that they extract feature directly to eliminates the region proposal process and predict bounding boxes and the class probabilities of these boxes in a single convolutional network. Thus, they maintain a fast running speed, and both enable real-time object detection, but they are struggle with small objects.

There are also some eddy detection methods based on deep learning. Lguensat *et al.* [24] proposed a deep learning-based architecture for pixel-wise detection and classification of ocean eddies from the sea surface height (SSH) maps dataset. Deng *et al.* [25] trained a vortex identification network that used an objective vortex definition method [26] to obtain the labels. Franz *et al.* [27] developed an eddy identification and tracking framework with a classification CNN that uses the sea level anomalies (SLA) dataset.



**FIGURE 1.** Workflow of our eddy detection method. (a) The flow field data. (b) The eddy image generated using the streampath transformation approach. (c) The SP-RCNN architecture. (d) Our predicted eddies. (e) The enhanced eddy image.

### III. PROPOSED METHOD

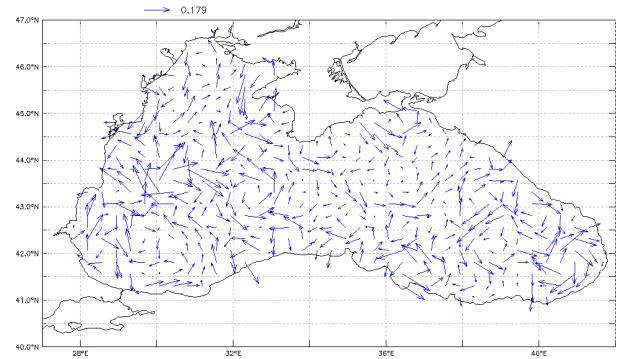
To automatically detect eddies from the flow field data, we combine the shape feature of the eddies to adopt a deep CNN model in this work. The results show that our method achieves excellent performance in eddy detection, which not only has comparable accuracy to the previous eddy detection approaches but can directly visualize the results on the image so that humans can detect eddies without professional knowledge. We further employ our method to visualize the flow field data and enhance the display of the eddies by placing more particles in the eddy domain. We believe our method is the first work to apply deep learning techniques to detect eddies in flow field data.

The workflow of our eddy detection method is illustrated in Figure 1. We first process the flow field data in Figure 1(a) and visualize them to a streampath image to obtain our initial training dataset Figure 1(b). We extract the feature maps of the eddy image through the convolution layer of the SP-RCNN network Figure 1(c) and send the feature maps to the region proposal network (RPN) to obtain the candidate region proposal of the eddy image.

Then, we use bounding box regression to obtain the exact position of the eddy Figure 1(d). In stage (e), we use the detection results of Figure 1(d) to place more particles in the eddy domain to enhance the eddies. We feed these enhanced eddy images to the dataset to make our dataset more robust and improve the detection accuracy of SP-RCNN. The following three subsections explain the detailed implementation.

#### A. STREAMPATH TRANSFORMATION

A high-accuracy eddy detection application requires a high-quality dataset and a high-performance deep learning model. However, the performance of the existing velocity-based eddy detection approach [19] requires further improvement in an eddy analysis application. Many approaches, such as CNN-based image classification and CNN-based object detection, have the potential to further improve eddy detection accuracy. However, the flow

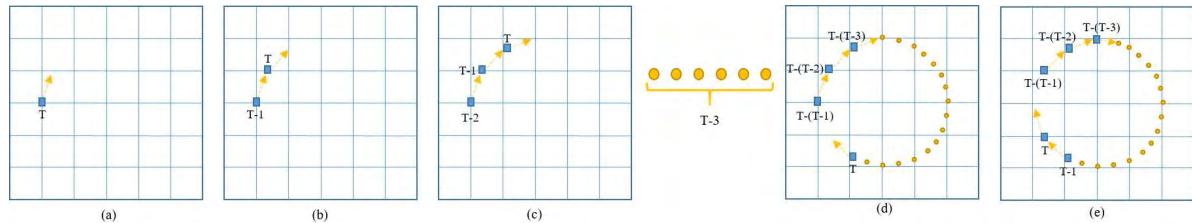


**FIGURE 2.** A schematic diagram of UV field data from AVISO, the direction of the blue arrow represents the direction of the speed, and the length represents the magnitude of the speed.

field data are not like the natural image because image camera can supply high resolution photography. The resolution of the flow field data is limited and they only include the limited features for training the neural network. AVISO <https://www.aviso.altimetry.fr/en/home.html> is the most commonly used flow field data for ocean eddy detection; it is combined with ocean topography experiment results and remote sensing satellite information [28], [29]. Figure 2 shows AVISO's absolute geostrophic velocity data (UV) including the zonal component (U) and meridian component (V).

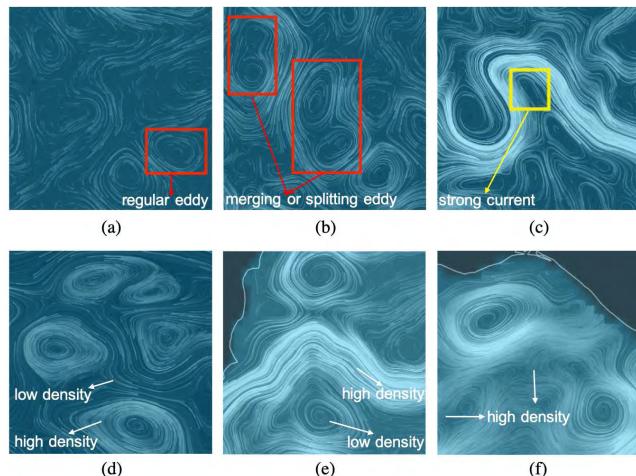
To address these concerns, we present a streampath transformation approach (SP-Tran) to generate abundant high-quality ocean eddy images from the flow field data. Our method is inspired by the work of LIC [10] from the flow visualization field. Eddy images generated via SP-Tran include more abundant eddy patterns than raw ocean velocity data. Moreover, it is easier to label ground truth data compared with the previous approaches because human systems are sensitive to the eddy pattern. SP-Tran can be formulated as follows:

$$\frac{ds(t)}{dt} = UV(sp(t), t), \quad sp(0) = sp_0, \quad 0 \leq t \leq T \quad (1)$$



