# AMERICAN UNIVERSITY OF BEIRUT

# EDDY DETECTION USING REANALYSIS DATASETS

by

## Omar Bassam Hakla

A thesis
submitted in partial fulfillment of the requirements
for the degree of Master of Engineering
to the Department of Electrical and Computer Engineering
of Maroun Semaan Faculty of Engineering and Architecture
at the American University of Beirut

Beirut, Lebanon
August 3, 2022

# AMERICAN UNIVERSITY OF BEIRUT

# EDDY DETECTION USING REANALYSIS DATASETS

by
## Omar Bassam Hakla

Approved by:

---

Dr. Issam Lakkis, Professor                    Advisor

Mechanical Engineering

---

Dr. Dany Abou Jaoude, Assistant Professor      Member of Committee

Mechanical Engineering                         Dany Abou Jaoude
                                               2022.09.02
                                               15:46:36 +03'00'

---

Dr. Ibrahim Hoteit, Professor                  Member of Committee

Mechanical Engineering


Date of thesis defense: August 3, 2022

# AMERICAN UNIVERSITY OF BEIRUT

# THESIS RELEASE FORM

Student Name: _____

Hakla                    Omar                    Bassam

I authorize the American University of Beirut, to: (a) reproduce hard or electronic copies of my thesis; (b) include such copies in the archives and digital repositories of the University; and (c) make freely available such copies to third parties for research or educational purposes

___ **As of the date of submission of my thesis**

X **After 1 year from the date of submission of my thesis .**

___ **After 2 years from the date of submission of my thesis .**

___ **After 3 years from the date of submission of my thesis .**

_____

Signature                                        Date 9/4/2022

# ACKNOWLEDGEMENTS

# ABSTRACT
# OF THE THESIS OF

Omar Bassam Hakla     for     Master of Engineering

Major: Mechanical Engineering

Title: EDDY DETECTION USING REANALYSIS DATASETS

Oceanic eddies are ubiquitous in oceans and play a major role in several parameters that include ocean energy transfer, nutrients distribution and air-sea interaction. Typically, eddy detection algorithms are based on single physical parameter, geometrics or other handcrafted features. To achieve better performances, we aim to develop a new approach to fuse multi-variable features for eddy detection. We will investigate lumping satellite datasets of Sea surface height, Sea surface temperature, Salinity in addition to full model solution velocity field through the inclusion of information (correlation) between the datasets.

# TABLE OF CONTENTS

# ILLUSTRATIONS

# TABLES

# CHAPTER 1

# INTRODUCTION

## 1.1 Ocean Vortices

Vortical activities in ocean is seen in plenty of regions in a mesoscale and sub-mesoscale as shown in figure 1.1 [1, 2]. These activities are known as eddies, which play key role in global energy and nutrient distribution throughout the ocean [3, 4, 5]. Eddies have a wide range of sizes that can vary between centime-



Figure 1.1: Ocean Eddy [1]

ters and hundreds of kilometers in diameter lasting for several days and sometimes more than a month [6, 7]. Moreover, their heights range up to 30 cm up or down the sea surface level depending on the type of the eddy as discussed further on [4, 8]. Add to that, due to their strong rotation in the ocean, they have high kinetic energy compared to ocean mean flow [9, 10]. Eddies have a major impact

on nutrient distribution within the ocean such as dissolved oxygen (DO) and water salinity [11, 12]. The two types of eddies to be studied further on are cyclonic and anticyclonic eddies [8]. These Eddies are detected based on 3 main parameters, which are the sea surface temperature (SST), sea surface height (SSH) and rotational direction [4]. Nutrients in deep water are transferred in cyclonic eddies through an upwelling technique, thus they are pumped to shallow areas [13, 14]. On the other hand, the opposite will occur when dealing with anticyclonic eddies where the nutrients are pushed to deep levels in the sea through the downwelling technique. The upwelling and downwelling phenomena are a major concern when studying eddies. This can be used as one of the key factors to determine eddies whether cyclonic or anticyclonic. Moreover, another important parameter is the core temperature of the eddy. It is found to be cold in cyclonic eddies because water from deeper depths is pumped up, however in anticyclonic eddies, due to the downwelling the opposite occurs where a warm core is observed. It is important to note that the direction of rotation is anticlockwise and clockwise in cyclonic and anticyclonic eddies respectively [15, 6]. The following is presented in the figure 1.2 bellow:
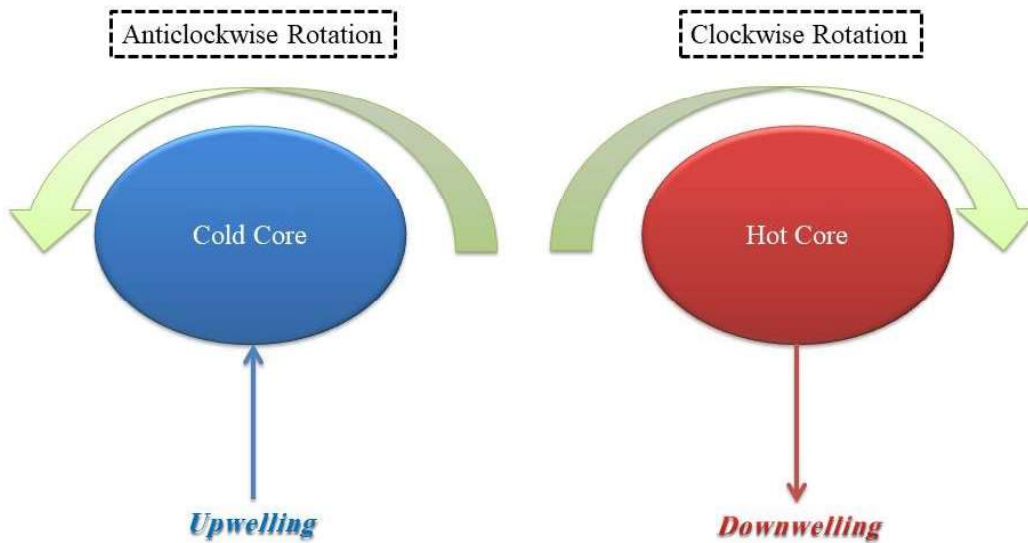


Figure 1.2: Upwelling and Downwellling phenomena associated with Temperature distribution and rotation direction

## 1.2 Reanalysis Data

Reanalysis is a scientific method used in weather and climate forecast predictions. It mainly relies on two parameters which are the observations and numerical

solutions as shown in figure 1.3. Observations are usually caught using satellites and are known as altimeter data. Altimeter data covers different properties such as sea level anomaly (SLA) patterns [16]. Recently, researchers have the access to higher resolution data [17]. Reanalysis data extends over years and in most cases for decades. Extensive climate researches demand a lot of information from the past which is achieved and calculated based on reanalysis data.
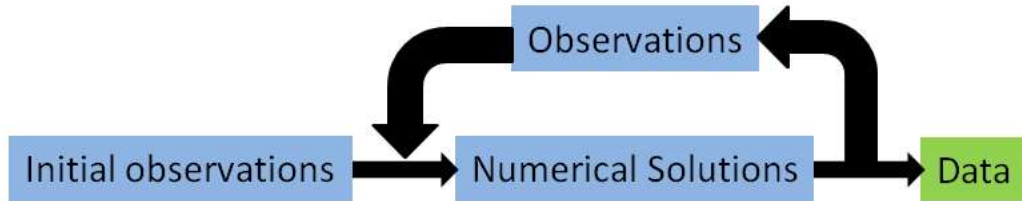


Figure 1.3: Reanalysis data model

## 1.3 Introduction to Deep Learning

Among the methods used to solve complex scenarios and understand the correlation between available parameters is deep learning [18, 19]. This new pillar is known for its high predictive modeling in plenty of fields and the fast processing it offers. Deep learning includes statistical and predictive modeling by training the model using large datasets to present a final algorithm that is built according to the required objective. These models can be trained in a supervised or unsupervised manner. Supervised learning trains the model by using labeled data, while in unsupervised training the model uses unlabeled data. Better models can be achieved when more training sets are available since deep learning is simply training by examples. Recently, deep learning is also used in medical research where some doctors are using it for cancer detection [20, 21]. Image detection using deep learning commonly uses Convolutional Neural Networks that include convolution, pooling and fully connected layers. In this study, deep learning techniques will be used to detect water vortices known as cyclonic and anticyclonic eddies. This helps in understanding the correlation found between them and between several parameters that will be referred to as the channels during the training process. In this study, the following framework in the below figure will be used to prepare a new model for eddy detection. The artificial neural network used is called the FCN-resnet50 network, which is known for image classification tasks [22]. Reanalysis data will be generated as images using both the velocity vectors and SST as shown below, then the eddies in each image will be labeled manually. Once this is done, this set of labeled images will be used in FCN-resnet50 to train the model and validate it. Once the model is trained and

validated, it will be able to identify the eddies present on any day by inputting SSH, SST, velocity and salinity data of the required day as illustrated in figure 1.4. The model will output segmented images of this day that include areas detected as non eddy, cyclonic or anticyclonic eddy as demonstarted in figure 1.5. For further knowledge about the artificial neural network, the following references can be used [23, 24, 25].
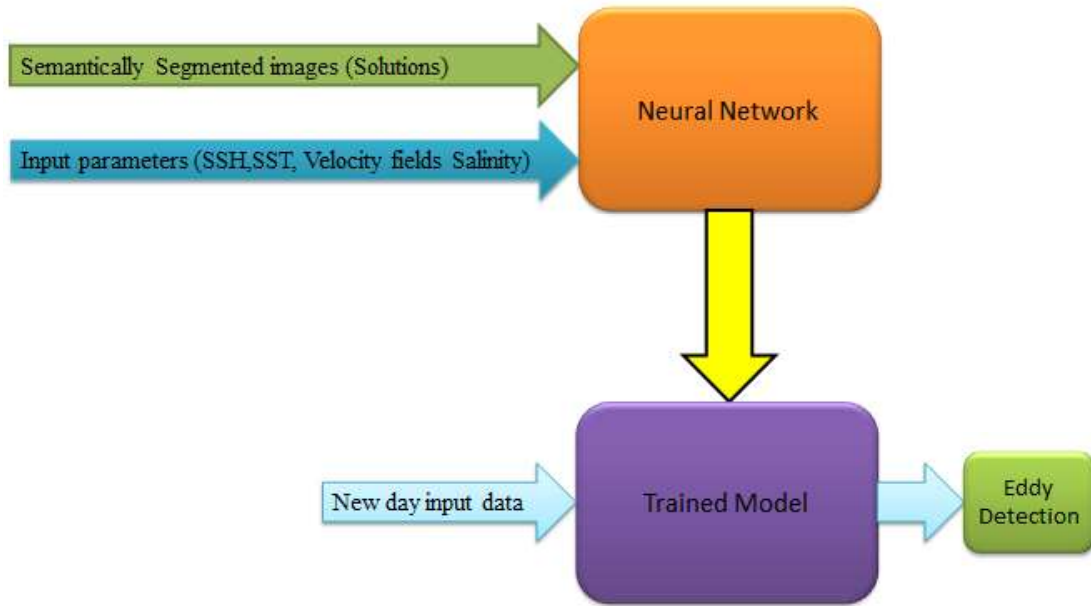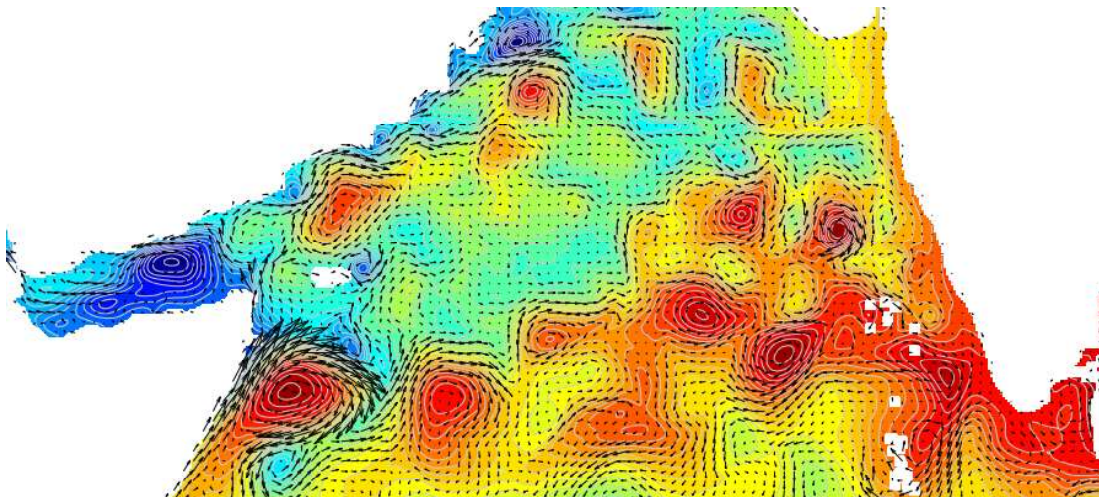


