A 2D fully convolutional neural network for nearshore and surfzone bathymetry inversion from synthetic imagery of the surfzone using the wave model Celeris

Adam Collins,[1] **Katherine L. Brodie,**[2] **Spicer Bak,**[2]
Tyler Hesser,[3] **Matthew W. Farthing,**[3] **Douglas W. Gamble,**[1] **Joseph W. Long,**[1]
[1]University of North Carolina at Wilmington, Earth and Ocean Sciences, Wilmington, NC
[2]U.S. Army Engineer Research and Development Center, Coastal and Hydraulics Laboratory, Duck, NC
[3]U.S. Army Engineer Research and Development Center, Coastal and Hydraulics Laboratory, Vicksburg, MS
[1]{amc3496, gambled, longjw}@uncw.edu,[2]{katherine.l.brodie, spicer.bak}@erdc.dren.mil,
[3]{tyler.hesser, matthew.w.farthing}@erdc.dren.mil

Abstract

Bathymetry has a first order impact on nearshore and surfzone hydrodynamics. Typical survey techniques are expensive and time-consuming, require specialized equipment, and are not feasible in a variety of situations (e.g. limited manpower and/or site access). However, the emergence of nearshore remote sensing platforms (e.g. Unmanned Aircraft Systems (UAS), towers, and satellites) from which high-resolution imagery of the sea-surface can be collected at frequent intervals, has created the potential for accurate bathymetric estimation from wave-inversion techniques without in-situ measurements. While a variety of physics-based algorithms have been applied to nearshore and surfzone bathymetric inversion problems, the commonly used approaches do not account for non-linear hydrodynamics that are prevalent during breaking waves. Models for estimating non-linear wave dynamics are slow and often require large amounts of computational power which make them unfeasible for rapid estimations of depth. Fully convolutional neural networks (FCNs) are a branch of artificial intelligence algorithms that have proven effective at computer vision tasks in semantic segmentation and regression problems. In this work, we consider the use of FCNs for inferring bathymetry from video-derived imagery. The FCN model presented shows the feasibility of using an AI system to perform bathymetric inversion on time-averaged images (timex) of realistic-looking, synthetically generated surfzone imagery from the hydrodynamic wave model Celeris (Tavakkol and Lynett 2017). Ongoing work includes extending the FCN to incorporate synthetic video frames as input as well as testing with actual tower and satellite imagery.

Introduction

Accurate knowledge of nearshore and surfzone water depths is important for a wide range of applications, ranging from enhancing the personal safety of beach-goers, to industrial and military applications such as identifying navigable areas for ships or other landing craft (Avera et al. 2002). The bottom boundary condition is one of the most important inputs for numerical simulations of nearshore and surfzone processes, with water depth and slope being principal parts of the governing wave equations in the nearshore. Currently the most accurate methods for determining bathymetry are in-situ observations involving physical contact with the bottom, or acoustic hydrographic surveys from vessels (Moulton, Elgar, and Raubenheimer 2014b). Both approaches are limited by the requirement of a physical presence at the site, which complicates their use in isolated environments or during unsafe water conditions (Birkemeier and Mason 1984). In addition, the surfzone bathymetry is constantly changing, and can vary considerably day-to-day making consistent measurement impractical using traditional methods (Moulton, Elgar, and Raubenheimer 2014a). An alternative approach to estimate bathymetry that would overcome some of these limitations is using remotely sensed data sources, which don't require a physical presence in the water at a site. A number of remote sensing approaches to estimate bathymetry have been developed including direct (e.g. bathymetric LiDAR, multi and hyper-spectral imagery) and inferred approaches (e.g. image or radar-derived observations of wave-kinematics and breaking)(Holland, Palmsten, and others 2018). Visible band imagery offers a low-cost approach which exploits the visible surface signature of shoaling and breaking waves in the nearshore – wave transformation processes that are largely controlled by water depth. Images record the location of wave breaking or speeds of wave propagation, which can be related to water depth using a bathymetry inversion algorithm (Holman, Lalejini, and Holland 2016; Van Dongeren et al. 2008). The use of different remote sensing platforms, such as satellites and unmanned aerial vehicles (UAVs) (Holland et al. 2010; Holman, Brodie, and Spore 2017; Brodie et al. 2019; Almar et al. 2019; Bergsma, Almar, and Maisongrande 2019), to collect this imagery offers opportunities to estimate bathymetry in areas that would normally be difficult or costly to assess with traditional methods, increasing data availability and reducing costs (both financial and temporal) compared to in-situ observation methods (Gao 2009).

While analyzing sequences of coastal video imagery with traditional signal processing and computer vision algorithms

to estimate bathymetry holds promise, the inherent complexity of the nearshore and surfzone, which includes many non-linear processes, will always lead to errors in any bathymetric inversion model that simplifies the effects of these processes through a linear approach. Machine learning algorithms, particularly deep neural networks, have previously demonstrated the ability to identify and classify pixels in complex images far beyond the quality of traditional hand-written algorithms (Simonyan and Zisserman 2014). Applying machine learning for classification of remote sensing images on a pixel-wise basis is referred to as semantic segmentation and has increasingly been utilized in remote sensing over the past decade. The combination of high-resolution data and faster computer processing has made this possible by allowing for the parallel processing of millions of parameters, which is required to process the increasing resolutions from remote sensing technologies, such as UASs and/or HD camera systems (Christophe et al. 2011).

