Towards an automated detection of the Gulf Stream North Wall from concurrent satellite images

Avijit Gangopadhyay
School for Marine Science and Technology
University of Massachusetts Dartmouth
Dartmouth, MA, USA
Avijit@umassd.edu

Kevin Lydon
Department of Computer Science
University of Massachusetts Dartmouth
Dartmouth, MA, USA
klydon@umassd.edu

Jeffrey A. Rezendes
Department of Computer Science
University of Massachusetts Dartmouth
Dartmouth, MA, USA
jrezendes@umassd.edu

Ramaprasad Balasubramanian
Department of Computer Science
University of Massachusetts Dartmouth
Dartmouth, MA, USA
r.bala@umassd.edu

Iren Valova
Department of Computer Science
University of Massachusetts Dartmouth
Dartmouth, MA USA
ivalova@umassd.edu

Abstract— Developing computational methods to automatically identify the Gulf Stream North Wall (GSNW) and similar currents in the ocean is a long-standing need for many types of operational ocean models. Specifically, the Feature-Oriented regional modeling system requires an accurate digitization of the GSNW and Rings (eddies) on a regular basis. Typical methods to determine its position and boundaries require skilled human operators to do a time-consuming manual extraction of visualized features. These experts are performing a feature extraction task that can be automated to save time, guarantee objectivity, and potentially increase precision.

In this paper we present first-results from two independent approaches of addressing this issue. In one of the approaches, the dynamical approach, the methodology begins by finding the most-likely bounds of iso-sea-surface-height contours within which the Gulf Stream north wall might fall. Other features, such as eddies, which are also captured, will be set-aside after a round of shape analysis. Any gap in the isoheight contours is filled with segments that are generated by combining the slopes from different heights. The second, a machine-learning approach uses an artificial neural network over a GSNW dataset, which has been generated weekly over past six years (2009-2015) by analysts. An artificial neural network is a type of learning algorithm designed as a system of neurons with connections among them. This neural network will first use the analyst-designated GSNW paths to determine the neural weights of the radial basis functions. Then the network will use the concurrent sea-surface height and temperature data that were used to identify those lines, and train itself to develop a smart network which will be able to identify GSNW paths from the concurrent satellite images on its own, with little to no human intervention.

In the long-term, we expect to merge the two techniques in a unique and unifying construct to be used operationally. A general approach of this methodology has the potential of being used for other similar operational modeling, reanalysis and skill assessment of numerical model system with data assimilation.

Keywords—Gulf Stream North Wall; Automated Detection; SSH; Neural Network

978-0-933957-43-5 ©2015 MTS

I. INTRODUCTION

Developing objective and automated methods to identify the Gulf Stream North Wall (GSNW) and similar currents in the ocean is a long-standing need for many types of operational ocean models such as those employed in MARACOOS and other similar operations [1,2,3,4,5]. Specifically, the Feature-Oriented regional modeling system requires an accurate digitization of the GSNW and Rings (eddies) on a regular basis [1,2,3]. To identify and extract the Gulf Stream North Wall (GSNW) from a multiple set of satellite images is time-consuming for manual operators. However, automating this process is not straightforward. While the Gulf Stream transports heat northward in the Atlantic Ocean, the North Wall is not an isotherm. In addition to this, sea surface temperature measurements are very susceptible to occlusion, with datasets often missing entire regions of importance needed for feature detection [5].

In addition to sea surface temperature, sea surface height and sea surface velocity (derived from height) can be an indicator of the stream's location. The stream travels more quickly than the surrounding water and due to its speed the height difference (or pressure gradient) is also greater than the surrounding region. Altimeter derived sea surface height anomalies (height, in short) are a possible identifier for the GS axis or north wall [6]. It is possible to find an isoheight contour within the bounds of the Gulf Stream, but there is no uniform height or velocity that can possibly be a synonym to the stream's north wall.

In difference to our previous work on utilizing SST and neural network for feature detection [7,8,9], we investigate here the possibility of using SSH and neural network in a synergistic way. In this paper, we present the results from a first attempt from both approaches starting with a dynamical approach using the sea surface height to approximate the position of the GSNW. Points along the isoheight contours are separated into groups based on proximity and a single group is selected as the backbone of our Gulf Stream model.