**FIGURE 3.** The generation process of a streampath image. The blue squares represent particles. The yellow arrow represents the direction of the current particle. The yellow dot represents a period of time.

where  $UV$  is the ocean velocity field data,  $t$  is the timestamp,  $T$  is the maximum timestamp, and  $sp$  is the function of the streampath. We use particle motions to simulate the streampaths of the velocity field. For a time-varying ocean velocity field, numerous particles are uniformly released from certain points at each moment. The particle's tangential direction at any point is the same as the ocean velocity value at that point. The particle position is updated by its tangential direction at an iteration, as shown in Figure 3. The particle's life cycle is  $T$ . Every iteration, the particle's life is reduced by one, and the particle disappears when the life is 0. After a certain number of iterations, we can obtain a streampath image for an ocean velocity data frame.



**FIGURE 4.** Examples of our dataset, the dark region is land. Inside the red box in (a) is a regular eddy. Inside the red box in (b) is a merging or splitting eddy. Inside the yellow box in (c) is a strong current. (d-f) These images have different density streampaths.

Hence, our training data generation consists of three stages. 1) Generating a streampath image by using the zonal component ( $U$ ) and meridian component ( $V$ ) flow field data, the vortex streampath that is composed of particles is considered to be an eddy, as shown in Figure 4. 2) Using the visualization results of the first step to crop the eddy images in different sub-regions to generate our eddy dataset. 3) Labeling this dataset manually by two professionals, one professional is responsible for the initial labeling and the other professional

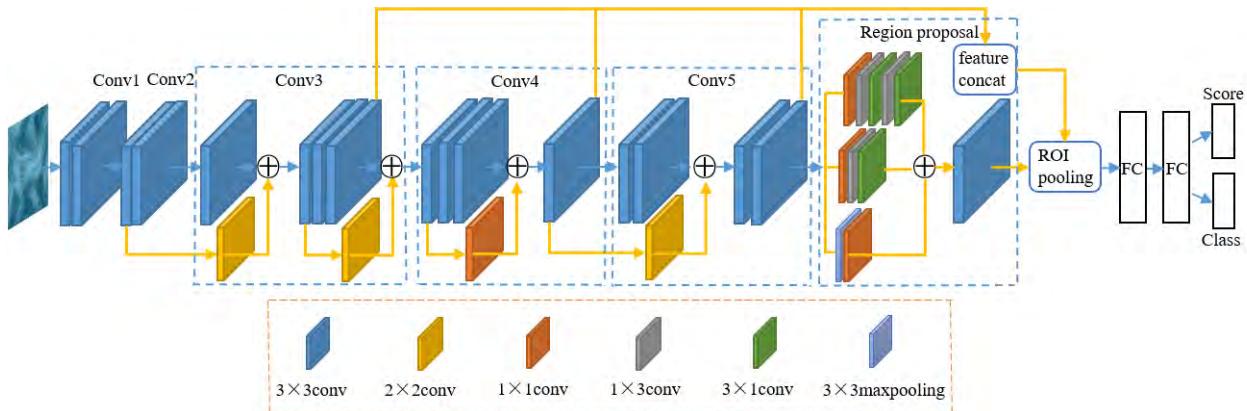
is responsible for correction to prevent mislabeling and to add missing labels in unclear streampath regions.

We applied SP-Tran to the AVISO and generated a dataset with a total of 3,500 eddy images. Our dataset includes many kinds of eddy images, as shown in Figure 4. An eddy is composed of a vortex streampath. Figure 4(a) shows an eddy image of a sparse streampath. Figure 4(b) is an eddy image of a medium density streampath. Figure 4(c) shows a confusing eddy image; the streampath inside the yellow box is a strong current, not an eddy. Figure 4(d) presents an eddy image of a high-density streampath in the eddy region and a low-density streampath around the eddy region. Figure 4(e) outlines an eddy image of a low-density streampath in the eddy region and a high-density streampath around the eddy region. Figure 4(f) shows an eddy image of a high-density streampath. We take 64% of our dataset for training, 16% of our dataset for verification and the remaining 20% is used for testing. To our knowledge, there is currently no publicly available eddy image dataset based on ocean velocity field data. We have established a relatively more sufficient dataset, and our dataset will be published for future research. We will continue enriching our dataset in the future.

## B. SP-RCNN

Inspired by the works of Ren *et al.* [6] and Szegedy *et al.* [30]–[32], we present a deep convolution neural network model, called SP-RCNN, to identify the shape feature of the eddies. Figure 5 shows an illustration of our SP-RCNN architecture. First, we input the image that is labeled with a set of ground truth eddies, and then we use different convolutional layers to generate feature maps. These feature maps are sent to the RPN to generate a set of manageable candidate eddy region proposals. Last, we concatenate different feature maps and these proposals to the region of interest (ROI) pooling layer and output classification and identification information.

In many previous object detection works, researchers have simply extracted feature maps from the last convolutional layer. This is reasonable for many natural scene images but is not appropriate for eddy detection. Some of the strong currents, especially meander currents in the Gulf Stream, as shown in Figure 4(c), have eddy-like features but should not be considered an eddy [14]. Research shows that the depth of the network is an important factor for achieving better



**FIGURE 5.** The SP-RCNN network architecture. In the orange box, there are different convolutional legends. It takes the image as input and generates the classification and identification information as output.

results [30]. To detect the eddies more accurately, we use residual connections for every two convolutional layers to deepen our network and improve the network performance. Furthermore, the residual connection can solve the degradation problem caused by increasing the depth of the network.

