Coastline Detection through Image Segmentation using UNet

1st Donovan Dahlin
Department of Computing Sciences
Texas A&M University - Corpus Christi
Corpus Christi, Texas
ddahlin@islander.tamucc.edu

2nd Kristine Veneles
School of Computer Science and Technology
Kean University
Union, New Jersey
venelesk@kean.edu

3rd Dr. Dulal Kar
Department of Computing Sciences
Texas A&M University - Corpus Christi
Corpus Christi, Texas
Dulal.Kar@tamucc.edu

4th Jacob Hopkins
Department of Computing Sciences
Texas A&M University - Corpus Christi
Corpus Christi, Texas
jhopkins2@islander.tamucc.edu

***Abstract*—Coastlines are an important yet volatile part of the environment and frequently change over time. Studying changes in coastlines can give important insights about the environment, so using efficient means to observe them is important. UAV's are cheap, fast and effective tools in gathering information, and segmentation can be used to detect and record the coastline. This paper proposes the use of CNN models to assist UAVs in detecting coastlines. Specifically, our research focuses on the combination of a neural network model, such as UNet, with transfer learning using a pre-trained model. The resulting model achieves semantic segmentation of coastlines to distinguish water and land. This allows UAVs' to better recognize coastlines and reinforce UAVs' role in remote sensing image processing.**

I. Introduction

Today, the world and its environments are experiencing great changes. Due to global warming, each year breaks more heat records than the one before. Rebecca Lindsey and LuAnn Dahlman have published records of the global temperatures, and they have revealed that 2020 was the warmest year on record. [1]

This specific research project focuses on the detection of coastlines. Given that the ocean regularly gives and takes sediments from beaches, coastlines are always in a state of constant flux. This perpetual change necessitates efficient methods for tracking the coastline. In addition to the changes introduced by global warming, the environment is also affected by direct changes from animals and people. For example, in Galveston, Texas, the beaches there are repeatably resupplied with sand. Otherwise, the beaches there would be washed away into nothing. [2]

Beaches are important for a variety of reasons, one of them being the effect they have on the psychological health of people. [Needs to be worked on]

Another stark example of changing coastlines is the Arctic permafrost. According to Cunliffe et al, the Arctic is warming up faster than any other part of the Earth. Naturally, this will cause the permafrost there to recede. One of the largest factors for the recession of permafrost is the erosion of the coastlines in the Arctic. This is because coastline erosion releases large amount of sediment, organic matter and nutrients into the ocean. This weakens the permafrost by taking away part of its composition. [3]

The importance of coastline tracking is clearly important, but the method of such tracking is important. One of the traditional methods of coastline monitoring is taking satellite images. However, there are some problems with this. Poor weather can obstruct the view of the satellite, hampering or ruining its ability to record data. Even without bad weather, the resolution of the satellite is often lower than that of other methods, such as local cameras. Bruno et al. also discuss about how constant GPS monitoring can be impractical in terms of logistics and costs. [4]

Other methods exist for tracking coastline. As Bruno et al. show, they include optical sensors and Li-DAR campaigns. Although they collect great amounts of accurate information, they also come with high inquisition costs. There discuss additional monitoring options, such as unmanned aerial vehicles and video-monitoring systems. Although significantly cheaper to use, they can only provide information about limited areas. [4]

We propose that unmanned aerial vehicles (UAV) are the most practical choice for this research project. The area that they can at once may be limited compared to other methods, but as already mentioned, drones cost less compared to other methods. They are also safe to use (if handled properly) and can take high quality pictures. Furthermore [additional research required.]These advantages make drones optimal tools for research.