The Dynamical approach and its results are discussed first. The Neural network methodology and its first results are described next. We present a comparison of both methods for a particular week in June 2014 then. Finally we discuss the findings of this preliminary study and outline a number of recommendations for carrying out this research in the near future to fruition.

II. DYNAMICAL APPROACH

Since our current technique relies on first extracting the isoheight contour, there is still an open question as to what height value would result in the closest approximation to the North Wall. Currently we are using a heuristic, extracting the isoheight contours at 25 centimeters above the reference ellipsoid. It seems that this contour is always found somewhere within the Gulf Stream and within 2014's historical data an extraction at this height always results in a feature within the bounds of the Gulf Stream.

It is possible that a closer contour can be chosen – optimized for each dataset – through some analysis of the surface speed or vorticity, given that the stream's shape is determined by its motion. This might be an important step in improving the current algorithm.

We begin by extracting the 25-centimeter contour from the sea surface height data supplied by Aviso. The height data has a ground sample distance (GSD) of 0.25 degrees. For each two consecutive data points in every row and column, we check to see if the contour of 25 centimeters should fall between their respective height values. If the contour falls between them, the arc length between the points and the difference in their height values are both calculated. Using these values and assuming a linear distribution of height across the arc, we approximate a point where the 25 cm value occurs. The new point is selected as part of the isoheight contour. This process results in a set of coordinates representing the isoheight contour.

The set of coordinates is then separated into groups based on proximity. First, each latitude/longitude pair is considered its own group. Then, groups are compared to determine if any latitude/longitude pair in one group is in close proximity to a latitude/longitude pair in the other. If two groups are in close proximity, they are combined to form a single group. Proximity of coordinates is determined based on great-circle distance. Any pairs within 34 kilometers of one another are considered "close". This distance is tentative and was chosen because of the height data's inherent granularity. Consecutive points in the region of interest can be anywhere between 17 and 39 kilometers apart. Typically, the isoheight contours are captured at slightly finer granularity, so distances higher than 34 kilometers have led to false positives in the group combination process.

Once groups have been separated, a procedure to determine selection of the Gulf Stream group begins. This is done by comparing the longitudinal lengths of the groups. The group that spans the longest west-east stretch is selected. The other groups, some of which are eddies, are set aside for future analysis.

III. RESULTS FROM DYNAMICAL APPROACH

The above methodology has been applied to a set of SSH images for selected weeks of January, March, June, August and October of 2014. These weeks were selected based on the level of complexity of the Gulf Stream path configuration. During January and March, the straight/wavy meandering of the GS path was reasonably well identified by this approach (See Figure 1) as matched with the analyst's path. An added advantage of the usage of SSH contouring idea is the avoidance of ambiguity in detection of the GSNW path for regions where the SST data is unavailable for a week of two due to clouds/winter storms. As an example, see Fig. 1 right panel for the region from 62W to 50W.