In addition, high-quality proposals are critical for the overall performance of eddy detection. RCNN [20], Fast RCNN [21] and Faster RCNN [6] are all two-stage detection methods, which generate region proposals in the first stage and classify them in the second stage. Although these methods have achieved good performance in object detection, they focus less on the quality of the region proposals generated in the first stage. For example, some of the eddies, especially very small eddies with a diameter of less than 100 km, are easily missed in the proposal stage. As a result, these eddies will not be detected in the detection stage. Therefore, we combine the RPN with an inception block to obtain more eddy features. Our inception block consists of three convolutional blocks, as shown in Figure 5. We split a  $3 \times 3$  convolution layer into a  $1 \times 3$  convolution layer and a  $3 \times 1$  convolution layer, which extracts features over Conv5\_4 feature maps at every sliding location.

Because there are many currents in the ocean, most of which are not eddies, filtering out the candidate region proposals is essential for our model. Many eddies are much smaller than objects in natural scene images. The size of the reference bounding box has a large impact on the candidate region proposals of our model. To make our reference bounding box more suitable for eddy detection, we consider the multiple shapes and sizes of eddies. We predict multiple region proposals at each location; the number of maximum proposals for each location is  $k$ . We design four scales (4,8,16,32) and six aspect ratios (0.2,0.5,0.8,1,1.5,2), yielding  $k = 24$  reference bounding boxes at every sliding location to prevent missing objects such as very small eddies.

Additionally, using only the proposals generated by the RPN easily causes the network to lose some details [33]; thus,

treating the strong current as an eddy and may cause deviation of the predicted object position. Because the low-level features have less semantic information but the object position is accurate, the semantic information of the high-level features is richer, but the object position is relatively rough. To solve this issue, we combine the feature maps of multilayer convolution (Conv3\_4, Conv4\_4, Conv5\_4) of the VGG19 model [34]. Then, we send these combined feature maps and the proposals generated by the RPN to the ROI pooling layer, which will help the classifier distinguish eddies from various backgrounds with more discriminative features increase the accuracy of eddy detection.

In our work, the proposal is considered as a foreground that has an intersection-over-union (IoU) above 0.7 with the ground truth bounding box, while the proposal that has an IoU lower than 0.3 with the ground truth bounding box is considered background. An interval proposal network (IPN) is our RPN with an inception block. SP-RCNN (SPN) is our main network with residual blocks and multilayer features concatenation. With these definitions, we define the loss function for an image as follows:

$$L_{spn} = L_{clsipn} + L_{locipn} + L_{clsroi} + L_{locroi} \quad (2)$$

$$\{L_{clsipn}, L_{clsroi}\} = -\log[e_i e'_i + (1 - e_i)(1 - e'_i)] \quad (3)$$

$$\{L_{locipn}, L_{locroi}\} = e' \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise} \end{cases} \quad (4)$$

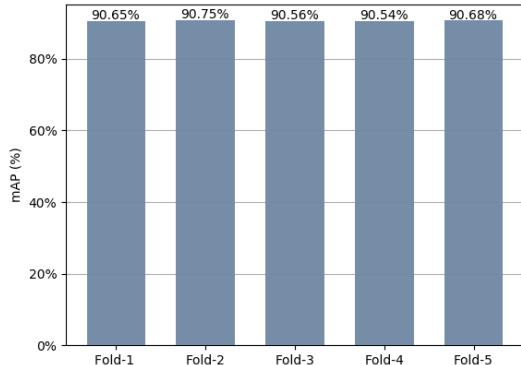
where  $L_{spn}$  is our main loss function.  $L_{clsipn}$  and  $L_{clsroi}$  are the classification loss functions of the IPN and ROI, respectively.  $L_{locipn}$  and  $L_{locroi}$  are the regression loss functions of the IPN and ROI, respectively. Our regression loss function is a robust L1 loss.  $e_i$  is the predicted probability of proposal  $i$  as an object.  $e'_i$  is the ground truth label. If the proposal is foreground, the value of  $e'_i$  is 1; otherwise, if the proposal is background, the value of  $e'_i$  is 0.  $x$  is the elementwise difference between the prediction box and ground truth.

### C. CROSS-VALIDATION

Cross-validation [35] is primarily used in model training or modeling applications. It can avoid models that have sufficient predictive capability on the training data; however, they cannot correctly predict future data. Therefore, cross-validation is the most common method for evaluating model performance [36]. The data are usually divided into two parts. Based on this division, one part is used to train, and the other part to test predictive performance. To obtain a reliable and stable model and avoid over-fitting and underfitting, we perform cross-validation.

The mean is calculated as equation 5. We divide the dataset into  $k$  parts,  $i$  is the different combination of  $k$  data parts. Considering the size of our dataset, we chose a 5-fold cross-validation that takes the value of  $k$  as 5. Our dataset is divided into five parts; we use four parts for training and one part for verification.  $mAP_i$  is the mean average precision of the different datasets, and the mean of five  $mAP$  is an estimate of the accuracy of our model. Figure 6 is the result of 5-fold experiments; the horizontal axis indicates the datasets, and the vertical axis is the  $mAP$  corresponding to different datasets. It can be seen that the accuracy of our model is approximately the same for the different data, which shows that our model is effective.

$$CV_k = \frac{1}{k} \sum_{i=1}^k mAP_i \quad (5)$$



**FIGURE 6.** The mAP of 5-fold cross-validation. The horizontal axis indicates the datasets. The vertical axis is the mAP corresponding to different datasets.