II. Related Works

A. Deep Learning

Coastline detection has been studied for a significant amount of time, and as such there have been many methods developed

Funded by the National Science Foundation.

and used for this purpose. Firat Erden et al. discuss these methods. In their discussion, they reveal that such methods include (but are not limited to): unsupervised clustering, band ratio, normalized difference water index (NDWI), thresholding and morphological filtering, supervised and unsupervised classification, Whale Optimization Model, among others. [5]

However, the above mentioned methods all have limitations in what they can do. Firat Erden et al. [5] go into detail about these limitations. Traditional image segmentation techniques may see wide use, but they need priori knowledge, and parameter settings would need to be re-defined for each image. As such, traditional image segmentation techniques do not offer comprehensive solutions. NDWI is well suited for deep water detection, but it produces unreliable results for shallow areas, which hinders its flexibility, especially since many coastlines are shallow. Edge detection methods are highly susceptible to noise, and they are also highly dependant on images. Pixel-based methods do not take into account spatial, textural and semantic differences.

To get around these limitations, we as a team have decided to use deep learning. At it's core, deep learning (DL) is a machine learning method learns learns much like the human brain. DL algorithms contain one or many layers of functions that are called neurons. Each functions accepts three parameters: The input, the weight, and the bias. The input is the data for the neuron to analyze. The weight is a multiplier assigned to the input by the DL algorithm. Finally, the bias is a constant that is added to (or subtracted from) the input multiplied by the weight.

Other researchers have developed deep learning techniques for detecting coastlines. For example, (list examples here)

According to Erden et al. [5], the biggest advantage of DL is the automatic feature extraction. They also reveal that DL is an end-to-end solution. DL is also extremely adaptable, being able to be used for nearly any study area, including coastline studies.

B. UNet Segmentation

Deep learning is a powerful analytical tool, but deep learning has multiple existing variations. As Shamsolmoali et al. [6] show, these variations include GoogleNet, Residual Net, and DenseNet. Other models include fully connected convolution networks, which is highly effective in image segmentation. One of those variations is U-Net (UN).

Ronneberger's work in biomedical image segmentation created UNet architecture to increase the training speed of classification and localization of pixels, which leads to image semantic segmentation, with minimal data set and precise predictions. UNet is an end to end fully convolutional network. The architecture is the arrangement of the deep learning tools, such as convolution and pooling, resulting in semantic segmentation of an image.

The UNet architecture involves symmetrical positioning of layers possible for concatenation of features of input images from contracting path to corresponding expansive path to produce an image with more context. The left side, the contracting path, also known as the encoder, consists of two 3x3 convolutions each followed by an activation function, ReLU and a 2x2 max pooling operation with a stride of 2 for down-sampling. The encoder is taking features of the input to help the decoder make accurate predictions. Simply, the encoder is looking at the features of the input and classifying, while decreasing the dimensions of the input. It is imperative that images be reshape to same dimensions, otherwise an error will occur during training.

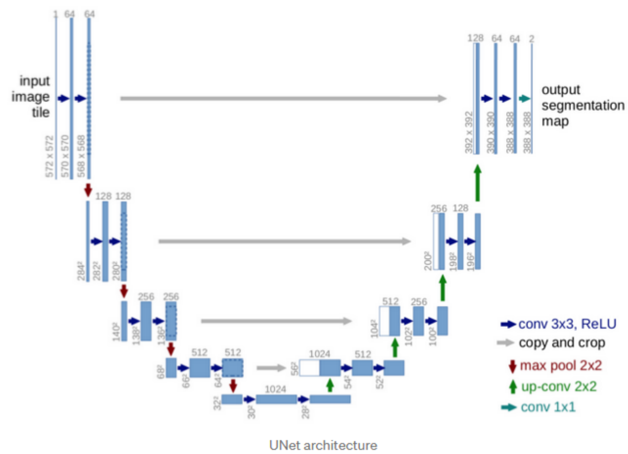

Fig. 1. UNet Architecture [7]

As it shifts into the right side, the expansive path also known as the decoder, the structure mirrors the contracting path. The decoder builds up into the original dimensions collecting the features, while receiving information from the encoder path to assist in localizing the pixels on the input. At each up-sampling step, the features coming from the corresponding contracting path will be concatenated to prevent loss of features to gain context of images. The final dense layer mapping images to each 1000 classes specified in ImageNet.