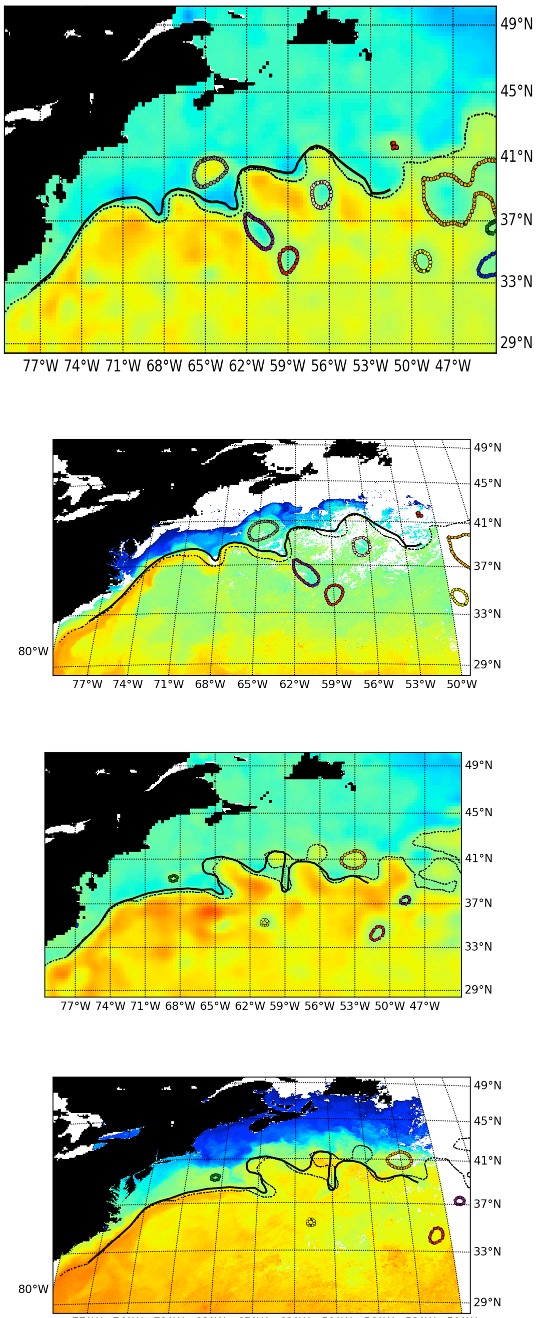


Figure 1: Visualization of feature extraction results for historical data from 24 March 2014 (Upper two panels) and for 20 October 2014 (lower two panels). The solid black line is the GSNW extracted by a manual operator. The dotted black line is the GSNW extracted by the automated algorithm. The other dotted lines are for other extracted features which could be eddies or other anomalous features. The lines are superposed on the corresponding Sea surface height field in color (top and third from top) and on Sea Surface temperature (second from top and bottom).

Our preliminary results are encouraging. Typically the isoheight contour that is being extracted is fairly comparable to the operator's manually extracted path from the historical data. The automated technique tends to deviate most from the operator's path in places where eddies are being formed or have recently formed. Another circumstance that seems to cause disagreement between our technique and the operator's path is westward stream movement, which happens over subsections of the stream.

It is difficult to determine when to classify some parts of the isoheight contour as eddies rather than as part of the stream since many eddies are parts of the stream which have diverged from its body. It might not be possible to do this accurately while considering only sea surface height. Another potential problem is data unavailability in time. While the height value we use for our contour seems quite good for some days, it deviates farther from the operator's path on others. Seasonal changes probably dictate the best fit for the height value, but our current technique does not take this time-dependence into account.

Another hurdle we have yet to overcome is that of validation. Currently human operators carry out identification of the GSNW. In order to validate our techniques, some comparison between our results and the results of manual extraction is necessary. We are still considering available distance measurements to calculate the "closeness" of the two paths. Even when this is completed, we will be comparing our results with manual extractions, which are prone to human error. We recognize that this is not the same as comparison to the actual GSNW coordinates, as the manual extraction is simply an approximation susceptible to the inaccuracies of the human eye.

Note that so far, the technique has only been applied to SSH. The results from applying the technique to SSH have been superimposed onto SSH images and SST images from the same day. The reason for this is to reinforce an understanding of what this technique is doing and compare the isoheight contour that is being extracted with another key GSNW indicator (SST).

IV. Neural Network Approach

Next, we present a technique for GSNW detection that is being developed using a multilayer perceptron as the basis for the detection. The complexity of a problem like this lends itself to being solved with a learning, approximation algorithm such as an artificial neural network. The long-term goal of this project is to test and compare a variety of neural network and other learning approaches to GSNW detection. The first technique chosen was an MLP because of its simplicity in comparison to other types of neural networks.

Multilayer Perceptrons (MLP) networks are commonly used to solve approximation problems. MLP are feed forward networks with one or more hidden layers of nonlinear perceptron elements. They usually employ sigmoid functions as nonlinearities and a version of backpropagation learning algorithm for training.

The MLP is designed to act as a nonlinear amplifier. The input sequence used for the training should be selected so that it spans the entire amplitude range of the input signal. The usual MLP topology is shown in Fig. 1. The NN is trained by backpropagation to eliminate the nonlinear signal distortions and improve the quality of signal transmission.

Backpropagation represents a supervised learning approach, where the weight adjustment is guided by the teaching signal and the penalty/reward of the error in the NN response. It adjusts its weights via the teaching signal (expected output) and usually utilizes least mean square error to perform the training process. MLP are feed forward networks with one or more hidden layers of nonlinear perceptron elements between the input and output layers. In our experiments we utilize a single hidden layer as multiple researchers have shown its suitability for majority of cases.

