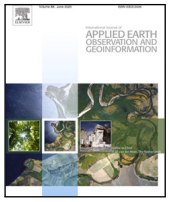
International Journal of Applied Earth Observations and Geoinformation 113 (2022) 102973

Contents lists available at ScienceDirect

International Journal of Applied Earth Observation and Geoinformation

journal homepage: www.elsevier.com/locate/jag



Oceanic mesoscale eddy detection and convolutional neural network complexity

Oliverio J. Santana [a,*], Daniel Hernández-Sosa [a], Ryan N. Smith [b]

[a] Institute of Intelligent Systems and Numeric Applications in Engineering, University of Las Palmas de Gran Canaria, 35001 Las Palmas de Gran Canaria, Spain
[b] Institute of Environment, Florida International University, Miami, FL 33199, USA

ARTICLE INFO

Keywords:
Oceanic mesoscale eddy
Deep learning
Convolutional neural network
Remote sensing
Satellite altimetry

ABSTRACT

Deep learning has drawn the attention of oceanographic researchers over the past few years, making the research community adopt computer vision techniques for oceanic mesoscale eddy detection on satellite altimetry gridded products. In this paper, we describe a convolutional neural network designed to detect eddies in satellite altimetry maps after being trained using segmentation masks provided by the OpenEddy detection algorithm. Against the current trend, in which increasingly complex neural networks are being proposed to address this problem, our design is relatively simple and yet provides competitive performance when compared to any of the previous deep learning methods reported in the literature. Furthermore, we show that our model is less sensitive to timely variations than the traditional models based on physical and geometric features defined by human experts, making it possible for our model to use the general data context to identify eddies that those traditional models would have missed. These results prove that overly complex neural network architectural designs are not required to solve the eddy detection problem on altimetry maps and generate a sufficiently good model for most practical applications in the field of marine sciences.

1. Introduction

Oceanic mesoscale eddies have a deep impact in diverse dynamic aspects of the marine systems. Due to their major role on physical, chemical, biological, and geophysical processes, the scientific community has devoted great efforts to develop eddy detection methods using different data sources. One of these sources are the satellite altimetry gridded products provided by entities like AVISO+ (Archiving, Validation, and Interpretation of Satellite Oceanographic Data). Fig. 1 shows an example of satellite altimetry map.

Early eddy detection methods relied on the selection of certain physical features to characterize vortex-like structures. Many approaches were based on the Okubo–Weiss parameter (Okubo, 1970; Weiss, 1991), a quantification of fluid rotation and deformation. The main disadvantage of this measurement is that it is highly sensitive to data noise, and thus it requires human experts to choose appropriate threshold values (Chelton et al., 2011).

To address this problem, systematic methods were designed on the basis of geometrical features (Mason et al., 2014; Faghmous et al., 2015). These methods are able to identify eddies by detecting local extrema and the surrounding contours around them. In this context, eddies can be divided into two categories: cyclonic eddies cause a depression in the ocean surface (the extreme is a minimum), while anticyclonic eddies cause a positive bump in the ocean surface (the extreme is a maximum). In either case, these algorithms are iterative in nature, and thus they still depend, in one way or another, on arbitrarily-chosen threshold values. Since eddy features have a large spatial variability, these arbitrary choices may cause eddy structures to be discarded or non-eddy structures to be wrongly identified as eddies.

The rise of deep learning has led to the proposal of different convolutional neural network (CNN) architectures for detection on objects from imagery. In the case of eddies, satellite altimetry data is processed to detect these structures in the same way as real images are processed to detect objects. Over the past few years, increasingly complex CNN architectures have been proposed to solve this particular problem.

Franz et al. (2018) use a traditional encoder–decoder CNN architecture to detect eddies. The encoder stage is composed of five building blocks with convolutional, batch normalization, and activation layers. Each block is followed by a max pooling layer that reduces the image size in order to identify eddy features in more detail. The decoder stage consists of similar blocks, but each preceded by an upsampling layer to bring the image back to its original size, making it possible to place the eddy features detected in the whole altimetry image. The CNN output is a probability value for each pixel that allows to identify it as belonging

* Corresponding author.
E-mail address: oliverio.santana@ulpgc.es (O.J. Santana).

https://doi.org/10.1016/j.jag.2022.102973
Received 9 February 2022; Received in revised form 3 August 2022; Accepted 8 August 2022
Available online 22 August 2022

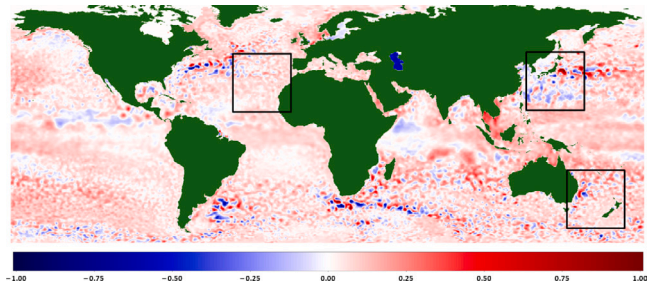


Fig. 1. Satellite altimetry data, measured in meters, corresponding to January 1st, 2013. Black boxes identify three ocean regions exhibiting high eddy activity. The Macaronesia region in the North Atlantic is integrated by the Canary Islands and other three volcanic archipelagos: Azores, Madeira, and Cape Verde. In the Pacific Ocean, the Kuroshio region spans the south and east of Japan, while the Eddy Ave extends off the coast of New South Wales in Australia.

or not to an eddy, although this method does not differentiate between cyclonic or anticyclonic behavior.

Lguensat et al. (2018b) propose EddyNet, a CNN architecture derived from U-Net (Ronneberger et al., 2015) that is able to detect eddies and distinguish them into the two categories. The U-Net architecture inserts paths between the encoding and decoding stages, permitting a combination of decoder's spatial placement information with encoder's more detailed features. The propagation of information from the initial layers makes it possible to train larger networks and leads to better performance. To train even deeper networks, Lguensat et al. (2018a) further improve this architecture replacing the traditional building blocks by residual units (He et al., 2016a), which extend the information flow by introducing additional paths within each block according to the principles of residual mappings.

This trend towards increasingly complex CNN architectures is continuing with the utilization of advanced architectures designed to parse real scenes and detect all kinds of elements in complex landscapes. Xu et al. (2019) adapt PSPNet (Zhao et al., 2017) for eddy detection. This architecture is based on a pyramid scene parsing module, which uses different pooling levels to examine the image at various resolutions and then fuses the features detected to gain a better understanding of the global context. Sun et al. (2021) take a similar approach, using the Xception architecture (Chollet, 2017) to obtain eddy features at multiple scales, which are provided to a pyramid pooling module upgraded with atrous convolutions (Chen et al., 2018) to enlarge the receptive field an collect more context information. Duo et al. (2019) built a CNN architecture based on RetinaNet (Lin et al., 2017), a single-stage object detection architecture. This architecture, called OEDNet, uses a residual network module to extract eddy features and a pyramid network module to refine the extracted features, which is followed by a third module devoted to pixel classification and positioning. Lu et al. (2020) also combine two different architectures: a high-resolution HR-Net architecture (Sun et al., 2019) for eddy detection and a CascadePSP module (Cheng et al., 2020) for border refinement. Fan et al. (2020) add another complexity factor: feeding complementary information to the CNN, such as ocean surface temperature and current flow velocity, to improve the CNN performance.

Nevertheless, while eddy detection involves some complexity, it should not be as challenging as analyzing a photographic image to recognize objects, animals, or people. As evidenced by histograms in Fig. 2, computed over a representative sample of 2013 year, eddy structures feature high values of circularity and solidity measurements. Circularity is a dimensionless shape roundness estimation that should give values between 0 and 1 (small irregular regions could have larger values). Solidity is a dimensionless shape compactness measure computed as the proportion of pixels in the region and in a convex hull around it, providing also 0 to 1 estimations. The high values obtained

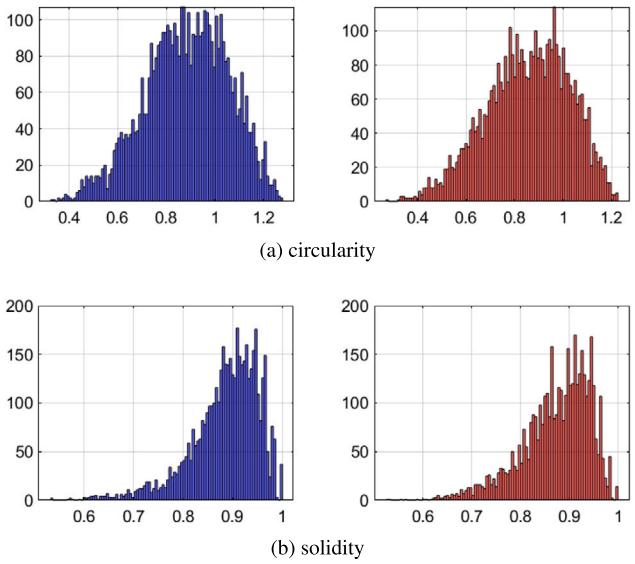


Fig. 2. Distribution of circularity and solidity measurements for eddies detected in the year 2013. Data is shown as separated histograms for cyclonic (blue) and anticyclonic (red) eddies. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

for both metrics imply that most eddies are relatively simple and homogeneous in shape.

On this basis, we present a novel CNN architecture that is simpler than those previously proposed, and yet provides better performance results. Section 2 introduces our CNN architecture proposal. Section 3 describes the experimental setup. Section 4 shows the performance results provided by our CNN architecture. Section 5 discusses the reasons behind the errors made by the proposed CNN. Finally, Section 6 presents our concluding remarks.