Figure 1.4: Framework schematic diagram



Figure 1.5: Reanalysis generated image

## 1.4   Literature Review

Classical methods consisted of calculating criteria to classify whether there are eddies or not, but they showed plenty of mislabeling. In the last decades, the most established metric used for eddy detection was the Okubo-Weiss criterion [26, 27]. This criterion determines the high deformation and vorticity over the whole domain using the following relations:

$$W(\mathbf{x}, t) = s^2(\mathbf{x}, t) - \zeta^2(\mathbf{x}, t),$$

$$s^2(\mathbf{x}, t) = (\frac{\partial u}{\partial x} - \frac{\partial v}{\partial y})^2 + (\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y})^2,$$

$$\zeta^2(\mathbf{x}, t) = (\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y})^2,$$

where $\mathbf{x}$ represents xy coordinates, $t$ represents time, $s^2$ represents normal strain, $\zeta^2$ represents shear stain also the $u$ and $v$ are the velocity components. An eddy is detected when having the value of the Okubo-Weiss criterion $W$ negative, then using SSH, the eddy is identified as cyclonic or anticyclonic. An alternative way was applied to the gulf of Alaska in 1994. High-resolution images supplied by a Synthetic Aperture Radar (SAR) were used to detect eddy presence in the gulf of Alaska. However, during strong winds and high sea tides, the surface signature of eddies was unclear, therefore the wave refractions of the eddy were observed and then implemented in a ray-tracing model [28]. In early 2000, an investigation was done regarding the eddies detection in the bay of Bengal where waters there are highly stratified. This study focused on SSH observations from altimeter data to detect eddies and classify them as cyclonic or anticyclonic. However, due to the highly stratified waters, the SST gradients were not much detectable [29]. These methods required expert-tuned parameters and led to high rates of eddy declassification. Having that said, deep learning techniques were implemented in the field of eddy detection and their results appeared to be more satisfying as shown further on. Currently enhancements in this field are being applied to have better results [19, 30].

In 2010, a model was designed to detect eddies based on the geometry of the velocity components [31]. The results were compared to manually detected eddies. The comparison was based on two main parameters, success detection rate (SDR) and excess detection rate (EDR), that were calculated as follows:

$$SDR = \frac{N_c \times 100}{N_{te}},$$

$$EDR = \frac{N_{oa} \times 100}{N_{te}},$$

where $N_{te}$ was referred to as true number of eddies and $N_c$ was the correct number of eddies detected by the algorithm, while $N_{oa}$ is number of incorrect eddies

predicted by the algorithm. The results for a sample of 10 days were shown in table 1.1:

| Day | 115 | 352 | 705 | 811 | 820 | 1476 | 1626 | 1787 | 1833 | 1870 | Total |
|---|---|---|---|---|---|---|---|---|---|---|---|
| True eddies | 19 | 20 | 15 | 20 | 16 | 18 | 16 | 24 | 26 | 23 | 197 |
| $N_c$ | 18 | 17 | 14 | 19 | 15 | 17 | 15 | 24 | 23 | 21 | 183 |
| $N_{oa}$ | 0 | 2 | 0 | 2 | 0 | 1 | 0 | 0 | 1 | 0 | 6 |
| Missed eddies | 1 | 3 | 1 | 1 | 1 | 1 | 1 | 0 | 3 | 2 | 14 |
| SDR (%) | 94.7 | 85.0 | 93.3 | 95.0 | 93.7 | 94.4 | 93.8 | 100.0 | 88.5 | 91.3 | $92.9 \pm 4.0$ |
| EDR (%) | 0.0 | 10.0 | 0.0 | 10.0 | 0.0 | 5.6 | 0.0 | 0.0 | 3.8 | 0 | $2.9 \pm 4.2$ |

Table 1.1: SDR and EDR sample results [31]

A model was implemented on the Southern Java Indian Ocean to study the characteristics of eddies on different vertical levels [32]. The model relied on both velocity fields and SST, which lead to visualization of the vertical structure of eddies and their presence at 4 different vertical levels which were 5, 109, 155 and 222 meters as demonstrated in figure 1.6 below:
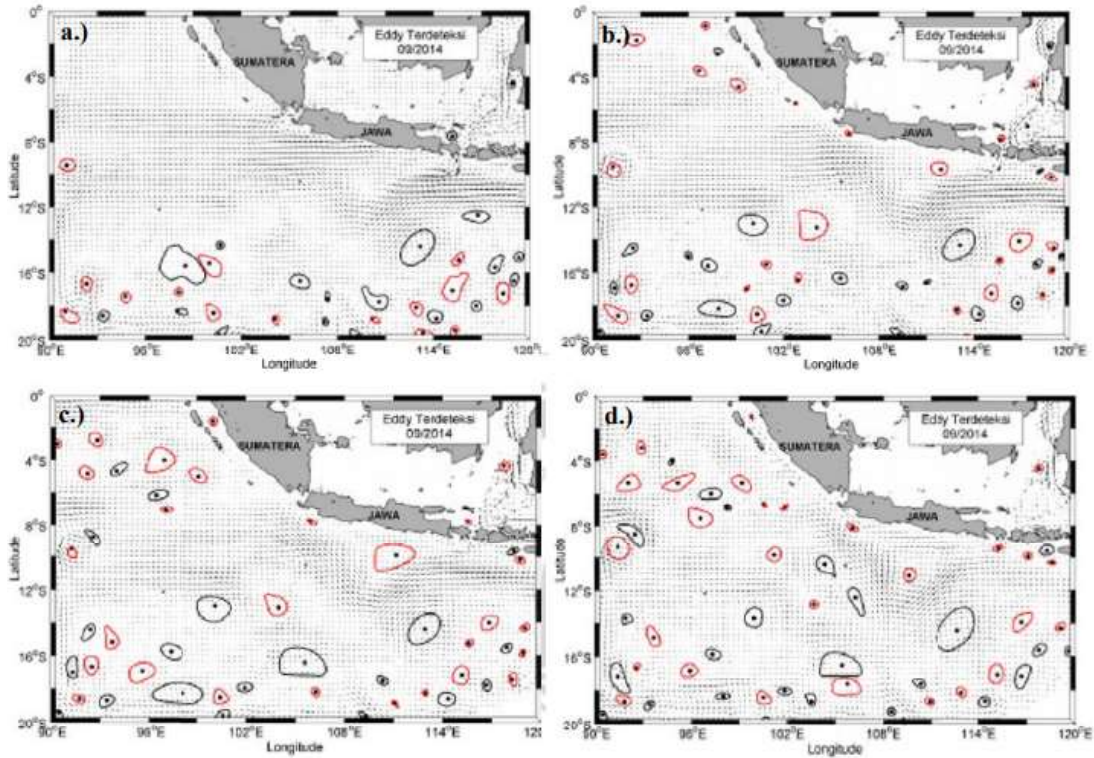


Figure 1.6: Eddy visualisation in September 2014 at four vertical levels [32] a.) 5 m, b.) 109 m, c.) 155 m, d.) 222 m

Moreover, the model also included calculating the eddy kinetic energy (EKE)

distribution over the region to understand how it is spread.

$$EKE = 0.5(u^2 + v^2).$$

Later on, through the use of Support Vector Machines (SVMs) classifier with radial basis function, a model was designed in 2016 to detect eddies. The model relied on the $u$ and $v$ components of the surface velocities only [33]. The phase angle between $v$ and $u$ components of velocity was calculated using the following relation:

$$\Phi = \tan^{-1} \frac{v}{u}.$$

Then, the value of the phase angle was plotted over the whole domain and eddies would be detected from the rotation field if present as shown in figure 1.7. Then
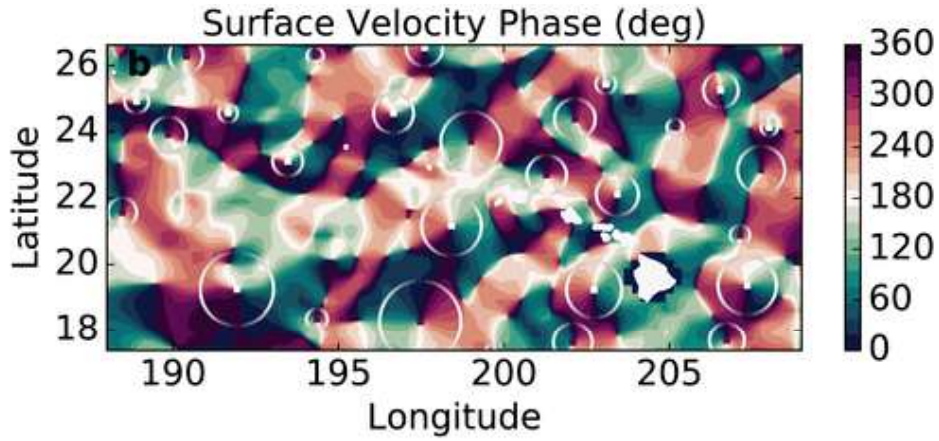