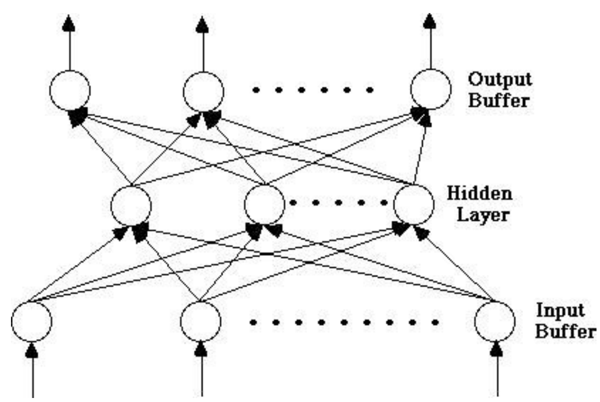


Figure.2: The MLP topology for performing the equalization of communication channels

The input layer introduces input values into the network. Input units are special, not applying any activation function or other processing to their values. The hidden layer performs processing and classification of features, which are patterns helping in determining the output. Any number of hidden layers can be used, but as the number of the layers increases, the learning algorithm (usually standard backpropagation or its different modifications) becomes more difficult to implement, which is another reason to use the smallest number of hidden layers, that are capable to solve the specific problem. The only difference is that the final outputs of the output are passed on to the world outside the neural network.

The output Y of a two hidden layer perceptron model with r units in the first layer, s units in the second layer, and input $X = [x_1, x_2 \ldots, x_n]$ is

$$(1) \quad Y = \sum_{i=1}^{s} \quad d_k \sigma(\sum_{i=1}^{r} \quad c_{ik} \sigma(W^{ik} X - \theta_{ik}) - \gamma_k)$$

where W^{ik} is the weight vector between the k-th of the input and the i-th units of the hidden layer, θ_{ik} is the threshold of the i-th unit of the k-th hidden layer, σ is the activation function (for example Heaviside function, logistic sigmoid, Gaussian sigmoid etc), c_{ik} is the weight between the i-th unit of the hidden layer and the output. The process of determining the weights and thresholds is called learning or training. In the multilayer feedforward perceptron model the basic learning algorithm is the backpropagation, which is a gradient descent method [10,11].

The backpropagation algorithm (BP) is used in order to minimize the mean square error between the actual and the desired output of a MLP network. The input patterns are presented to the network in some sequence, during the learning phase. Then each of these training patterns is propagated forward layer by layer and then an output pattern is computed. This output is then compared to the desired output and according to this comparison an error value is determined. In order to reduce the error, the calculated error is used to adjust the weights on the connections coming into the output layer. Then the errors are propagated backwards towards the input layer, adjusting weights between hidden layers and between the input and first hidden layer, which results to the further reducing of the error. This procedure is repeated until no further reduction in error occurs, or the level of error encountered is between the acceptable limits.

V. Application of NN for GSNW Detection

The method used for this project involves two steps: data preprocessing and the MLP. Significant preprocessing of the data is required to convert it all into a format that is compatible with the neural network. That data is then fed into the MLP, which produces a classification of points - at a granularity of 0.25° of latitude and longitude - as being part of the north wall of the Gulf Stream or not. Figure 3 shows the GSNW network as implemented for this study.

Note that the number of hidden nodes is independent of the number of parameters. Adding more nodes will usually increase the accuracy of the network while also increasing the time for training. We tested values between 35 and 55 for the network and settled on 55 because it produced the best results. We considered larger numbers, but the network took too long to train. Training the best network with 55 hidden nodes took between 10 and 12 hours, which was the upper limit for what we considered was practical for our tests.

Also, it is not easy to define the relationships between the physical parameters and the hidden layers in the network. The network starts out with random biases and then adjusts them all during training in an effort to move toward a better classification rate. There isn't really one bias associated with each input parameter. The biases are associated with the connections between nodes, so, for each input parameter there is a separate bias on each of its 55 connections to the hidden nodes.