2. Methods

Eddy detection using a CNN over satellite altimetry images is configured as a supervised learning problem. The CNN is trained combining two datasets: satellite images and their corresponding segmentation masks. The goal of this training process is to maximize the value of a given performance metric. Once trained, the CNN proposes a segmentation map for each image in a separate test dataset.

We use satellite altimetry data provided by AVISO+. To generate these data, an Earth reference ellipsoid value is subtracted from the altitude above the surface measured by the satellite to obtain the sea surface height (SSH). Next, sea level anomaly (SLA) datasets are generated by comparing the geoid value and the mean dynamic topography (MDT) with the SSH, removing a multi-year averaged stationary component. These SLA datasets constitute the input provided to the CNN, which treats them as if they were images.

Training labels are generated as image segmentation masks by the OpenEddy algorithm (Faghmous et al., 2015). These segmentation masks classify each image pixel into three categories: pixels belonging to cyclonic eddies, pixels belonging to anticyclonic eddies, and pixels not belonging to any eddy (background pixels). Consequently, we adopt the eddy definition provided by Faghmous et al. (2015): the outermost closed-contour SLA containing a single extreme. This is a geometric definition and might introduce some uncertainty, since it is not related to any eddy physical property, but Faghmous et al. (2015) show that the OpenEddy algorithm detects more than 96% of the structures identified as eddies by domain experts.

In essence, eddies physically create a local sea level anomaly, and thus the use of SLA datasets allows for a higher precision in

the structure detection and yields better performance than using SSH datasets (Santana et al., 2020). Nevertheless, it is important to keep in mind that SLA is not necessarily the best option in all cases. When a recurrent eddy exists for long periods of time, it would likely be present in the MDT and, as a consequence, may alter the SLA value and lead the model to wrong detection results, since SLA represents only the variable part of the sea surface. In such an environment, it may be preferable to train the CNN using absolute dynamic topography (ADT) data, which adds MDT and SLA to combine the mean ocean behavior with its temporal variability (Pegliasco et al., 2021).

2.1. Ocean area selection

To conduct our study, we have selected ocean areas widely known for their intense eddy activity. All these areas are delimited by a 32° square. Since the resolution of the satellite altimetry images is 0.25°, the dimensions of the images are 128 pixels high and 128 pixels wide.

For the training process, we selected the Macaronesia region in the North Atlantic Ocean, location of the Canary Islands' eddy corridor (Sangrà et al., 2009), which is shown in Fig. 3.a. This particular eddy system is originated by the interactions between the local ocean current, the Trade Winds and the islands bodies, constituting a main source of westward propagating, long-lived eddies. A rich variety of ocean structures interact in this region, mixing fronts and filaments emerging from Northwest Africa coast upwelling.

To assess the generalization skill of our model, we also evaluate its performance in other regions of the ocean. First, we selected a 128 × 128 pixel region centered on Kuroshio that is shown in Fig. 3.b. This region exhibits a complex interaction between water masses, originating cyclonic and anticyclonic eddies, with an important warming effect on the south and south-east coasts of Japan (Cheng et al., 2014), resulting in remarkable biological and climatic implications.

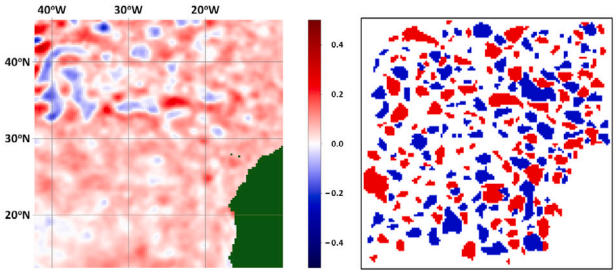
Additionally, since both the Macaronesia and Kuroshio regions are in the Northern Hemisphere, we have also chosen a region from the Southern Hemisphere shown in Fig. 3.c. This is a quite unusual region off the coast of New South Wales, Australia, which is interesting because it has two strong boundary currents: the Leeuwin in the west and the East Australian Current in the east. Eddies peel off from those currents, and form persistent energetic structures, justifying the area been nicknamed Eddy Ave (Everett et al., 2012).

Fig. 4 shows the distribution of the eddy surface measures during the year 2013 in the three regions selected for this work. As can be observed, the vast majority of structures fall in the interval between 3×10^4 km^2 and 2×10^5 km^2 areas. The lower limit, marked by a vertical red line in the figure, corresponds to the lower range of the mesoscale, i.e. 50 km in diameter eddies, while the higher limit is equivalent to 200 km in diameter structures.

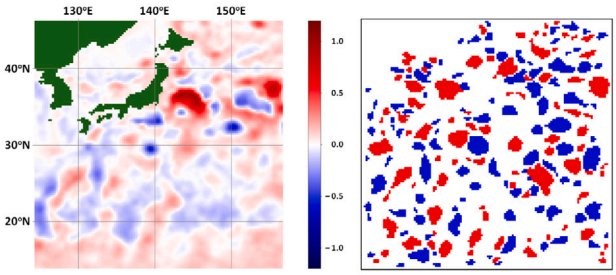
2.2. Data augmentation

Training a CNN requires large amounts of data. In cases where there is not enough data available, it is common to implement strategies to augment the dataset. The work by Duo et al. (2019) presents a good example, since they only had 48 images manually labeled by experts to train their CNN proposal. These images were standardized and then rotated at different angles using bilinear interpolation. Through these rotations, 11 additional versions of each image were generated, thus raising the total of 48 images to 576. As a last step, a new version of each image was generated after adding Gaussian noise, yielding a total of 1152 images.

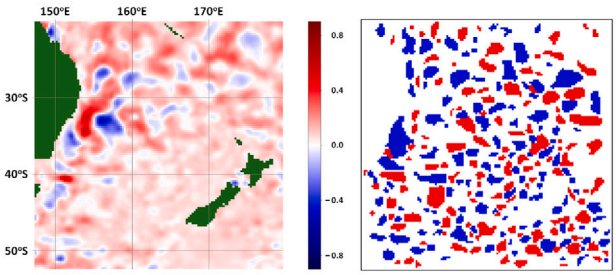
This method is particularly helpful when there is a limited number of available images to train a CNN, but it should be noted that the images generated by this method are related to each other, thereby reducing the variability of the training dataset. Using an automated algorithm as OpenEddy (Faghmous et al., 2015) to label the images



(a) Macaronesia region in the North Atlantic Ocean



(b) Kuroshio region in the Northwest Pacific Ocean



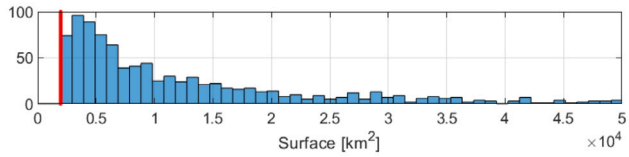
(c) New South Wales region in the Southwest Pacific Ocean

Fig. 3. Ocean regions under study. The image on the left shows the SLA data measured in meters. The image on the right shows the segmentation masks, cyclonic eddies in blue and anticyclonic eddies in red. Data refers to January 1st, 2013. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

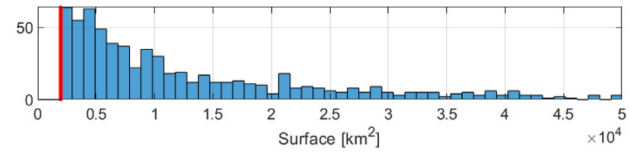
provides much larger datasets to train the networks, making it possible for the training process to learn from a wide variety of real data, independent of each other, instead of artificially generated data.

However, despite the large amount of eddy data available, it is common in the literature to focus on a specific area, such the ones commented above, and train the CNN on it. Focusing on a particular area means limiting the amount of data available for training, which leads to lower detection performance. To address this issue, we propose to train the CNN using larger regions and then clip the region of interest from the results provided by the CNN.

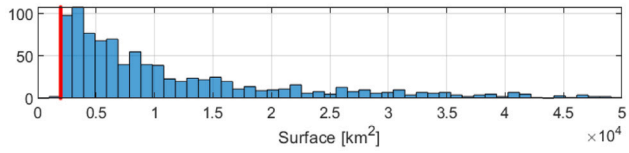
Fig. 5 shows a general outline of our processing pipeline. Instead of training our model using the 128 × 128 pixel Macaronesia region, we train the network using a 256 × 256 pixel region, that is, a region four times larger that practically encompasses the entire North Atlantic Ocean. For test purposes, our model proposes a segmentation for the whole 256 × 256 pixel North Atlantic Ocean region and clips the Macaronesia region from it. The applied procedure is similar for the Kuroshio and New South Wales regions, though proposing a 256 × 256 pixel segmentation respectively for the Northwest and Southwest Pacific Ocean regions.