### D. ENHANCED VISUALIZATION

For the initial flow field data, we were unable to determine the position of the eddies before detection. In the initial visualization stage, we scattered the particles evenly in the SP-Tran process, which causes sparse streampaths in small velocity regions, particularly the center of the eddies, as shown in Figure 4(a). Therefore, sparse streampaths will result in an inaccurate definition of the eddy boundary, and it is difficult for a person to directly identify the specific positions of the eddies, which is likely to cause missed detection.

Applying the detection results of our network, we can improve our SP-Tran approach in the visualization stage. We specifically scatter more particles in the eddies region and scatter fewer particles in the noneddies region. Thus, the streampaths will not be too sparse even in the small velocity region, and the eddy region will be enhanced, and the noneddy region will be weakened. The enhanced eddy image will reduce the noise impact of the training dataset, helping the network detect more accurate eddy boundaries. It not only helps humans identify eddies intuitively but also increases the quality of our training dataset.

### IV. EXPERIMENTS

In this section, we introduce the experimental results of our proposed approach. More results are shown in the additional demo video. Many factors affect the effectiveness of eddy detection. We carried out three experiments to evaluate the performance of our approach. The first experiment evaluated the design of our network architecture. The second experiment reported the effectiveness of our model compared with state-of-the-art methods. The last experiment applied our detected results to the visualization process and demonstrated our enhanced eddies.

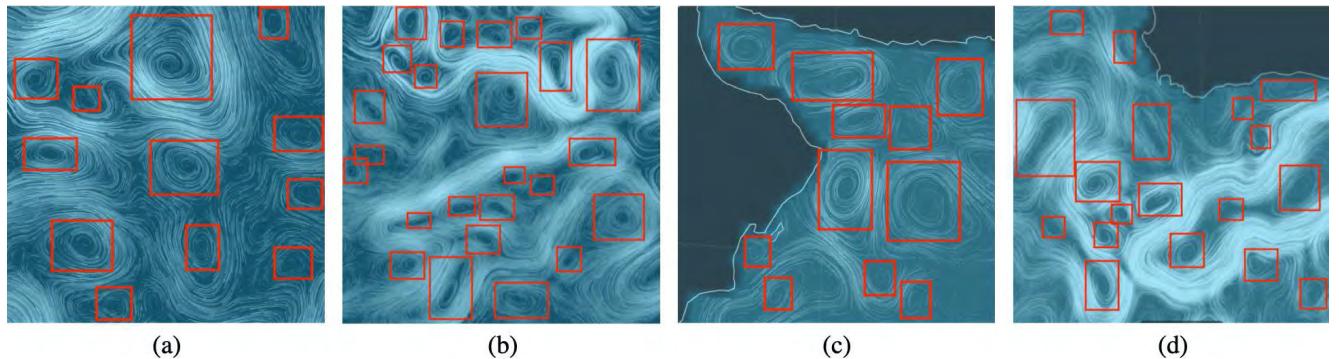
#### A. EVALUATION OF SP-RCNN

We considered eddy characteristics and evaluated the performance of our network using the reference bounding boxes designed in Section III-B. Table 1 shows the settings of the bounding boxes with different scales and aspect ratios. The result is consistent with our expectations. We can see that our model with four scales [4,8,16,32] and six aspect ratios [0.2,0.5,0.8,1,1.5,2] performs better, and the network can achieve an  $mAP$  of 88.94%. If using three scales and three aspect ratios at each position, the  $mAP$  decreases by approximately 1%. This proves that large-scale reference bounding boxes will cause the network to miss several small eddies, and our reference bounding boxes are suitable for eddy characteristics and cover almost all eddy shapes.

**TABLE 1.** The comparison between reference bounding boxes with different scales and aspect ratios.

| mAP(%)                | Scales | [8,16,32] | [4,8,16,32]  |
|-----------------------|--------|-----------|--------------|
| Aspect ratios         |        |           |              |
| [0.5,1,2]             |        | 87.96     | 88.16        |
| [0.5,1,1.5,2]         |        | 88.15     | 88.32        |
| [0.5,0.8,1,1.5,2]     |        | 88.47     | 88.73        |
| [0.2,0.5,0.8,1,1.5,2] |        | 88.75     | <b>88.94</b> |

To evaluate the effectiveness of the proposed method in Section III-B, we conducted control experiments on different components of our network. Table 2 demonstrates the performance of different components through mean average precision ( $mAP$ ) calculation. The baseline model is our



**FIGURE 7.** Results of SP-RCNN. Inside the red box is the detected eddies. (a) and (b) is the area without land boundaries, (c) and (d) is the area with land boundaries.

**TABLE 2.** Control experiments on different components of our network (dataset with 3,500 regions).

| Components Setting         | mAP(%)       |
|----------------------------|--------------|
| Baseline model             | 87.96        |
| Baseline model+EBB         | 88.94        |
| Baseline model+EBB+IPN     | 89.57        |
| Baseline model+EBB+IPN+SPN | <b>90.64</b> |

basic network with VGG19 model. The EBB is our reference bounding boxes with four scales[4, 8, 16, 32] and six aspect ratios[0.2, 0.5, 0.8, 1, 1.5, 2]. IPN is our RPN with an inception block. SP-RCNN (SPN) is our main network with residual blocks and multi-layer features concatenation. We find that the full model with EBB, SPN and IPN performs better. This indicates that our network can detect almost all eddies.

We show more test results of SP-RCNN in Figure 7. We tested areas with land boundaries and no land boundaries. Figure 7(a) is a sparse eddy area without land boundaries. Figure 7(b) shows a dense eddy area without land boundaries. Figure 7(c) is a sparse eddy area with land boundaries. Figure 7(d) shows a dense eddy area with land boundaries. Our method is robust, and we can detect almost all eddies.