Figure 1.7: Phase angle distribution over the domain [33]

using the SST (referred as sea level anomaly), eddies were classified as cyclonic or anticyclonic ones as shown in figure 1.8.
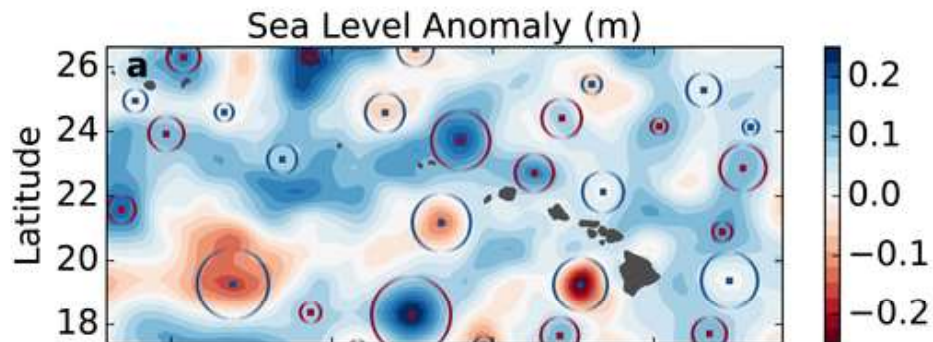


Figure 1.8: Cyclonic eddies in blue and anticyclonic eddies in red [33]

12

A U-Net like network named EddyNet was used in the year 2018 which is based on pixel-wise identification and also uses a convolutional encoder-decoder [34]. This architecture was trained using only SSH data that was provided by Copernicus Marine and Environment Monitoring Service (CMEMS). It was composed of 15 years where each year is $128 \times 128$ pixels. The prediction map consisted of 3 labels that represent non eddy, cyclonic and anticyclonic eddy areas. Moreover, the results were studied according to the dice coefficient (DC) metric which is calculated as follows:

$$DC = \frac{2 \times \text{Common area between prediction and true labels}}{\text{Sum of predicted and true areas}}.$$

In 2019, another approach to detect eddies using deep learning technique was held using pyramid scene parsing network (PSPNet) architecture. The model relied only on SSH data. The results showed better detection when compared with a traditional vector geometry-based (VG) approach [35] as presented in figure 1.9. The data was composed of 5 years that were between 2011 and 2015
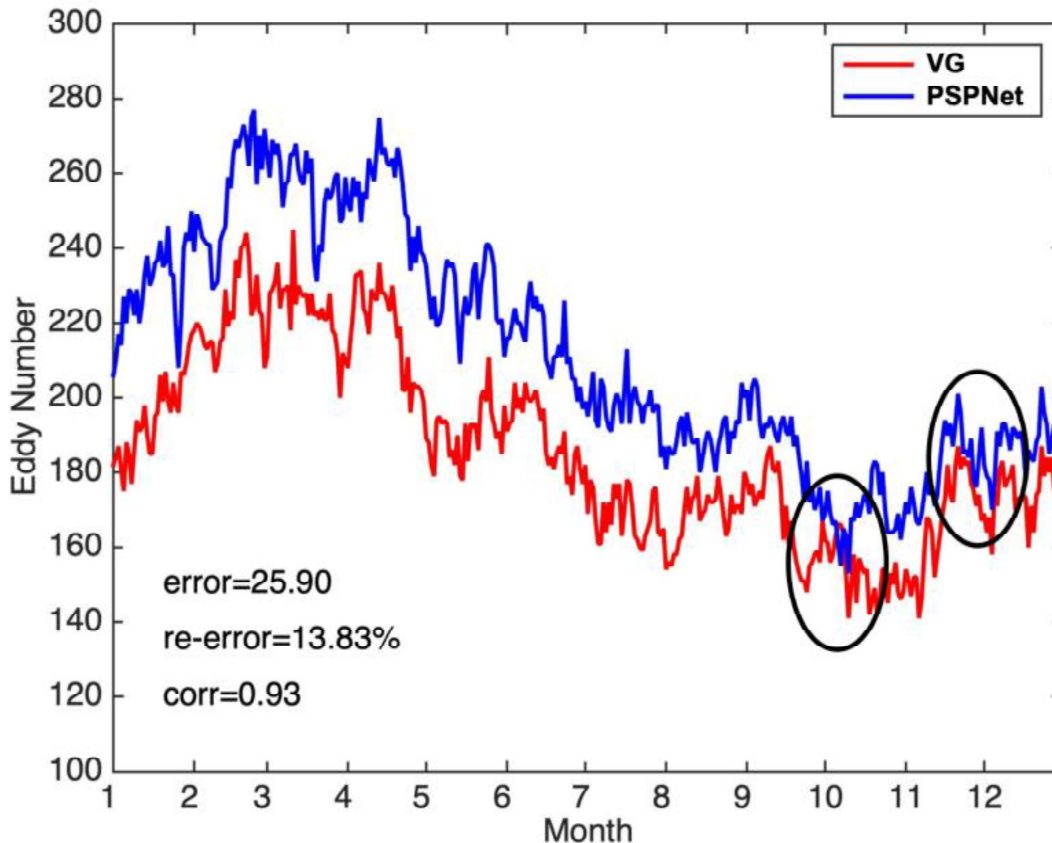


Figure 1.9: Comparison of monthly results between PSPNet and VG architectures [35]

13

covering the east of the Luzon Strait to the Hawaii Islands. Only the year 2015 was used for validation further on. The VG approach was based on the velocity fields vector, which has two components the u and v that were used to determine areas of rotation. Upon using the PSPNET model more eddies were detected during 2015 than VG architecture. The average eddy difference was 26 eddies approximately per day and a relative error of 13.83%, which was the error divided by VG results. Both architectures showed a correlation of 0.93 which was very high.

Lately in 2021, in order to study the applicability of deep learning techniques in eddy detection, three techniques were implemented that rely SSH data. The data was distributed over 5 years. The results showed how each of the three techniques had it's perk. PSPNet resulted in largest number of eddies predicted. Bilateral Segmentation Network (BISENET) showed higher accuracy in finding large scale eddies. However, DeepLabV3+ technique didn't track much eddies as the previous two techniques [36].

# Chapter 2

# Segmentation

In this chapter, deep explanation will be available to understand how the labelling of the satellite images was done and what types of image segmentation can be used.

## 2.1 Image Segmentation

There are 3 main types of image segmentation: Semantic, instance and panoptic segmentation. The aim is to understand what objects are available in each image and each technique serves a certain result of classification [37, 38]. In outline, the first type to be understood is the semantic segmentation; all image pixels will be classified of a certain class. For example, if there is a picture of a normal straight road all cars will be treated as one class and the people as another calss as shown in figure 2.1. As it can be seen, this is a pixel-by-pixel identification



Figure 2.1: Semantic Segmentation

where each type of object belonging to the same class was identified by a certain color. Another type of image segmentation was introduced further on which deals with available objects as distinct individuals and this is called the instance segmentation technique [38]. In this technique, even if the objects were of the same primary class, however they will be dealt with as separate objects. Each object has its id and properties to be studied such as presented in figure 2.2 below:



Figure 2.2: Instance Segmentation

Although the image includes several detections that include humans, each individual was treated differently than the other. This technique differentiates each one of them. Therefore, there were several objects labeled and identified as humans but each has its specific id and properties although they belong to the same primary class. Further on, a new concept was introduced which combines both instance and semantic segmentation. This new concept was known as panoptic segmentation. It assigns for each pixel two values. One of them will be the same as the semantic case and the other same as the instance segmentation case presented in figure 2.3 [38].
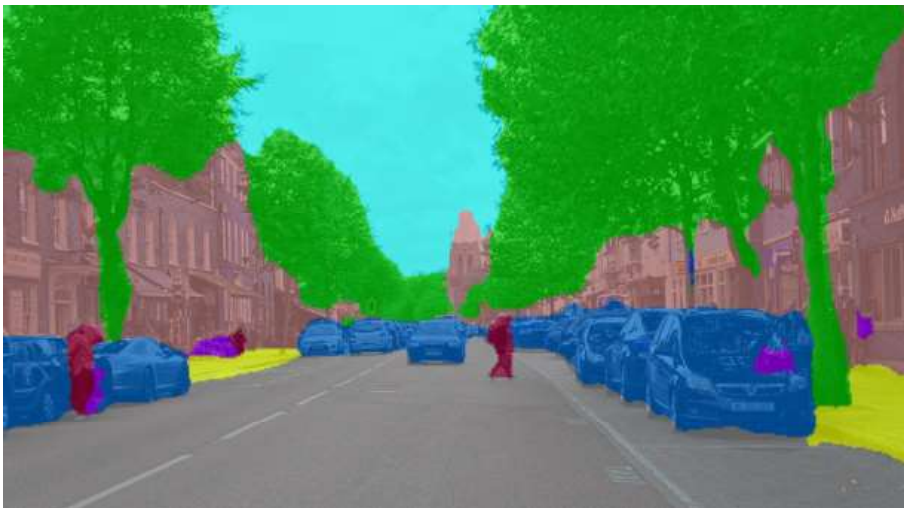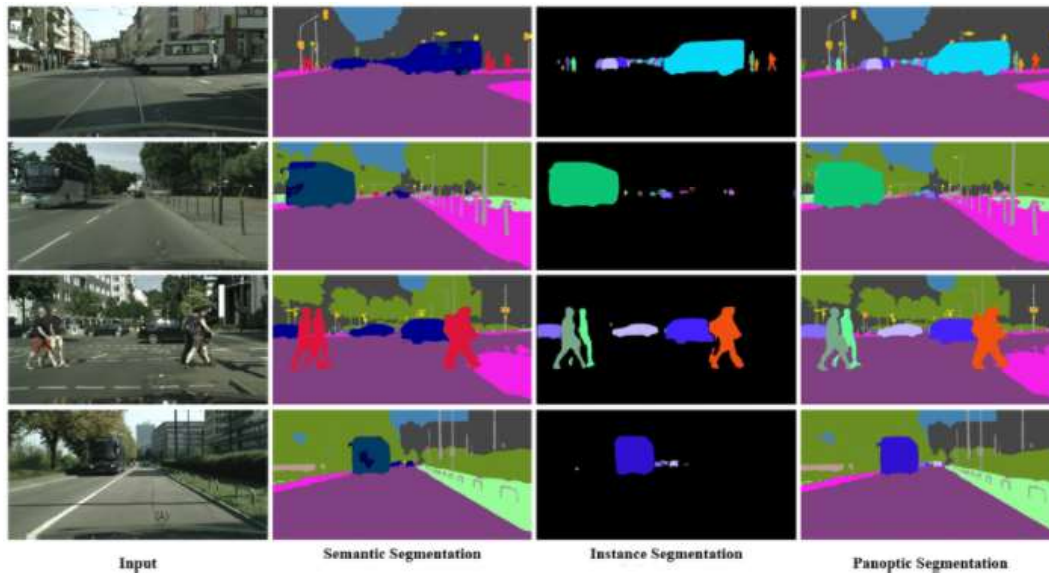
Figure 2.3: Panoptic Segmentation

## 2.2 Eddy Detection using Semantic Segmentation

The aim of this project is to classify where there are eddies and of what type are they. The interest is not studying eddy development over days and years. Neither the aim is to calculate the kinetic energy of eddies. All pixels should be categorized under three categories: background, cyclonic and anticyclonic eddy. Therefore, the best segmentation technique that matches this project's aim is semantic segmentation.