(a) Macaronesia region in the North Atlantic Ocean



(b) Kuroshio region in the Northwest Pacific Ocean



(c) New South Wales region in the Southwest Pacific Ocean

Fig. 4. Distribution of the eddy surface area measured during the year 2013. The vertical red line marks the lower range of the mesoscale. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

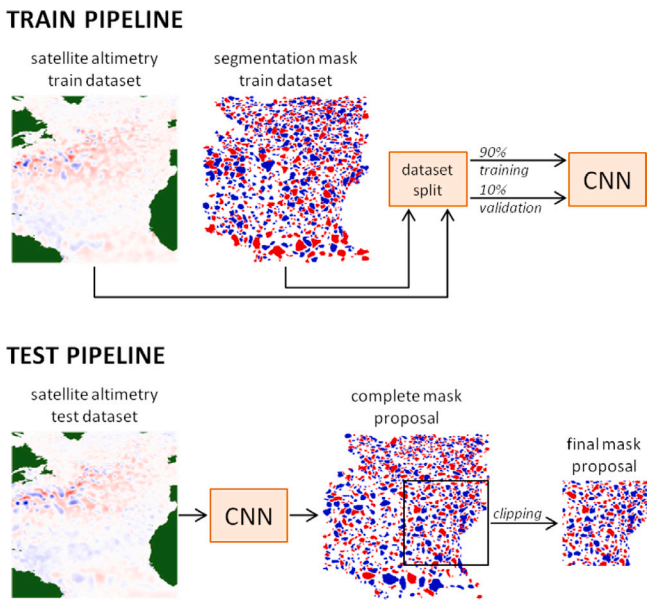


Fig. 5. Processing pipeline proposed for eddy detection.

2.3. Residual eddy detection network

Our proposed CNN architecture, shown in Fig. 6, is an encoder–decoder design based on residual blocks. This residual eddy detection network (REDN) is an evolution of our previous work (Santana et al., 2020) and thus we call it REDN2 to distinguish it from our earlier design, subsequently named REDN1.

The REDN2 encoding stage consists of three convolution blocks, followed by a max pooling layer that reduces the image resolution in order to allow the CNN to capture more contextual details. This layer is followed by a central stage, with another three convolution blocks. Finally, an upsampling layer restores the original resolution of the image, enabling feature localization by the decoding stage, which is formed again by three convolution blocks.

In line with the structure of the U-Net architecture (Ronneberger et al., 2015), we have established a link between the encoding stage and the decoding stage, enabling the latter to generate more accurate outputs by combining the data arriving from the central stage with the higher resolution information provided by the encoding stage. It is interesting to note that, while the architectures designed to process real-life images typically have multiple levels of max pooling and upsampling layers, our design has only one, preserving the traditional u-shape that gives the architecture its name. The aspect of an eddy captured from satellite sensors is much simpler than the aspect that an average day-to-day object, taken in perspective in a photorealistic image, might have. As a consequence, we have found that further reducing the image resolution does not provide any benefit and may even be detrimental to the model performance.

The convolution blocks that form each stage consist of a sequence of convolution, batch normalization, and activation layers. Each convolution layer applies a 3×3 bidimensional convolution window to the input images and includes 128 filters. We have found that increasing the number of convolution filters provides no significant benefit, as we discuss in Section 4.1. All the activation layers apply a ReLU function with the exception of the last one, which applies a softmax function to provide the final result.

The layers in each convolution block are organized to fit a residual mapping directly, with respect to the input layer, introducing feedback points within each block that allow the architecture to re-process information from precedent layers. In order to design our residual mapping, we began from the typical ResNet mapping (He et al., 2016a) and added a 1×1 convolution in the shortcut path to compensate the resolution changes between the different stages. After considering different alternatives (He et al., 2016b), we displaced the last batch normalization layer after the addition layer, enabling it to affect both the convolutions on the main path and the shortcut.

2.4. Comparison baseline networks

For comparison purposes, we have selected two eddy detection networks clearly established in the literature: EddyNet (Lguensat et al., 2018b) and PSPNet (Xu et al., 2019). In addition, we include our previous REDN1 proposal (Santana et al., 2020) to distinguish it from our new REDN2 proposal. Fig. 7 shows the architectural design of these three networks, making it possible to compare them in terms of layout and complexity.

EddyNet (Lguensat et al., 2018b) is built with basic blocks formed by two sets of convolution, batch normalization and activation layers but, unlike our proposal, these blocks do not follow a residual mapping. Another difference is that, at the end of each EddyNet block, a dropout layer (Srivastava et al., 2014) randomly deactivates 20% of the neurons to prevent training overfitting. While Eddynet benefits from this approach, there are no dropout layers in our REDN2 architecture because we have found that they do not improve the results provided by our proposal. On the other hand, although EddyNet originally used convolutions with 32 filters, we have found that increasing the number of filters improves its performance, and thus we have configured EddyNet convolutions with 128 filters to ensure a fair comparison with our proposal.

Like REDN2, EddyNet is organized around two paths, one for encoding and one for decoding, with connections linking them (Ronneberger et al., 2015). However, the number of max pooling and upsampling layers is higher, having fewer convolution layers between each of them. The encoding path consists of three blocks separated from one another by a max pooling layer that halves the image dimensions. After a central block, the decoding path is formed by three other blocks separated by upsampling layers, recovering the original size of the

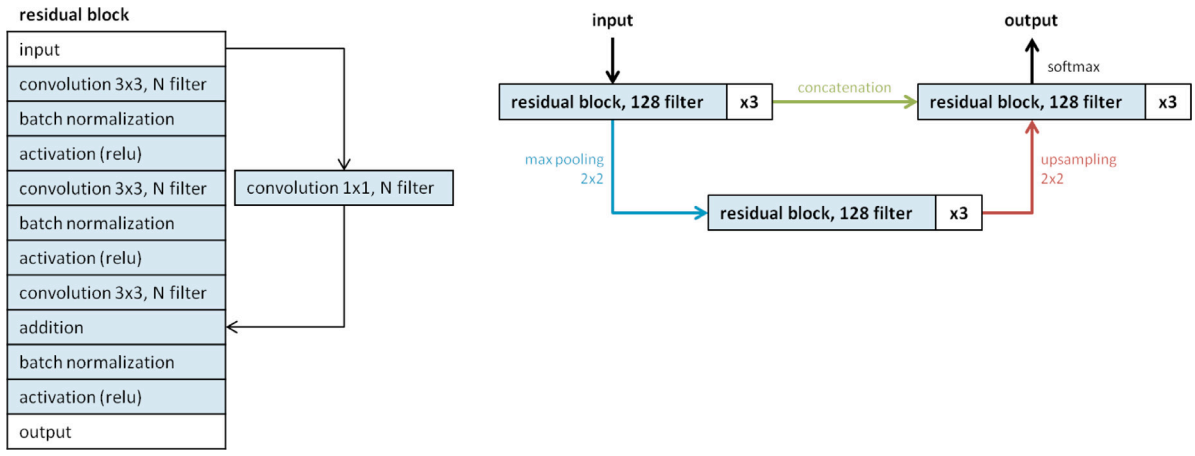


Fig. 6. Architectural design proposed for REDN2.

image. For this purpose, the convolution layers of these three decoding blocks are replaced by transposed convolutions.

Our previous REDN1 architecture (Santana et al., 2020) is also u-shaped (Ronneberger et al., 2015) and has two interconnected paths, one for encoding and one for decoding. The overall REDN1 layout is halfway between EddyNet and our new REDN2 proposal, as there are only two pairs of max pooling and upsampling layers, while the number of convolution layers between them has been increased. In fact, there are two consecutive residual blocks at each stage of the encoding and decoding paths, while EddyNet has only one and REDN2 has three.

The REDN1 building blocks follow a residual mapping, but they differ from our current approach in several aspects. First, there are only two sets of convolution, batch normalization and activation layers, while there are three in our new proposal. In addition, the REDN1 batch normalization layers are placed after the activation layers, while in our new design they are placed before, as indicated in the original proposal (Ioffe and Szegedy, 2015). Finally, the 1×1 convolution layer of the shortcut path is not present in all residual blocks but only in those blocks that are just behind an upsampling layer.

The PSPNet architecture (Xu et al., 2019) uses ResNet (He et al., 2016a) as backbone. This CNN has a first block formed by a 3×3 convolution layer with 64 filters, followed by a batch normalization layer and an activation layer using a classical ReLU function. These layers are usually followed by a max pooling layer but, in line with our findings for REDN2, we have concluded that removing it provides better results for this particular problem.

The first block is followed by a long sequence of residual blocks. As in REDN2, these blocks consist of three sets of convolution, batch normalization and activation layers. However, in contrast to our proposal, these three sets of layers are not homogeneous. Only the second convolution applies a 3×3 filter, while the first and third convolutions apply a 1×1 filter. In addition, the number of filters is not the same in all the convolution layers, as the third convolution multiplies the number of filters of the others by a factor of four.

As suggested by Xu et al. (2019), we have improved the ResNet design using dilated convolutions (Yu and Koltun, 2015). In particular, we have replaced the second 3×3 convolution layer in each residual block by a dilated convolution to expand the receptive field without losing image resolution. In addition, given the large number of layers in the network, we have replaced the functions of the first two activation layers of each block by leaky ReLU functions in order to avoid the vanishing gradient problem during training.

The ResNet residual blocks are distributed in four groups: a group of 3 blocks with 64-filter convolutions, a group of 4 blocks with 128-filter convolutions, a group of 6 blocks with 256-filter convolutions, and a final group of 3 blocks with 512-filter convolutions. The residual block shortcut path has a 1×1 convolution like REDN2, however, it is not present in all blocks but only in the first of each group. Another difference is that the shortcut path convolution has four times the number of filters to match the last convolution of the block. In addition, the shortcut path convolution is followed by its own batch normalization layer, thus the shortcut path ends after the batch normalization layer of the main block instead of before.

After this sequence, there is a final block consisting of a 512-filter 3×3 convolution, a batch normalization layer, and a ReLU activation function, resulting in a total of 50 convolutions in the architecture; hence it is often referred to as ResNet50. Xu et al. (2019) used another variant, called ResNet101, which has 23 blocks in the fourth group resulting, as the name suggests, in a total of 101 convolutions.