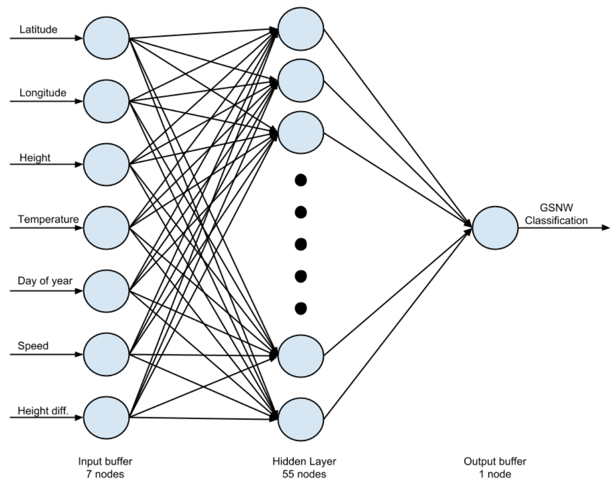


Figure 3. The GSNW Network. Each of the input attributes (on the left) for each point is fed into the network. The values are modified by the weights associated with each connection and each of the 55 hidden nodes. The output buffer produces a value between 0 and 1, indicating whether to classify the point as part of the GSNW (closer to 1) or not (closer to 0).

The data preprocessing starts with reading in all of the relevant data. For each week's worth of data, this included sea surface height, sea surface temperature, and, in the case of training and testing data, coordinates that made up the line plotted by an expert representing the gulf stream north wall. The granularity of each of these is different from the others. All of this information is mapped to a grid of coordinates, each 0.25° apart. For fields that exceeded this granularity, this is achieved by averaging all values corresponding to points within a range of 0.125° of each point. Once a value is determined in this manner for each field for each point in the grid, missing data points have to be filled in. There is significant missing data, particularly in the sea surface temperature. This is resolved for each point with a missing value by averaging values of each point adjacent to it. Once this is all done, the data is normalized to work better with the network. This normalized data is converted into training and testing datasets for running through the network. Multiple datasets are generated, using various combinations of the inputs: sea surface height, sea surface temperature, latitude, longitude, date (converted to day-of-year), speed, and largest height difference between a point and the points immediately to its left, top, and top-left. Multiples of each of these configurations is also generated with different ratios of GSNW to non-GSNW points, ranging from the true ratio to an even ratio for better representation of GSNW points.

For our network, we utilize a hidden layer with 55 neurons, 7 inputs and 1 output unit. Research has shown that two different learning rates (between input and hidden and between hidden and output) facilitate faster and better convergence. We also employ a momentum term designed to prevent wild swings of the neuron weight corrections due to outliers. These parameters are set to 0.0010 for learning rate input/hidden, 0.0005 for learning rate hidden/output and 0.0010 for momentum term. The data produced through preprocessing is fed into the network with a roughly 75/25 split of training to testing data. Training has been done using data for spring, summer, and fall of 2014.

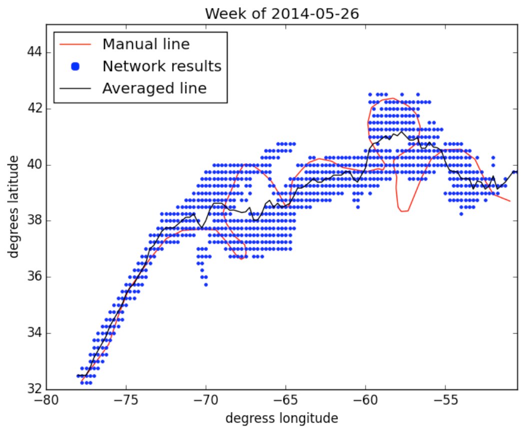
VI. Preliminary Results from the Neural Network

In this section we present the results from our first attempt using the above methodology. The best classification percentage has so far been achieved using all of the above-mentioned input attributes. The network that achieved this has been applied to data for the weeks of 2014-05-26, 2014-06-02, 2014-06-09, and 2014-06-16. Selected results are shown in Figure 4 for the weeks of 2014-5-26 and for 2014-06-16. The NN was trained using the data during May-June of 2014.

The most immediately noticeable aspect of these results is that there is not a great deal of variation in the points plotted by the network from week to week. Similarly noticeable is how many there are points there are. The network incorrectly classifies many points as being part of the GSNW. To attempt to account for this, a simple post-processing algorithm has been applied that takes the average of the latitude values of the plotted points at each longitude and plots those as a line, shown in the figures in black.