## 2.3 Training Data

The data is based on the Copernicus Marine Environment Monitoring Service (CMEMS) global ocean eddy-resolving reanalysis product with a horizontal resolution of $1/12^o$ (approximately 8 km). The reanalysis system is based on version 3.1 of the NEMO (Nucleus for European Models of the Ocean) ocean model, driven at the surface by ECMWF ERA-Interim reanalysis. Observations are assimilated with a reduced-order Kalman filter, including along-track altimeter data, satellite Sea Surface Temperature (SST), and in situ T/S vertical profiles. Daily mean temperature, salinity, current, and sea level fields are analyzed. The velocities are full model solutions, not from the geostrophic relationship.

17

## 2.4  Methods Used

This study relies on two main tasks to be achieved. The first one will be preparing a well labelled training set which can be done using an image annotation tool such as a Labelme where each satellite image available will be labelled and then the data will be extracted. There was no interest in examining land areas. Therefore, they were marked white throughout the whole studied domain. Cyclonic and anticyclonic eddies were labeled using two main parameters: eddy velocity fields and the SST. SST contours were plotted accompanied by the velocity field to help identify eddy presence and type as shown in figure 2.4. Labeling the training set involves the representation of each eddy by a polygon created using the image annotation tool Labelme.
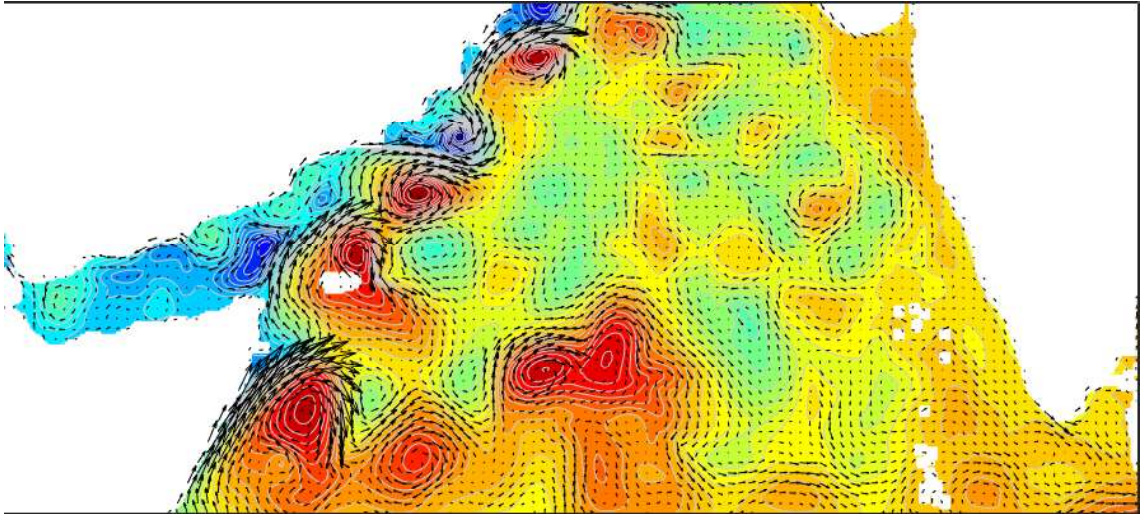


Figure 2.4: Satellite Image Sample

The largest contour which shows both high vorticity and temperature relevance was identified as an eddy. After the labelling of an image, it will then be turned into an annotation image. The stored xy coordinates of the polygons were semantically segmented using labelme as shown in figure 2.5. Through this, the solution for the training set was prepared. After image annotations are available, usually they are distributed into two parts. The first part includes 80% of the prepared annotations, which will be the bulk of training. The remaining 20% will be used to validate the operation of the neural network. However, in this investigation since the available images were distributed over 11 years, it was decided to use 10 years as training data sets and the last year for validation. This was done to ensure validation over a whole year accompanied with maximum training datasets. Moreover, the more accurate the images were labeled and revised carefully, the better the results will appear. So, it is very important to have well-labeled images because this is reflected in the predicted results. The
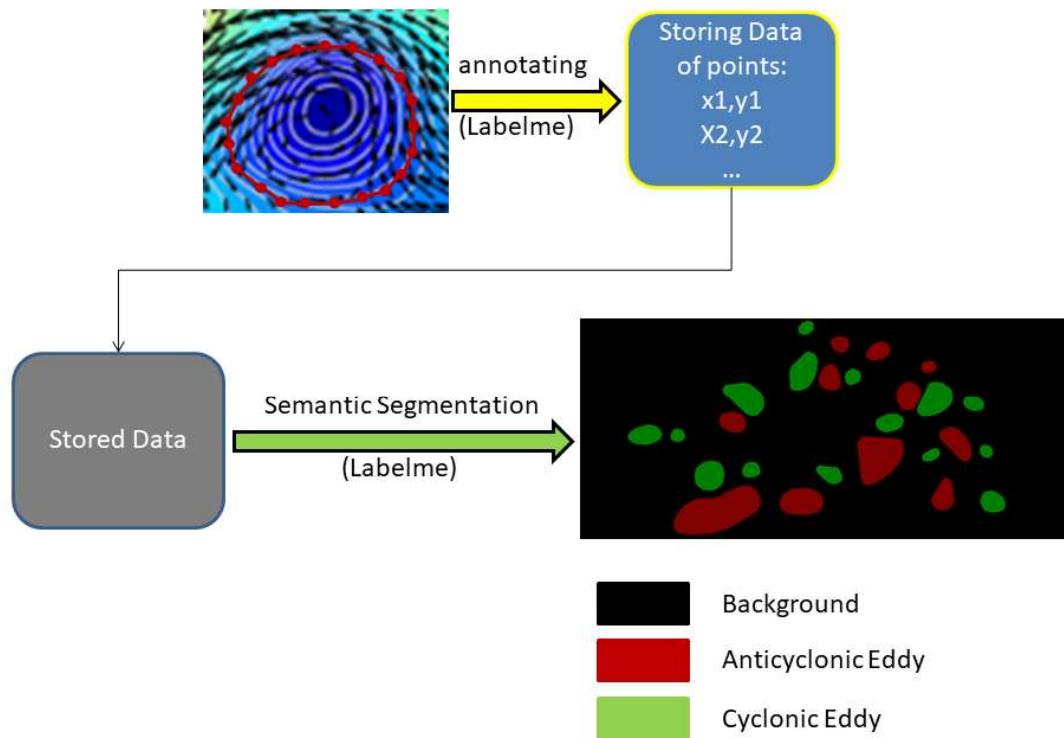
18

Figure 2.5: Annotation image prepation

aim of the project is to classify the pixels of an image whether it's an eddy and of what type or not an eddy. Hence, the semantic segmented data was extracted for all days as provided below in figure 2.6. Once there is a well prepared training dataset, the neural network can be actuated and the model can be trained. In this study, PyTorch framework was used to create the neural network. PyTorch is an open source framework used in variety of research fields that include deep learning, object detection and plenty of other projects can be done using this framework.
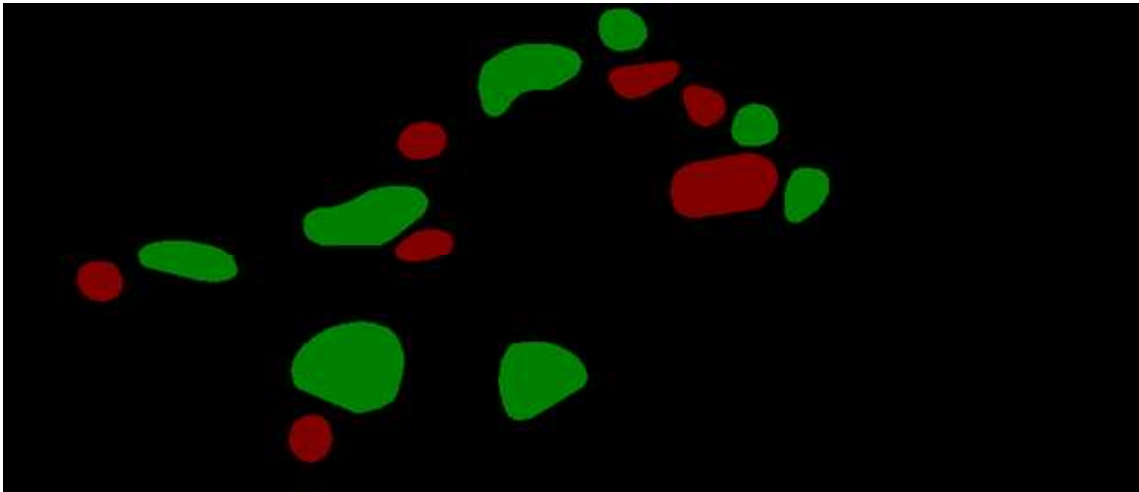
Figure 2.6: Semantic Segmented Image

# Chapter 3

# Training the network

In this chapter, there will be explanation about how the training was done and results analysis will be shown.

## 3.1   Code architecture

The data is composed of 4015 labelled image distributed into two folders, one for training and one for validation datasets. The training file included the data from 2001 till 2010 and the validation file included the data of the final year 2011. Moreover, the training file included requirements to launch the code on KAUST IBEX. It was executed using the sbatch command that initiates the code and inputs all training data in the FCN architecture. Once training is completed, the results are stored in the form of tensorboard files. They were viewed further on using google colab to select the best model. During the training phase, the training dataset, which was composed of 3650 day was divided into batches. A batch size of 5 was used in all experiments. This means that after working and training the model through 5 images, in the training phase, the model gets updated. The total number of batches was calculated as follows:

$$\text{Total number of batches} = \frac{\text{Total data files}}{\text{Batch size}} = \frac{3650}{5} = 730.$$

Moreover, the epoch number chosen was 25, which means that the model will go over the training datasets for 25 times. Therefore, the weights of the model will be updated at the end of each batch as shown in figure 3.1. Each epoch will update the model 730 times because the total number of the batches is 730. Consequently, the weights of the model will be updated 18,250 times as shown below:

$$\text{Model weights updates} = \text{Epoch number} \times \text{Total batches} = 25 \times 730 = 18,250.$$
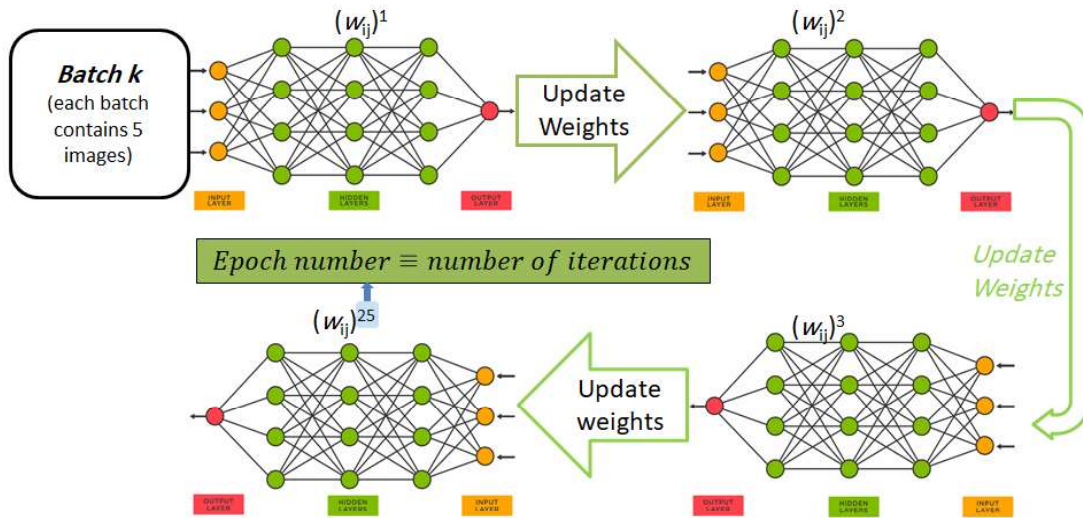
Figure 3.1: Model Training