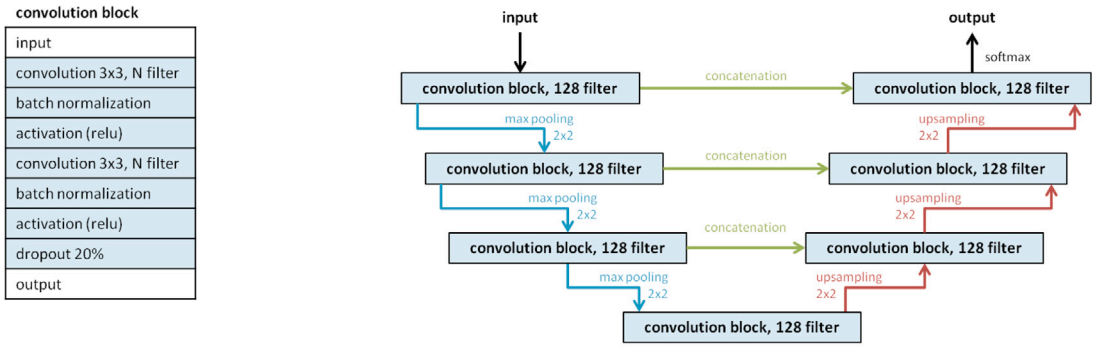
PSPNet is created from this architecture by introducing a pyramid pooling module (Zhao et al., 2017) before the final block. This module scans the images processed by ResNet at four different scales. The coarsest path applies 1×1 global average pooling and the other paths applies average pooling. We have achieved the best results using 2×2, 4×4, and 8×8 scales for these average pooling paths. After the pooling operation, a convolution layer restores the original size of the image and then all the context information obtained at different scales is gathered by merging the four paths before generating the final segmentation map.

2.5. Performance metric

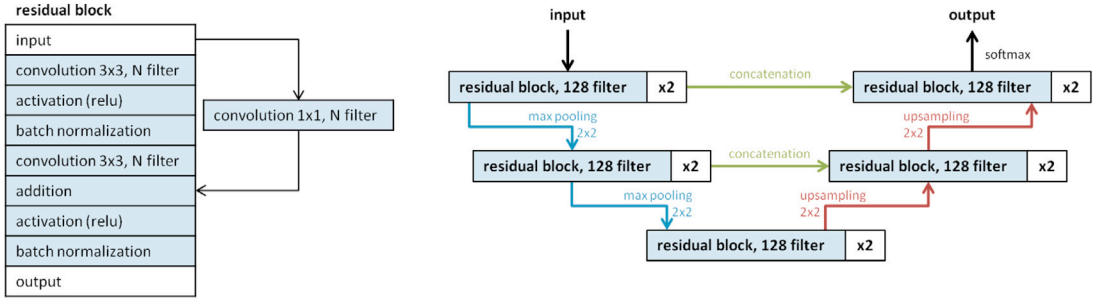
Our performance metric of choice is the F1 score (Chinchor, 1992), a popular statistical measure for classification problems that is calculated as the harmonic mean of precision and recall. Accuracy is a misleading metric for eddy detection methods (Santana et al., 2020) because its value is distorted due to the fact that even a simple CNN is able to easily classify most background pixels, precisely those we are not interested in.

In this context, the F1 score measures the level of intersection between the set of pixels labeled as eddies by the ground truth segmentation masks and the set of pixels identified as eddies by the CNN model (Santana et al., 2020). A value of one would mean that the image segmentation proposed by the CNN model matches perfectly the segmentation provided by the OpenEddy algorithm, while a value of zero would mean that they do not match at all.

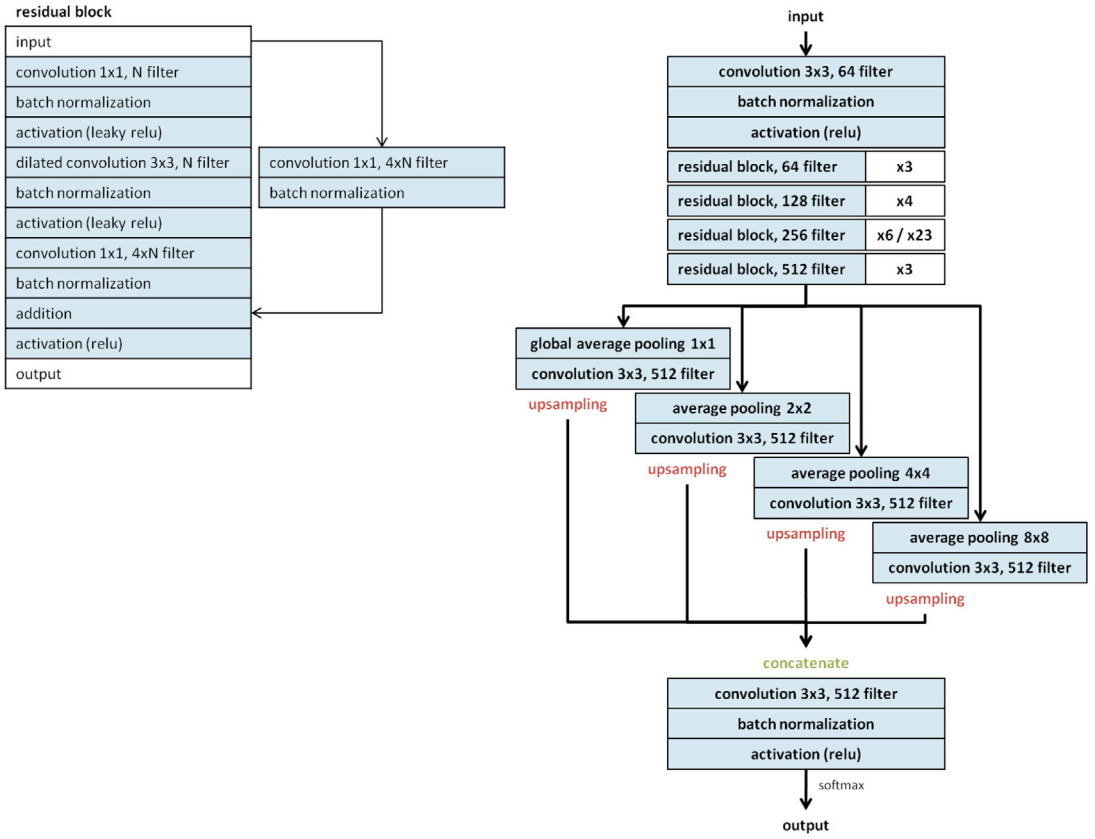
It is important to note that eddy detection is a highly unbalanced problem. Table 1 shows the percentage of pixels per category in the Macaronesia dataset and in the whole North Atlantic dataset for the year 2013. Most of the pixels correspond to the background, while the number of pixels belonging to each eddy category are very similar. We checked different ocean areas and different years and found that the ratio is always close to 70-15-15.



(a) EddyNet



(b) REDN1



(c) PSPNet

Fig. 7. Architectural design of the CNN comparison baselines.

Due to this imbalance, we train all the evaluated CNN architectures to maximize a weighted average of the F1 score metric. A weight inversely proportional to its frequency of occurrence is assigned to each category (Lguensat et al., 2018b), so the eddy categories have much more contribution in the calculation of the CNN loss function value than the background.

Table 1
Percentage of pixels per category for 2013.

	Macaronesia	North Atlantic
Background	68.84%	70.27%
Cyclonic eddy	15.87%	15.08%
Anticyclonic eddy	15.29%	14.65%

3. Experimental setup

We have implemented our residual CNN architecture using Python 3.6.9 and TensorFlow 2.3.1. The model was trained in an Ubuntu 18.04 system with four GeForce GTX 1080Ti GPUs under CUDA 10.0 v7.6.4. The CNN training process was performed using the Adam optimization algorithm (Kingma and Ba, 2015) and the He initialization method (He et al., 2015). The training dataset features 4749 images, one for every day between 2000 and 2012. These images are randomly divided into two sets, with 90% of the images used for training and the remaining 10% used for validation. The test dataset is made of the 365 images corresponding to each day of the year 2013. The example SLA images and segmentation masks presented throughout this paper correspond to the first day of the test dataset: January 1st, 2013.

At the beginning of the training process, the learning rate is set to an initial value of 0.001; this value is then divided by 2 whenever 10 epochs are completed without detecting any significant improvement in the validation loss function. In this sense, an improvement is considered not significant when the variation is below 0.001. The training process ends after 25 consecutive epochs have been completed without detecting any improvement in the loss function; at this point the value reached is considered optimal.

4. Results

To evaluate our REDN2 proposal both in terms of performance and complexity, we compare it with four different eddy detection networks. As representatives of similar sized networks we have chosen EddyNet (Lguensat et al., 2018b) and our own REDN1 (Santana et al., 2020). To represent the current trend towards proposing large networks, we have chosen two variants of PSPNet (Xu et al., 2019), PSPNet50 and PSPNet101, which respectively use ResNet50 and ResNet101 as their backbone.

4.1. Convolution filters

Bidimensional convolution layers are the essential component of a CNN. Each layer learns a series of output filters that are applied on the image to extract the relevant information for the detection process. Table 2 shows the F1 score obtained by our REDN2 model as we change the number of convolution filters. Increasing the number of filters leads to a significant performance improvement for the lower filter numbers, but this improvement becomes progressively smaller as we reach larger filter numbers.

Increasing the number of filters also involves increasing the number of trainable parameters of the model. This aspect is relevant, since convolution filters are the vast majority of the parameters that the model needs to learn. As shown in Table 2, increasing the number of filters to the next power of 2 roughly implies increasing the number of parameters by a factor of 4. Based on these results, we consider that the optimal number of convolution filters is 128, since the improvement obtained by increasing the number of convolution filters to 256 is limited but involves a leap from 4 million trainable parameters to 16 million.