Possible causes for the limited results so far include the following. First, the lack of variation in the points plotted by the network from week to week likely indicate an over fitting of the network to the training data. This could potentially be improved upon by expanding the training dataset to include multiple years worth of data. This would expose the network to more variation in the line and possibly allow it to better identify some of the less common features of it, like the sharp dips around -57° longitude in figures 2 and 3.

Another potential cause of problems is the way the network addresses each point individually. In a problem like this one, the point's context is important. This context is taken into account very little in this network, in the height difference attribute. Because the network is setup to address each point individually, it is not ideal for considering this context in a more meaningful way. This could potentially be addressed with a more complex training algorithm or by taking an altogether different approach. Beyond these, other learning algorithms may be better suited to solving a problem of this complexity, including other types of neural networks. Major next steps for this project will focus on these concerns.



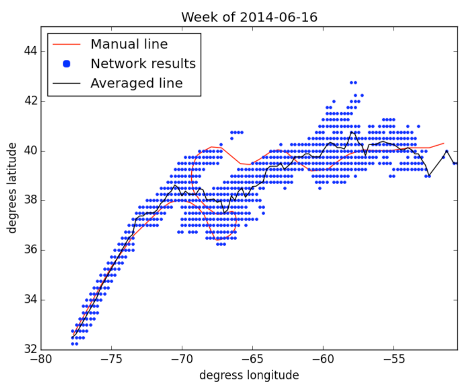


Figure 4 - Network applied to week of 2014-05-26 (left) and for 2014-06-16 (right). Note (i) how the region identification for GSNW is similar between the two due to the training being restricted to a season, and (ii) how the identified regions are different in the high-amplitude, large-meandering GS regions to the east of 65W.

VII. A COMPARISON OF THE TWO APPROACHES

Next, we present a case where both of the above methods (dynamical and NN) were applied to the images of week of June 02, 2014. The results are presented in Figure 5.

There are several things to note from the comparison. First, The straight-line extension drawn by the human operator was necessary for providing suitable boundary outflow condition to the numerical model, which was restricted to 53W. The NN method might interpret this as "human errors" and should thus exclude these particular instances from training. Second, note that the dip in the digitized path at around 65°W, was captured by both NN and dynamic algorithm as a separate feature. A closer look at the velocity field appeared to indicate a closed contour – possibly an eddy. Such discrepancies elucidate the necessity of using both NN and dynamic algorithm in a synergistic manner.

An interesting difference between the two approaches is that the dynamical approach currently seeks to output a path, whereas the neural network approach is outputting a range of points. The range of points represents an area of uncertainty wherein the path most probably lies. Finding the path within this area of uncertainty through post-processing is a viable research direction.

The type of neural network used in this case serves very well in generalizing decisions, surfaces, etc. As such, it is understandable that the output would be a general view of where the north wall has appeared throughout the studied period. Whether due to human error, eddies or other interferences, the swings in the dynamical approach are present and accounted for in the multitude of classified points, east of 65W. While we utilize a well-studied NN in this work, it is obvious that the approach needs adjustments and, possibly, post-processing. Given the nature of the data, however, the network has performed very well in determining the region of the north wall. Hence, the conjecture that additional post-processing is necessary to provide the closer approximation of the GSNW.