Traditional low-resolution algorithms used to analyze remote sensing imagery do not maintain their effectiveness at these higher resolutions of present-day interest, while the abundance of parameters in the high spatial and spectral resolution data make a traditional analytical algorithm more difficult to develop when classifying complex features (Zhu et al. 2017). Image processing algorithms to simplify these datasets are often time consuming to run and require substantial investment in powerful computer hardware. However, the performance of semantic segmentation of high-resolution scenes has increased rapidly since 2012, which was the beginning of the domination of supervised deep learning with the introduction of the deep convolutional neural network (DCNN) AlexNet (Krizhevsky, Sutskever, and Hinton 2012; Alom et al. 2018). In addition, deep neural networks have the advantage of being extremely fast to compute targets once trained, yielding portability to run on a relatively modest processor. For example, deep neural networks allow for near real-time semantic segmentation on board a UAS or sea-based vessel to aid autonomous navigation (Tian et al. 2018).

However, the downside of typical supervised training with deep neural networks is the requirement for extremely large labeled datasets. Because of this, many classifiers and segmentation networks start with pre-trained parameters as opposed to the typical machine learning approach where the parameters start out as random values. These pre-trained parameters are then transferred to the current task, changing only a small subset of them with the training data for the new problem looking to be solved. Parameters that have been pre-trained on large image datasets will be able to identify vague features, such as edges in an image. These vague features are then used as inputs into the final layers, which are the layers whose parameters will be adjusted by the new training dataset. This application of a trained network being adjusted and then applied to another task is commonly referred to as transfer learning (Huh, Agrawal, and Efros 2016).

Oceanographic data sets of coastal imagery coincident to highly accurate bathymetric measurements are extremely rare, and generally occur only during small waves. Available training data sets using real imagery are likely too small to find proper parameters from randomly initialized DCNN parameters, which would likely lead to over-fitting (Kemker, Luu, and Kanan 2018). This study seeks to both utilize the non-linear prediction powers of a deep neural network and explore the use of synthetic data to approach the bathymetry inversion problem through the development of a deep learning network using synthetic surfzone imagery derived from a photorealistic visualization of the nearshore wave model, Celeris (Tavakkol and Lynett 2017).

Background

Parametric equations have a long history of use to approximate beach slopes and are based on the model

$$h = Ax^{2/3}$$

where h is the water depth, A is a constant, and x is distance in the cross-shore direction (Bruun 1954). While parametric beach models are good for quantifying large-scale trends, such as regional inundation due to sea level rise; in smaller regions of interest, surfzone and nearshore variations in bathymetry, like sandbars, are not accounted for. To address the limitations of parametric bathymetry models, the location of the shoreline and sandbars can be added to parametric models using time-averaged imagery (timex) (Holman et al. 2014). Sandbars are identified by time-averaging sequences of video imagery of the surfzone, to generate a timex image (Lippmann and Holman 1989). Timex images are used to identify regions of persistent wave breaking. Waves break in areas with reduced water depth over sandbars, when the water depth decreases to be between 0.4 and 0.8 of their wave height (Komar and Gaughan 1973). Persistent regions of wave breaking appear as a white-band that can then be manually digitized from timex images to identify the position of the surf zone sandbars. Exposure times to generate the time-averaged images can range from a minimum of 10 minutes to full day exposures, using a variety of video capture techniques (Guedes et al. 2011).

Parametric bathymetry models can also be used in two-dimensions (2D) to generate more complex bathymetry (Holman, Lalejini, and Holland 2016). Their parametric beach tool requires twelve parameters to create a shoreline morphology, however eight of the parameters are evaluated to constants in practical implementation. The remaining variable inputs are the climatological beach slope at the shoreline, the depth and bottom slope at some location seaward of the active bar zone, and the cross-shore location of the sand bar crest. In the 2D implementation, a mean shoreline is input by the user, and normal transects from the shoreline are calculated. The distance from the shoreline to the sandbar, using expert identification with time lapsed images, is also input into the model, along with an estimated offshore depth and beach slope. This inversion model was tested by (Holman, Lalejini, and Holland 2016) and showed to have a mean bias and RMSE error of 0.27 m and 0.49 m, respectively, over the study area at Duck, NC.

Beyond simple parametric representations, a number of efforts have been made to directly measure surf-zone pa-

rameters of interest in order to estimate bathymetry. Direct inversion techniques have focused on measuring wave speeds from image sequences and estimating bathymetry using linear wave theory (Stockdon and Holman 2000; Plant, Holland, and Haller 2008; Holman, Plant, and Holland 2013; Bergsma and Almar 2018; Bergsma, Almar, and Maisongrande 2019), whereas other inversion schemes have utilized data assimilation techniques which combine the remotely sensed parameters with numerical models. Data assimilation techniques have ranged from classical variational methods and Kalman Filters (Holman, Plant, and Holland 2013; Wilson and Berezhnoy 2018) to ensemble approaches (Wilson, Özkan Haller, and Holman 2010) and more recent nonlinear extensions of the Kalman Filter (Ghorbanidehno et al. 2019). The types of surface observations that have been explored includes wave speeds as well as wave heights, currents (Holman, Plant, and Holland 2013; Wilson et al. 2014; Moghimi et al. 2016) and estimates of wave energy dissipation from timex images (Van Dongeren et al. 2008; Aarninkhof, Ruessink, and Roelvink 2005). In general, approaches combining modern inversion techniques with high fidelity models of nearshore hydrodynamics have shown the potential to provide higher accuracy estimates under a wider set of hydrodynamic regimes. However, this accuracy introduces added complexity and computational expense, which are potential barriers to fielding these approaches for real-time application in limited resource environments like mobile platforms.