## B. PERFORMANCE

To fully validate our model, we compare our method with the traditional VG method. Because VG is mainly for detecting mesoscale eddies with a radius between 100-300 kilometers (the pixels are between 25-60 in our dataset), for consistency of comparison, we compared mesoscale eddies and other scale eddies. There are many different shapes of eddies near the Gulf Stream. Therefore, we take the eddies for 6 days in the region near the Gulf Stream for validation. We validate our method by three different parameters [18], the success of detection rate  $SDR = N_c/N_t$ , the excess of detection rate  $EDR = N_e/N_t$  and the miss of detection rate  $MDR = N_m/N_t$ .

where  $N_t$  is the total number of ground truth eddies,  $N_c$  is the number of eddies identified by the method that has an eddy center corresponding to the position of the ground truth eddy center, and  $N_e$  is the number of eddies identified by the method in which the position of eddy center does not correspond to the ground truth eddy center,  $N_m$  is the number of eddies which should be detected, but were not detected by the method.

Table 3 reports the results of the VG method and our method. For the detection of mesoscale eddies, our method achieves an average SDR of 98.91%, which is higher than the SDR of the VG method. The VG method performs better on mesoscale eddies than on other scale eddies, but it has a lower SDR and a higher MDR than the original VG paper [19], which has an SDR of 92.9%; however, the dataset of the original paper is not disclosed. Our dataset is different, so the accuracy is different. The detection accuracy of our method is also higher than the SDR of the WA method (92.7%) and OW method (86.8%) observed by Chaigneau et al. [18]. It is not possible to perform a direct comparison with these two methods because our dataset is different from these two methods. Furthermore, the EDR in our study is only 0.72%, which is also lower than the traditional method.

To fully understand our results, we visualize the detection result of day 1 in Figure 8. The red boxes present the correctly detected eddies, the white boxes outline the missed eddies, and the black box indicates the incorrectly detected eddy. We can clearly see that the VG method cannot detect irregularly shaped eddies or eddies with diameters less than 100 km (25 pixels) because these eddies do not meet the predefined constraints in a limited area. Our method can detect almost all eddies, except for several eddies with sparse streakpaths. The incorrectly detected eddies of the VG method are mainly due to the region meeting the predefined constraints, but it is not a vortex streakpath. The incorrectly detected eddies in our method are mainly due to an unclear streakpath.

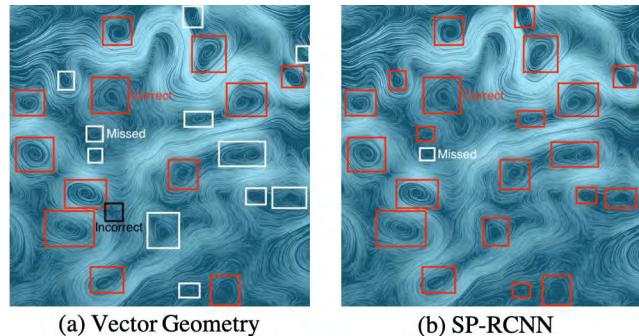
To further validate our approach, we compare SP-RCNN with two typical deep learning methods Faster RCNN [6] and SSD [8]. For a comprehensive comparison, we used 500 images, 2,000 images, and 3,500 images as our dataset,

**TABLE 3.** Validation results of VG (marked with 1) and SP-RCNN (marked with 2) in six days.

| Day          | day1      |        | day2      |        | day3      |        | day4      |        | day5      |        | day6      |        | Total        |        |
|--------------|-----------|--------|-----------|--------|-----------|--------|-----------|--------|-----------|--------|-----------|--------|--------------|--------|
|              | mesoscale | others | mesoscale    | others |
| $N_t$        | 14        | 9      | 17        | 8      | 17        | 5      | 16        | 8      | 15        | 8      | 13        | 9      | 92           | 47     |
| $N_{c1}$     | 11        | 1      | 15        | 2      | 12        | 2      | 13        | 3      | 13        | 2      | 11        | 2      | 75           | 12     |
| $N_{c2}$     | 14        | 8      | 17        | 8      | 16        | 5      | 16        | 7      | 15        | 8      | 13        | 8      | 91           | 44     |
| $SDR_1(\%)$  | 78.57     | 11.11  | 88.24     | 25.00  | 70.59     | 40.00  | 81.25     | 37.50  | 86.67     | 25.00  | 84.62     | 22.22  | <b>81.52</b> | 25.54  |
| $SDR_2(\%)$  | 100.00    | 88.88  | 100.00    | 100.00 | 94.12     | 100.00 | 100.00    | 87.50  | 100.00    | 100.00 | 100.00    | 88.88  | <b>98.91</b> | 93.62  |
| $N_{m1}$     | 3         | 8      | 2         | 6      | 5         | 3      | 3         | 5      | 3         | 6      | 2         | 7      | 17           | 35     |
| $N_{m2}$     | 0         | 1      | 0         | 0      | 1         | 0      | 0         | 1      | 0         | 0      | 0         | 1      | 1            | 3      |
| $MDR_1(\%)$  | 21.43     | 88.88  | 11.77     | 75.00  | 29.41     | 60.00  | 18.75     | 62.50  | 20.00     | 75.00  | 15.38     | 77.78  | <b>18.48</b> | 74.47  |
| $MDR_2(\%)$  | 0         | 11.11  | 0         | 0      | 5.88      | 0      | 0         | 12.5   | 0         | 0      | 0         | 11.11  | <b>1.09</b>  | 6.38   |
| $N_{e1}(\%)$ | 1         |        | 1         |        | 2         |        | 0         |        | 1         |        | 1         |        | 6            |        |
| $N_{e2}(\%)$ | 0         |        | 0         |        | 0         |        | 0         |        | 0         |        | 1         |        | 1            |        |
| $EDR_1(\%)$  | 4.35      |        | 4.00      |        | 9.09      |        | 0         |        | 4.35      |        | 4.55      |        | 4.32         |        |
| $EDR_2(\%)$  | 0         |        | 0         |        | 0         |        | 0         |        | 0         |        | 4.55      |        | 0.72         |        |