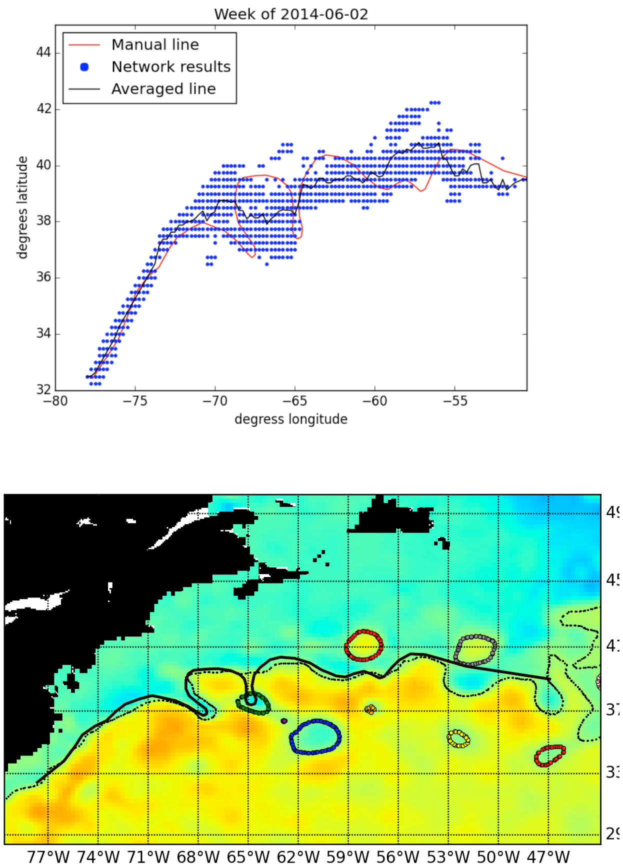


Figure 5. A comparison of the NN-derived GSNW region (top) with the dynamical approach derived GSNW path (bottom) for the week of 2014-06-02. The NN captures the region of GSNW reasonably well to the west of 70W, where the GS maintains a rather straight meandering configuration. However, the region identification algorithms need further refinement to the east of 65W.

VIII. Discussion and Future Work

It is clear from the above results that both dynamical method (with SSH) and the NN algorithm show promises for automatic detection of the GSNW based on concurrent SSH, SST and other images. In this section, we first outline a few key points that have been identified and are being investigated currently. We then describe some of immediate research areas for the near future.

First or all, we note that the SSH certainly helps when there are clouds and SST is not available. This is clear from Figure 1 (bottom-right panel), when the SST images are not available for the human operator to detect the GS. In future, we would like to quantify the difference between SSH derived path and SST derived path (by gradient method and/or by human operator) for six years of data to understand the implications of the dynamic method and use this information with the NN technique for synergistic detection of the GSNW.

Second, there could be a seasonal dependence of the GSNW on its isoheight contour specification. What is that range? We will carry out a statistical analysis of all five years of data to get this range and fix the isoheights (probably month-by-month).

Third, how do we identify eddies/Rings/loops? What kind of criteria should we use? Eddies can be described as features maintaining a rotational flow within their closed contours. Detecting a closed contour is an indication that the feature being analyzed is an eddy. Finding that the vorticity within the closed contour is consistently in the same direction – especially if the vorticity just outside the feature is in the opposite direction – seems fairly conclusive. It bears experimentation but we should be able to identify eddies by detecting the closed contours and through vorticity comparison. Note that the network also does not seem to be very influenced by the two eddies (at 59W and between 53 and 56W in the bottom panel of Figure 5).

Fourth, what is the role of vorticity and speed in defining the GSNW signature? Is it supplementary to SSH and SST? After taking a look at vorticity, I have wondered if it is a more appropriate baseline for identifying the Gulf Stream than SSH. I believe that it is possible to separate the Gulf Stream's zero-vorticity line from that around it by shaving false positives. Most of the false positives will be at points with lower speed/temperature than the GS so you could use those as selection criteria (this would likely leave some eddies).

Fifth, how do we use both dynamical method and the neural network to come up with a synergistic and objective way of identifying the GSNW? Are there cases when we should use one over the other? -- There might be a way to narrow down the search region using the network, and then search within this reduced region using the dynamical method. Also, if the NN method proves better at deciding when an eddy is separate from the stream that will be very valuable to guide the dynamical method chose an isoheight after eddy-detection and eliminating such regions from the final image. Incidentally, preliminary analysis suggests that it might also be possible to separate the GSNW's zero-valued isovorticity contour from similar contours of nearby features by utilizing other datasets and information such as speed, temperature and SSH itself.

Finally, a few comments about the future of the NN research are outlined next. The NN approach utilized here is performing well for its goal – classification and generalization of and based on past data. However, it is obvious that a refinement is necessary to better pinpoint the GSNW. In view of this, two possible options are envisioned. Option one entails the utilization of unsupervised and supervised learning, in that order. The first step would handle the clustering of the general area of GSNW positioning. The second, supervised step would then act on that area along and provide the function approximation of the actual GSNW line. In the first step, we would utilize self-organizing map (SOM) for its excellent grouping skills [12]. A considerable amount of work has been done to prove the utility of SOM in such kind of applications [13]. In step two, we propose radial basis functions, another known neural network suitable for function approximation [14]. As a second option, we conjecture that the multilayer perceptron (MLP) utilized here will benefit significantly by a post-processing step, which will probably require statistical methodology. The goal is to significantly refine the generalized conclusion of the MLP by the use of image processing methodology used in face recognition, for example.