In this effort we explore the ability of machine learning algorithms to learn the relationship between locations of persistent wave breaking in timex images and surfzone bathymetry, removing the need for manual digitization of the sandbar location (e.g. (Holman, Lalejini, and Holland 2016)) or a numerical wave dissipation model (e.g. (Van Dongeren et al. 2008)).

Methodology

Wave Modeling Software Selection

Celeris is an open source Bousinessq wave model that runs on a GPU cluster and creates visually realistic simulations of nearshore and surfzone waves in near real-time on a typical desktop computer (Tavakkol and Lynett 2017). Celeris generates and visualizes different wave interactions, such as shoaling, refraction, reflection, and breaking. These are the relevant processes influencing the visual expression of wave propagation in the nearshore, and therefore the wave model results provide a relevant corollary to observations collected by remote video platforms. This wave model was selected not only for its efficient run time, but also its pseudo-realistic visualizations of wave transformation and breaking, which can be used as a proxy for coastal video imagery. Through video capture of the wave model results, a 20 minute video is created after an initial 10 minute spinup time. These video files of the Celeris visualization are then averaged in time to produce a timex image (similar to the timex images typically created by nearshore video monitoring stations (Holman and Stanley 2007)) for that bathymetry and wave condition (Figure 1).

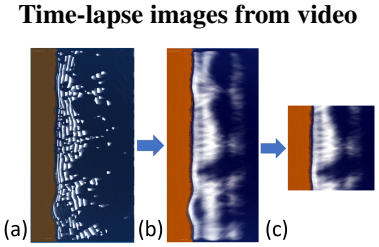


Figure 1: A single frame of the Celeris visualization (a), 8000 frames are averaged together to form the time-averaged image (timex) (b), with brighter areas showing where waves break more often. The far offshore part of the visualization in (a), which is the deepest section, is not included in the final timex (b) due to the lack of breaking waves. This produces a timex image (b) of 1795 m in the alongshore direction, and 860 m in the cross-shore direction. Finally, the timex is then cropped to be (512, 512) pixels and centered on the alongshore of the image (c). The shape is for ease of input to the ML model, while the position is due to edge effects from wave generation.

Bathymetry Selection

While there are multiple parameters to adjust in the Celeris wave model, two inputs have the largest effect on the generation of synthetic video imagery of surfzone processes: bathymetry and offshore wave boundary conditions. A set of 100 statistically driven bathymetries were generated using an Empirical Orthogonal Function approach on 40 years of in-situ bathymetry surveys collected at the U.S. Army Corps of Engineers Field Research Facility (Braud and Obled 1991). This set of bathymetries were then divided into separate sets of training (80 bathymetries), validation (10 bathymetries), and testing (10 bathymetries). These bathymetries extend 1795 m in the alongshore direction (parallel to the beach), and 970 m in the cross-shore direction (perpendicular to the beach), with an average shoreline position of about 220 m in the cross-shore direction. The cross-shore distance is chosen due to its correspondence with the location of the FRF's 8 m water depth pressure sensor array. In addition, 100 synthetically generated bathymetries were created, using parametric beach slopes, with sandbars, troughs, and depressions created by perturbing the slope at random locations, frequencies, and intensities (Figure 2). These bathymetries are introduced to allow the ML model to learn different breaking patterns, such as multiple sandbars, and their correspondence with water depth that are not usually visible in the bathymetries statistically driven from the observed data set from Duck, NC. It also serves as a preventative measure against over-fitting, intending to generalize the ML model's ability to accurately assess water depths for breaking wave intensities from imagery, by using a wider range of inputs and depths at different cross-shore locations beyond that of the statistics from the historical dataset (Figure 3). Perturbations from the mean profile were generated and added between 200 m from the shoreward domain edge and 200 m from the offshore bound-

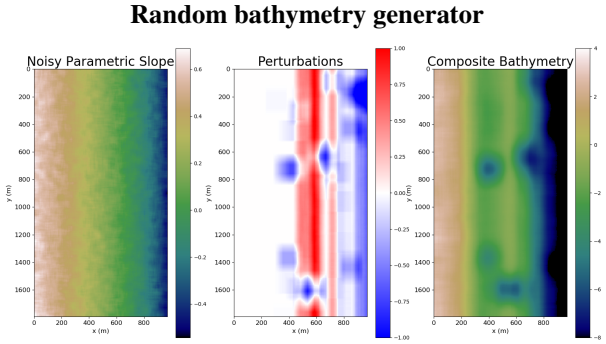


Figure 2: The different parts of the random bathymetry generator. This script was created to model different beaches and slopes than typically found in Duck, NC, such as multi-sandbar beaches. First a parametric profile is created, from that trough, sandbar, and spot variations are combined onto the original slope. This is then blurred to create the final bathymetry.