C. Transfer Learning and Data Augmentation

Deep learning is a very powerful tool, but it also requires a great deal of data in order to function properly. Creating and/or gathering this data can be a labor and time intensive task. Fortunately, there are several solutions to this issue, one of which is data augmentation. Data augmentation can transform existing data to create new data, thus creating additional data for the machine to work with.

There is also transfer learning to consider. Cui et al. [8] studied transfer learning in their efforts to improve semantic segmentation. They describe the key idea in transfer learning as simply using the data and learning from other deep learning machines and using it to improve their own data and learning.

The research done by Cui et al. shows proof that transfer learning works effectively as well. They have shown that deep convolutions networks that have already been trained on data works can be transferred to machines with insufficient data sets and improve accuracy more than deep learning techniques. A research team that tested their transfer learning machine on the UC Merced Land Use benchmark found that the accuracy went from 83.1% to 92.4%. [8]

Transfer learning is implemented to speed up training and improve the performance of the model. [9] With transfer learning, models already trained on a specific task can be used to accelerate training the neural network. Visual Geometry Group from the University of Oxford created VGG16. This is a 16 layer trained on localization and classification on the large database of images with annotations called ImageNet. Pre-trained models have learned weights that can be used to achieve other segmentation goals as well as classification. Keras provides libraries of pre-trained models to fine tune pre-trained models, such as VGG16 to accomplish other neural network's goals.

D. Unmanned Aerial Vehicles

While the usage of deep learning, UNet segmentation, transfer learning and data augmentation have all been discussed, the tools used to apply these techniques to the real world is just as important. There are several options to this, but our research team decided the UAVs would be the best choice for this project.

There are many advantages of UAV's, which shall be discussed here at more length than in the introduction. As already mentioned, UAVs cost less than other methods. UAVs are also safe to use, provided that they are handled properly by either the pilot or its programming. If used correctly, a UAV is not likely to crash into or hurt any persons or animals. Drones can also be easily equipped with high-quality cameras in order to record images and data more accurately. Drones also have the advantage of being much closer to the source of the study, which lessens the possibility of obstructions.

III. Methodology

A. Data Set Sub-section

The data set consists of 326 images of shoreline. 261 images were taken randomly from 2 coastline videos taken by UAV over bob hall pier and cape cod. The video of Cape Cod beach in Massachusetts, USA. The Bob Hall Pier video is taken in Corpus Christi, TX. The UAV, DJI Phantom 4 Pro, was used to take both videos. Masks were created for each image using an application called Labelme provided by Github. The classes we chose to specify pixels are notWater and Water. Red is water and green is notWater. This data will be used in out training model. Collectively, there are 3 folders: train_img, test_img, and valid_img. These folders have corresponding folders consisting of their masks: train_masks, test_masks, and valid_masks.

B. Data Preparation

Due to our minimal amount of data, data augmentation will be used to increase our data set. Data augmentation is a popular technique in image classification and segmentation to increase data to create variables in the data to reduce over fitting. Within our data, the images are altered, by rotated 90 degrees and flipped horizontally and vertically. Data augmentation has shown to be important in remaining invariant when extracting features from images. [?] The input images in our data set are resized to 500x500 dimension with RGB channels. The datatype have been converted to float32 and normalized. Using Keras train_test_split module, we split our train data set with test size as 10% and the remaining as training data. The X_test and y_test will be the validation set during training process.

Fig. 2. Sample images and masks

C. UNet Implementation

The source code of UNet with pre-trained mode, VGG16, are downloaded from Kaggle as preliminary guide to fine-tuning our own model. [10] The No-top pre-trained model, VGG16, is imported which exempts the last fully connected layers to provide outputs of our own binary predicted masks. This is done to fine tune the model to our own goal, which is semantic segmentation of our input images of shorelines as land and water.