Table 2
F1 score value for our REDN2 model configured with different numbers of convolution filters. F1 score values correspond to the Macaronesia region, shown separately for each eddy category. The last column shows the total number of trainable parameters in the model.

Filters	Cyclonic	Anticyclonic	Parameters
16	0.9366	0.9329	65,584
32	0.9518	0.9512	260,192
64	0.9583	0.9562	1,036,480
128	0.9625	0.9605	4,137,344
256	0.9640	0.9629	16,532,224

Table 3
Summary of the CNN models evaluated.

	Layers	Parameters	Training time
EddyNet	62	2.364.416	0 h 59 m
REDN1	82	3.168.768	1 h 15 m
REDN2	106	4.137.344	1 h 46 m
PSPNet50	192	46.570.432	26 h 09 m
PSPNet101	362	65.562.560	41 h 22 m

Table 4
F1 score value obtained for the Macaronesia region; results are shown separately for each eddy category.

	Cyclonic	Anticyclonic
EddyNet	0.9307	0.9268
REDN1	0.9509	0.9491
REDN2	0.9625	0.9605
PSPNet50	0.9597	0.9581
PSPNet101	0.9605	0.9597

4.2. Model complexity

The CNN models evaluated differ widely in their layout. Table 3 presents a summary of various criteria that make it possible to compare the complexity of these designs: the number of layers, the number of trainable parameters, and the total training time. The number of layers and the number of trainable parameters determine how much memory is required to train the networks, while the training time points out the computational cost involved in tuning the structural hyperparameters to optimize CNN performance.

There are clearly two distinct groups. The first three networks require a moderate amount of memory and can be trained in a manageable amount of time, thus allowing many variations to be tested in order to optimize their design. On the other hand, the two PSPNet variants require a much larger amount of memory, over 10 times higher. It should be kept in mind that, in current computer systems, we are not referring to the ordinary system memory but to the memory of the GPUs where the CNN will be trained, which nowadays is much more expensive and difficult to obtain. With a training time of one to two days, optimizing these models becomes a much more complex and time-consuming task.

4.3. Performance comparison

Table 4 shows the F1 score achieved by all the CNN models under evaluation. To enable a direct comparison between all of them, the REDN2 F1 score values presented in this table have been obtained without using our data augmentation methodology, that is, all these networks have been trained using the data from the 128 × 128 pixel Macaronesia region. As can be observed, REDN2 provides the best results for both eddy categories, outperforming even much larger networks like the two PSPNet variants.

To assess the generalization skill of the models trained in the Macaronesia region, Table 5 shows the results obtained in the Kuroshio region and Table 6 shows the results obtained in the New South Wales region. Again, REDN2 provides the best results for cyclonic eddies and

Table 5
F1 score value obtained for the Kuroshio region; results are shown separately for each eddy category.

	Cyclonic	Anticyclonic
EddyNet	0.8906	0.9008
REDN1	0.9195	0.9222
REDN2	0.9327	0.9354
PSPNet50	0.9305	0.9313
PSPNet101	0.9319	0.9387

Table 6
F1 score value obtained for the New South Wales region; results are shown separately for each eddy category.

	Cyclonic	Anticyclonic
EddyNet	0.8968	0.9111
REDN1	0.9255	0.9305
REDN2	0.9358	0.9430
PSPNet50	0.9339	0.9392
PSPNet101	0.9330	0.9443

Table 7
F1 score value for the REDN2 model trained with and without our data augmentation method; results are shown separately for each eddy category.

	Cyclonic	Anticyclonic
Without data augmentation	0.9625	0.9605
With data augmentation	0.9767	0.9761

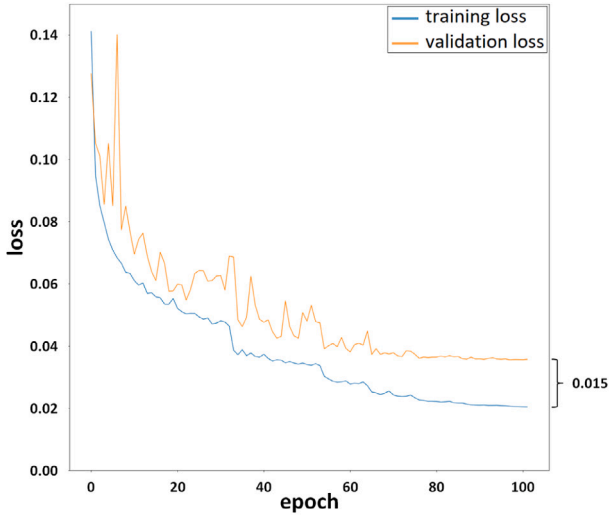
it is outperformed only by PSPNet101 for anticyclonic eddies, though it is a modest improvement considering that this CNN has almost 16 times more trainable parameters than our model. These results confirm that a relatively simple CNN tuned for eddy detection can achieve comparable or even better results than much larger and complex networks.

4.4. Performance with data augmentation

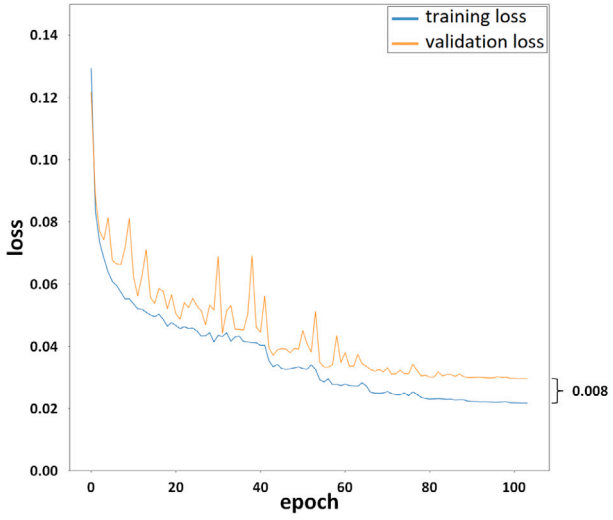
Once we have established that our REDN2 model provides competitive state-of-the-art performance, we proceed to evaluate our data augmentation method. Table 7 compares the performance of the REDN2 model with and without data augmentation. The first line of the table shows the performance of REDN2 without data augmentation, that is, trained using the Macaronesia 128 × 128 pixel dataset. The second line of the table shows the performance of REDN2 with data augmentation: the training process used the 256 × 256 pixel dataset covering the whole North Atlantic Ocean and the 256 × 256 pixel segmentation proposed by the model is clipped to match the 128 × 128 pixel Macaronesia region under study. The improvement is significant, making it possible for REDN2 to provide values over 0.976 for both eddy categories.

These results illustrate the importance of using as much data as possible to train CNN architectures. Fig. 8 shows that, when training our model with the 128 × 128 pixel Macaronesia dataset, the deviation between the training loss and the validation loss is higher. The model trained with the 256 × 256 pixel North Atlantic dataset provides better results, but not only because it has been trained with more data; besides, it has been trained in a wider region, preventing the model from focusing on the specific features of the eddies in a particular ocean area, and thus being able to generalize the learned model better.

The REDN2 model with data augmentation also provides the best generalization. Table 8 shows that the results obtained for both the Kuroshio and New South Wales regions are almost equally good, being closer to the values obtained for the Macaronesia region than in any of the previous models. It is interesting to note that both regions are in different hemispheres. Eddy rotation differs in the two hemispheres due to the Coriolis effect: cyclonic eddies rotate counterclockwise and anticyclonic eddies rotate clockwise in the northern hemisphere; their behavior is the opposite in the southern hemisphere. However, as the presented results show, this fact has no relevant impact on the performance of our model.



(a) without data augmentation



(b) with data augmentation

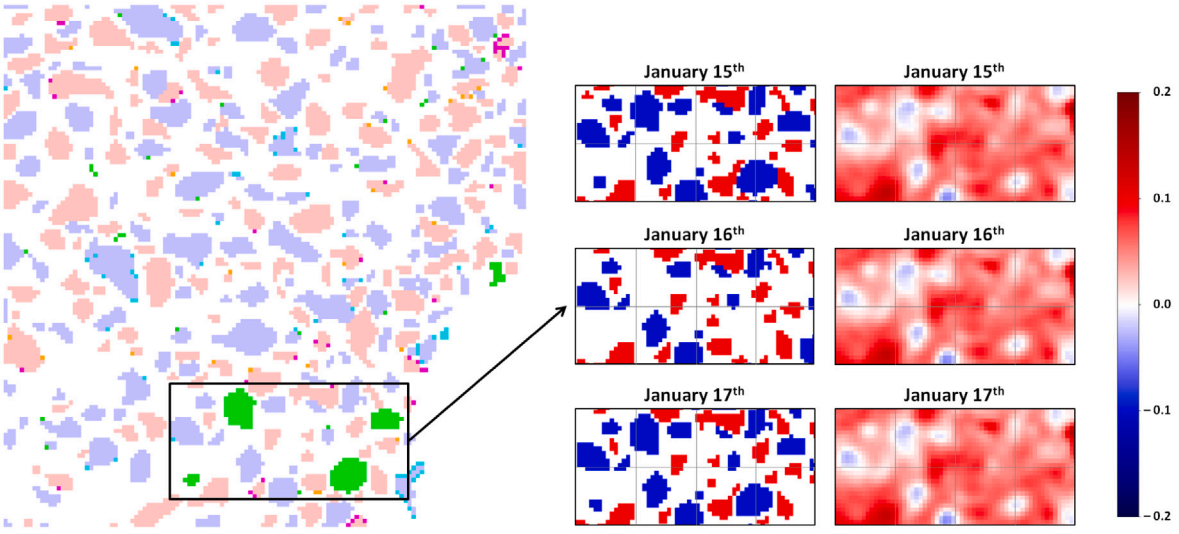
Fig. 8. Learning curves for our CNN model trained using (a) the 128 × 128 pixel Macaronesia dataset and (b) the 256 × 256 pixel North Atlantic dataset. The blue line shows the training loss value and the orange line shows the validation loss value. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