ary. Between 0-25 bar-trough (alongshore uniform) features, of random amplitudes and spacing, are generated and applied to the mean profile. At least 50 and up to 100 alongshore non-uniform, circular features of various radii and amplitude (positive and negative) are applied to the same portion of the profile area. The bathymetry is smoothed and stretched with a length scale of up to 20, and then the entire profile is shifted so that the average depth at the offshore boundary is at the desired depth. Bathymetries were conditioned to be centered at 8 m water depth at the offshore boundary since Celeris was setup to force with wave observations observed in 8 m depth. This set of bathymetries was similarly divided into separates set for training (80), validation (10), and testing (10).

Wave Condition Selection

To force the wave model, we selected the most highly probable wave conditions that were measured at the FRF's phased array of pressure sensors in 8 m water depth (about 950m from the shoreline) (Long and Oltman-Shay 1991). The wave rose, (Figure 4), bins the historical wave conditions by significant wave height and direction over the course of 10 years. Individual simulations were performed using the most frequent wave conditions observed in Duck, NC. While additional conditions were initialized by using the probabilistic wave conditions as boundary conditions for a Latin hypercube sub-sampling of the data plotted in Figure 3. The wave height affects where in the domain the waves break, whereas the wave frequency will affect how often the waves break (and resultant image intensity). The wave direction also affects the final timex image by varying the direction waves travel toward the shore, and thus the direction in which breaking occurs. These ranges are: wave heights between 0.7 m and 2.5 m, peak frequencies between 0.09 Hz and .2 Hz, and peak wave direction between 45° and 112.5° True North). Wave directions outside of these ranges only occur 14.7% of the time over the sam-

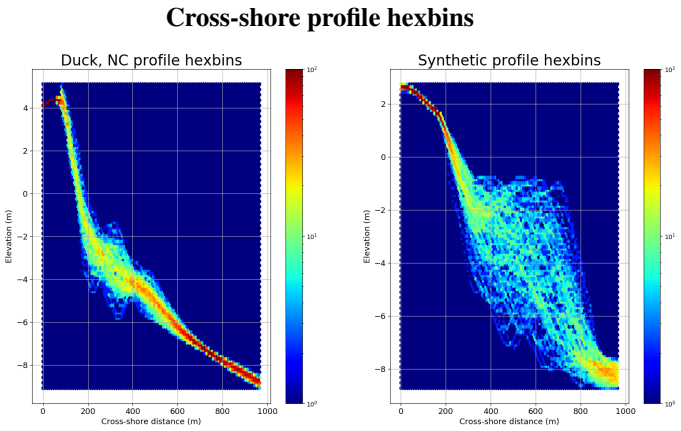


Figure 3: Density plot of the range of water depths at each location that were derived within a standard deviation of historical Duck, NC bathymetries. A similar density plot of the randomly generated bathymetries. Red areas are where nearly all of the 100 bathymetries had the same average cross-shore depth at that point.

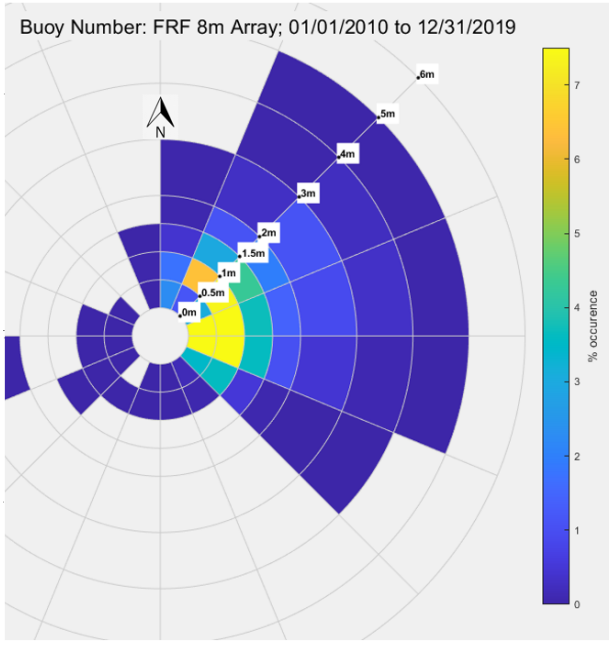


Figure 4: Binned histogram of observed wave conditions at the FRF's 8 m offshore array in Duck, NC from 1/1/2010 to 12/31/2019. The radial direction shows the incoming direction of waves. The outward axis shows the conditions binned by their wave height. For example you can see that Duck, NC, where normally incident waves travel from 71.8°, experienced waves of a 0.5 - 1 m average significant wave height from the 67.5° to 90° interval around 6% of the time during the queried interval.

pled time period. While wave conditions with wave heights smaller than 0.7 m are quite common (greater than 80% of the time), they are not considered in this study due to the very small observable surfzone features produced by low energy wave conditions and lack of wave breaking. These three conditional inputs were used as inputs to the TMA equation to generate a 2D wave spectra (Bouws et al. 1985; Hughes 1984) that is used as the input wave condition to the Celeris wave model.