The UNet model consists of 5 blocks of 2 convolutional layers followed by max pooling layers. Layers are frozen to prevent weights from being modified during backpropagation. Other layers will be updated during backpropagation so that the model can learn and provide better output performance. The model will train for 500 epochs with steps_per_epoch equals to 2.

The model will use Adam as the optimizer with a learning rate of 0.0001. Adam, adaptive moment estimation, will calculate the way weights should be altered so that the loss function, binary_crossentropy, can reach a minimized loss. The learning rate determines how fast the model will learn to reach minimized loss. The binary_crossentropy calculates how far or how close the predicted values are from the actual values. The activation function used in our model is LeakyReLu. LeakyReLu allows weights less than 0 to stay active and contribute to training process even if it becomes too negative. Within the model, batch normalization is included to normalize the output of the previous layers, which keeps input values and hidden layers within a certain range with an advantage of an improvement in training speed.

D. Model Evaluation

Metrics used to evaluate how well our model performed will be intersection over union and pixel accuracy. Pixel accuracy is most commonly seen. TP represents the true positives, in which the model's pixel prediction is correct. TN, true negatives, are the pixel predictions made correctly that are not a certain class. FP, false positives, are pixel predictions misclassified into a certain class. FN, false negatives, are pixel predictions classified incorrectly as that are not in a certain class. It is the percent of pixels in the image that is classified to its actual labels.

$$PixelAccuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

However, this metric can lead to misleading results. When the pixel representation of TN is too high in an image, it can lead to a high accuracy, misrepresenting the actual prediction mask.

The intersection over union metric, also known as Jaccard index, compares the ground truth mask and the result of the prediction mask based on how much of the pixels are classified and localized over the total amount of the pixels of both the mask and the prediction. The intersection (A∩B) are the pixels found in both the prediction mask and the ground truth mask. The union (A∪B) are all the pixels found in either the prediction mask and the ground truth mask. Each class will have an IoU score and averaged over all classes to produce the IoU score of a semantic segmentation prediction. Source code to produce IoU score are available in Github repository due to Keras' issues in implementing its own IoU score module. [11]

$$IoU = \frac{Intersection}{Union} = \frac{TP}{TP + FP + FN}$$

[12]

F1-score, also known as Dice Coefficient, provides the harmonic mean of precision and recall. This is useful in cases with uneven class distribution, such as high amount of True Negative predictions. Precision is the TP over the Total Predicted Positive and Recall is the TP over the Total Actual Positive. In our data set, TP are the predicted pixels classified correctly as water, FP are pixels classified incorrectly as water, FN are pixels are incorrectly classified as land and TN are pixels correctly classified as land. It measures the weighted average of precision and recall. F1-score and IoU scores are commonly used metrics used in image segmentation.

$$F1score = \frac{2 * Precision * Recall}{Precision + Recall}$$

[13]

IV. Results

A data set of 326 original images (and their corresponding masks) were provided for training the model. Most of the images are frames from a pair of videos taken from a previous REU. The rest come from the royalty-free website Pixabay. Data augmentation was applied to the data set increase 8 times into a set of 2168 images. 90% of this data set was reserved for training data, 10% was used for validation, while the last 10% was used for testing the model. The model was trained for 500 epochs, each epoch consisting of two training steps. The table below shows the final metrics that the model has. Source code from Github repository assisted in visualizing predicted masks, as well as the learning curves, that our model have produced. [14] [11]