5. Discussion

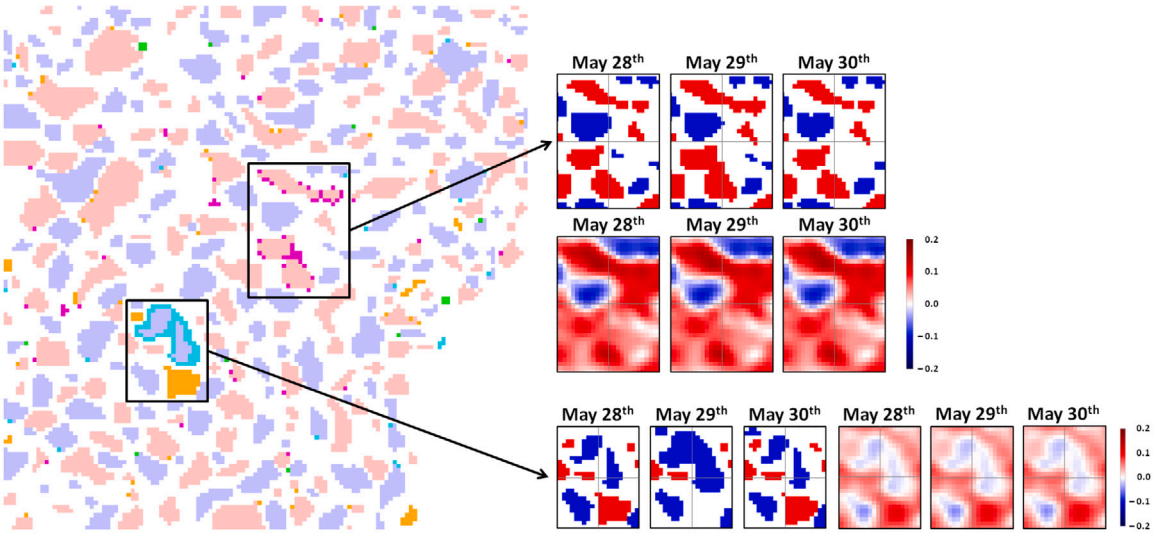
Our REDN2 architecture consistently provides F1 score values over 0.970 for all the ocean regions studied. To the best of our knowledge, no better results have been reported for any previously proposed eddy detection CNN in the literature. This high performance implies that there is little room for improvement. Consequently, it is important to analyze the reasons behind the remaining errors to determine the best way to address them.

5.1. Classification error analysis

Overall, the errors can be divided into false negatives (pixels classified as background but identified as eddies in the ground truth segmentation) and false positives (pixels classified as eddies but identified as background by the ground truth segmentation). Table 9 shows



(a) January 16th



(b) May 29th

Fig. 9. Classification errors in the segmentation proposed by our CNN model for the Macaronesia region. Correctly classified pixels are shown faded, cyclonic structures in blue and anticyclonic in red. False negatives are shown in cyan (cyclonic) and magenta (anticyclonic), while false positives are shown in green (cyclonic) and orange (anticyclonic). The small boxes on the right show the ground truth segmentation for a particular area on different days and the corresponding SLA values measured between −0.2 m and 0.2 m. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 8
F1 score value for the REDN2 model using data augmentation at different ocean areas; results are shown separately for each eddy category. The value in parentheses shows the difference with the results obtained for the Macaronesia region.

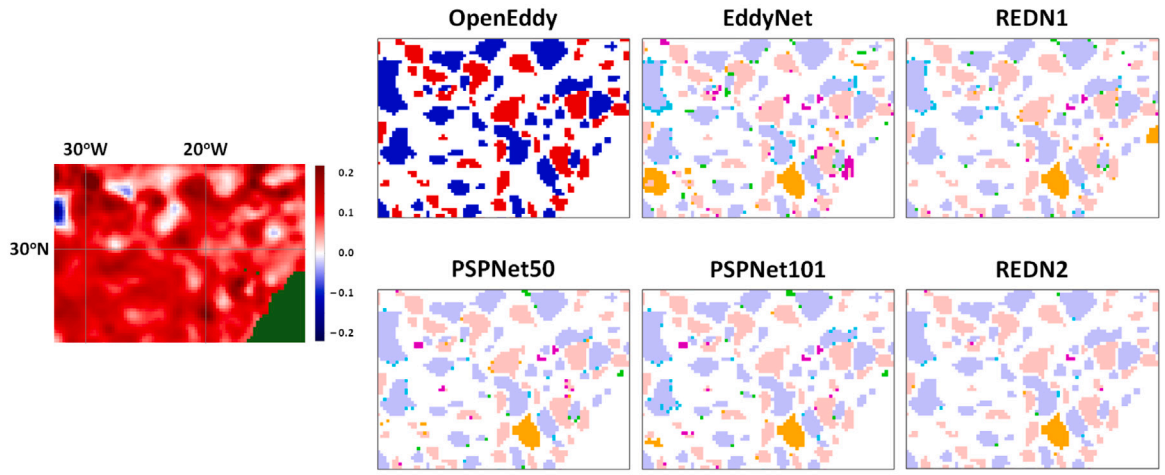
	Cyclonic	Anticyclonic
Kuroshio	0.9707 (−0.0060)	0.9704 (−0.0057)
New South Wales	0.9703 (−0.0064)	0.9731 (−0.0030)

the distribution of these errors in the three ocean regions we have studied.

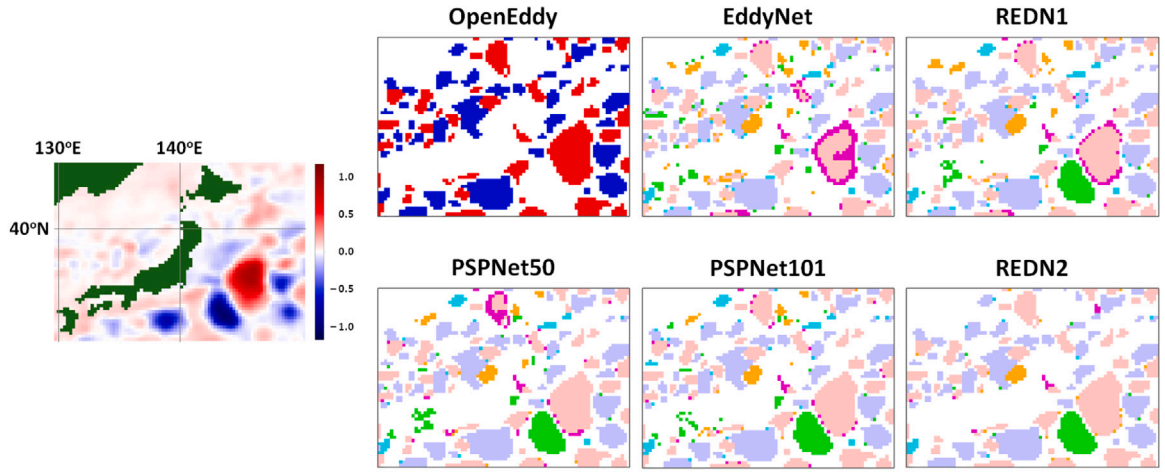
We divide the misclassified pixels into four categories: firstly we divide them into false positives and false negatives and then into cyclonic and anticyclonic eddies. As a general rule, between 24% and 32% of the misclassified pixels are isolated errors, that is, none of the eight neighboring pixels belong to the same error category. These isolated errors are small discrepancies in eddy edge detection that will have little relevance to oceanographic research.

Most of the misclassified pixels belong to a cluster where there are two or more pixels of the same category in close contact. We have observed two different behaviors in these clusters. On the one hand, false negatives tend to arrange themselves as edge clusters surrounding eddies of the same category. On the other hand, false positives tend to be organized as independent clusters forming eddies that do not appear in the ground truth segmentation. Fig. 9 shows a series of illustrative examples we have selected from the Macaronesia test dataset.