## 3.2 Learning rate

Initially, the weights of the neural network between the layers will hold arbitrary values. Further on, these weights will updated after each iteration using the optimization function. The ideal scenario is to reach a zero loss, which represents the error in the neural network. The faster the loss is reached and the lower it becomes, the more optimized is the model. Learning rate is a hyper-parameter that controls the speed of model and how it adapts to new changes after each iteration. As such, the learning rate is chosen to maximize accuracy and minimize the error. After conducting several tests, the graph of the loss function can be drawn and the optimal value of this hyper-parameter can be captured as shown in figure 3.2. The same idea pops up when dealing with neural networks and talking about the learning rate. Upon conducting several tests the results will be captured and analyzed to choose which model shows lower losses and higher level of accuracy. It is highly recommended to know that not only the learning rate will have a reflect on the speed of training but also on the results appearing. Upon using low learning rates, a lot of time steps will be required to reach the optimal value of this hyper-parameter. On the other hand, large values of leaning rates will lead to a divergence. Therefore, the conducted experiments in this research included different values of learning rate. Based on results of each experiment, the best model was chosen. The whole training was carried out for 19 different values of learning rate. Their results were stored and included the loss and accuracy over the three studied classes that are:

- Background: class 0 that represents areas with no eddies

22

**Too low**

A small learning rate requires many updates before reaching the minimum point

**Just right**

The optimal learning rate swiftly reaches the minimum point

**Too high**

Too large of a learning rate causes drastic updates which lead to divergent behaviors
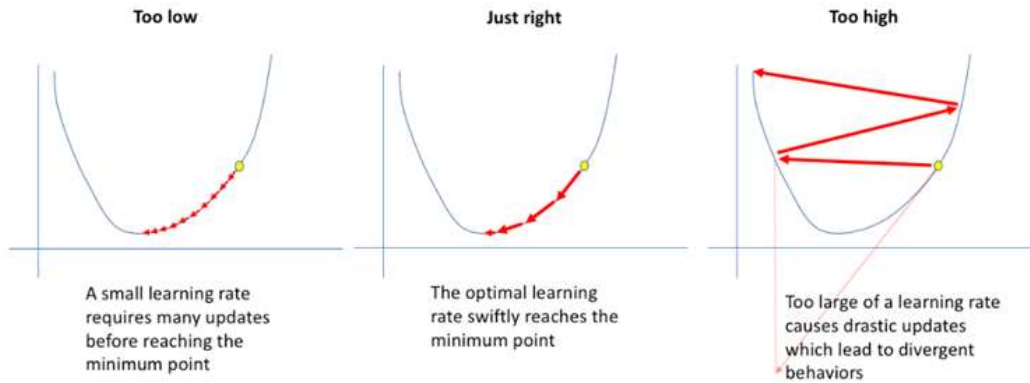
Figure 3.2: Learning rate effect

- Anticyclonic eddies: class 1 that represents areas with anticyclonic eddies

- Cyclonic eddies: class 2 that represents areas with cyclonic eddies

The results of these conducted experiments were observed using tensorboard and were presented based on six metrics. The metrics are: training accuracy, training intersection over union (IOU), validation accuracy, validation IOU, training loss and validation loss.

Training accuracy helps understand how well the model is accurate in the training phase. Validation accuracy gives a prediction how well the model will operate when new data is to be used. Both accuracies represent the percentage of correct labeled pixels. If the model shows high training accuracy but low validation accuracy this is an indication that the model is losing the ability to provide better predictions when using new data.

The IOU is a metric used to evaluate the detection accuracy of the model. Once the model is used for a certain day, there will be areas detected as eddies similar to the provided solutions and other areas that weren't part of the solution. Therefore, dividing the area that overlaps the prediction results with segmented images by the union of them results in the IOU metric. The ideal case is to have it equal to 1 and the worst case is to be equal to 0 as shown in figure 3.3.

The loss metric refers to the use of cross entropy loss function that is calculated based on following formula:

$$L_{CE} = -\sum_{i=1}^{N} t_i log(p_i),$$

where ti is true value and pi is the probability value appearing at the end. Based on the final probability values a pixel can be identified of what class as show in figure 3.4 that includes a pixel identification example. The highest probability class was chosen to represent the pixel, which was in the example cyclonic class.

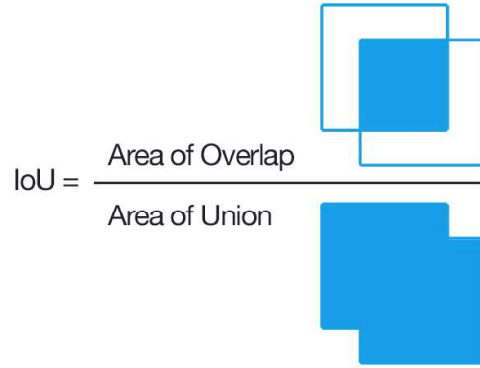The metric results were plotted and demonstrated in figures (3.5, 3.6, 3.7) and tables (3.1, 3.2, 3.3).
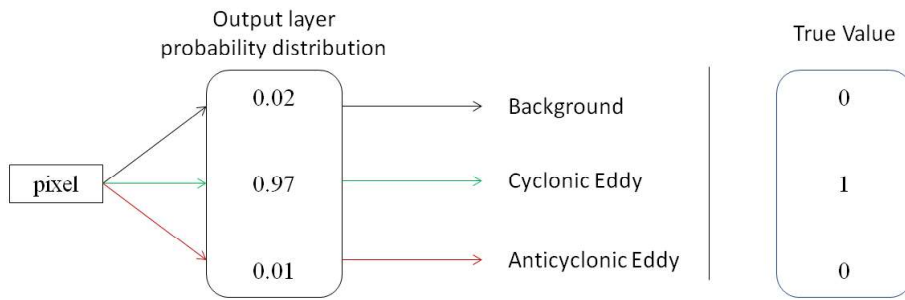


Figure 3.3: IOU calculation



Figure 3.4: Loss function example

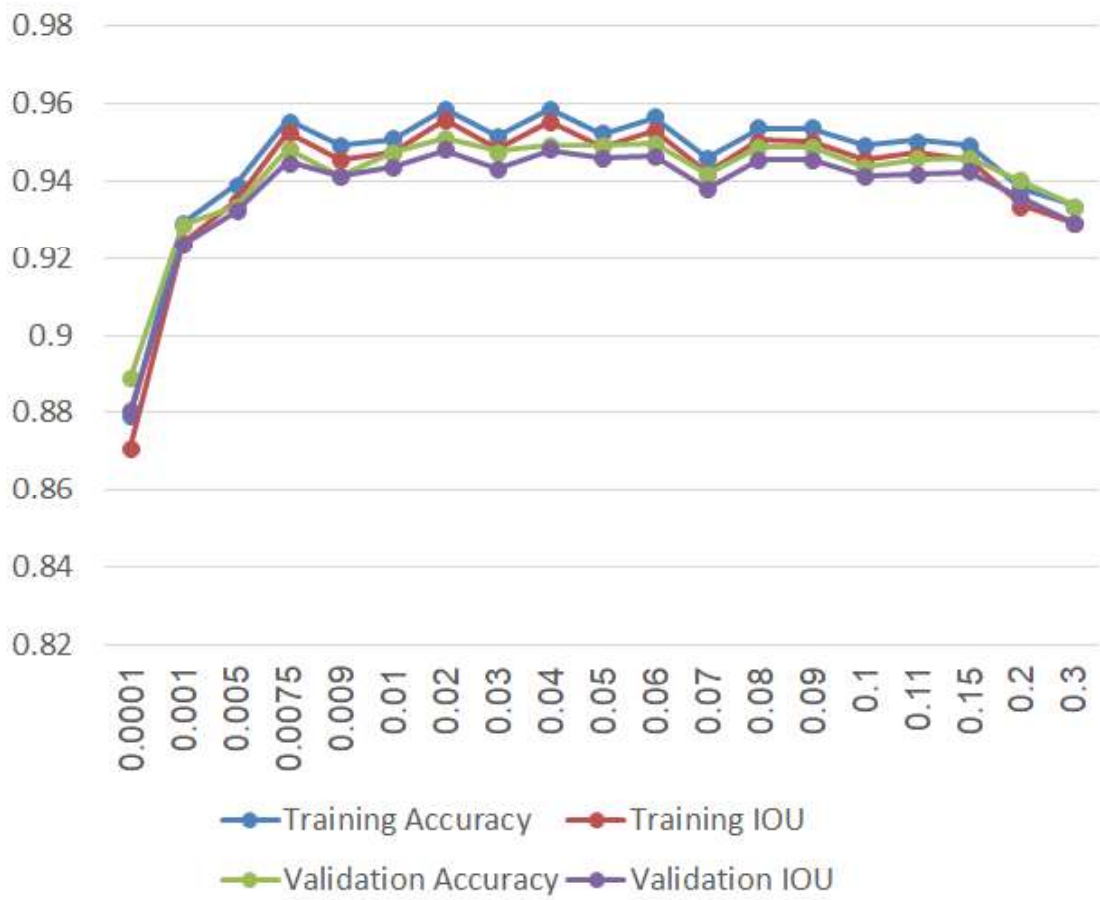$$L_{CE} = -\sum_{i=1}^{N} t_i log(p_i) = (0log(0.02) + 1log(0.97) + 0log(0.01)) = 0.0132$$