**TABLE 4.** The detection accuracy comparison between Faster-RCNN, SSD and SP-RCNN on different datasets.

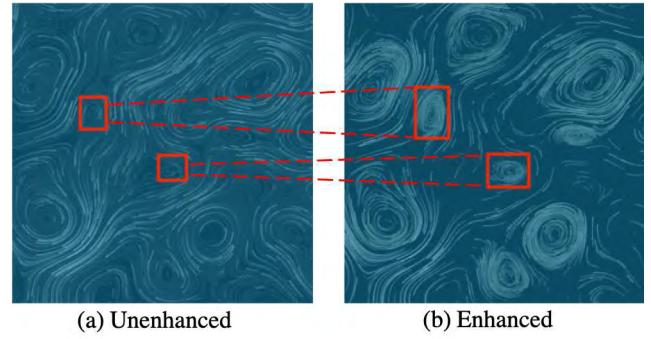
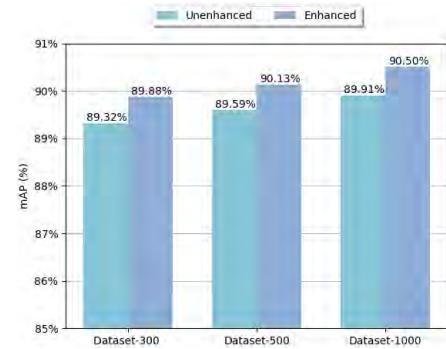
| Methods          | Dataset-500 | Dataset-2000 | Dataset-3500 |
|------------------|-------------|--------------|--------------|
| Faster R-CNN [6] | 86.75       | 87.03        | 87.52        |
| SSD [8]          | 71.28       | 74.14        | 75.50        |
| SP-RCNN          | 89.59       | 90.41        | <b>90.64</b> |

**FIGURE 8.** Detection results of two methods. The red boxes present the correctly detected eddies, the white boxes outline the missed eddies, and the black box indicates the incorrectly detected eddy.

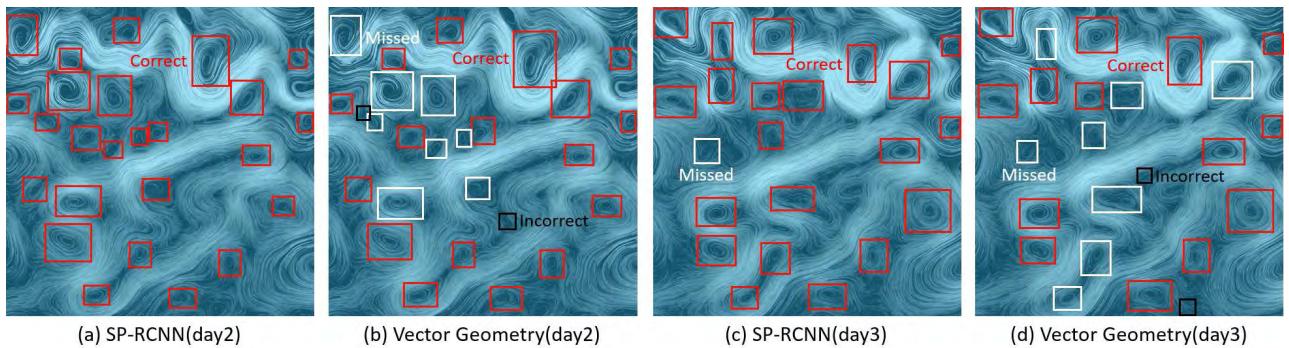
and they are compared by mAP(%). Table 4 shows the detection accuracy comparison on our dataset. This indicates that SP-RCNN is more competitive with an mAP of 90.64%, outperforming the SSD by approximately 25%. Compared with SSD, the Faster RCNN has a higher mAP, which corresponds with our expectation because SSD is not effective for small objects and Faster RCNN is more competitive for small objects.

### C. ENHANCED VISUALIZATION RESULT

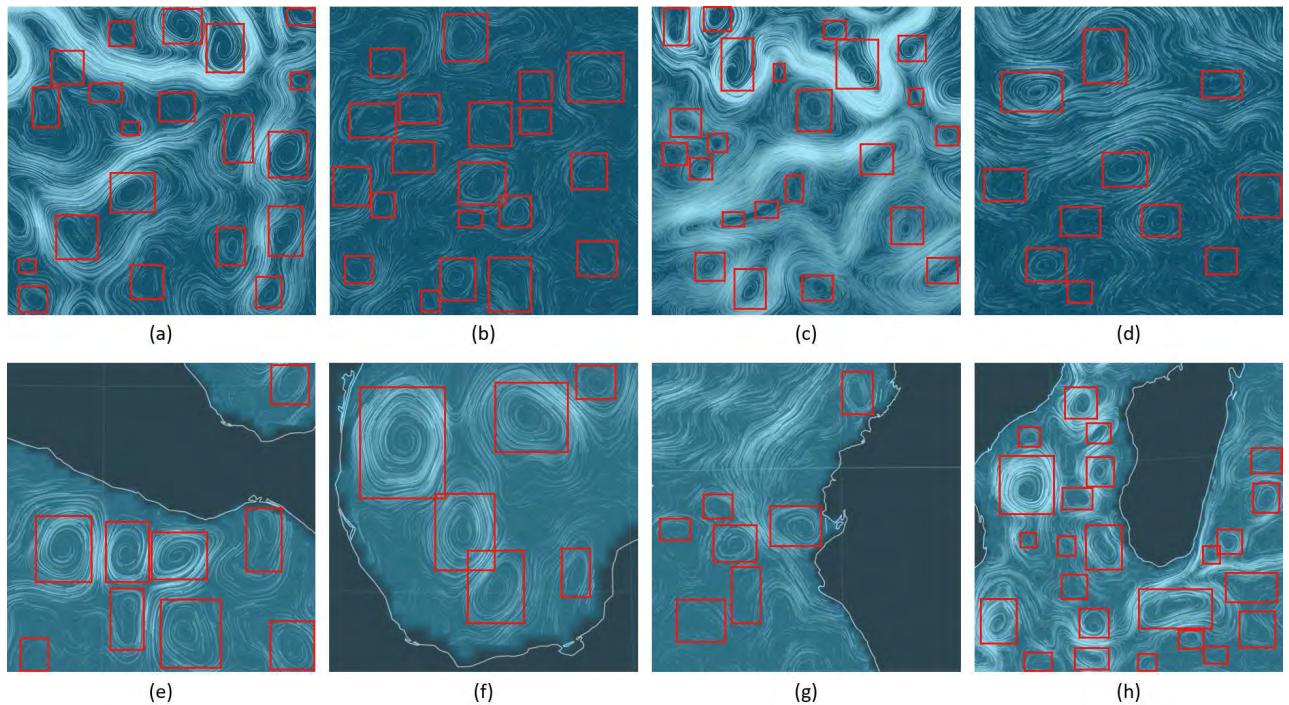
To prove the necessity for enhanced visualization, we selected a sparse streampath region for comparison, as shown in Figure 9. Using the detection results of our method,