Network Architecture

The need to directly convert the visual signal of breaking waves in an image to water depth from the visual input features partially motivates the usage of a 2D fully convolutional neural network (FCN), which has proven to be effective in pixel-wise regression and semantic segmentation applications in other remote sensing fields (Wu et al. 2019). Another motivation for this network selection is the potential for transfer learning. By attempting to generate and use synthetic data as visually close to measured optical data as possible, the potential exists for transfer learning, where the found bathymetric inversion ML model can be only slightly modified with the smaller subset of true coastal imagery data that exists, compared to the near limitless availability of synthetic data. The FCN setup has been demonstrated as a particularly apt network architecture for transfer learning, with examples of it being successful for semantic segmentation with similar remotely sensing data (Kemker, Salvaggio, and Kanan 2018; Kim et al. 2018; Sakurai et al. 2018; Wurm et al. 2019). The type of FCN model chosen is a modified U-net (Ronneberger, Fischer, and Brox 2015) architecture, which can be easily modified to do pixel-wise regression as well as its original use for semantic segmentation (Yao et al. 2018). The current architecture is modified to accept images of (512, 512, 4) size by adding two layers. The traditional dropout rates and upsampling methods used originally are also modified for better generalization to our domain.

Numerical Experiments

The FCN model was trained, validated, and tested using PyTorch, Tensorflow, scipy, numpy, cv2, and tifffile libraries. The PyTorch.Dataset class was overwritten to perform simultaneous loading and augmenting of the dataset to include an additional channel with information on the offshore beach slope. The FCN model and training/validating/testing functions were implemented in Tensorflow 2.0, due to the smaller memory imprint than when using a PyTorch model. The Celeris model simulations were ran on a Dell Precision 5820 with 64GB of RAM and a NVIDIA RTX 2080. The FCN model was trained on a custom built PC with 64GB of RAM and a NVIDIA RTX Titan V with 24GB of VRAM.

The final timex image used for training is a subset of the entire Celeris wave model domain (Figure 1). This timex image is stored with 3 red, green, and blue (RGB) channels as a (512, 512, 3) tiff file. When these files are loaded during training an additional channel is added to provide additional input features (slope) for a more accurate prediction, resulting in a final image size of (512, 512, 4). The constant value for slope is written to the last channel. The RGB channels are normalized across the training set. The slope is calculated by finding the physical slope from the alongshore averaged shoreline elevation to the alongshore averaged end of image depth for each bathymety. Estimated offshore beach slope is also an input to the latest parametric beach model (Holman, Lalejini, and Holland 2016). With all the inputs into the model the role of the trained FCN model is to estimate the existence and extent of perturbations from the parametric slopes by examining the breaking wave pattern observable through the timex imagery.

Training

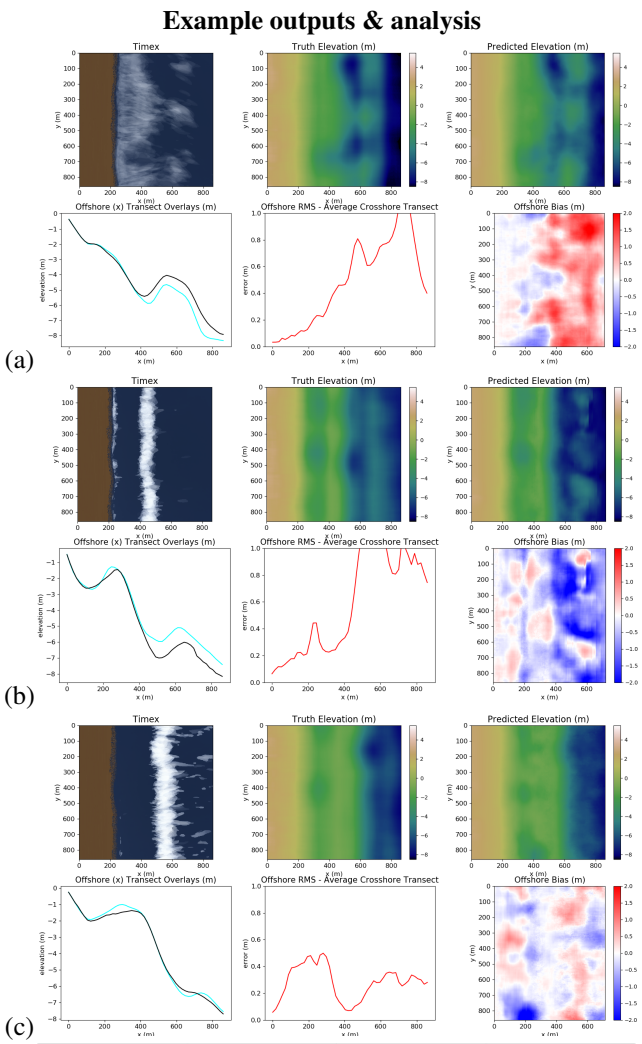
Training was performed using the timex images from 80 randomly generated bathymetries and 10 of the most highly probable wave conditions measured at Duck, NC, yielding 800 training samples, of which there are 80 unique targets. The training was done with Tensorflow 2.0's train_on_batch function and random images were selected using the modified PyTorch Dataset class for a mini-batch size of 15, which was chosen because it was the largest mini-batch size that could fit into GPU memory on current local hardware. During training, mini-batches were randomly selected from the training dataset until the end of the epoch. The validation dataset was created similarly to the training dataset but consists of 10 different bathymetries ran over the same 10 wave conditions used in the training set. At the end of each of these randomly sampled epochs validation was ran over 50 images randomly selected from the validation dataset.