TABLE I
METRICS OF THE FINAL MODEL

Metric	Measurement
Accuracy	0.6382
Mean IOU	0.2013
F1-Score	0.5084

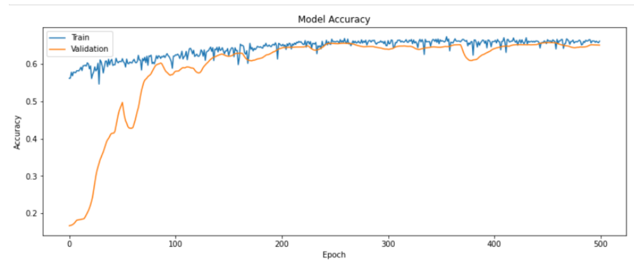

Fig. 3. Graph of Accuracy

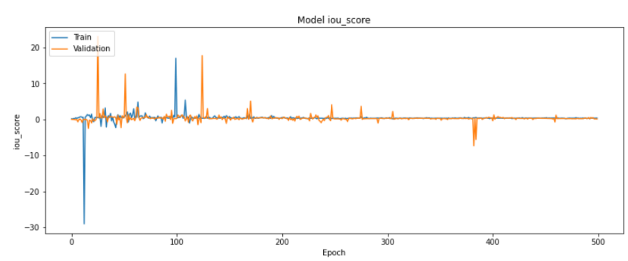

Fig. 4. Graph of Mean IOU

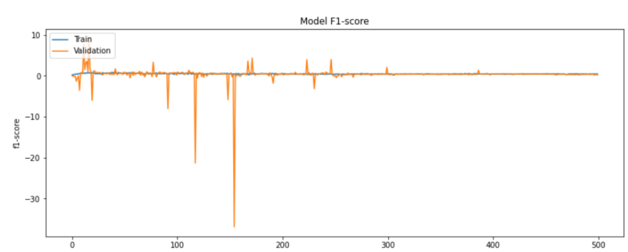

Fig. 5. Graph of F1-Score

V. Discussion

Figure 6 shows the test images the model was given, the prediction the model gave for that image, and the mask for the

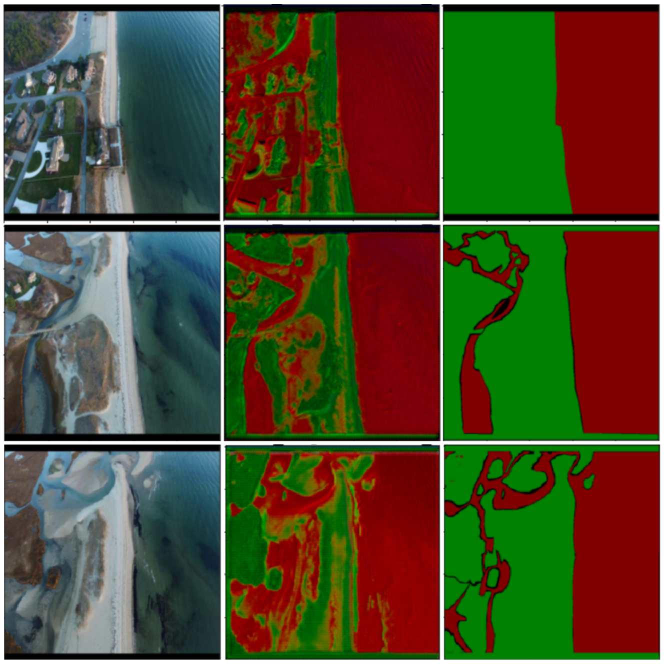

Fig. 6. Image, Model Prediction and Mask

image, respectively. The model has little trouble detecting the water, but it experiences difficulty when detecting land. This difficulty seems to arise in land areas that have similar colors to water. For example, the roads in the model results have colors that are close to that of the ocean. It is close enough that the machine mistakes the roads for water instead of land and labels them as such.

VI. Future Work

Plans for future work involve using a data set consisting of HSV images rather than RGB images. Training the segmentation model with HSV images may result in a higher accuracy than using RGB images. Additional future plans include implementing path programming for a drone so that it can autonomously record the coastline. This would involve drawing a counter line to distinguish the coastline, and then generating directions (such as directional vectors) for the drone based on that coastline. DeepUNet, proposed by Li, et al., shows promising results in pixel segmentation of satellite images of coastlines with an F1 score of 95.39%. [13] Due to time constraints, we are unable to implement this strategy.