**FIGURE 9.** Streampath enhancement. (a) Example of sparse streampath eddies. (b) Enhanced eddies.**FIGURE 10.** Comparison of detection accuracy between unenhanced dataset and enhanced dataset.

we enhance the eddies region in the result of the SP-Tran process to make it more visible and weaken the display of the noneddies region to highlight the eddies region. The result shows that the enhanced eddies are more easily recognized by a human, as shown in Figure 9(b), which corresponds with our expectation. For further comparison fairness, we mix the original dataset and enhanced dataset to re-train our network. For the accuracy of the results, we tested them on different datasets, as shown in Figure 10.



**FIGURE 11.** Detection results of our method and VG method in day 2 and day 3.



**FIGURE 12.** The eddies detected by our method in different scenarios.

The evaluation results show that training the enhanced eddy image can increase the detection accuracy (mAP) by approximately 0.6%.

## V. CONCLUSION

We present a deep learning-based method (SP-RCNN) for high-accuracy eddy detection, which shows a high accuracy with an mAP of 90.64%. To the best of our knowledge, there is no publicly available eddy image dataset based on flow field data. We have established a relatively more sufficient dataset, and our dataset will be published for future research. Our method is the first to apply deep learning techniques to eddy detection in flow field data. The experimental results demonstrate that SP-RCNN performs better than previous eddy detection methods. It can directly visualize the detection results on the image so that humans can detect eddies without professional knowledge. The enhanced eddy

images can also expand our dataset to make it more robust and further improve the accuracy of eddy detection. In the future, we plan to extend our approach to explore other eddy-based applications. Eddy tracking and eddy event finding may be potential directions.

## APPENDIX

### A. MACHINE CONFIGURATION

Our machine is a Desktop PC with an Intel i7-8700K CPU, 32GB RAM, and NVIDIA GeForce GTX 1080Ti GPU. We trained our dataset on Ubuntu 18.04.

### B. MORE RESULTS

Figure 11 shows the day2 and day3 results of our method and VG method. It can be seen that our method is more robust than VG. More results of our detected eddies are shown in Figure 12. It can be seen that the eddies can be detected in various complicated scenarios.