The optimizer that found the best convergence was NAdam with all parameters at default settings except the starting learning rate is modified to .00008. In addition, a custom learning rate decay is introduced where the learning rate is reduced by 10 percent after the validation loss has not decreased for 8 straight epochs. Convergence with these parameters takes around 12 hours of training time on the hardware described above.

Testing

Testing was done by using timex images from 10 bathymetries and 10 wave conditions selected using Latin hypercube sampling within the realistic boundary conditions measured in Duck, NC, yielding 100 testing samples, where there were 10 unique targets. The bathymetries used for the test set were generated with the same bathymetry generation code used to make the random training bathymetry sets, but differed visually from the training and validation samples, and were not used during those processes. The wave conditions were also unique to the test set.

The testing was done with Tensorflow 2.0's predict function, with visualization done with matplotlib.pyplot. Example outputs are shown in Figure 5. In Figure 5a and 5b, the largest RMSE and a significant amount of Bias error occurs offshore of the sandbar/breaking wave visual signature (right side of images). These areas will only occasionally see breaking waves and in turn the estimates are biased by the algorithm as a result. Additionally, errors grow in the trough between the sandbar and the shoreline, where the



MeanAE	MeanSE	10% error	50% error
0.449m	0.390m	0.050m	0.341m
65% Error	80% Error	95% Error	>> Error
0.516m	0.759m	1.230m	2.251m

Figure 5: Three example input features and outputs are shown above (a;b;c). Each prediction has six frames of data shown here. The first frame shows the input RGB channels of the timex image. The second frame is the ground truth elevation. The third frame is the predicted elevation. The fourth frame shows two average offshore cross-shore transects, with the black line being the predicted transect, and the cyan line being ground truth. The fifth frame shows the average offshore cross-shore RMSE error. The sixth frame shows the offshore Bias with areas predicted too shallow in red, and areas predicted too deep in blue. The tables (d) give summary statistics on each particular input: Mean Absolute Error, Mean Square Error, Mean Bias, Median Absolute Error, Greatest Pixel Error, and percentile error, such that '95% of the pixels have less error than this value' over the offshore portion of the test set.

waves have dissipated enough energy to stop breaking before re-breaking near the shoreline, and thus little information about depth is observable in this region (Figure 5c). Over the entire test set of 10 unique bathymetries and 10 unique wave conditions the mean bias and RMSE of water depth were 0.449 m and 0.390 m (Figure 5d). In most instances across the test set, the prediction was too shallow (negative bias), exceptions to this rule are commonly seen in nearshore troughs and the seaward side of the sandbar when there are no breaking waves and the resulting prediction is often too deep (positive bias).

Conclusions

Initial results show promise in the ability of the trained FCNs to estimate nearshore water depths from synthetic wave breaking signatures expressed in timex imagery, generated with the wave model Celeris. The FCN model shows a clear ability to identify the differences between deeper and shallower areas, identifying the location of sandbars, troughs, and depressions not seen in the original training dataset, and that they are directly related the amount of breaking waves in that particular location. For this study, the bias (0.449 m) and RMSE (0.390 m) over the test set is encouraging, and comparable to other remotely sensed inversion techniques. Some error is inherent as the timex images extend up to 660m offshore, where waves are generally not breaking. We chose to use unique wave conditions during the testing phase to determine if the FCN model could show understanding that timex images that varied greatly depending on different wave patterns can still point to the same target water depth. Testing with conditions not seen during training is also important because it would be impossible to train for all possible combination of wave conditions that could be seen at a given location due to the wide ranges in wave heights, frequencies, and directions.

Future Work

Modifications to the model are currently in development. The most promising is the inclusion of wave condition features as inputs to the U-net architecture by including their values along with slope in the additional channel. These environmental parameters are hypothesized to help with the algorithm because they directly impact the resultant (RGB) timex image generated by the wave model. Development of these input features would also be advantageous for the transfer to real datasets, as they are available at most locations worldwide from global wave and tide models.

Additional future work will improve the FCN model and analysis by 1) comparing the bias and RMSE of the FCN model on the test set to predictions made by the parametric beach tool introduced by (Holman, Lalejini, and Holland 2016); 2) expand the synthetic training and testing dataset in the form of more bathymetries and wave conditions to significantly increase the ranges of slopes and wave conditions seen during training/testing; 3) introduce video frame data as a feature input into the FCN model, likely improving the accuracy in areas with little to no breaking waves by allowing the algorithm to utilize observations of wave speed

in these areas (which is proportional to water depth) in addition to wave breaking; and 4) curate and compile a real timex and bathymetry dataset that can be similarly represented by Celeris to then test the ability of the FCN model to transfer to real datasets.