ACKNOWLEDGMENT

This research has been supported in part through a NOAA grant NA07NOS4730221 for the implementation of the Mid Atlantic Regional Association Coastal Ocean Observing System (MARACOOS). Our sincere thanks to Andre Schmidt, M. Monim, and Agata Braga for their help with GSNW digitized paths and Python usage.

REFERENCES

[1] Schmidt, A. C. K., Gangopadhyay, A., 2012: An operational Circulation Ocean Prediction System for the Northwest Atlantic: hindcasting during July-September of 2006. Continental Shelf Research, http://dx.doi.org/10.1016/j.csr.2012.08.017

[2] Schofield, O., S.M. Glenn, J. Orcutt, M. Arrott, M. Meisinger, A. Gangopadhyay, W. Brown, R. Signell, M. Moline, Y. Chao, S. Chien, D. Thompson, A. Balasuriya, P. Lermusiaux and M. Oliver, 2010: Automated Sensor Networks to Advance Ocean Science, EOS, 91 (39), 28 Sept, 2010.

[3] Brown, W. S, A. Gangopadhyay, F. L. Bub, Z. Yu and G. Strout, and A. R. Robinson, 2007a. "An Operational Circulation Modeling System for the Gulf of Maine/Georges Bank Region, Part 1: The Basic Elements", IEEE J. Oceanic Engineering, 32 (4), 807-822.

[4] Brown, W. S, A. Gangopadhyay, and Z. Yu, 2007b. "An Operational Circulation Modeling System for the Gulf of Maine/Georges Bank Region, Part 2: Applications", IEEE J. Oceanic Engineering, 32 (4), 823-838. [7] Fairall, C.W., E.F. Bradley, J.E. Hare,

[5] J.F. Cayula and P. Cornillon, Edge Detection Algorithm for SST Images, Journal of Atmospheric and Oceanic Technology, 1992, 9(1), 67-80

[6] Kelly, K. A. (1991), The Meandering Gulf Stream as Seen by the Geosat timeter: Surface Transport, Position, and Velocity Variance from 73° to 46°W, J. Geophys. Res., 96(C9), 16,721–16,738.

[7] Swapnil Chaudhari, Ramprasad Balasubramanian, Avijit Gangopadhyay, Upwelling Detection in AVHRR Sea Surface Temperature (SST) Images Using Neural-Network Framework, 2008 IEEE International Geoscience & Remote Sensing Symp. July 6-11, 2008 Boston, Massachusetts, U.S.A, v. IV 926-929.

[8] Samarth Patel, Ramprasad Balasubramanian, Avijit Gangopadhyay, Automatic Detection Of Oceanic Eddies In SeaWiFS-Derived Color Images Using Neural Networks and Shape Analysis, 2008 IEEE International Geoscience & Remote Sensing Symposium July 6-11, 2008 Boston, Massachusetts, U.S.A, v II 835-838.

[9] Chaudhuri, A., Gangopadhyay, R. Balasubramanian and S. Ray, Automated Oceanographic feature detection from high resolution satellite images, Proceedings of the Seventh IASTED International Conference on Computer Graphics and Imaging (CGIM 2004), 2004, Hawaii, August, 2004, 217-223.

[10] Principe J. C., Euliano N. R., and Lefebvre W. C. (2000) Neural and Adaptive Systems-Fundamentals Through Simulations, John Wiley & Son, US.

[11] Chen T., and Chen H. (1995) Approximation Capability To Functions Of Several Variables Nonlinear Functionals And Operators By Radial Basis Function Neural Networks. IEEE Transactions on Neural Networks, 6(4):904-910.

[12] Kohonen, T. (1990) The self-organizing map. Proceedings of the IEEE, 78, 1464–1480.

[13] Furao, S., Ogura, T., and Hasegawa, O. (2007) An enhanced self-organizing incremental neural network for online unsupervised learning. Neural networks : the official journal of the International Neural Network Society, 20, 893–903.

[14] Buhmann, M.D. (2003) Radial Basis Functions: Theory and Implementations, Cambridge, 2003.