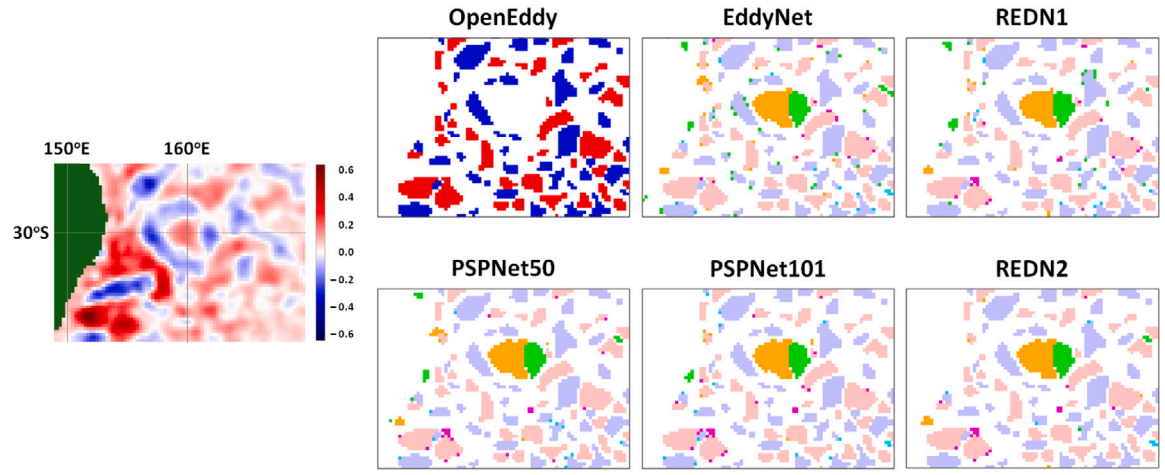
The segmentation proposed by our model for January 16th shows four groups of false positives in the area of the Cape Verde Islands that are clearly identifiable as cyclonic eddies. These eddies are not present in the ground truth segmentation of January 16th, but they do exist the day before and the day after. After analyzing the satellite altimetry data, we conclude that our CNN architecture is correctly identifying the presence of these eddies, and thus, these eddies are not the result of a pixel classification error but an error in the pixel labeling generated by the OpenEddy algorithm (Faghmous et al., 2015).



(a) Macaronesia region in the North Atlantic Ocean, November 20th



(b) Kuroshio region in the Northwest Pacific Ocean, May 20th



(c) New South Wales region in the Southwest Pacific Ocean, October 9th

Fig. 10. Classification errors in the segmentation proposed by all the evaluated CNN models. Correctly classified pixels are shown faded, cyclonic structures in blue and anticyclonic in red. False negatives are shown in cyan (cyclonic) and magenta (anticyclonic), while false positives are shown in green (cyclonic) and orange (anticyclonic). (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

Table 9
Distribution of misclassified pixels: false negatives and false positives for cyclonic eddies (fnc & fpc) and anticyclonic eddies (fna & fpa).

	Macaronesia			
	fnc	fna	fpc	fpa
Isolated	30.8%	30.9%	32.2%	32.9%
Border cluster	55.6%	55.8%	16.0%	13.4%
Independent cluster	13.6%	13.3%	51.8%	53.7%
	Kuroshio			
	fnc	fna	fpc	fpa
Isolated	25.0%	25.8%	26.6%	24.9%
Border cluster	55.6%	54.7%	19.1%	12.6%
Independent cluster	19.4%	19.5%	54.3%	62.5%
	New South Wales			
	fnc	fna	fpc	fpa
Isolated	26.6%	29.3%	27.8%	29.4%
Border cluster	47.9%	53.6%	17.8%	14.3%
Independent cluster	25.5%	17.1%	54.4%	56.3%

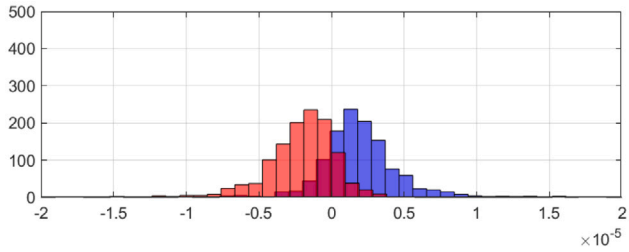
The proposed segmentation for May 29th shows another example of an independent cluster, this time an anticyclonic eddy, which spuriously disappears from the ground truth segmentation one day but is present on the preceding and following days. We can also observe some examples of border clusters, either cyclonic or anticyclonic, that merge two independent eddies in the ground truth segmentation just during that day, while our model keeps them correctly separated.

Fig. 10 presents some additional examples selected from the three areas under study, showing the segmentation proposed by all the evaluated models to allow a direct comparison between the generated results. In the Macaronesia region (a), the productive Canary Islands' eddy corridor (Sangrà et al., 2009) shows a sequence of cyclonic and anticyclonic eddies spreading westward from the southwest of the Canary Islands on November 20th. While the OpenEddy algorithm fails to identify one of the anticyclonic structures, all the CNN models identify this eddy correctly, although EddyNet struggles to correctly define the edges of some of these eddies.

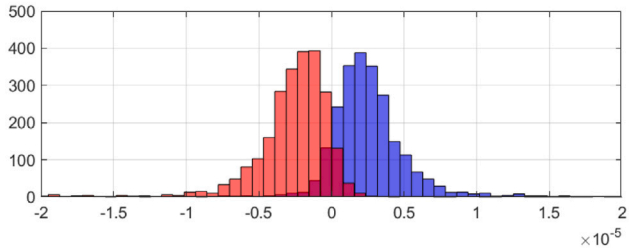
In the Kuroshio extension (b), focusing in the repetitive appearance of structures propagating eastward from the southeast coast of Japan (Cheng et al., 2014), two clear eddies can be identified on May 20th, a cyclonic eddy and an anticyclonic eddy. The OpenEddy segmentation correctly detects the anticyclonic structure but fails to identify the cyclonic one. EddyNet also fails to detect the cyclonic eddy and is not able to properly identify the borders of the anticyclonic eddy. REDN1 correctly identifies the cyclonic eddy but fails to properly define the borders of the anticyclonic eddy. The remaining networks do a good job identifying the pixels belonging to both eddies.

In the New South Wales region (c), the active Eddy Ave (Everett et al., 2012) shows on October 9th two eddies in close proximity that the OpenEddy algorithm is not able to detect. This algorithm assumes that an eddy has a single extreme and iterates around its closed-contour to generate the segmentation mask. If the algorithm extends the eddy contour to contain two extrema, it reverts back to a previous iteration (Faghmous et al., 2015). In this case, the contours of the two eddies interfere with each other and cause the algorithm to fail at identifying either, whereas models based on CNNs are able to identify both.

These examples illustrate the main drawback of iterative eddy detection algorithms based on geometric features: they depend on certain thresholds set by experts and, once these thresholds are exceeded, their behavior changes without taking into account the wider context. This kind of disruptions on specific days is a problem that has been observed before in the literature and has been attributed to a variety of reasons related to sampling errors and measurement noise (Chelton et al., 2011). Previously proposed solutions generally rely on interpolating the position of the missing eddy using data from the day before and the day after. Our results show that a CNN model is able to overcome these issues, providing a more robust eddy detection.



(a) eddies detected by OpenEddy that do not match REDN2 segmentation



(b) eddies detected by REDN2 that do not match OpenEddy segmentation

Fig. 11. Centroid vorticity distribution for the structures identified as eddies by only one of the segmentation maps; anticyclonic eddies are shown in red and cyclonic eddies are shown in blue. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

5.2. Model quality assessment

Following this argument, it is reasonable to think that many of the pixel classification errors made by our CNN architecture could be avoided, either because they are not actual errors or because the model has been mislead to learn some erroneous patterns.

To compare our eddy detection model with the original OpenEddy segmentation (Faghmous et al., 2015) used to train our CNN architecture, we have measured the amount of global structures detected by both approaches. Taking into account the full set of SLA fields corresponding to the 2013 year in the Macaronesia region, our model has identified a total of 101,164 compact structures: 50,331 cyclonic eddies and 50,833 anticyclonic eddies. For the same region and period of time, the OpenEddy segmentation mask dataset contains a total of 98,854 structures: 49,207 cyclonic and 49,647 anticyclonic.

To analyze the discrepancies between both segmentations, we have computed the vorticity values corresponding to the associated current velocity field (zonal and meridional components) of the processed region for all the non matching structures. There are 4076 structures identified as eddies by the REDN2 segmentation that do not correspond to eddies in the OpenEddy segmentation, while there are 2399 structures identified as eddies by the OpenEddy segmentation that do not correspond to eddies in the REDN2 segmentation. Fig. 11 shows the histogram of the centroid vorticity values for these structures. Analyzing the coherence of the vorticity values for these structures, we have found that the structures identified as eddies by OpenEddy that do not match the REDN2 segmentation present a 16.55% misclassification rate. In contrast, the structures identified as eddies by REDN2 that do not match the OpenEddy segmentation only present a 9.33% misclassification rate. These results reinforce the idea that the pixels in those regions have been misclassified due to the limitations of the original OpenEddy algorithm, while our model is able to correctly identify them as eddies.

6. Conclusions

Over the last decade, deep learning has taken artificial intelligence to unprecedented levels. However, with the continued increase in global energy consumption and the shortage of electronic components, applying deep learning as a brute-force strategy is an unsustainable approach.

In contrast with the trend towards proposing increasingly complex architectures to detect oceanic mesoscale eddies, we have shown in this paper that a relatively simple CNN design can provide comparable or even better results. In order to accomplish this, it is key to use as much data as possible. Even if we are interested in studying the eddies in a particular region of the ocean, training the model with a larger region and then clipping the target area allows the training process to work with much more information, producing a model that is able to avoid overfitting and provide a better generalization.

Our results show that the proposed CNN design is able to correctly classify more than 97% of the pixels corresponding to eddies on SLA gridded maps for different regions of the ocean labeled using the OpenEddy algorithm (Faghmous et al., 2015), while requiring a considerably smaller amount of parameters than previous proposals. Our results also show that most of the pixel classification errors are not due to the model itself, but to robustness issues in the labeling algorithm.

On this basis, it becomes clear that improving the input data would have a positive impact on CNN performance. Pegliasco et al. (2021) have shown recently that it may be preferable to rely on ADT maps in those regions where certain eddies appear recurrently, since SLA maps may lead to detection errors. In addition, Stegner et al. (2021) have shown that eddy detection is not completely reliable on the basis of gridded altimetry maps like the AVISO+ products. The low resolution of these altimetry maps also has a negative impact on eddy detection (Amores et al., 2018), making current detection algorithms to miss some of the smaller eddies or even aggregate them into larger structures, as confirmed by our results.