Figure 3.5: Class 0 metrics

Table 3.1: Class 0 results

| Learning Rate | Train ACC | Train IOU | valid ACC | valid IOU |
|:---:|:---:|:---:|:---:|:---:|
| 0.0001 | 0.8792 | 0.8712 | 0.889 | 0.8807 |
| 0.0075 | 0.9556 | 0.9526 | 0.9483 | 0.945 |
| 0.001 | 0.929 | 0.924 | 0.9286 | 0.9236 |
| 0.005 | 0.9389 | 0.9347 | 0.9336 | 0.9322 |
| 0.009 | 0.9491 | 0.9454 | 0.9415 | 0.9412 |
| 0.01 | 0.9507 | 0.9473 | 0.9474 | 0.9439 |
| 0.02 | 0.9585 | 0.9557 | 0.9511 | 0.948 |
| 0.03 | 0.9515 | 0.9482 | 0.9477 | 0.9433 |
| 0.04 | 0.9585 | 0.9555 | 0.9494 | 0.9481 |
| 0.05 | 0.9523 | 0.949 | 0.9493 | 0.946 |
| 0.06 | 0.9562 | 0.9532 | 0.9497 | 0.9465 |
| 0.07 | 0.9458 | 0.9421 | 0.9418 | 0.9379 |
| 0.08 | 0.9537 | 0.9506 | 0.9486 | 0.9454 |
| 0.09 | 0.9535 | 0.9503 | 0.9487 | 0.9454 |
| 0.1 | 0.9491 | 0.9455 | 0.9438 | 0.9413 |
| 0.11 | 0.9506 | 0.9472 | 0.9456 | 0.942 |
| 0.15 | 0.9491 | 0.9456 | 0.946 | 0.9424 |
| 0.2 | 0.938 | 0.9337 | 0.9401 | 0.9359 |
| 0.3 | 0.9337 | 0.9291 | 0.9337 | 0.9291 |
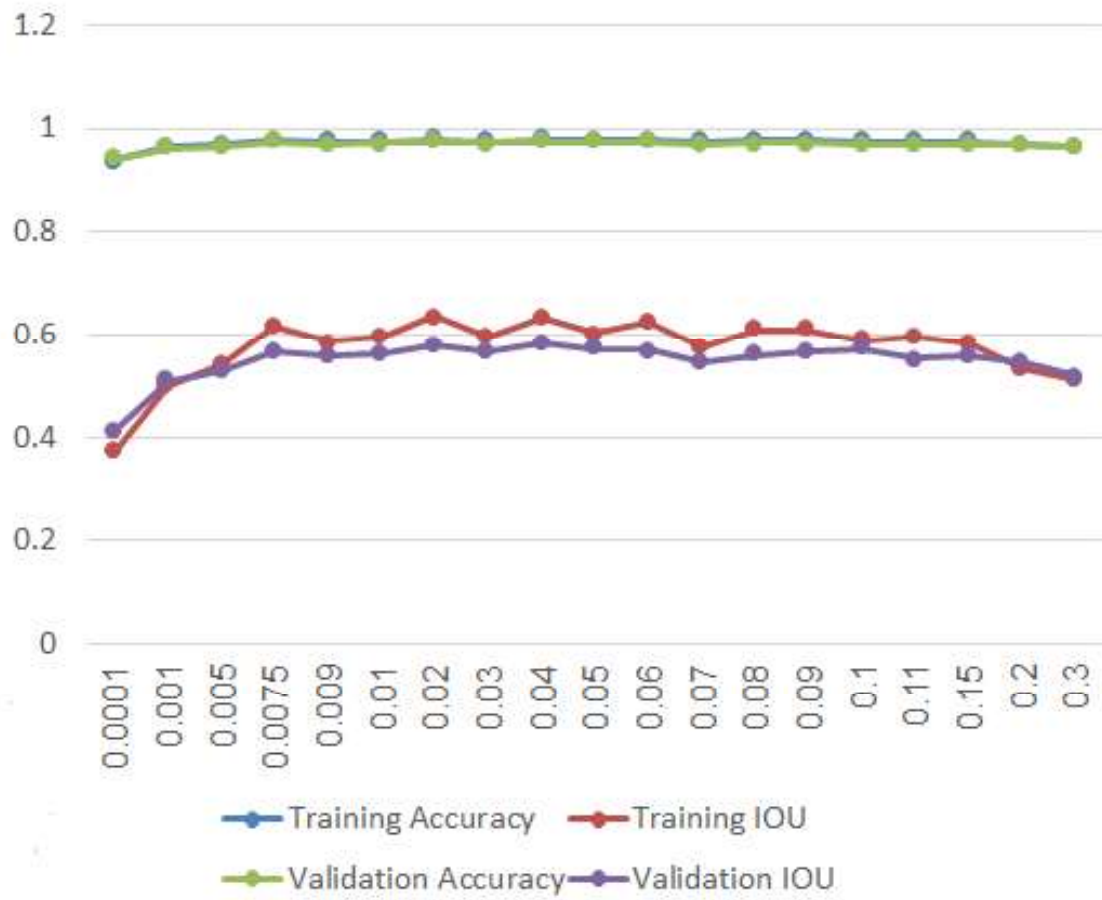
Figure 3.6: Class 1 metrics

Table 3.2: Class 1 results

| Learning Rate | Train ACC | Train IOU | valid ACC | valid IOU |
|:---:|:---:|:---:|:---:|:---:|
| 0.0001 | 0.939 | 0.3702 | 0.9417 | 0.4119 |
| 0.0075 | 0.9773 | 0.6191 | 0.9726 | 0.5659 |
| 0.001 | 0.9638 | 0.5011 | 0.9626 | 0.5092 |
| 0.005 | 0.9691 | 0.541 | 0.9662 | 0.5289 |
| 0.009 | 0.9741 | 0.5867 | 0.9695 | 0.5575 |
| 0.01 | 0.9749 | 0.5945 | 0.9716 | 0.5639 |
| 0.02 | 0.9789 | 0.6362 | 0.9741 | 0.5793 |
| 0.03 | 0.9752 | 0.5977 | 0.9718 | 0.5651 |
| 0.04 | 0.9788 | 0.6357 | 0.9741 | 0.5837 |
| 0.05 | 0.9758 | 0.6036 | 0.9733 | 0.5714 |
| 0.06 | 0.9777 | 0.6237 | 0.9732 | 0.57 |
| 0.07 | 0.9727 | 0.5738 | 0.9691 | 0.5462 |
| 0.08 | 0.9764 | 0.6095 | 0.9723 | 0.5593 |
| 0.09 | 0.9763 | 0.6095 | 0.972 | 0.5657 |
| 0.1 | 0.9742 | 0.5874 | 0.9711 | 0.5726 |
| 0.11 | 0.9749 | 0.595 | 0.9709 | 0.5532 |
| 0.15 | 0.9738 | 0.5844 | 0.9703 | 0.5583 |
| 0.2 | 0.968 | 0.5339 | 0.9678 | 0.5439 |
| 0.3 | 0.9654 | 0.5137 | 0.9641 | 0.5193 |
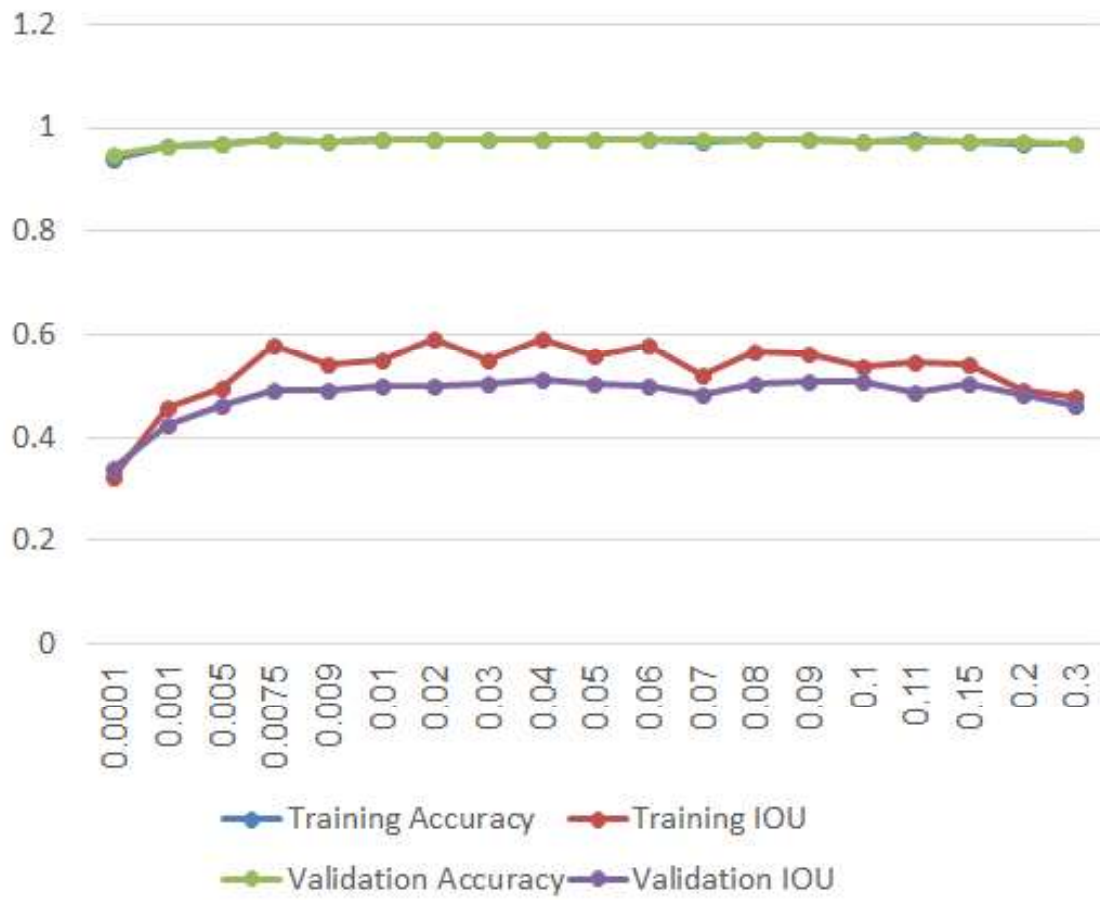
Figure 3.7: Class 2 metrics

| Table 3.3: Class 2 results | | | | |
|---|---|---|---|---|
| Learning Rate | Train ACC | Train IOU | valid ACC | valid IOU |
| 0.0001 | 0.9391 | 0.3219 | 0.9462 | 0.3423 |
| 0.0075 | 0.9781 | 0.5774 | 0.9755 | 0.4906 |
| 0.001 | 0.9648 | 0.4561 | 0.96533 | 0.425 |
| 0.005 | 0.9695 | 0.4936 | 0.97 | 0.4616 |
| 0.009 | 0.9747 | 0.5396 | 0.9737 | 0.4918 |
| 0.01 | 0.9755 | 0.549 | 0.9754 | 0.5007 |
| 0.02 | 0.9795 | 0.5926 | 0.9767 | 0.4982 |
| 0.03 | 0.976 | 0.5509 | 0.9756 | 0.5018 |
| 0.04 | 0.9794 | 0.591 | 0.9767 | 0.512 |
| 0.05 | 0.9762 | 0.5567 | 0.9756 | 0.5023 |
| 0.06 | 0.9783 | 0.5785 | 0.9767 | 0.4982 |
| 0.07 | 0.9727 | 0.5206 | 0.9758 | 0.483 |
| 0.08 | 0.9771 | 0.5657 | 0.9761 | 0.5049 |
| 0.09 | 0.9768 | 0.5619 | 0.9764 | 0.5096 |
| 0.1 | 0.9745 | 0.5376 | 0.9744 | 0.507 |
| 0.11 | 0.9754 | 0.5467 | 0.9743 | 0.4877 |
| 0.15 | 0.9749 | 0.5422 | 0.9753 | 0.5047 |
| 0.2 | 0.9695 | 0.4925 | 0.9719 | 0.4833 |
| 0.3 | 0.9678 | 0.4785 | 0.969 | 0.4599 |