Acknowledgement

We would like to thank the U.S. Army Corps of Engineers for providing the main source of funding for the research through the Deputy Assistant Secretary of the Army for Research and Technology under ERDC's Military Engineering research program titled "Force Projection Entry Operations", as well as the Geological Society of America for their funding support. We would also like to thank the many faculty and graduate students at University of North Carolina - Wilmington that provided their input and feedback on the research, as well as Dr. Jonghyun Harry Lee at the University of Hawaii at Manoa for his support.

References

Aarninkhof, S.; Ruessink, B.; and Roelvink, J. 2005. Nearshore subtidal bathymetry from time-exposure video images. *Journal of Geophysical Research: Oceans* 110(C6).

Almar, R.; Bergsma, E. W.; Maisongrande, P.; and de Almeida, L. P. M. 2019. Wave-derived coastal bathymetry from satellite video imagery: A showcase with pleiades persistent mode. *Remote Sensing of Environment* 231:111263.

Alom, M. Z.; Taha, T. M.; Yakopcic, C.; Westberg, S.; Sidike, P.; Nasrin, M. S.; Van Esesn, B. C.; Awwal, A. A. S.; and Asari, V. K. J. a. p. a. 2018. The history began from alexnet: A comprehensive survey on deep learning approaches.

Avera, W. E.; Harris, M. M.; Bibee, L. D.; Lingsch, S.; and Sample, J. T. 2002. Multibeam bathymetry from a minehunting military sonar. Report, NAVAL RESEARCH LAB STENNIS SPACE CENTER MS.

Bergsma, E. W.; Almar, R.; and Maisongrande, P. 2019. Radon-augmented sentinel-2 satellite imagery to derive wave-patterns and regional bathymetry. *Remote Sensing* 11(16):1918.

Bergsma, E. W., and Almar, R. 2018. Video-based depth inversion techniques, a method comparison with synthetic cases. *Coastal Engineering* 138:199–209.

Birkemeier, W. A., and Mason, C. 1984. The crab: A unique nearshore surveying vehicle. *Journal of Surveying Engineering* 110(1):1–7.

Bouws, E.; Günther, H.; Rosenthal, W.; and Vincent, C. 1985. Similarity of the wind wave spectrum in finite depth water: 1. spectral form. *Journal of Geophysical Research: Oceans* 90(C1):975–986.

Braud, I., and Obled, C. 1991. On the use of empirical orthogonal function (eof) analysis in the simulation of random fields. *Stochastic Hydrology and Hydraulics* 5(2):125–134.

Brodie, K. L.; Bruder, B. L.; Slocum, R. K.; and Spore, N. J. 2019. Simultaneous mapping of coastal topography and bathymetry from a lightweight multicamera uas. *IEEE Transactions on Geoscience and Remote Sensing* 57(9):6844–6864.

Bruun, P. 1954. *Coast erosion and the development of beach profiles*, volume 44. US Beach Erosion Board.

Christophe, E.; Michel, J.; Inglada, J. J. I. J. o. S. T. i. A. E. O.; and Sensing, R. 2011. Remote sensing processing: From multicore to gpu. 4(3):643–652.

Gao, J. J. P. i. P. G. 2009. Bathymetric mapping by means of remote sensing: methods, accuracy and limitations. 33(1):103–116.

Ghorbanidehno, H.; Lee, J.; Farthing, M.; Hesser, T.; Kitanidis, P. K.; and Darve, E. F. 2019. Novel data assimilation algorithm for nearshore bathymetry. *Journal of Atmospheric and Oceanic Technology* 36(4):699–715.

Guedes, R. M.; Calliari, L. J.; Holland, K. T.; Plant, N. G.; Pereira, P. S.; and Alves, F. N. 2011. Short-term sandbar variability based on video imagery: Comparison between time–average and time–variance techniques. *Marine Geology* 289(1-4):122–134.

Holland, K. T.; Lalejini, D. M.; Spansel, S. D.; and Holman, R. A. 2010. Littoral environmental reconnaissance using tactical imagery from unmanned aircraft systems. In *Ocean Sensing and Monitoring II*, volume 7678, 767806. International Society for Optics and Photonics.

Holland, K. T.; Palmsten, M. L.; et al. 2018. Remote sensing applications and bathymetric mapping in coastal environments. *Advances in Coastal Hydraulics* 375–411.

Holman, R. A., and Stanley, J. 2007. The history and technical capabilities of argus. *Coastal engineering* 54(6-7):477–491.

Holman, R. A.; Lalejini, D. M.; Edwards, K.; and Veeramony, J. 2014. A parametric model for barred equilibrium beach profiles. *Coastal engineering* 90:85–94.

Holman, R. A.; Brodie, K. L.; and Spore, N. J. 2017. Surf zone characterization using a small quadcopter: Technical issues and procedures. *IEEE Transactions on Geoscience and Remote Sensing* 55(4).

Holman, R. A.; Lalejini, D. M.; and Holland, T. J. C. E. 2016. A parametric model for barred equilibrium beach profiles: Two-dimensional implementation. 117:166–175.

Holman, R.; Plant, N.; and Holland, T. J. J. o. G. R. O. 2013. cbathy: A robust algorithm for estimating nearshore bathymetry. 118(5):2595–2609.

Hughes, S. A. 1984. The tma shallow-water spectrum description and applications. Technical report, COASTAL ENGINEERING RESEARCH CENTER VICKSBURG MS.