Therefore, we believe that the solution to the eddy detection problem is not to keep proposing more and more complex networks. Instead, a mindful design of the training phase, a careful selection of the input data, and the improvement of the labeling process will most likely lead to a more efficient CNN model yielding near-optimal results.

CRediT authorship contribution statement

Oliverio J. Santana: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Resources, Data curation, Writing – original draft, Writing – review & editing, Visualization, Supervision, Project administration. **Daniel Hernández-Sosa:** Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Resources, Data curation, Writing – review & editing, Visualization, Supervision, Project administration. **Ryan N. Smith:** Conceptualization, Formal analysis, Writing – review & editing, Supervision, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The authors do not have permission to share data.

Acknowledgments

This work is supported in part by the NSF grant IIS-2024733. The authors would like to thank James H. Faghmous for releasing the OpenEddy segmentation masks.

References

Amores, A., Jordà, G., Arsouze, T., Le Sommer, J., 2018. Up to what extent can we characterize ocean eddies using present-day gridded altimetric products? J. Geophys. Res. Oceans 123 (10), 7220–7236. http://dx.doi.org/10.1029/2018JC014140.

Chelton, D.B., Schlax, M.G., Samelson, R.M., 2011. Global observations of nonlinear mesoscale eddies. Prog. Oceanogr. 91 (2), 167–216. http://dx.doi.org/10.1016/j.pocean.2011.01.002.

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L., 2018. DeepLab: semantic image segmentation with deep convolutional Nets, atrous convolution, and fully connected CRFs. IEEE Trans. Pattern Anal. Mach. Intell. 40 (4), 834–848. http://dx.doi.org/10.1109/TPAMI.2017.2699184.

Cheng, H.K., Chung, J., Tai, Y.-W., Tang, C.-K., 2020. CascadePSP: toward class-agnostic and very high-resolution segmentation via global and local refinement. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 8890–8899.

Cheng, Y., Ho, C., Zheng, Q., Kuo, N., 2014. Statistical characteristics of mesoscale eddies in the North Pacific derived from satellite altimetry. Remote Sens. 6 (6), 5164–5183. http://dx.doi.org/10.3390/rs6065164.

Chinchor, N., 1992. MUC-4 evaluation metrics. In: MUC4 '92: Proceedings of the 4th Conference on Message Understanding. pp. 22–29. http://dx.doi.org/10.3115/1072064.1072067.

Chollet, F., 2017. Xception: deep learning with depthwise separable convolutions. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1800–1807. http://dx.doi.org/10.1109/CVPR.2017.195.

Duo, Z., Wang, W., Wang, H., 2019. Oceanic mesoscale eddy detection method based on deep learning. Remote Sens. 11 (16), 1921. http://dx.doi.org/10.3390/rs11161921.

Everett, J.D., Baird, M.E., Oke, P.R., Suthers, I.M., 2012. An avenue of eddies: quantifying the biophysical properties of mesoscale eddies in the Tasman Sea. Geophys. Res. Lett. 39 (16), L16608. http://dx.doi.org/10.1029/2012GL053091.

Faghmous, J.H., Frenger, I., Yao, Y., Warmka, R., Lindell, A., Kumar, V., 2015. A daily global mesoscale ocean eddy dataset from satellite altimetry. Sci. Data 2 (1), 150028. http://dx.doi.org/10.1038/sdata.2015.28.

Fan, Z., Zhong, G., Wei, H., Li, H., 2020. EDNet: a mesoscale eddy detection network with multi-modal data. In: 2020 International Joint Conference on Neural Networks (IJCNN). pp. 1–7. http://dx.doi.org/10.1109/IJCNN48605.2020.9206613.

Franz, K., Roscher, R., Milioto, A., Wenzel, S., Kusche, J., 2018. Ocean eddy identification and tracking using neural networks. In: IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium. pp. 6887–6890. http://dx.doi.org/10.1109/IGARSS.2018.8519261.

He, K., Zhang, X., Ren, S., Sun, J., 2015. Delving deep into rectifiers: surpassing human-level performance on ImageNet classification. In: 2015 IEEE International Conference on Computer Vision (ICCV). pp. 1026–1034. http://dx.doi.org/10.1109/ICCV.2015.123.

He, K., Zhang, X., Ren, S., Sun, J., 2016a. Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 770–778. http://dx.doi.org/10.1109/CVPR.2016.90.

He, K., Zhang, X., Ren, S., Sun, J., 2016b. Identity mappings in deep residual networks. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (Eds.), Computer Vision – ECCV 2016. Springer International Publishing, Cham, pp. 630–645.

Ioffe, S., Szegedy, C., 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37. In: ICML'15, JMLR.org, pp. 448–456. http://dx.doi.org/10.5555/3045118.3045167.

Kingma, D.P., Ba, J., 2015. Adam: a method for stochastic optimization. arXiv:1412.6980. Published as a conference paper at ICLR 2015.

Lguensat, R., Rjiba, S., Mason, E., Fablet, R., Sommer, J., 2018a. Convolutional neural networks for the segmentation of oceanic eddies from altimetric maps. http://dx.doi.org/10.13140/RG.2.2.23386.90564.

Lguensat, R., Sun, M., Fablet, R., Tandeo, P., Mason, E., Chen, G., 2018b. EddyNet: a deep neural network for pixel-wise classification of oceanic eddies. In: IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium. pp. 1764–1767. http://dx.doi.org/10.1109/IGARSS.2018.8518411.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., Dollár, P., 2017. Focal loss for dense object detection. In: 2017 IEEE International Conference on Computer Vision (ICCV). pp. 2999–3007. http://dx.doi.org/10.1109/ICCV.2017.324.

Lu, X., Guo, S., Zhang, M., Dong, J., Chen, X., Sun, X., 2020. Mesoscale ocean eddy detection using high-resolution network. In: 11th International Conference on Awareness Science and Technology (ICAST). pp. 1–6. http://dx.doi.org/10.1109/iCAST51195.2020.9319490.

Mason, E., Pascual, A., McWilliams, J.C., 2014. A new sea surface height-based code for oceanic mesoscale eddy tracking. J. Atmos. Ocean. Technol. 31 (5), 1181–1188. http://dx.doi.org/10.1175/JTECH-D-14-00019.1.

Okubo, A., 1970. Horizontal dispersion of floatable particles in the vicinity of velocity singularities such as convergences. Deep Sea Res. Oceanogr. Abstr. 17 (3), 445–454. http://dx.doi.org/10.1016/0011-7471(70)90059-8.

Pegliasco, C., Chaigneau, A., Morrow, R., Dumas, F., 2021. Detection and tracking of mesoscale eddies in the Mediterranean Sea: a comparison between the sea level anomaly and the absolute dynamic topography fields. Adv. Space Res. 68 (2), 401–419. http://dx.doi.org/10.1016/j.asr.2020.03.039.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (Eds.), Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. Springer International Publishing, Cham, pp. 234–241.

Sangrà, P., Pascual, A., Rodríguez-Santana, A., Machín, F., Mason, E., McWilliams, J., Pelegrí, J., Dong, C., Rubio, A., Arístegui, J., Marrero-Díaz, A., Hernández, A., Martínez-Marrero, A., Auladell, M., 2009. The Canary eddy corridor: a major pathway for long-lived eddies in the subtropical North Atlantic. Deep Sea Res. I 56 (12), http://dx.doi.org/10.1016/j.dsr.2009.08.008, 2100–2014.

Santana, O.J., Hernández-Sosa, D., Martz, J., Smith, R.N., 2020. Neural network training for the detection and classification of oceanic mesoscale eddies. Remote Sens. 12 (16), 2625. http://dx.doi.org/10.3390/rs12162625.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. 15 (1), 1929–1958. http://dx.doi.org/10.5555/2627435.2670313.

Stegner, A., Le Vu, B., Dumas, F., Ghannami, M.A., Nicolle, A., Durand, C., Faugere, Y., 2021. Cyclone-anticyclone asymmetry of eddy detection on gridded altimetry product in the Mediterranean Sea. J. Geophys. Res. Oceans 126 (9), http://dx.doi.org/10.1029/2021JC017475, e2021JC017475.

Sun, K., Xiao, B., Liu, D., Wang, J., 2019. Deep high-resolution representation learning for human pose estimation. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5686–5696. http://dx.doi.org/10.1109/CVPR.2019.00584.

Sun, X., Zhang, M., Dong, J., Lguensat, R., Yang, Y., Lu, X., 2021. A deep framework for eddy detection and tracking from satellite sea surface height data. IEEE Trans. Geosci. Remote Sens. 59 (9), 7224–7234. http://dx.doi.org/10.1109/TGRS.2020.3032523.

Weiss, J., 1991. The dynamics of enstrophy transfer in two-dimensional hydrodynamics. Physica D 48 (2), 273–294. http://dx.doi.org/10.1016/0167-2789(91)90088-Q.

Xu, G., Cheng, C., Yang, W., Xie, W., Kong, L., Hang, R., Ma, F., Dong, C., Yang, J., 2019. Oceanic eddy identification using an AI scheme. Remote Sens. 11 (11), 1349. http://dx.doi.org/10.3390/rs11111349.

Yu, F., Koltun, V., 2015. Multi-scale context aggregation by dilated convolutions. http://dx.doi.org/10.48550/ARXIV.1511.07122.

Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J., 2017. Pyramid scene parsing network. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6230–6239. http://dx.doi.org/10.1109/CVPR.2017.660.