The following results showed that the best models were the ones working under 0.02, 0.04 and 0.06 learning rates. They achieved the highest training and validation accuracy in addition to the highest IOU. The loss data of the 0.02 learning rate model was plotted below in figure 3.8. After Epoch number 22, there was approximately no change in the loss results that appeared in epoch number 23 so we stopped before the last two iterations. Although Validation loss is 0.7, the objective of detecting eddies was completed and this was verified when using the model and through the accuracy results. Moreover, the model is tested in the second section and the aim of detecting eddy location was achieved.
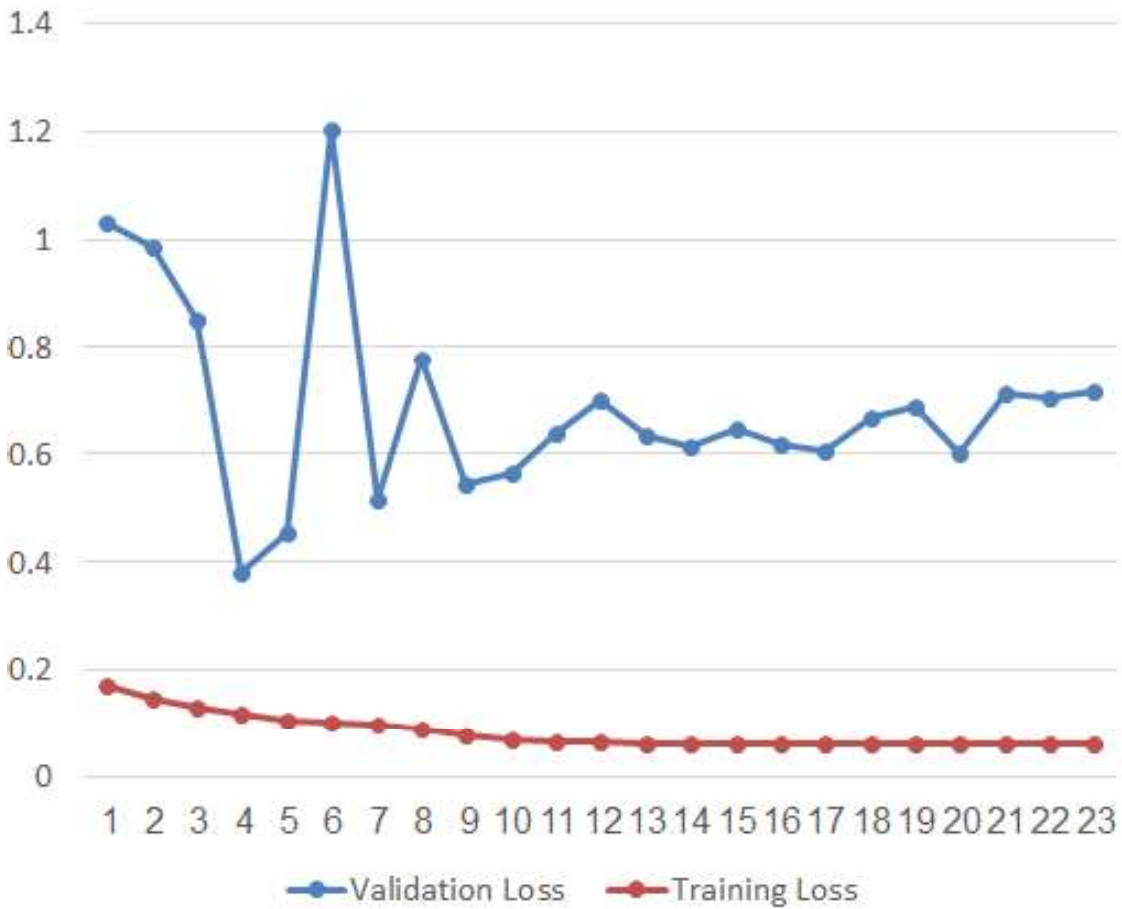


Figure 3.8: Class 2 metrics

## 3.3    Testing the model

Upon using any of the available models that were trained, what will be predicted at the end for each pixel is three probability values for each class. The difference

between each trained model will be in the weight of the connections between the layers that was affected by the learning rate. A sigmoid function is used in the last layer of the neural network as an activation function. The sigmoid function was used to represent results at the end as probabilities due to its shape represented in figure 3.9. Therefore, the class with highest probability is chosen
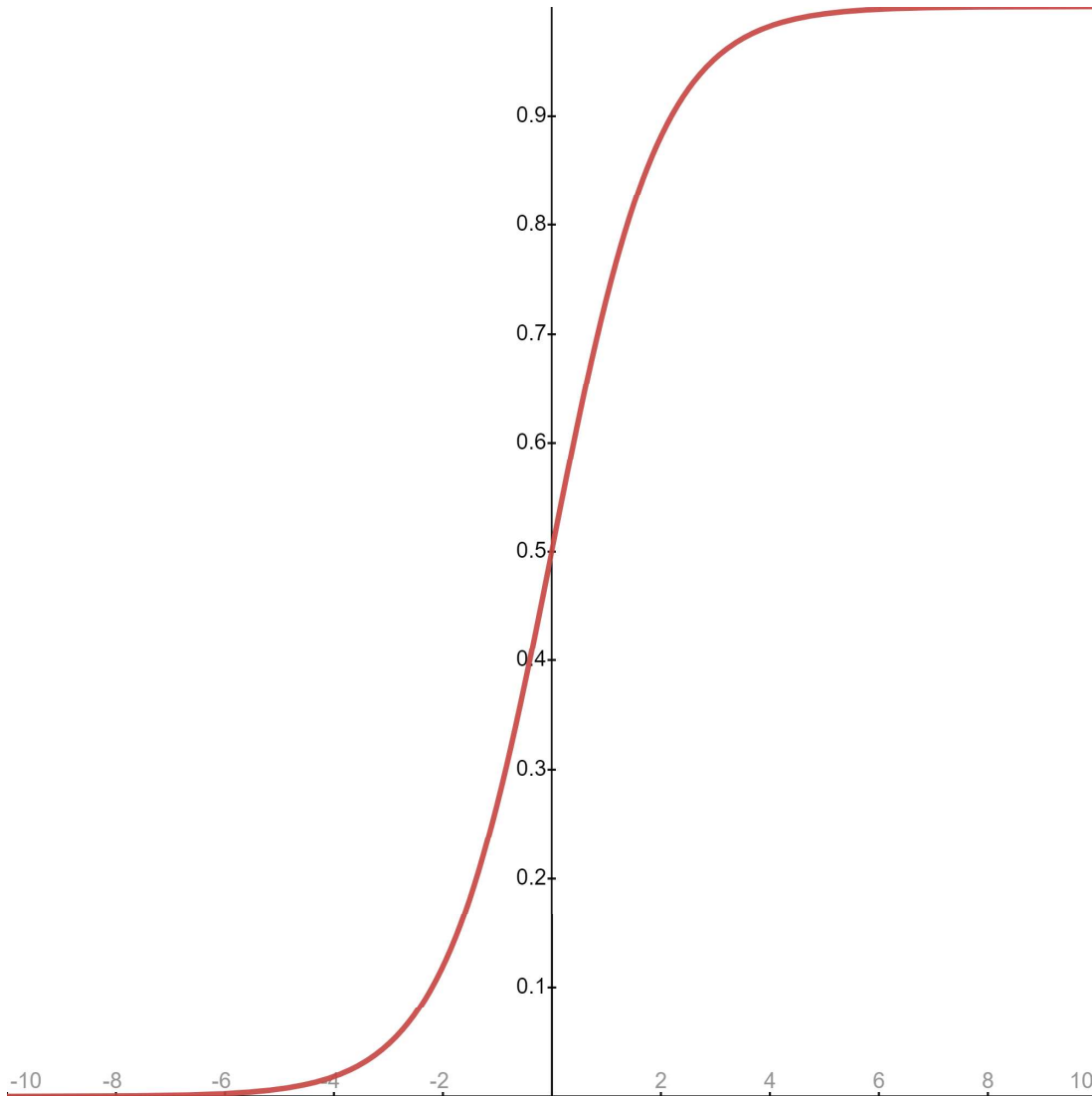


Figure 3.9: Sigmoid function

to represent the pixel and at the end new images were generated accordingly as shown in figure 3.10.

All detected eddies show a circular and elliptic shape and there is a big range in the sizes. The island and land areas show no eddy detection as desired. This was due to having no labelled area there during the manual labelling phase and results appeared to be convenient with what intended.
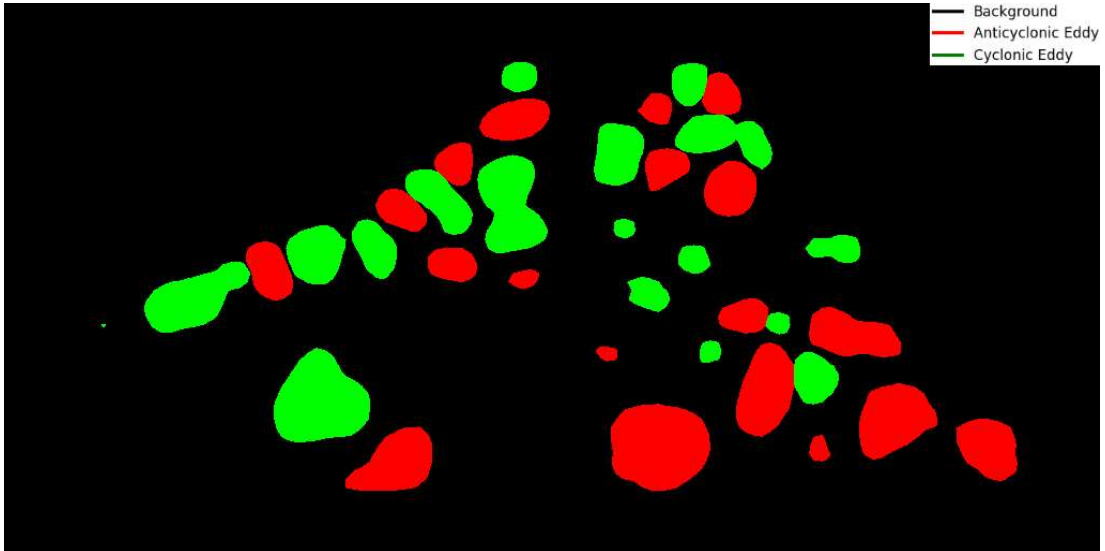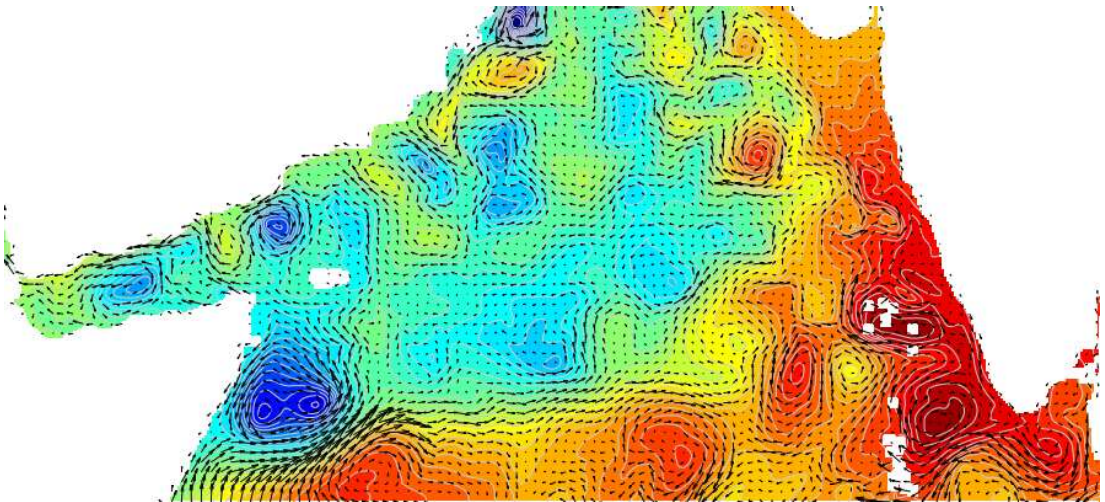
Figure 3.10: Model Prediction