Huh, M.; Agrawal, P.; and Efros, A. A. J. a. p. a. 2016. What makes imagenet good for transfer learning?

Kemker, R.; Luu, R.; and Kanan, C. 2018. Low-shot learning for the semantic segmentation of remote sensing imagery. *Ieee Transactions on Geoscience and Remote Sensing* 56(10):6214–6223.

Kemker, R.; Salvaggio, C.; and Kanan, C. 2018. Algorithms for semantic segmentation of multispectral remote sensing

imagery using deep learning. *ISPRS Journal of Photogrammetry and Remote Sensing*.

Kim, H.; Kim, H.; Hong, Y. W.; and Byun, H. 2018. Detecting construction equipment using a region-based fully convolutional network and transfer learning. *Journal of computing in Civil Engineering* 32(2):04017082.

Komar, P. D., and Gaughan, M. K. 1973. Airy wave theory and breaker height prediction. In *Coastal Engineering 1972*. 405–418.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, 1097–1105.

Lippmann, T. C., and Holman, R. A. 1989. Quantification of sand bar morphology: A video technique based on wave dissipation. *Journal of Geophysical Research* 94(C1):995.

Long, C. E., and Oltman-Shay, J. M. 1991. Directional characteristics of waves in shallow water. Technical report, COASTAL ENGINEERING RESEARCH CENTER VICKSBURG MS.

Moghimi, S.; Ozkan-Haller, H. T.; Wilson, G. W.; and Kurapov, A. 2016. Data assimilation for bathymetry estimation at a tidal inlet. *Journal of Atmospheric and Oceanic Technology* 33(10):2145–2163.

Moulton, M.; Elgar, S.; and Raubenheimer, B. 2014a. Improving the time resolution of surfzone bathymetry using in situ altimeters. *Ocean Dynamics* 64(5):755–770.

Moulton, M.; Elgar, S.; and Raubenheimer, B. J. G. R. L. 2014b. A surfzone morphological diffusivity estimated from the evolution of excavated holes. 41(13):4628–4636.

Plant, N. G.; Holland, K. T.; and Haller, M. C. 2008. Ocean wavenumber estimation from wave-resolving time series imagery. *IEEE Transactions on Geoscience and Remote Sensing* 46(9):2644–2658.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, 234–241. Springer.

Sakurai, S.; Uchiyama, H.; Shimada, A.; Arita, D.; and Taniguchi, R.-i. 2018. Two-step transfer learning for semantic plant segmentation. In *ICPRAM*, 332–339.

Simonyan, K., and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Stockdon, H. F., and Holman, R. A. J. J. o. G. R. O. 2000. Estimation of wave phase speed and nearshore bathymetry from video imagery. 105(C9):22015–22033.

Tavakkol, S., and Lynett, P. 2017. Celeris: A GPU-accelerated open source software with a Boussinesq-type wave solver for real-time interactive simulation and visualization. *Computer Physics Communications* 217:117–127.

Tian, Y.; Pei, K.; Jana, S.; and Ray, B. 2018. Deeptest: Automated testing of deep-neural-network-driven autonomous cars. In *Proceedings of the 40th international conference on software engineering*, 303–314. ACM.

Van Dongeren, A.; Plant, N.; Cohen, A.; Roelvink, D.; Haller, M. C.; and Catalán, P. 2008. Beach wizard: Nearshore bathymetry estimation through assimilation of model computations and remote observations. *Coastal engineering* 55(12):1016–1027.

Wilson, G., and Berezhnoy, S. 2018. Surfzone state estimation, with applications to quadcopter-based remote sensing data. *Journal of Atmospheric and Oceanic Technology* 35(10):1881–1896.

Wilson, G. W.; Özkan Haller, H. T.; Holman, R. A.; Haller, M. C.; Honegger, D. A.; and Chickadel, C. C. 2014. Surf zone bathymetry and circulation predictions via data assimilation of remote sensing observations. *Journal of Geophysical Research: Oceans* 119(3):1993–2016.

Wilson, G. W.; Özkan Haller, H. T.; and Holman, R. A. 2010. Data assimilation and bathymetric inversion in a two-dimensional horizontal surf zone model. *Journal of Geophysical Research: Oceans* 115(C12).

Wu, G.; Guo, Y.; Song, X.; Guo, Z.; Zhang, H.; Shi, X.; Shibasaki, R.; and Shao, X. 2019. A stacked fully convolutional networks with feature alignment framework for multi-label land-cover segmentation. *Remote Sensing* 11(9):1051.

Wurm, M.; Stark, T.; Zhu, X. X.; Weigand, M.; and Taubenböck, H. 2019. Semantic segmentation of slums in satellite images using transfer learning on fully convolutional neural networks. *ISPRS journal of photogrammetry and remote sensing* 150:59–69.

Yao, W.; Zeng, Z.; Lian, C.; and Tang, H. 2018. Pixel-wise regression using u-net and its application on pansharpening. *Neurocomputing* 312:364–371.

Zhu, X. X.; Tuia, D.; Mou, L. C.; Xia, G. S.; Zhang, L. P.; Xu, F.; and Fraundorfer, F. 2017. Deep learning in remote sensing. *Ieee Geoscience and Remote Sensing Magazine* 5(4):8–36.