VII. Conclusion

In this research study, we proposed the use of convolutional neural network to train a model to distinguish between water and land to find the coastline. Finding the coastline will assist UAVs to follow its path. Collectively, our dataset consists of coastline images from Pixabay and frames from two videos, Bob Hall Pier and Cape Cod used in previous REU research studies. Data augmentation is used to increase our data-set. The CNN model follows the structure of a UNet model proposed by Dr. Ronneberger, used commonly in image segmentation tasks [7]. Transfer learning has also been adopted, using VGG16 pre-trained model to speed up training and improve model performance. In its entirety, the model has been modified and fine-tuned to achieve better output masks. Although, the accuracy, the IoU and F1-score are low, the predicted segmented images have shown that further fine-tuning of hyper-parameters can lead to better results most useful in detecting coastlines, as well as implementing recommendations suggested in future works section.

References

[1] R. Lindsey and L. Dahlman, "Climate change: Global temperature," March 2021. [Online]. Available: https://www.climate.gov/news-features/understanding-climate/climate-change-global-temperature.

[2] [Online]. Available: https://galvestonparkboard.org/288/Babes-Beach-Nourishment-Project-2021

[3] A. Cunliffe, G. Tanski, B. Radosavljevic, W. Palmer, T. Sachs, H. Lantuit, J. Kerby, and I. Myers-Smith, "Rapid retreat of permafrost coastline observed with aerial drone photogrammetry," *The cryosphere*, vol. 13, no. 5, pp. 1513–1528, 2019.

[4] M. Yasir, H. Sheng, H. Fan, S. Nazir, A. J. Niang, M. Salauddin, and S. Khan, "Automatic coastline extraction and changes analysis using remote sensing and gis technology," *IEEE access*, vol. 8, pp. 180 156–180 170, 2020.

[5] F. Erdem, B. Bayram, T. Bakirman, O. C. Bayrak, and B. Akpinar, "An ensemble deep learning based shoreline segmentation approach (waternet) from landsat 8 oli images," *Advances in space research*, vol. 67, no. 3, pp. 964–974, 2021.

[6] P. Shamsolmoali, M. Zareapoor, R. Wang, H. Zhou, and J. Yang, "A novel deep structure u-net for sea-land segmentation in remote sensing images," *IEEE journal of selected topics in applied earth observations and remote sensing*, vol. 12, no. 9, pp. 3219–3232, 2019.

[7] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, Eds. Cham: Springer International Publishing, 2015, pp. 234–241.

[8] B. Cui, X. Chen, and Y. Lu, "Semantic segmentation of remote sensing images using transfer learning and deep convolutional neural network with dense connection," *IEEE access*, vol. 8, pp. 116 744–116 755, 2020.

[9] S. Piao and J. Liu, "Accuracy improvement of UNet based on dilated convolution," *Journal of Physics: Conference Series*, vol. 1345, p. 052066, nov 2019. [Online]. Available: https://doi.org/10.1088/1742-6596/1345/5/052066

[10] V. Basu, "Transfer learning and unet to segment rocks and boulders on the lunar surface." [Online]. Available: https://www.kaggle.com/basu369victor/transferlearning-and-unet-to- segment-rocks-on-moon/execution

[11] P. Yakubovskiy, "Segmentation models," https://github.com/qubvel/segmentation_models, 2019.

[12] R. Singh and R. Rani, "Semantic segmentation using deep convolutional neural network: A review," *Proceedings of the International Conference on Innovative Computing Communications*, April 2020.

[13] R. Li, W. Liu, L. Yang, S. Sun, W. Hu, F. Zhang, and W. Li, "Deepunet: A deep fully convolutional network for pixel-level sea-land segmentation," *IEEE journal of selected topics in applied earth observations and remote sensing*, vol. 11, no. 11, pp. 3954–3962, 2018.

[14] S. Bhattiprolu, "python_for_microscopists." [Online]. Available: https://github.com/bnsreenu/python_for_microscopists/blob/master/207-simple_unet_model_with_jacard.py