Figure 3.11: Selected day image

# CHAPTER 4

# CONCLUSION

Earlier eddy detection algorithms were only based on one or two parameters. Upon understanding the effects that an eddy shows in water, manual segmentation using labelme was done. It covered 4015 images that include SSH, SST, velocity field and Salinity datasets available. The new model receives these semantically segmented images as solutions and 5 input parameters. The training data included images from the year 2001 till 2010 (3650 days) under different learning rates. Then the model was validated using the last year's (2011) images. The data was divided in this order to make sure that training and validation were done during all times of the year. It was proven that the most accurate model was the one operating with a learning rate of 0.02. On the other hand, accuracy increased by nearly 10% on average for all classes. Add to that, the model can be used at any sea surface level, which gives it a privilege over other models. Further on, what will be added as a sort of filtering mechanism that removes undesirable detection that appears to be not an eddy. More tests will also be conducted to understand the correlations and relations between the input parameters. Moreover, different types of neural networks will be used and experimented with the addition of instance segmented images used as solutions.

# Appendix A

# TITLE OF APPENDIX

| | |
|---|---|
| SSH | Sea Surface Height |
| SST | Sea Surface Temperature |
| AC | Anticyclonic |
| C | Cyclonic |
| SAR | Synthetic Aperture Radar |
| SDR | success detection rate |
| EDR | excess detection rate |
| EKE | eddy kinetic energy |
| SVM | Support Vector Machines |
| NEMO | Nucleus for European Models of the Ocean |
| CMEMS | Copernicus Marine and Environment Monitoring Service |
| DC | dice coefficient |
| PSPNet | pyramid scene parsing network |
| VG | vector geometry based |
| BISENET | Bilateral Segmentation Network |
| FCN | Fully Convolutional Network |

# BIBLIOGRAPHY

[1] E. Underwood, "What happens when ocean eddies hit a wall?," *Eos*, Apr 2017.

[2] D. Rivas, A. Mitre-Apaez, and E. García-Mendoza, "Formation of a subsurface anticyclonic eddy from the california undercurrent off northwestern baja california, mexico," *Journal of Marine Systems*, p. 103707, 2022.

[3] C. Dong, J. C. McWilliams, Y. Liu, and D. Chen, "Global heat and salt transports by eddy movement," *Nature Communications*, vol. 5, no. 1, pp. 1–6, 2014.

[4] Y. Cotroneo, G. Aulicino, S. Ruiz, A. Pascual, G. Budillon, G. Fusco, and J. Tintoré, "Glider and satellite high resolution monitoring of a mesoscale eddy in the algerian basin: Effects on the mixed layer depth and biochemistry," *Journal of Marine Systems*, vol. 162, pp. 73–88, 2016.

[5] P. Gaube, D. J. McGillicuddy Jr, D. B. Chelton, M. J. Behrenfeld, and P. G. Strutton, "Regional variations in the influence of mesoscale eddies on near-surface chlorophyll," *Journal of Geophysical Research: Oceans*, vol. 119, no. 12, pp. 8195–8220, 2014.

[6] A. Bakun, "Fronts and eddies as key structures in the habitat of marine fish larvae: opportunity, adaptive response and competitive advantage," *Scientia Marina*, vol. 70, no. S2, pp. 105–122, 2006.

[7] G. Chen, J. Gan, Q. Xie, X. Chu, D. Wang, and Y. Hou, "Eddy heat and salt transports in the south china sea and their seasonal modulations," *Journal of Geophysical Research: Oceans*, vol. 117, no. C5, 2012.

[8] D. B. Chelton, M. G. Schlax, R. M. Samelson, and R. A. de Szoeke, "Global observations of large oceanic eddies," *Geophysical Research Letters*, vol. 34, no. 15, 2007.

[9] C. Xu, X.-D. Shang, and R. X. Huang, "Estimate of eddy energy generation/dissipation rate in the world ocean from altimetry data," *Ocean Dynamics*, vol. 61, no. 4, pp. 525–541, 2011.

[10] L. Brannigan, "Intense submesoscale upwelling in anticyclonic eddies," *Geophysical Research Letters*, vol. 43, no. 7, pp. 3360–3369, 2016.

[11] E. Lee-Sánchez, V. F. Camacho-Ibar, J. A. Velásquez-Aristizábal, J. A. Valencia-Gasti, and G. Samperio-Ramos, "Impacts of mesoscale eddies on the nitrate distribution in the deep-water region of the gulf of mexico," *Journal of Marine Systems*, p. 103721, 2022.

[12] D. J. McGillicuddy Jr, "Mechanisms of physical-biological-biogeochemical interaction at the oceanic mesoscale," *Annual Review of Marine Science*, vol. 8, pp. 125–159, 2016.

[13] D. J. McGillicuddy Jr, L. A. Anderson, N. R. Bates, T. Bibby, K. O. Buesseler, C. A. Carlson, C. S. Davis, C. Ewart, P. G. Falkowski, S. A. Goldthwait, *et al.*, "Eddy/wind interactions stimulate extraordinary mid-ocean plankton blooms," *Science*, vol. 316, no. 5827, pp. 1021–1026, 2007.

[14] G. Xu, C. Dong, Y. Liu, P. Gaube, and J. Yang, "Chlorophyll rings around ocean eddies in the north pacific," *Scientific reports*, vol. 9, no. 1, pp. 1–8, 2019.

[15] A. Arur, P. Krishnan, G. George, M. Goutham Bharathi, M. Kaliyamoorthy, K. Hareef Baba Shaeb, A. Suryavanshi, T. S. Kumar, and A. Joshi, "The influence of mesoscale eddies on a commercial fishery in the coastal waters of the andaman and nicobar islands, india," *International Journal of Remote Sensing*, vol. 35, no. 17, pp. 6418–6443, 2014.

[16] L. Bengtsson, S. Hagemann, and K. I. Hodges, "Can climate trends be calculated from reanalysis data?," *Journal of Geophysical Research: Atmospheres*, vol. 109, no. D11, 2004.

[17] H. Murakami, "Tropical cyclones in reanalysis data sets," *Geophysical Research Letters*, vol. 41, no. 6, pp. 2133–2141, 2014.

[18] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[19] L. Zhang, L. Zhang, and B. Du, "Deep learning for remote sensing data: A technical tutorial on the state of the art," *IEEE Geoscience and Remote Sensing Magazine*, vol. 4, no. 2, pp. 22–40, 2016.

[20] B. Pandey, D. K. Pandey, B. P. Mishra, and W. Rhmann, "A comprehensive survey of deep learning in the field of medical imaging and medical natural language processing: Challenges and research directions," *Journal of King Saud University-Computer and Information Sciences*, 2021.

[21] Y. Tian and S. Fu, "A descriptive framework for the field of deep learning applications in medical images," *Knowledge-Based Systems*, vol. 210, p. 106445, 2020.

[22] H. Jung, M.-K. Choi, J. Jung, J.-H. Lee, S. Kwon, and W. Young Jung, "Resnet-based vehicle classification and localization in traffic surveillance systems," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 61–67, 2017.

[23] Z. Zhang, "Artificial neural network," in *Multivariate Time Series Analysis in Climate and Environmental Research*, pp. 1–35, Springer, 2018.

[24] H. Kukreja, N. Bharath, C. Siddesh, and S. Kuldeep, "An introduction to artificial neural network," *Int J Adv Res Innov Ideas Educ*, vol. 1, pp. 27–30, 2016.

[25] S. B. Maind, P. Wankar, *et al.*, "Research paper on basic of artificial neural network," *International Journal on Recent and Innovation Trends in Computing and Communication*, vol. 2, no. 1, pp. 96–100, 2014.

[26] A. Okubo, "Horizontal dispersion of floatable particles in the vicinity of velocity singularities such as convergences," in *Deep Sea Research and Oceanographic Abstracts*, vol. 17, pp. 445–454, Elsevier, 1970.

[27] J. Weiss, "The dynamics of enstrophy transfer in two-dimensional hydrodynamics," *Physica D: Nonlinear Phenomena*, vol. 48, no. 2-3, pp. 273–294, 1991.

[28] A. K. Liu, C. Y. Peng, and J. D. Schumacher, "Wave-current interaction study in the gulf of alaska for detection of eddies by synthetic aperture radar," *Journal of Geophysical Research: Oceans*, vol. 99, no. C5, pp. 10075–10085, 1994.

[29] A. Gopalan, V. Krishna, M. Ali, and R. Sharma, "Detection of bay of bengal eddies from topex and in situ observations," *Journal of Marine Research*, vol. 58, no. 5, pp. 721–734, 2000.

[30] X. X. Zhu, D. Tuia, L. Mou, G.-S. Xia, L. Zhang, F. Xu, and F. Fraundorfer, "Deep learning in remote sensing: A comprehensive review and list of resources," *IEEE Geoscience and Remote Sensing Magazine*, vol. 5, no. 4, pp. 8–36, 2017.

[31] 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," *Journal of Atmospheric and Oceanic Technology*, vol. 27, no. 3, pp. 564–579, 2010.

[32] A. Tussadiah, M. L. Syamsuddin, W. S. Pranowo, N. P. Purba, and I. Riyantini, "Eddy vertical structure in southern java indian ocean: Identification using automated eddies detection," *International Journal of Science and Research*, vol. 5, no. 3, pp. 967–971, 2016.

[33] M. D. Ashkezari, C. N. Hill, C. N. Follett, G. Forget, and M. J. Follows, "Oceanic eddy detection and lifetime forecast using machine learning methods," *Geophysical Research Letters*, vol. 43, no. 23, pp. 12–234, 2016.

[34] R. Lguensat, M. Sun, R. Fablet, P. Tandeo, E. Mason, and G. Chen, "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, IEEE, 2018.

[35] G. Xu, C. Cheng, W. Yang, W. Xie, L. Kong, R. Hang, F. Ma, C. Dong, and J. Yang, "Oceanic eddy identification using an ai scheme," *Remote Sensing*, vol. 11, no. 11, p. 1349, 2019.

[36] G. Xu, W. Xie, C. Dong, and X. Gao, "Application of three deep learning schemes into oceanic eddy detection," *Frontiers in Marine Science*, vol. 8, p. 715, 2021.

[37] H. Zhang, K. Dana, J. Shi, Z. Zhang, X. Wang, A. Tyagi, and A. Agrawal, "Context encoding for semantic segmentation," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 7151–7160, 2018.

[38] A. Kirillov, K. He, R. Girshick, C. Rother, and P. Dollár, "Panoptic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9404–9413, 2019.