## REFERENCES

- [1] J. C. Mcwilliams, *The Nature and Consequences of Oceanic Eddies*. Washington, DC, USA: American Geophysical Union, 2013.
- [2] L. Liu, D. Silver, K. Bemis, D. Kang, and E. Curchitser, "Illustrative visualization of mesoscale ocean eddies," *Comput. Graph. Forum*, vol. 36, no. 3, pp. 447–458, Jun. 2017.
- [3] Y. S. Chang and Y.-G. Park, "Variation of flow properties during a collision event of two mesoscale eddies in the gulf stream region from numerical simulation," *Ocean Sci. J.*, vol. 50, no. 3, pp. 567–579, Sep. 2015.
- [4] D. Matsuoka, F. Araki, Y. Inoue, and H. Sasaki, "A new approach to ocean eddy detection, tracking, and event visualization—Application to the northwest pacific ocean," *Procedia Comput. Sci.*, vol. 80, pp. 1601–1611, Jun. 2016.
- [5] C. Dong, F. Nencioli, Y. Liu, and J. C. Mcwilliams, "An automated approach to detect oceanic eddies from satellite remotely sensed sea surface temperature data," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 6, pp. 1055–1059, Nov. 2011.
- [6] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2015, pp. 91–99.
- [7] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. Comput. Vis. Pattern Recognit.*, 2016, pp. 779–788.
- [8] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 21–37.
- [9] G. H. Vatistas, V. Kozel, and W. C. Mih, "A simpler model for concentrated vortices," *Exp. Fluids*, vol. 11, no. 1, pp. 73–76, Apr. 1991. doi: [10.1007/BF00198434](https://doi.org/10.1007/BF00198434).
- [10] B. Cabral and L. C. Leedom, "Imaging vector fields using line integral convolution," in *Proc. ACM 20th Annu. Conf. Comput. Graph. Interact. Techn. (SIGGRAPH)*, New York, NY, USA, 1993, pp. 263–270.
- [11] B. Kim and T. Günther, "Robust reference frame extraction from unsteady 2D vector fields with convolutional neural networks," *Comput. Graph. Forum*, vol. 38, no. 3, pp. 285–295, 2019.
- [12] J. Grant, G. Erlebacher, and J. O'Brien, "Case study: Visualizing ocean flow vertical motions using Lagrangian–Eulerian time surfaces," in *Proc. IEEE Vis. VIS*, Oct. 2002, pp. 529–532.
- [13] D. Banesh, J. A. Schoonover, J. P. Ahrens, and B. Hamann, "Extracting, visualizing and tracking mesoscale ocean eddies in two-dimensional image sequences using contours and moments," in *Proc. Workshop Visualisation Environ. Sci. (EnvirVis)*, K. Rink, A. Middel, D. Zeckzer, and R. Bujack, Eds. Aire-la-Ville, Switzerland: The Eurographics Association, 2017, pp. 43–47.
- [14] E. Williams, J. Sean, Hecht, W. Matthew, Petersen, Mark, Strelitz, Richard, Maltrud, and E. Mathew, "Visualization and analysis of eddies in a global ocean simulation," *Comput. Graph. Forum*, vol. 30, no. 3, pp. 991–1000, 2011.
- [15] A. Okubo, "Horizontal dispersion of floatable particles in the vicinity of velocity singularities such as convergences," *Deep Sea Res. Oceanogr. Abstracts*, vol. 17, no. 3, pp. 445–454, 1970.
- [16] J. Weiss, "The dynamics of enstrophy transfer in two-dimensional hydrodynamics," *Phys. D, Nonlinear Phenomena*, vol. 48, nos. 2–3, pp. 273–294, 1991.
- [17] I. A. Sadarjoen and F. H. Post, "Detection, quantification, and tracking of vortices using streamline geometry," *Comput. Graph.*, vol. 24, no. 3, pp. 333–341, 2000.
- [18] A. Chaigneau, A. Gizolme, and C. Grados, "Mesoscale eddies off Peru in altimeter records: Identification algorithms and eddy spatio-temporal patterns," *Prog. Oceanogr.*, vol. 79, no. 2, pp. 106–119, 2008.
- [19] F. Nencioli, C. Dong, T. Dickey, L. Washburn, and J. C. Mcwilliams, "A vector geometry-based eddy detection algorithm and its application to a high-resolution numerical model product and high-frequency radar surface velocities in the Southern California bight," *J. Atmos. Ocean. Technol.*, vol. 27, no. 3, p. 564, 2010.
- [20] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.
- [21] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1440–1448.
- [22] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," *CoRR*, vol. abs/1612.08242, pp. 1–9, Dec. 2016.
- [23] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," Apr. 2018, *arXiv:1804.02767*. [Online]. Available: <https://arxiv.org/abs/1804.02767>
- [24] R. Lguensat, M. Sun, R. Fablet, E. Mason, P. Tandeo, and G. Chen, "EddyNet: A deep neural network for pixel-wise classification of oceanic eddies," *CoRR*, vol. abs/1711.03954, pp. 1–5, Nov. 2017.
- [25] L. Deng, Y. Wang, Y. Liu, F. Wang, S. Li, and J. Liu, "A CNN-based vortex identification method," *J. Vis.*, vol. 22, no. 1, pp. 65–78, Feb. 2019.
- [26] G. Haller, A. Hadjighasem, M. Farazmand, and F. Huhn, "Defining coherent vortices objectively from the vorticity," *J. Fluid Mech.*, vol. 795, pp. 136–173, May 2016.
- [27] K. Franz, R. Roscher, A. Milioto, S. Wenzel, and J. Kusche, "Ocean eddy identification and tracking using neural networks," *CoRR*, vol. abs/1803.07436, pp. 1–4, May 2018.
- [28] P. Y. Le Traon, F. Nadal, and N. Ducet, "An improved mapping method of multisatellite altimeter data," *J. Atmos. Ocean. Technol.*, vol. 15, no. 2, pp. 522–534, 1998.
- [29] N. Ducet, P. Y. Le Traon, and G. Reverdin, "Global high-resolution mapping of ocean circulation from TOPEX/Poseidon and ERS-1 and -2," *J. Geophys. Res. Oceans*, vol. 105, no. C8, pp. 19477–19498, 2000.
- [30] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1–9.
- [31] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2818–2826.
- [32] C. Szegedy, S. Ioffe, A. A. Alemi, and V. Vanhoucke, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Proc. AAAI*, 2017, pp. 4278–4284.
- [33] K.-C. Peng and T. Chen, "A framework of extracting multi-scale features using multiple convolutional neural networks," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jun./Jul. 2015, pp. 1–6.
- [34] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: <https://arxiv.org/abs/1409.1556>
- [35] M. Stone, "Cross-validation choice and assessment of statistical predictions," *J. Roy. Stat. Soc. B, Methodol.*, vol. 36, no. 2, pp. 111–133, 1974.
- [36] Z. Q. J. Lu, "The elements of statistical learning: Data mining, inference, and prediction," *J. Roy. Stat. Soc. A, Statist. Soc.*, vol. 173, no. 3, pp. 693–694, 2010.



**XUE BAI** received the B.Eng. degree from Hainan University, in 2017. She is currently pursuing the master's degree with the School of Computer Science and Software Engineering, East China Normal University. Her main research interests include information visualization and computer vision.



**CHANGBO WANG** received the Ph.D. degree with the State Key Laboratory of CADCG, Zhejiang University, in 2006. He held a postdoctoral position at The State University of New York, in 2010. He is currently a Professor with the School of Computer Science and Software Engineering, East China Normal University. His research interests mainly include computer graphics, information visualization, and visual analytics. He is serving as the Young AE of Frontiers of Computer Science and a PC member of several international conferences.



**CHENHUI LI** received the Ph.D. degree from the Department of Computing, The Hong Kong Polytechnic University. He is currently a Lecturer with the School of Computer Science and Software Engineering, East China Normal University. He has received the CHENHUI Scholar Honor, in 2018. He was involved in the research of computer graphics and information visualization.